

Alibaba Cloud

Object Storage Service SDK Code Samples

Document Version: 20220615

Legal disclaimer

Alibaba Cloud reminds you to carefully read and fully understand the terms and conditions of this legal disclaimer before you read or use this document. If you have read or used this document, it shall be deemed as your total acceptance of this legal disclaimer.

1. You shall download and obtain this document from the Alibaba Cloud website or other Alibaba Cloud-authorized channels, and use this document for your own legal business activities only. The content of this document is considered confidential information of Alibaba Cloud. You shall strictly abide by the confidentiality obligations. No part of this document shall be disclosed or provided to any third party for use without the prior written consent of Alibaba Cloud.
2. No part of this document shall be excerpted, translated, reproduced, transmitted, or disseminated by any organization, company or individual in any form or by any means without the prior written consent of Alibaba Cloud.
3. The content of this document may be changed because of product version upgrade, adjustment, or other reasons. Alibaba Cloud reserves the right to modify the content of this document without notice and an updated version of this document will be released through Alibaba Cloud-authorized channels from time to time. You should pay attention to the version changes of this document as they occur and download and obtain the most up-to-date version of this document from Alibaba Cloud-authorized channels.
4. This document serves only as a reference guide for your use of Alibaba Cloud products and services. Alibaba Cloud provides this document based on the "status quo", "being defective", and "existing functions" of its products and services. Alibaba Cloud makes every effort to provide relevant operational guidance based on existing technologies. However, Alibaba Cloud hereby makes a clear statement that it in no way guarantees the accuracy, integrity, applicability, and reliability of the content of this document, either explicitly or implicitly. Alibaba Cloud shall not take legal responsibility for any errors or lost profits incurred by any organization, company, or individual arising from download, use, or trust in this document. Alibaba Cloud shall not, under any circumstances, take responsibility for any indirect, consequential, punitive, contingent, special, or punitive damages, including lost profits arising from the use or trust in this document (even if Alibaba Cloud has been notified of the possibility of such a loss).
5. By law, all the contents in Alibaba Cloud documents, including but not limited to pictures, architecture design, page layout, and text description, are intellectual property of Alibaba Cloud and/or its affiliates. This intellectual property includes, but is not limited to, trademark rights, patent rights, copyrights, and trade secrets. No part of this document shall be used, modified, reproduced, publicly transmitted, changed, disseminated, distributed, or published without the prior written consent of Alibaba Cloud and/or its affiliates. The names owned by Alibaba Cloud shall not be used, published, or reproduced for marketing, advertising, promotion, or other purposes without the prior written consent of Alibaba Cloud. The names owned by Alibaba Cloud include, but are not limited to, "Alibaba Cloud", "Aliyun", "HiChina", and other brands of Alibaba Cloud and/or its affiliates, which appear separately or in combination, as well as the auxiliary signs and patterns of the preceding brands, or anything similar to the company names, trade names, trademarks, product or service names, domain names, patterns, logos, marks, signs, or special descriptions that third parties identify as Alibaba Cloud and/or its affiliates.
6. Please directly contact Alibaba Cloud for any errors of this document.

Document conventions

Style	Description	Example
 Danger	A danger notice indicates a situation that will cause major system changes, faults, physical injuries, and other adverse results.	 Danger: Resetting will result in the loss of user configuration data.
 Warning	A warning notice indicates a situation that may cause major system changes, faults, physical injuries, and other adverse results.	 Warning: Restarting will cause business interruption. About 10 minutes are required to restart an instance.
 Notice	A caution notice indicates warning information, supplementary instructions, and other content that the user must understand.	 Notice: If the weight is set to 0, the server no longer receives new requests.
 Note	A note indicates supplemental instructions, best practices, tips, and other content.	 Note: You can use Ctrl + A to select all files.
>	Closing angle brackets are used to indicate a multi-level menu cascade.	Click Settings > Network > Set network type .
Bold	Bold formatting is used for buttons, menus, page names, and other UI elements.	Click OK .
<code>Courier font</code>	Courier font is used for commands	Run the <code>cd /d C:/window</code> command to enter the Windows system folder.
<i>Italic</i>	Italic formatting is used for parameters and variables.	<code>bae log list --instanceid</code> <i>Instance_ID</i>
[] or [a b]	This format is used for an optional value, where only one item can be selected.	<code>ipconfig [-all -t]</code>
{ } or {a b}	This format is used for a required value, where only one item can be selected.	<code>switch {active stand}</code>

Table of Contents

1. Overview	30
2. Java	31
2.1. Introduction	31
2.2. Installation	34
2.3. Initialization	35
2.4. Quick start	43
2.5. Buckets	51
2.5.1. Create a bucket	51
2.5.2. List buckets	51
2.5.3. Determine whether a bucket exists	58
2.5.4. Query the regions of buckets	59
2.5.5. Query bucket information	61
2.5.6. Manage the ACL of a bucket	61
2.5.7. View whether the hierarchical namespace feature is en...	65
2.5.8. Delete a bucket	65
2.5.9. Bucket tagging	67
2.5.10. Bucket inventory	75
2.5.11. Bucket policy	83
2.5.12. Retention policies	87
2.5.13. Lifecycle	93
2.5.14. Enable pay-by-requester	102
2.5.15. Hotlink protection	106
2.5.16. Logging	111
2.5.17. Static website hosting (mirroring-based back-to-origin)	115
2.5.18. Cross-region replication	125
2.5.19. CORS	129

2.5.20. Transfer acceleration	134
2.6. Upload objects	137
2.6.1. Overview	137
2.6.2. Simple upload	137
2.6.3. Form upload	145
2.6.4. Append upload	149
2.6.5. Resumable upload	152
2.6.6. Multipart upload	154
2.6.7. Progress bars	169
2.6.8. Upload callback	171
2.7. Download objects	173
2.7.1. Overview	173
2.7.2. Streaming download	173
2.7.3. Download objects to local computers	175
2.7.4. Range download	176
2.7.5. Resumable download	180
2.7.6. Conditional download	181
2.7.7. Progress bars	183
2.8. Manage objects	186
2.8.1. Overview	186
2.8.2. Determine whether an object exists	186
2.8.3. Manage object ACLs	187
2.8.4. Manage object metadata	191
2.8.5. Convert the storage classes of objects	197
2.8.6. List objects	200
2.8.7. Query objects	236
2.8.8. Rename objects	238
2.8.9. Delete objects	239

2.8.10. Copy objects -----	245
2.8.11. Prevent objects from being overwritten by objects tha...-----	251
2.8.12. Restore objects -----	259
2.8.13. Manage symbolic links -----	262
2.9. Manage directories -----	266
2.10. Versioning -----	269
2.10.1. Versioning -----	269
2.10.2. Upload objects -----	272
2.10.3. Download objects -----	278
2.10.4. Copy objects -----	280
2.10.5. List objects -----	284
2.10.6. Delete objects -----	294
2.10.7. Restore objects -----	302
2.10.8. Query object metadata -----	304
2.10.9. Manage object ACLs -----	305
2.10.10. Manage symbolic links -----	308
2.11. Object tagging -----	311
2.11.1. Configure object tagging -----	311
2.11.2. Query the tags of an object -----	328
2.11.3. Delete the tags of an object -----	331
2.11.4. Object tagging and lifecycle management -----	335
2.12. Single-connection bandwidth throttling -----	338
2.13. Data encryption -----	342
2.13.1. Server-side encryption -----	342
2.13.2. Client-side encryption -----	344
2.14. Data security -----	362
2.15. Authorized access -----	367
2.16. IMG -----	381

2.17. Error handling	389
2.18. FAQ	393
3.PHP	409
3.1. Preface	409
3.2. Installation	410
3.3. Initialization	411
3.4. Quick start	416
3.5. Buckets	421
3.5.1. Create buckets	421
3.5.2. List buckets	422
3.5.3. Determine whether buckets exist	426
3.5.4. Query the regions of buckets	427
3.5.5. Query bucket information	428
3.5.6. Query the metadata of buckets	429
3.5.7. Manage bucket ACLs	430
3.5.8. Delete a bucket	432
3.5.9. Bucket tagging	434
3.5.10. Manage lifecycle rules	437
3.5.11. Bucket policy	441
3.5.12. Retention policy	444
3.5.13. Enable pay-by-requester	449
3.5.14. Hotlink protection	453
3.5.15. Logging	456
3.5.16. Static website hosting	459
3.5.17. CORS	462
3.5.18. Map custom domain names	465
3.6. Upload objects	468
3.6.1. Overview	468

3.6.2. Simple upload	469
3.6.3. Append upload	471
3.6.4. Multipart upload	474
3.6.5. Upload callback	483
3.7. Download objects	487
3.7.1. Overview	487
3.7.2. Download objects to local files	487
3.7.3. Download to local memory	489
3.7.4. Range download	489
3.7.5. Conditional download	490
3.8. Manage objects	493
3.8.1. Overview	493
3.8.2. Determine whether a specified object exists	494
3.8.3. Manage the ACL of an object	494
3.8.4. Manage object metadata	497
3.8.5. Convert the storage classes of objects	501
3.8.6. Prevent objects from being overwritten by objects with... ..	504
3.8.7. List objects	507
3.8.8. Delete objects	510
3.8.9. Copy objects	514
3.8.10. Restore objects	518
3.8.11. Manage symbolic links	520
3.8.12. Enable MD5 verification	522
3.9. Versioning	523
3.9.1. Versioning	523
3.9.2. Upload objects	527
3.9.3. Download objects	531
3.9.4. Copy objects	533

3.9.5. List objects	537
3.9.6. Delete objects	544
3.9.7. Restore objects	548
3.9.8. Query object metadata	549
3.9.9. Manage object ACLs	550
3.9.10. Manage symbolic links	553
3.10. Object tagging	555
3.10.1. Configure object tagging	555
3.10.2. Query the tags of an object	561
3.10.3. Remove the tags of an object	563
3.11. Single-connection bandwidth throttling	565
3.12. Data encryption	567
3.12.1. Server-side encryption	567
3.13. Authorized access	571
3.14. IMG	576
3.15. Exception handling	582
4. Node.js	586
4.1. Installation	586
4.2. Initialization	587
4.3. Configuration items	588
4.4. Quick Start	589
4.5. Buckets	591
4.5.1. Create a bucket	591
4.5.2. List buckets	592
4.5.3. Determine whether a bucket exists	594
4.5.4. Query the regions of buckets	595
4.5.5. Query bucket information	596
4.5.6. Manage bucket ACLs	597

4.5.7. Delete buckets	601
4.5.8. Pay-by-requester mode	602
4.5.9. Manage lifecycle rules	605
4.5.10. Bucket policy	610
4.5.11. Retention policy	612
4.5.12. Bucket tagging	616
4.5.13. Bucket inventory	619
4.5.14. CORS	623
4.5.15. Access log	626
4.5.16. Hotlink protection	627
4.5.17. Static website hosting (mirroring-based back-to-origin)	629
4.5.18. Use custom domain names	635
4.6. Upload objects	636
4.6.1. Overview	636
4.6.2. Upload a local file	636
4.6.3. Upload from local memory	638
4.6.4. Stream upload	638
4.6.5. Multipart upload	640
4.6.6. Append upload	648
4.6.7. Resumable upload	649
4.6.8. Upload callback	650
4.7. Download objects	651
4.7.1. Overview	652
4.7.2. Download objects as files	652
4.7.3. Download to local memory	652
4.7.4. Streaming download	653
4.7.5. Range download	654
4.7.6. Conditional download	656

4.8. Manage objects	660
4.8.1. Overview	660
4.8.2. Determine whether an object exists	661
4.8.3. Manage Object Meta	661
4.8.4. Manage the ACL of an object	663
4.8.5. List objects	664
4.8.6. Copy objects	675
4.8.7. Disable overwrite for objects with the same name	680
4.8.8. Delete objects	683
4.8.9. Convert the storage classes of objects	686
4.8.10. Restore objects	687
4.8.11. Manage symbolic links	690
4.9. Configure versioning	692
4.9.1. Versioning	692
4.9.2. Upload objects	694
4.9.3. Download objects	697
4.9.4. Copy objects	697
4.9.5. List objects	699
4.9.6. Delete objects	705
4.9.7. Restore objects	708
4.9.8. Obtain object metadata	709
4.9.9. Manage object ACL	710
4.9.10. Manage symbolic links	712
4.10. Object tagging	714
4.10.1. Configure object tagging	714
4.10.2. Query the tags of an object	723
4.10.3. Delete the tags of an object	725
4.10.4. Object tagging and lifecycle rules	727

4.11. Common operations on LiveChannels	729
4.12. Single-connection bandwidth throttling	735
4.13. Server-side encryption	736
4.14. Authorized access	739
4.15. IMG	745
4.16. Error handling	751
4.17. FAQ	752
5. Python	754
5.1. Preface	754
5.2. Installation	755
5.3. Initialization	758
5.4. Quick start	762
5.5. Buckets	764
5.5.1. Create buckets	764
5.5.2. List buckets	765
5.5.3. Query the regions of buckets	766
5.5.4. Determine whether a bucket exists	767
5.5.5. Obtain bucket information	768
5.5.6. Manage bucket ACLs	769
5.5.7. Delete a bucket	770
5.5.8. Bucket tagging	772
5.5.9. Bucket inventory	774
5.5.10. Bucket policy	779
5.5.11. Retention policy	781
5.5.12. Enable pay-by-requester	783
5.5.13. Lifecycle	785
5.5.14. Hotlink protection	789
5.5.15. Logging	790

5.5.16. Static website hosting (mirroring-based back-to-origin)	792
5.5.17. CORS	797
5.5.18. Transfer acceleration	799
5.5.19. Cross-region replication	800
5.6. Upload objects	804
5.6.1. Overview	804
5.6.2. Simple upload	805
5.6.3. Append upload	809
5.6.4. Resumable upload	811
5.6.5. Multipart upload	812
5.6.6. Upload progress bars	818
5.6.7. Upload callback	818
5.7. Download objects	819
5.7.1. Overview	819
5.7.2. Streaming download	820
5.7.3. Download objects to local files	821
5.7.4. Range download	822
5.7.5. Resumable download	824
5.7.6. Download progress bars	827
5.7.7. Conditional download	827
5.8. Manage objects	828
5.8.1. Overview	828
5.8.2. Determine whether an object exists	829
5.8.3. Manage the ACL of an object	829
5.8.4. Manage object metadata	831
5.8.5. Convert the storage classes of objects	834
5.8.6. List objects	837
5.8.7. Query objects	847

5.8.8. Delete objects	852
5.8.9. Copy objects	854
5.8.10. Disable overwriting object with the same name	858
5.8.11. Restore objects	863
5.8.12. Manage symbolic links	865
5.9. Versioning	867
5.9.1. Versioning	867
5.9.2. Upload objects	868
5.9.3. Download objects	871
5.9.4. Copy objects	872
5.9.5. List objects	875
5.9.6. Delete objects	882
5.9.7. Restore objects	887
5.9.8. Obtain object metadata	888
5.9.9. Manage object ACL	889
5.9.10. Manage symbolic links	890
5.10. Object tagging	892
5.10.1. Configure object tagging	892
5.10.2. Query the tags of an object	902
5.10.3. Remove the tags of an object	904
5.10.4. Object tagging and lifecycle rules	906
5.11. Single-connection bandwidth throttling	908
5.12. Data encryption	910
5.12.1. Client-side encryption	911
5.12.2. Server-side encryption	920
5.13. Authorized access	921
5.14. Data security	925
5.15. Enable log recording	927

5.16. IMG	929
5.17. Chinese character and time	934
5.18. Exception handling	936
5.19. FAQ	938
6. Browser.js	943
6.1. Installation	943
6.2. Parameters	948
6.3. Quick start	949
6.4. Upload objects	955
6.4.1. Overview	955
6.4.2. Simple upload	955
6.4.3. Multipart upload	957
6.4.4. Resumable upload	964
6.4.5. Upload callback	966
6.5. Download objects	968
6.5.1. Overview	968
6.5.2. Preview or download an object	968
6.5.3. Range download	970
6.5.4. Conditional download	971
6.6. Manage objects	973
6.6.1. Overview	973
6.6.2. List objects	974
6.6.3. Copy an object	977
6.6.4. Set ACLs	980
6.6.5. Prevent objects from being overwritten by objects with...	980
6.6.6. Convert the storage classes of an object	984
6.6.7. Single-connection bandwidth throttling	986
6.6.8. Restore an object	987

6.6.9. Manage object metadata	991
6.6.10. Manage symbolic links	992
6.6.11. Delete objects	994
6.7. Map custom domain names	997
6.8. Authorized access	997
6.9. IMG	1001
6.10. Application in browsers	1006
6.11. Exception handling	1010
6.12. FAQ	1010
6.13. Log collection	1014
7.. NET	1015
7.1. Preface	1015
7.2. Installation	1016
7.3. Initialization	1017
7.4. Quick start	1020
7.5. Buckets	1024
7.5.1. Create buckets	1024
7.5.2. List buckets	1025
7.5.3. Determine whether buckets exist	1026
7.5.4. Query the regions of buckets	1027
7.5.5. Query bucket information	1028
7.5.6. Manage bucket ACLs	1029
7.5.7. Delete a bucket	1032
7.5.8. Bucket tagging	1033
7.5.9. Bucket inventory	1039
7.5.10. Lifecycle	1044
7.5.11. Pay-by-requester	1049
7.5.12. Bucket policy	1053

7.5.13. Logging	1055
7.5.14. CORS	1058
7.5.15. Static website hosting	1061
7.5.16. Hotlink protection	1064
7.6. Upload objects	1067
7.6.1. Overview	1067
7.6.2. Simple upload	1067
7.6.3. Append upload	1072
7.6.4. Resumable upload	1074
7.6.5. Multipart upload	1075
7.6.6. Progress bars	1085
7.6.7. Upload callback	1086
7.7. Download objects	1089
7.7.1. Overview	1089
7.7.2. Streaming download	1089
7.7.3. Range download	1090
7.7.4. Resumable download	1092
7.7.5. Download progress bars	1093
7.8. Manage objects	1095
7.8.1. Overview	1095
7.8.2. Determine whether a specified object exists	1095
7.8.3. Manage object ACLs	1096
7.8.4. Manage object metadata	1097
7.8.5. List objects	1099
7.8.6. Delete objects	1107
7.8.7. Copy objects	1109
7.8.8. Restore objects	1115
7.8.9. Manage symbolic links	1118

7.9. Versioning	1119
7.9.1. Versioning	1119
7.9.2. Upload objects	1121
7.9.3. Download objects	1125
7.9.4. Copy objects	1126
7.9.5. Delete objects	1130
7.9.6. Restore objects	1136
7.9.7. Query object metadata	1137
7.9.8. Manage the ACL of an object	1138
7.9.9. Manage symbolic links	1140
7.10. Object tagging	1142
7.10.1. Configure object tagging	1142
7.10.2. Query the tags added to an object	1156
7.10.3. Delete the tags added to an object	1156
7.10.4. Object tagging and lifecycle rules	1157
7.11. Single-connection bandwidth throttling	1159
7.12. Server-side encryption	1162
7.13. Authorized access	1165
7.14. IMG	1169
7.15. Exception handling	1176
8. Android	1178
8.1. Preface	1178
8.2. Installation	1179
8.3. Initialization	1180
8.4. Quick start	1183
8.5. Buckets	1187
8.5.1. Create buckets	1187
8.5.2. List buckets	1188

8.5.3. Query bucket ACLs	1192
8.5.4. Query bucket information	1194
8.5.5. Delete buckets	1194
8.5.6. Manage lifecycle rules	1195
8.5.7. Hotlink protection	1198
8.5.8. Access logging	1200
8.6. Upload objects	1202
8.6.1. Overview	1202
8.6.2. Simple upload	1202
8.6.3. Append upload	1207
8.6.4. Multipart upload	1209
8.6.5. Resumable upload	1216
8.6.6. Progress bar	1223
8.6.7. Upload callback	1224
8.7. Download objects	1225
8.7.1. Overview	1225
8.7.2. Streaming download	1226
8.7.3. Range download	1228
8.7.4. Download objects to local files	1229
8.7.5. Conditional download	1230
8.7.6. Progress bars	1231
8.8. Manage objects	1232
8.8.1. Overview	1232
8.8.2. Determine whether a specified object exists	1233
8.8.3. Query the ACL of an object	1233
8.8.4. Copy objects	1234
8.8.5. List objects	1236
8.8.6. Prevent objects from being overwritten by objects with...	1243

8.8.7. Delete objects	1247
8.8.8. Set the Content-Type of an object	1249
8.8.9. Obtain the metadata of an object	1250
8.8.10. Restore an archive object	1251
8.8.11. Manage symbolic links	1252
8.9. Data security	1254
8.10. Sign a URL	1255
8.11. Authorized access	1256
8.12. IMG	1259
8.13. Exception handling	1261
9.Go	1263
9.1. Preface	1263
9.2. Installation	1264
9.3. Initialization	1265
9.4. Quick start	1267
9.5. Buckets	1272
9.5.1. Create buckets	1272
9.5.2. List buckets	1273
9.5.3. Determine whether buckets exist	1277
9.5.4. Query the regions of buckets	1278
9.5.5. Query bucket information	1279
9.5.6. Manage bucket ACLs	1280
9.5.7. Delete a bucket	1282
9.5.8. Configure bucket tagging	1283
9.5.9. Bucket inventory	1287
9.5.10. Transfer acceleration	1292
9.5.11. Bucket policy	1294
9.5.12. Lifecycle	1296

9.5.13. Pay-by-requester	1300
9.5.14. Retention policies	1304
9.5.15. Hotlink protection	1309
9.5.16. Logging	1312
9.5.17. Static website hosting (mirroring-based back-to-origin)	1314
9.5.18. CRR	1321
9.5.19. CORS	1327
9.6. Upload objects	1330
9.6.1. Overview	1330
9.6.2. Simple upload	1331
9.6.3. Append upload	1336
9.6.4. Resumable upload	1338
9.6.5. Multipart upload	1340
9.6.6. Progress bars	1349
9.7. Download objects	1351
9.7.1. Overview	1351
9.7.2. Streaming download	1351
9.7.3. Range download	1354
9.7.4. Download objects	1357
9.7.5. Resumable download	1358
9.7.6. Conditional download	1361
9.7.7. Download a compressed object	1362
9.7.8. Use progress bars	1363
9.8. Manage objects	1365
9.8.1. Overview	1365
9.8.2. Determine whether a specified object exists	1365
9.8.3. Manage the ACL of an object	1366
9.8.4. Manage object metadata	1368

9.8.5. Convert the storage class of an object	1372
9.8.6. List objects	1374
9.8.7. Delete objects	1408
9.8.8. Copy objects	1412
9.8.9. Disable overwrite for objects with the same name	1417
9.8.10. Restore objects	1421
9.8.11. Manage symbolic links	1424
9.9. Versioning	1427
9.9.1. Versioning	1428
9.9.2. Upload objects	1430
9.9.3. Download objects	1436
9.9.4. Copy objects	1436
9.9.5. List objects	1440
9.9.6. Delete objects	1451
9.9.7. Restore objects	1458
9.9.8. Obtain object metadata	1459
9.9.9. Manage object ACL	1461
9.9.10. Manage symbol links	1462
9.10. Object tagging	1464
9.10.1. Configure object tagging	1464
9.10.2. Obtain the tags added to an object	1477
9.10.3. Delete the tags added to an object	1478
9.10.4. Object tagging and lifecycle rules	1478
9.11. Common operations on LiveChannels	1480
9.12. Data encryption	1491
9.12.1. Server-side encryption	1491
9.12.2. Client-side encryption	1492
9.13. Single-connection bandwidth throttling	1500

9.14. Data security	1504
9.15. Authorized access	1508
9.16. IMG	1513
9.17. Error handling	1519
10.iOS	1524
10.1. Preface	1524
10.2. Installation	1524
10.3. Quick start	1525
10.4. Initialization	1530
10.5. Buckets	1532
10.5.1. Create buckets	1532
10.5.2. List buckets	1533
10.5.3. Query bucket ACLs	1535
10.5.4. Query bucket information	1536
10.5.5. Delete buckets	1537
10.6. Upload objects	1537
10.6.1. Overview	1537
10.6.2. Simple upload	1538
10.6.3. Multipart upload	1540
10.6.4. Append upload	1546
10.6.5. Resumable upload	1547
10.6.6. Upload callback	1549
10.6.7. Upload progress bars	1550
10.7. Download objects	1551
10.7.1. Overview	1551
10.7.2. Simple download	1551
10.7.3. Streaming download	1552
10.7.4. Range download	1553

10.7.5. Resumable download	1554
10.7.6. Conditional download	1563
10.7.7. Download progress bars	1563
10.8. Manage objects	1564
10.8.1. Overview	1564
10.8.2. Determine whether the specified object exists	1564
10.8.3. Obtain object metadata	1565
10.8.4. List objects	1565
10.8.5. Copy objects	1570
10.8.6. Delete objects	1572
10.8.7. Convert the storage classes of objects	1573
10.8.8. Prevent objects from being overwritten by objects tha.....	1575
10.8.9. Manage symbolic links	1579
10.8.10. Restore Archive objects	1580
10.8.11. Manage object ACLs	1581
10.9. Data security	1582
10.10. Signed URLs	1584
10.11. Authorized access	1585
10.12. IMG	1589
10.13. Handle exceptions	1591
11.C++	1593
11.1. Preface	1593
11.2. Installation	1593
11.3. Initialization	1596
11.4. Quick start	1601
11.5. Buckets	1606
11.5.1. Create buckets	1606
11.5.2. List buckets	1607

11.5.3. Determine whether buckets exist	1609
11.5.4. Query the regions of buckets	1609
11.5.5. Query bucket information	1610
11.5.6. Manage bucket ACLs	1611
11.5.7. Delete a bucket	1614
11.5.8. Configure bucket tagging	1616
11.5.9. Bucket inventory	1623
11.5.10. Bucket policy	1628
11.5.11. Retention policy	1632
11.5.12. Lifecycle	1637
11.5.13. Pay-by-requester	1642
11.5.14. Hotlink protection	1645
11.5.15. Logging	1648
11.5.16. Static website hosting	1651
11.5.17. CORS	1654
11.6. Upload objects	1658
11.6.1. Overview	1658
11.6.2. Simple upload	1659
11.6.3. Append upload	1662
11.6.4. Resumable upload	1663
11.6.5. Multipart upload	1666
11.6.6. Interrupt an upload task	1675
11.6.7. Use a progress bar	1676
11.6.8. Upload callback	1677
11.7. Download objects	1678
11.7.1. Overview	1679
11.7.2. Download objects to local files	1679
11.7.3. Download objects to the local memory	1680

11.7.4. Range download	1682
11.7.5. Resumable download	1684
11.7.6. Use a progress bar	1687
11.8. Manage objects	1688
11.8.1. Overview	1688
11.8.2. Determine whether the specified object exists	1688
11.8.3. Manage object ACLs	1689
11.8.4. Manage object metadata	1691
11.8.5. Convert the storage class of an object	1694
11.8.6. List objects	1697
11.8.7. Delete objects	1715
11.8.8. Copy objects	1719
11.8.9. Disable overwrite for objects with the same name	1723
11.8.10. Restore objects	1729
11.8.11. Manage symbolic links	1733
11.9. Versioning	1735
11.9.1. Versioning	1736
11.9.2. Upload objects	1738
11.9.3. Download objects	1743
11.9.4. Copy objects	1744
11.9.5. Delete objects	1748
11.9.6. Restore objects	1753
11.9.7. Obtain object metadata	1754
11.9.8. Manage object ACLs	1756
11.9.9. Manage symbolic links	1757
11.10. Object tagging	1760
11.10.1. Configure object tagging	1760
11.10.2. Obtain the tags added to an object	1772

11.10.3. Delete the tags added to an object	1773
11.10.4. Object tagging and lifecycle rules	1774
11.11. Data encryption	1777
11.11.1. Server-side encryption	1777
11.11.2. Client-side encryption	1781
11.12. Single-connection bandwidth throttling	1793
11.13. Data security	1795
11.14. Authorized access	1798
11.15. Image processing	1801
12.C	1807
12.1. Preface	1807
12.2. Installation	1808
12.3. Initialization	1812
12.4. Quick start	1817
12.5. Buckets	1825
12.5.1. Create buckets	1825
12.5.2. List buckets	1826
12.5.3. Query the regions of buckets	1828
12.5.4. Query bucket information	1829
12.5.5. Manage bucket ACLs	1830
12.5.6. Delete buckets	1833
12.5.7. Manage lifecycle rules	1835
12.5.8. CORS	1840
12.5.9. Set logging	1844
12.5.10. Hotlink protection	1848
12.5.11. Static website hosting	1852
12.6. Upload objects	1856
12.6.1. Overview	1856

12.6.2. Simple upload	1856
12.6.3. Append upload	1861
12.6.4. Resumable upload	1865
12.6.5. Multipart upload	1867
12.6.6. Upload progress bars	1872
12.6.7. Upload callback	1873
12.7. Download objects	1875
12.7.1. Overview	1875
12.7.2. Download objects to local computers	1875
12.7.3. Download to your local memory	1877
12.7.4. Range download	1879
12.7.5. Resumable download	1880
12.7.6. Download progress bars	1882
12.8. Manage objects	1884
12.8.1. Overview	1884
12.8.2. Manage object ACLs	1884
12.8.3. Manage object metadata	1886
12.8.4. List objects	1889
12.8.5. Delete objects	1904
12.8.6. Copy objects	1909
12.8.7. Restore Archive objects	1913
12.8.8. Manage symbolic links	1916
12.9. Authorized access	1918
12.10. IMG	1923
12.11. Error handling	1929
13. Ruby	1932
13.1. Installation	1932
13.2. Quick start	1934

13.3. Buckets	1937
13.3.1. Create buckets	1937
13.3.2. List buckets	1938
13.3.3. Determine whether buckets exist	1938
13.3.4. Manage bucket ACLs	1938
13.3.5. Delete buckets	1939
13.3.6. Specify ACLs	1940
13.3.7. Manage lifecycle rules	1941
13.3.8. Hotlink protection	1943
13.3.9. Set logging	1944
13.3.10. CORS	1945
13.3.11. Static website hosting	1946
13.3.12. Map custom domain names	1947
13.4. Upload objects	1948
13.5. Download objects	1951
13.6. Manage objects	1954
13.7. Use STS to authorize temporary access	1960
13.8. Exception handling	1961
13.9. Rails application	1962

1. Overview

This topic describes Object Storage Service (OSS) SDKs in mainstream programming languages.

Prerequisites

- OSS is activated. For more information, see [Activate OSS](#).
- An AccessKey pair is created. For more information, see [Obtain an AccessKey pair](#).

OSS SDKs

The following table describes OSS SDKs in mainstream programming languages. The OSS SDKs describe the installation, operations, parameters, example code, and FAQ.

Programming language	References
Java	OSS SDK for Java
Python	OSS SDK for Python
Go	OSS SDK for Go
C++	OSS SDK for C++
PHP	OSS SDK for PHP
C	OSS SDK for C
Android	OSS SDK for Android
iOS	OSS SDK for iOS
.NET	OSS SDK for .NET
Node.js	OSS SDK for Node.js
Browser.js	OSS SDK for Browser.js
Ruby	OSS SDK for Ruby

2.Java

2.1. Introduction

This topic is written based on Object Storage Service (OSS) SDK for Java V3.15.0.

Compatibility

Version compatibility of OSS SDK for Java:

- OSS SDK for Java V3.x.x:
 - API operations: compatible
 - Namespace: compatible
- OSS SDK for Java V2.x.x:
 - API operations: compatible
 - Namespace: compatible
- OSS SDK for Java V1.0.x:
 - API operations: compatible
 - Namespace: incompatible. The code for Tablestore in OSS SDK for Java V1.0.x is removed from OSS SDK for Java V2.0.0 and the package names `com.aliyun.openservices.*` and `com.aliyun.openservices.oss.*` are changed to `com.aliyun.oss.*`.

Source code and API documentation

For more information about the source code of OSS SDK for Java, visit [GitHub](#). For more information about the API documentation, see [Aliyun OSS SDK for Java 3.13.2 API](#).

Sample code

OSS SDK for Java provides various sample code files for your reference or use. The following table lists the sample code files provided by OSS SDK for Java.

Sample code file	Content
GetStartedSample.java	Quick start

Sample code file	Content
BucketOperationsSample.java	• Create a bucket • List buckets • Query bucket information • Determine whether a bucket exists • Query the regions of buckets • View whether the hierarchical namespace feature is enabled • Manage the ACL of a bucket • Manage lifecycle rules • Logging • Hotlink protection • CORS • Static website hosting (mirroring-based back-to-origin) • Delete a bucket
BucketTaggingSample.java	Bucket tagging
BucketInventorySample.java	Bucket inventory
BucketPolicySample.java	Bucket policy
BucketWormSample.java	Retention policies
SetRequestPaymentSample.java	Enable pay-by-requester
BucketReplicationSample.java	Cross-region replication
BucketTransferAccelerationSample.java	Transfer acceleration
CreateFolderSample.java	Simple upload
PostObjectSample.java	Form upload  Note Use PostObject for form upload. The implementation of form upload is independent of OSS SDK for Java.
AppendObjectSample.java	Append upload
UploadSample.java	Resumable upload
MultipartUploadSample.java	Multipart upload
CallbackSample.java	Upload callback
SimpleGetObjectSample.java	Download objects

Sample code file	Content
DownloadSample.java	Resumable download
ConcurrentGetObjectSample.java	Resumable download
GetProgressSample.java	• Upload progress bars • Download progress bars
GetStartedSample.java	• Determine whether an object exists • Manage object ACLs
ObjectMetaSample.java	Manage object metadata
StorageTypeSample.java	Convert the storage classes of objects
ListObjectsSample.java	List objects
SelectObjectSample.java	Query objects
ObjectOperationSample.java	Rename objects
DeleteObjectsSample.java	Delete objects
UploadPartCopySample.java	Copy objects
ObjectOperationSample.java	Prevent objects from being overwritten by objects that have the same names
RestoreObjectSample.java	Restore objects
SymLinkSample.java	Manage symbolic links
DirectoryManageSample.java	Manage directories
BucketVersioningSample.java	Versioning
ObjectTaggingSample.java	• Configure object tagging • Query the tags of an object • Delete the tags of an object • Object tagging and lifecycle management
TrafficLimitSample.java	Single-connection bandwidth throttling
• EncryptionClientRsaSample.java • EncryptionClientKmsSample.java	Client-side encryption
EncryptionServiceSample.java	Server-side encryption
AuthorizedAccessSample.java	Authorized access

Sample code file	Content
CRCSample.java	CRC-64
ImageSample.java	Image Processing (IMG)

2.2. Installation

Before you perform operations on OSS, you must install OSS SDK for Java. This topic describes how to install OSS SDK for Java in various scenarios. Install OSS SDK for Java based on your requirements.

Prerequisites

Java 7 or later is installed.

You can run the `java -version` command to check the Java version.

Download OSS SDK for Java

- [Download SDK installation package](#)
- [Download from Git Hub](#)
- [Download previous versions](#)

Install OSS SDK for Java

You can use one of the following method to install OSS SDK for Java:

- Method 1: (Recommended) Add dependencies to your Maven project

To use OSS SDK for Java in Maven, you only need to add the corresponding dependencies to the `pom.xml` file. For OSS SDK for Java V3.10.2, add the following content to `<dependencies>`:

```
<dependency>
  <groupId>com.aliyun.oss</groupId>
  <artifactId>aliyun-sdk-oss</artifactId>
  <version>3.10.2</version>
</dependency>
```

If you use Java 9 or later versions, you must add dependencies on `jaxb`. Add the following content to `<dependencies>`:


```
<dependency>
  <groupId>javax.xml.bind</groupId>
  <artifactId>jaxb-api</artifactId>
  <version>2.3.1</version>
</dependency>
<dependency>
  <groupId>javax.activation</groupId>
  <artifactId>activation</artifactId>
  <version>1.1.1</version>
</dependency>
<!-- no more than 2.3.3-->
<dependency>
  <groupId>org.glassfish.jaxb</groupId>
  <artifactId>jaxb-runtime</artifactId>
  <version>2.3.3</version>
</dependency>
```

- Method 2: Import the JAR package to your Eclipse project

For OSS SDK for Java V3.10.2, perform the following operations to import the JAR package to your Eclipse project:

- i. Download the [OSS SDK for Java package](#).
- ii. Decompress the package.
- iii. Copy the *aliyun-sdk-oss-3.10.2.jar* file and all files in the lib folder to your project.
- iv. Right-click your project name in Eclipse and choose **Properties > Java Build Path > Add JARs**.
- v. Select all JAR files that you copy and import them to Libraries.

- Method 3: Import the JAR package to IntelliJ IDEA

For OSS SDK for Java V3.10.2, perform the following operations to import the JAR package to IntelliJ IDEA:

- i. Download the [OSS SDK for Java package](#).
- ii. Decompress the package.
- iii. Copy the *aliyun-sdk-oss-3.10.2.jar* file and all JAR files in the lib folder to your project.
- iv. In IntelliJ IDEA, select your project and choose **File > Project Structure > Modules > Dependencies > + > JARs or directories** in the right-click menu.
- v. Select all JAR files that you copy and import them to External Libraries.

2.3. Initialization

OSSClient serves as the Java client to manage Object Storage Service (OSS) resources such as buckets and objects. To use OSS SDK for Java to initiate a request, you must initialize an OSSClient instance and modify the default configuration items of ClientConfiguration.

Create an OSSClient instance

To create an OSSClient instance, you must specify an endpoint. For more information about endpoints, see [Regions and endpoints](#) and [Map a custom domain name](#).

- Create the OSSClient instance based on an OSS endpoint

The following code provides an example on how to create an OSSClient instance based on an OSS endpoint. OSSClient supports concurrent tasks. Therefore, a project can contain one or more OSSClient instances.

```
// Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com.
String endpoint = "yourEndpoint";
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a Resource Access Management (RAM) user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
String accessKeyId = "yourAccessKeyId";
String accessKeySecret = "yourAccessKeySecret";
// Create an OSSClient instance.
OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);
// Shut down the OSSClient instance.
ossClient.shutdown();
```

- Create an OSSClient instance by using a custom domain name

The following code provides an example on how to create an OSSClient instance by using a custom domain name:

```
// Set yourEndpoint to the custom domain name.
String endpoint = "yourEndpoint";
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
String accessKeyId = "yourAccessKeyId";
String accessKeySecret = "yourAccessKeySecret";
// Create a ClientConfiguration instance and modify the default parameters.
ClientBuilderConfiguration conf = new ClientBuilderConfiguration();
// Configure whether CNAME is enabled. CNAME is used to map the custom domain name to the bucket.
conf.setSupportCname(true);
// Create an OSSClient instance.
OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret, conf);
// Shut down the OSSClient instance.
ossClient.shutdown();
```

 **Note** ossClient.listBuckets is unavailable when a custom domain name is used.

- Create an OSSClient instance based on Apsara Stack or a private region

The following code provides an example on how to create an OSSClient instance based on Apsara Stack or a private region:

```
// Set yourEndpoint to the endpoint of the region in which the bucket is located.
String endpoint = "yourEndpoint";
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to
access OSS because the account has permissions on all API operations. We recommend that y
ou use a RAM user to call API operations or perform routine O&M. To create a RAM user, lo
g on to the RAM console.
String accessKeyId = "yourAccessKeyId";
String accessKeySecret = "yourAccessKeySecret";
// Create a ClientConfiguration instance and modify the default parameters.
ClientBuilderConfiguration conf = new ClientBuilderConfiguration();
// Disable CNAME.
conf.setSupportCname(false);
// Create an OSSClient instance.
OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret, conf
);
// Shut down the OSSClient instance.
ossClient.shutdown();
```

- Create an OSSClient instance based on an IP address

The following code provides an example on how to create an OSSClient instance based on an IP address:

```
// In some special scenarios such as a private region, an IP address is used as the endpo
int. In such scenarios, specify the actual IP address.
String endpoint = "https://10.10.10.10";
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to
access OSS because the account has permissions on all API operations. We recommend that y
ou use a RAM user to call API operations or perform routine O&M. To create a RAM user, lo
g on to the RAM console.
String accessKeyId = "yourAccessKeyId";
String accessKeySecret = "yourAccessKeySecret";
// Create a ClientConfiguration instance.
ClientBuilderConfiguration conf = new ClientBuilderConfiguration();
// Enable access from a second-level domain because access from a second-level domain is
disabled by default. You must configure this parameter for OSS SDK for Java V2.1.2 or ear
lier, because versions later than 2.1.2 automatically detect IP addresses.
conf.setSLDEnabled(true);
// Create an OSSClient instance.
OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret, conf
);
// Shut down the OSSClient instance.
ossClient.shutdown();
```

- Create an OSSClient instance based on STS

The following code provides an example on how to create an OSSClient instance based on Security Token Service (STS):

 **Note** For more information about how to set up STS, see [Use a temporary credential provided by STS to access OSS](#) in OSS Developer Guide. You can call the [AssumeRole](#) operation or use [STS SDKs for various programming languages](#) to obtain a temporary access credential. The temporary access credential contains a security token and a temporary AccessKey pair that consists of an AccessKey ID and an AccessKey secret. The minimum validity period of a temporary access credential is 900 seconds. The maximum validity period of a temporary access credential is the maximum session duration specified for the current role. For more information, see [Specify the maximum session duration for a RAM role](#).

```
// Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com.
String endpoint = "yourEndpoint";
// Specify the temporary AccessKey pair obtained from STS.
String accessKeyId = "yourAccessKeyId";
String accessKeySecret = "yourAccessKeySecret";
// Specify the security token obtained from STS.
String securityToken = "yourSecurityToken";
// Create an OSSClient instance.
OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret, securityToken);
// Shut down the OSSClient instance.
ossClient.shutdown();
```

For more information, see [Authorized access](#).

- Create an OSSClient instance by using STSAssumeRole

The following code provides an example on how to create an OSSClient by using STSAssumeRole:

```
// Specify the region that STSAssumeRole can access. In this example, the China (Hangzhou)
// region is used. For other regions, specify this parameter based on the actual region.
String region = "cn-hangzhou";
// Specify the AccessKey ID and AccessKey secret of the RAM user.
String accessKeyId = "yourAccessKeyId";
String accessKeySecret = "yourAccessKeySecret";
// Specify the Alibaba Cloud Resource Name (ARN) of STSAssumeRole, which indicates the ID
// of the role. Format: acs:ram::$accountId:role/$roleName.
// $accountId indicates the Alibaba Cloud account ID. To view the Alibaba Cloud account ID,
// perform the following steps: Log on to the OSS console, and move the pointer over the
// profile picture in the upper-right corner to view and copy the account ID, or click Basic
// Information to view the account ID.
// $roleName indicates the name of the RAM role. To view the RAM role name, perform the following
// steps: Log on to the RAM console. In the left-side navigation pane, choose Identities > Roles.
// On the page that appears, view the name in the Role Name column.
String roleArn = "acs:ram::17464958****:role/ossststest";
// Create an STSAssumeRoleSessionCredentialsProvider instance.
STSAssumeRoleSessionCredentialsProvider provider = CredentialsProviderFactory.newSTSAssumeRoleSessionCredentialsProvider(
    region, accessKeyId, accessKeySecret, roleArn);
// Set yourEndpoint to the endpoint of the region in which the bucket is located. For example,
// if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com.
String endpoint = "yourEndpoint";
// Create a ClientConfiguration instance.
ClientBuilderConfiguration conf = new ClientBuilderConfiguration();
// Create an OSSClient instance.
OSS ossClient = new OSSClientBuilder().build(endpoint, provider, conf);
// Shut down the OSSClient instance.
ossClient.shutdown();
```

For more information, see [Use a temporary credential provided by STS to access OSS](#).

- Create an OSSClient instance by using EcsRamRole

You can access OSS from an Elastic Compute Service (ECS) instance by using a RAM role bound to the instance. You can bind a RAM role to an ECS instance to access OSS from the instance by using STS temporary credentials. STS temporary credentials are automatically generated and updated. Applications can obtain the STS temporary credentials by using the instance metadata URL.

The following code provides an example on how to create an OSSClient by using EcsRamRole:

```
// Use the EcsRamRole role to access OSS.
// In this example, the China (Hangzhou) region is used. For other regions, specify the endpoint of the actual region.
String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
// Use EcsRamRole to obtain the access credential. The following code shows how to obtain the access credential by using a role named ecs-ram-role.
InstanceProfileCredentialsProvider provider = CredentialsProviderFactory.newInstanceProfileCredentialsProvider("ecs-ram-role");
// Create an OSSClient instance.
OSS ossClient = new OSSClientBuilder().build(endpoint, provider, new ClientBuilderConfiguration());
// Shut down the OSSClient instance.
ossClient.shutdown();
```

Configure an OSSClient instance

ClientConfiguration is a configuration class of OSSClient. You can use ClientConfiguration to configure parameters such as host proxies, connection timeouts, and the maximum number of connections. The following table describes the parameters that you can configure when you configure an OSSClient instance.

Parameter	Description	Method
MaxConnections	Specifies the maximum number of HTTP connections that are allowed. Default value: 1024.	ClientConfiguration.setMaxConnections
SocketTimeout	Specifies the timeout period for data transmission at the socket layer. Unit: milliseconds. Default value: 50000.	ClientConfiguration.setSocketTimeout
ConnectionTimeout	Specifies the timeout period for establishing a connection. Unit: milliseconds. Default value: 50000.	ClientConfiguration.setConnectionTimeout
ConnectionRequestTimeout	Specifies the timeout period for obtaining a connection from the connection pool. Unit: milliseconds. By default, the timeout period is unlimited.	ClientConfiguration.setConnectionRequestTimeout
IdleConnectionTime	Specifies the timeout period for idle connections. The connection is closed when the timeout period ends. Unit: milliseconds. Default value: 60000.	ClientConfiguration.setIdleConnectionTime
MaxErrorRetry	Specifies the maximum number of retry attempts that are allowed when a request error occurs. Default value: 3.	ClientConfiguration.setMaxErrorRetry
SupportCname	Specifies whether CNAME can be used as an endpoint. By default, CNAME can be used as an endpoint.	ClientConfiguration.setSupportCname

Parameter	Description	Method
SLDEnabled	Specifies whether access from second-level domains is enabled. By default, access from second-level domains is disabled.	ClientConfiguration.setSLDEnabled
Protocol	The protocol used to connect to OSS. Default value: HTTP. Valid values: HTTP and HTTPS.	ClientConfiguration.setProtocol
UserAgent	Specifies the user agent (the User-Agent field in the HTTP header). Default value: <code>aliyun-sdk-java</code> .	ClientConfiguration.setUserAgent
ProxyHost	Specifies the IP address to access the proxy host.	ClientConfiguration.setProxyHost
ProxyPort	Specifies the port for the proxy server.	ClientConfiguration.setProxyPort
ProxyUsername	Specifies the username verified by the proxy server.	ClientConfiguration.setProxyUsername
ProxyPassword	Specifies the password verified by the proxy server.	ClientConfiguration.setProxyPassword
RedirectEnable	Specifies whether HTTP redirection is enabled.  Note You can configure whether HTTP redirection is enabled for OSS SDK for Java V3.10.1 or later. By default, HTTP redirection is enabled for OSS SDK for Java V3.10.1 or later.	ClientConfiguration.setRedirectEnable
VerifySSLEnable	Specifies whether SSL-based authentication is enabled.  Note You can configure whether SSL-based authentication is enabled for OSS SDK for Java V3.10.1 or later. By default, SSL-based authentication is enabled for OSS SDK for Java V3.10.1 or later.	ClientConfiguration.setVerifySSLEnable

The following code provides an example on how to configure parameters for an `OSSClient` instance by using `ClientConfiguration`:

```
// Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-
```

```

// If the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-
s-cn-hangzhou.aliyuncs.com.
String endpoint = "yourEndpoint";
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to ac-
cess OSS because the account has permissions on all API operations. We recommend that you u-
se a RAM user to call API operations or perform routine O&M. To create a RAM user, log on t-
o the RAM console.
String accessKeyId = "yourAccessKeyId";
String accessKeySecret = "yourAccessKeySecret";
// Create a ClientConfiguration instance. ClientConfiguration is a configuration class of O-
SSClient. You can use ClientConfiguration to configure parameters such as host proxies, con-
nection timeouts, and the maximum number of connections.
ClientBuilderConfiguration conf = new ClientBuilderConfiguration();
// Configure the maximum number of HTTP connections allowed by an OSSClient instance. Defau-
lt value: 1024.
conf.setMaxConnections(200);
// Configure the timeout period for data transmission at the socket layer. Unit: millisecon-
ds. Default value: 50000.
conf.setSocketTimeout(10000);
// Configure the timeout period for establishing a connection. Unit: milliseconds. Default
value: 50000.
conf.setConnectionTimeout(10000);
// Configure the timeout period for obtaining a connection from the connection pool. Unit:
milliseconds. By default, the timeout period is unlimited.
conf.setConnectionRequestTimeout(1000);
// Configure the timeout period for idle connections. The connection is closed when the tim-
eout period ends. Unit: milliseconds. Default value: 60000.
conf.setIdleConnectionTime(10000);
// Configure the maximum number of allowed retry attempts when a request error occurs. Defa-
ult value: 3.
conf.setMaxErrorRetry(5);
// Configure whether CNAME can be used as an endpoint. By default, CNAME can be used as an
endpoint.
conf.setSupportCname(true);
// Configure whether access from second-level domains is enabled. By default, access from s-
econd-level domains is disabled.
conf.setSLDEnabled(true);
// Configure the protocol used to connect to OSS. Valid values: HTTP and HTTPS. Default val-
ue: HTTP.
conf.setProtocol(Protocol.HTTP);
// Configure User-Agent in the HTTP header. Default value: aliyun-sdk-java.
conf.setUserAgent("aliyun-sdk-java");
// Configure the port for the proxy host.
conf.setProxyHost("<yourProxyHost>");
// Configure the username verified by the proxy host.
conf.setProxyUsername("<yourProxyUserName>");
// Configure the password verified by the proxy host.
conf.setProxyPassword("<yourProxyPassword>");
// Configure whether HTTP redirection is enabled. By default, HTTP redirection is enabled.
conf.setRedirectEnable(true);
// Configure whether SSL-based authentication is enabled. By default, SSL-based authenticat-
ion is enabled.
conf.setVerifySSLEnable(true);
// Create an OSSClient instance.
OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret, conf);

```

```
// Shut down the OSSClient instance.
ossClient.shutdown();
```

Retry policies

When an error occurs, OSS uses different retry policies for different request types.

- If a request is a POST request, OSS does not retry the request by default.
- If a non-POST request matches one of the following scenarios, OSS retries the request by based on the default retry policy. OSS retries three times at most.
 - The exception is `ClientException` and the returned error code is `ConnectionTimeout`, `SocketTimeout`, `ConnectionRefused`, `UnknownHost`, or `SocketException`.
 - The exception is `OSSEException`, and the returned error code is not `InvalidResponse`.
 - The returned status code is 500, 502, or 503.

If the default retry policy cannot meet your requirements, you can customize a retry policy and the maximum number of retries by using the `ClientConfiguration` class. Generally, we recommend that you do not customize retry policies. The following code provides an example on how to customize a retry policy:

```
// Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com.
String endpoint = "yourEndpoint";
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
String accessKeyId = "yourAccessKeyId";
String accessKeySecret = "yourAccessKeySecret";
// Create a ClientConfiguration instance.
// ClientConfiguration is a configuration class of OSSClient. ClientConfiguration can be used to customize retry policies and configure parameters such as retry times and the connection timeout period.
ClientConfiguration conf= new ClientConfiguration();
// Configure the maximum number of allowed retry attempts when a request error occurs. Default value: 3.
conf.setMaxErrorRetry(5);
// Customize a retry policy. Generally, we recommend that you do not customize retry policies.
// Assume that the TestRetryStrategy class is a retry policy that you customized. Then, the TestRetryStrategy class must inherit RetryStrategy.
conf.setRetryStrategy(new TestRetryStrategy());
// Create an OSSClient instance.
OSS ossClient=new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret, conf);
// Shut down the OSSClient instance.
ossClient.shutdown();
```

2.4. Quick start

This topic describes how to use Object Storage Service (OSS) SDK for Java to perform routine operations, such as creating a bucket, uploading an object, and downloading an object.

Sample projects

OSS SDK for Java provides sample projects for Maven and Ant. You can compile and run the sample projects on local devices. You can also develop your own applications based on the sample projects. For more information about how to compile and run a project, see README.md in the project directory.

- Sample project for Maven: [aliyun-oss-java-sdk-demo-mvn.zip](#)
- Sample project for Ant: [aliyun-oss-java-sdk-demo-ant.zip](#)

Create a bucket

A bucket is a global namespace in OSS. A bucket is a container for objects stored in OSS.

The following code provides an example on how to create a bucket:

```

public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify the actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the name of the bucket. Example: examplebucket.
        String bucketName = "examplebucket";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);

        try {
            // Create the bucket.
            ossClient.createBucket(bucketName);
        } catch (OSSException oe) {
            System.out.println("Caught an OSSException, which means your request made it to OSS, "
                + "but was rejected with an error response for some reason.");
            System.out.println("Error Message:" + oe.getErrorMessage());
            System.out.println("Error Code:" + oe.getErrorCode());
            System.out.println("Request ID:" + oe.getRequestId());
            System.out.println("Host ID:" + oe.getHostId());
        } catch (ClientException ce) {
            System.out.println("Caught an ClientException, which means the client encountered "
                + "a serious internal problem while trying to communicate with OSS, "
                + "such as not being able to access the network.");
            System.out.println("Error Message:" + ce.getMessage());
        } finally {
            if (ossClient != null) {
                ossClient.shutdown();
            }
        }
    }
}

```

Upload objects

The following code provides an example on how to upload objects to OSS by using stream upload:

```

import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import java.io.ByteArrayInputStream;

public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify the actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the name of the bucket. Example: examplebucket.
        String bucketName = "examplebucket";
        // Specify the full path of the object. Example: exampledir/exampleobject.txt. The full path of the object cannot contain the bucket name.
        String objectName = "exampledir/exampleobject.txt";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);

        try {
            String content = "Hello OSS";
            ossClient.putObject(bucketName, objectName, new ByteArrayInputStream(content.getBytes()));
        } catch (OSSException oe) {
            System.out.println("Caught an OSSException, which means your request made it to OSS, "
                + "but was rejected with an error response for some reason.");
            System.out.println("Error Message:" + oe.getErrorMessage());
            System.out.println("Error Code:" + oe.getErrorCode());
            System.out.println("Request ID:" + oe.getRequestId());
            System.out.println("Host ID:" + oe.getHostId());
        } catch (ClientException ce) {
            System.out.println("Caught an ClientException, which means the client encountered "
                + "a serious internal problem while trying to communicate with OSS, "
                + "such as not being able to access the network.");
            System.out.println("Error Message:" + ce.getMessage());
        } finally {
            if (ossClient != null) {
                ossClient.shutdown();
            }
        }
    }
}

```

Download objects

The following code provides an example on how to download objects from OSS by using stream download:

```
import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.OSSObject;
import java.io.BufferedReader;
import java.io.InputStream;
import java.io.InputStreamReader;
public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify the actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the name of the bucket. Example: examplebucket.
        String bucketName = "examplebucket";
        // Specify the full path of the object. Example: exampledir/exampleobject.txt. The full path of the object cannot contain the bucket name.
        String objectName = "exampledir/exampleobject.txt";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);
        try {
            // Call ossClient.getObject to obtain an OSSObject instance. The OSSObject instance contains the content and metadata of the object.
            OSSObject ossObject = ossClient.getObject(bucketName, objectName);
            // Call ossObject.getObjectContent to obtain an input stream of the object. The input stream can be read to obtain the content of the object.
            InputStream content = ossObject.getObjectContent();
            if (content != null) {
                BufferedReader reader = new BufferedReader(new InputStreamReader(content));
                while (true) {
                    String line = reader.readLine();
                    if (line == null) break;
                    System.out.println("\n" + line);
                }
                // You must close the obtained stream after the object is read. Otherwise, connection leaks may occur. Consequently, no connections are available and an exception occurs.
                content.close();
            }
        } catch (OSSException oe) {
            System.out.println("Caught an OSSException, which means your request made it to OSS, "
                + "but was rejected with an error response for some reason.");
            System.out.println("Error Message:" + oe.getErrorMessage());
        }
    }
}
```

```
        System.out.println("Error Code:" + oe.getErrorCode());
        System.out.println("Request ID:" + oe.getRequestId());
        System.out.println("Host ID:" + oe.getHostId());
    } catch (ClientException ce) {
        System.out.println("Caught an ClientException, which means the client encounter  
ed "
            + "a serious internal problem while trying to communicate with OSS, "
            + "such as not being able to access the network.");
        System.out.println("Error Message:" + ce.getMessage());
    } finally {
        if (ossClient != null) {
            ossClient.shutdown();
        }
    }
}
```

List objects

The following code provides an example on how to list objects stored in a bucket named `examplebucket`. By default, a maximum of 100 objects are listed.

```

import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.OSSObjectSummary;
import com.aliyun.oss.model.ObjectListing;
public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify the actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the name of the bucket. Example: examplebucket.
        String bucketName = "examplebucket";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);

        try {
            // Call ossClient.listObjects to obtain an ObjectListing instance. The ObjectListing instance contains the response to the listObject request.
            ObjectListing objectListing = ossClient.listObjects(bucketName);
            // Call objectListing.getObjectSummaries to obtain the descriptions of all objects.
            for (OSSObjectSummary objectSummary : objectListing.getObjectSummaries()) {
                System.out.println("- " + objectSummary.getKey() + " " +
                    "size = " + objectSummary.getSize() + " ");
            }
        } catch (OSSException oe) {
            System.out.println("Caught an OSSException, which means your request made it to OSS, "
                + "but was rejected with an error response for some reason.");
            System.out.println("Error Message:" + oe.getErrorMessage());
            System.out.println("Error Code:" + oe.getErrorCode());
            System.out.println("Request ID:" + oe.getRequestId());
            System.out.println("Host ID:" + oe.getHostId());
        } catch (ClientException ce) {
            System.out.println("Caught an ClientException, which means the client encountered "
                + "a serious internal problem while trying to communicate with OSS, "
                + "such as not being able to access the network.");
            System.out.println("Error Message:" + ce.getMessage());
        } finally {
            if (ossClient != null) {
                ossClient.shutdown();
            }
        }
    }
}

```

Delete an object

The following code provides an example on how to delete an object:

```
import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;

public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify the actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the name of the bucket. Example: examplebucket.
        String bucketName = "examplebucket";
        // Specify the full path of the object. Example: exampledir/exampleobject.txt. The full path of the object cannot contain the bucket name.
        String objectName = "exampledir/exampleobject.txt";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);

        try {
            // Delete the object.
            ossClient.deleteObject(bucketName, objectName);
        } catch (OSSException oe) {
            System.out.println("Caught an OSSException, which means your request made it to OSS, "
                + "but was rejected with an error response for some reason.");
            System.out.println("Error Message:" + oe.getErrorMessage());
            System.out.println("Error Code:" + oe.getErrorCode());
            System.out.println("Request ID:" + oe.getRequestId());
            System.out.println("Host ID:" + oe.getHostId());
        } catch (ClientException ce) {
            System.out.println("Caught an ClientException, which means the client encountered "
                + "a serious internal problem while trying to communicate with OSS, "
                + "such as not being able to access the network.");
            System.out.println("Error Message:" + ce.getMessage());
        } finally {
            if (ossClient != null) {
                ossClient.shutdown();
            }
        }
    }
}
```

References

- For more information about the complete sample code for the operations described in this topic, visit [GitHub](#).
- For more information about the API operation that you can call to create a bucket, see [PutBucket](#).
- For more information about the API operation that you can call to upload an object, see [PutObject](#).
- For more information about the API operation that you can call to download an object, see [GetObject](#).
- For more information about the API operation that you can call to list objects, see [GetBucket \(ListObjects\)](#).
- For more information about the API operation that you can call to delete an object, see [DeleteObject](#).

2.5. Buckets

2.5.1. Create a bucket

A bucket is a container used to store objects in Object Storage Service (OSS). Each object is contained in a bucket. This topic describes how to create a bucket.

Sample code

The following code provides an example on how to create a bucket named `examplebucket`:

```
// Create an OSSClient instance.
OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);
// Create the bucket and enable the hierarchical namespace feature for the bucket.
CreateBucketRequest createBucketRequest = new CreateBucketRequest(bucketName).withHnsStatus(HnsStatus.Enabled);
// The following code provides an example on how to specify the storage class, access control list (ACL), and redundancy type of the bucket that you want to create.
// In this example, the storage class of the bucket is set to Standard.
createBucketRequest.setStorageClass(StorageClass.Standard);
// By default, the redundancy type of a bucket is set to DataRedundancyType.LRS, which indicates locally redundant storage (LRS).
createBucketRequest.setDataRedundancyType(DataRedundancyType.LRS);
// Set the ACL of the bucket to public-read. The default value is private.
createBucketRequest.setCannedACL(CannedAccessControlList.PublicRead);
ossClient.createBucket(createBucketRequest);
// Shut down the OSSClient instance.
ossClient.shutdown();
```

References

- For the complete sample code that is used to create a bucket, visit [GitHub](#).
- For more information about the API operation that you can call to create a bucket, see [PutBucket](#).

2.5.2. List buckets

A bucket is a container that is used to store objects in Object Storage Service (OSS). All objects in OSS are contained in buckets. Bucket names are listed in alphabetical order. You can list all buckets that belong to the current Alibaba Cloud account. You can also list a specified number of buckets or only the buckets that meet specified conditions.

 **Note** To list buckets of the Infrequent Access (IA) or Archive storage class, use OSS SDK for Java V2.6.0 or later.

List all buckets that belong to an Alibaba Cloud account

The following sample code provides an example on how to list all buckets that belong to an Alibaba Cloud account:

```

import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.Bucket;
import java.util.List;
public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);

        try {
            // List all buckets that belong to the current account.
            List<Bucket> buckets = ossClient.listBuckets();
            for (Bucket bucket : buckets) {
                System.out.println("- " + bucket.getName());
            }
        } catch (OSSException oe) {
            System.out.println("Caught an OSSException, which means your request made it to OSS, "
                + "but was rejected with an error response for some reason.");
            System.out.println("Error Message:" + oe.getErrorMessage());
            System.out.println("Error Code:" + oe.getErrorCode());
            System.out.println("Request ID:" + oe.getRequestId());
            System.out.println("Host ID:" + oe.getHostId());
        } catch (ClientException ce) {
            System.out.println("Caught an ClientException, which means the client encountered "
                + "a serious internal problem while trying to communicate with OSS, "
                + "such as not being able to access the network.");
            System.out.println("Error Message:" + ce.getMessage());
        } finally {
            if (ossClient != null) {
                ossClient.shutdown();
            }
        }
    }
}

```

List buckets whose names contain a specified prefix

The following sample code provides an example on how to list buckets whose names contain the "example" prefix:


```
import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.Bucket;
import com.aliyun.oss.model.BucketList;
import com.aliyun.oss.model.ListBucketsRequest;
public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);

        try {
            ListBucketsRequest listBucketsRequest = new ListBucketsRequest();
            // List the buckets whose names contain the specified prefix.
            // Specify the prefix.
            listBucketsRequest.setPrefix("example");
            BucketList bucketList = ossClient.listBuckets(listBucketsRequest);
            for (Bucket bucket : bucketList.getBucketList()) {
                System.out.println("- " + bucket.getName());
            }
        } catch (OSSException oe) {
            System.out.println("Caught an OSSException, which means your request made it to OSS, "
                + "but was rejected with an error response for some reason.");
            System.out.println("Error Message:" + oe.getErrorMessage());
            System.out.println("Error Code:" + oe.getErrorCode());
            System.out.println("Request ID:" + oe.getRequestId());
            System.out.println("Host ID:" + oe.getHostId());
        } catch (ClientException ce) {
            System.out.println("Caught an ClientException, which means the client encountered "
                + "a serious internal problem while trying to communicate with OSS, "
                + "such as not being able to access the network.");
            System.out.println("Error Message:" + ce.getMessage());
        } finally {
            if (ossClient != null) {
                ossClient.shutdown();
            }
        }
    }
}
```

List buckets whose names are alphabetically after the value of marker

The following sample code provides an example on how to list buckets whose names are alphabetically after examplebucket:

```
import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.Bucket;
import com.aliyun.oss.model.BucketList;
import com.aliyun.oss.model.ListBucketsRequest;
public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the name of the bucket. Example: examplebucket.
        String bucketName = "examplebucket";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);

        try {
            ListBucketsRequest listBucketsRequest = new ListBucketsRequest();
            // List buckets whose names are alphabetically after the value of marker.
            // Specify the value of marker. Buckets whose names are alphabetically after the bucket specified by the marker are returned.
            listBucketsRequest.setMarker(bucketName);
            BucketList bucketList = ossClient.listBuckets(listBucketsRequest);
            for (Bucket bucket : bucketList.getBucketList()) {
                System.out.println("- " + bucket.getName());
            }
        } catch (OSSException oe) {
            System.out.println("Caught an OSSException, which means your request made it to OSS, "
                + "but was rejected with an error response for some reason.");
            System.out.println("Error Message:" + oe.getErrorMessage());
            System.out.println("Error Code:" + oe.getErrorCode());
            System.out.println("Request ID:" + oe.getRequestId());
            System.out.println("Host ID:" + oe.getHostId());
        } catch (ClientException ce) {
            System.out.println("Caught an ClientException, which means the client encountered "
                + "a serious internal problem while trying to communicate with OSS, "
                + "such as not being able to access the network.");
            System.out.println("Error Message:" + ce.getMessage());
        } finally {
            if (ossClient != null) {
                ossClient.shutdown();
            }
        }
    }
}
```

```
        if (ossClient != null) {  
            ossClient.shutdown();  
        }  
    }  
}
```

List a specified number of buckets

The following sample code provides an example on how to list 500 buckets:

```

import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.Bucket;
import com.aliyun.oss.model.BucketList;
import com.aliyun.oss.model.ListBucketsRequest;
public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);

        try {
            ListBucketsRequest listBucketsRequest = new ListBucketsRequest();
            // Set the maximum number of buckets that can be listed to 500. The default value is 100, and the maximum value is 1000.
            listBucketsRequest.setMaxKeys(500);
            BucketList bucketList = ossClient.listBuckets(listBucketsRequest);
            for (Bucket bucket : bucketList.getBucketList()) {
                System.out.println("- " + bucket.getName());
            }
        } catch (OSSException oe) {
            System.out.println("Caught an OSSException, which means your request made it to OSS, "
                + "but was rejected with an error response for some reason.");
            System.out.println("Error Message:" + oe.getErrorMessage());
            System.out.println("Error Code:" + oe.getErrorCode());
            System.out.println("Request ID:" + oe.getRequestId());
            System.out.println("Host ID:" + oe.getHostId());
        } catch (ClientException ce) {
            System.out.println("Caught an ClientException, which means the client encountered "
                + "a serious internal problem while trying to communicate with OSS, "
                + "such as not being able to access the network.");
            System.out.println("Error Message:" + ce.getMessage());
        } finally {
            if (ossClient != null) {
                ossClient.shutdown();
            }
        }
    }
}

```

References

- For more information about the complete sample code that is used to list buckets, visit [GitHub](#).
- For more information about the API operation that you can call to list buckets, see [GetService \(ListBuckets\)](#).

2.5.3. Determine whether a bucket exists

A bucket is a container used to store objects in Object Storage Service (OSS). Every object is contained in a bucket. This topic describes how to determine whether a bucket exists.

Sample code

The following sample code shows how to determine whether a bucket named examplebucket exists:

```
package com.aliyun.oss.demo;
import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify the actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);

        try {
            // Determine whether the bucket named examplebucket exists. If true is returned, the bucket exists. If false is returned, the bucket does not exist.
            boolean exists = ossClient.doesBucketExist("examplebucket");
            System.out.println(exists);
        } catch (OSSException oe) {
            System.out.println("Caught an OSSException, which means your request made it to OSS, "
                + "but was rejected with an error response for some reason.");
            System.out.println("Error Message:" + oe.getErrorMessage());
            System.out.println("Error Code:" + oe.getErrorCode());
            System.out.println("Request ID:" + oe.getRequestId());
            System.out.println("Host ID:" + oe.getHostId());
        } catch (ClientException ce) {
            System.out.println("Caught an ClientException, which means the client encountered "
                + "a serious internal problem while trying to communicate with OSS, "
                + "such as not being able to access the network.");
            System.out.println("Error Message:" + ce.getMessage());
        } finally {
            if (ossClient != null) {
                ossClient.shutdown();
            }
        }
    }
}
```

References

For more information about the complete sample code for determining whether a bucket exists, visit [GitHub](#).

2.5.4. Query the regions of buckets

A bucket is a container for objects stored in Object Storage Service (OSS). Every object is contained in a bucket. This topic describes how to query the region of a bucket.

Sample code

The following sample code shows how to query the region or location of a bucket:

```
import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;

public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify the actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the name of the bucket. Example: examplebucket.
        String bucketName = "examplebucket";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);

        try {
            String location = ossClient.getBucketLocation(bucketName);
            System.out.println(location);
        } catch (OSSException oe) {
            System.out.println("Caught an OSSException, which means your request made it to OSS, "
                + "but was rejected with an error response for some reason.");
            System.out.println("Error Message:" + oe.getErrorMessage());
            System.out.println("Error Code:" + oe.getErrorCode());
            System.out.println("Request ID:" + oe.getRequestId());
            System.out.println("Host ID:" + oe.getHostId());
        } catch (ClientException ce) {
            System.out.println("Caught an ClientException, which means the client encountered "
                + "a serious internal problem while trying to communicate with OSS, "
                + "such as not being able to access the network.");
            System.out.println("Error Message:" + ce.getMessage());
        } finally {
            if (ossClient != null) {
                ossClient.shutdown();
            }
        }
    }
}
```

References

- For more information about the complete sample code that is used to query the region of a bucket, visit [Git Hub](#).
- For more information about the API operation that you can call to query the region of a bucket, see [GetBucketLocation](#).

2.5.5. Query bucket information

A bucket is a container for objects stored in Object Storage Service (OSS). Every object is contained in a bucket. This topic describes how to query the information about a bucket.

The following code provides an example on how to query the information about a bucket, including the region and creation date of the bucket:

```
// Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com.
String endpoint = "yourEndpoint";
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a Resource Access Management (RAM) user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
String accessKeyId = "yourAccessKeyId";
String accessKeySecret = "yourAccessKeySecret";
// Create an OSSClient instance.
OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);
// Information about a bucket includes the region (Region or Location), creation date (CreationDate), and owner (Owner) of the bucket.
// Specify the name of the bucket. Example: examplebucket.
BucketInfo info = ossClient.getBucketInfo("examplebucket");
// Query the region.
info.getBucket().getLocation();
// Query the creation date of the bucket.
info.getBucket().getCreationDate();
// Query the information about the owner of the bucket.
info.getBucket().getOwner();
// Query the redundancy option for the bucket.
info.getDataRedundancyType();
// Shut down the OSSClient instance.
ossClient.shutdown();
```

For more information about how to query bucket information, see [GetBucketInfo](#).

2.5.6. Manage the ACL of a bucket

A bucket is a container used to store objects in Object Storage Service (OSS). Every object is contained in a bucket. This topic describes how to configure and query the access control list (ACL) of a bucket.

Configure the ACL of a bucket

The following table describes the bucket ACLs.

ACL	Description	Method
Private	Only the owner or authorized users of the bucket have the read and write permissions on objects in the bucket. Other users cannot access the objects in the bucket.	<code>CannedAccessControlList.Private</code>
Public read	Only the owner or authorized users of the bucket have the read and write permissions on objects in the bucket. Other users have only the read permissions on the objects in the bucket. Exercise caution when you use this permission.	<code>CannedAccessControlList.PublicRead</code>
Public read/write	All users have the read and write permissions on the objects in the bucket. Exercise caution when you use this permission.	<code>CannedAccessControlList.PublicReadWrite</code>

The following code provides an example on how to configure the ACL of a bucket:

```

import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.CannedAccessControlList;
public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the name of the bucket. Example: examplebucket.
        String bucketName = "examplebucket";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);

        try {
            // Configure the ACL of the bucket. In this example, the ACL of the examplebucket is set to private.
            ossClient.setBucketAcl(bucketName, CannedAccessControlList.Private);
        } catch (OSSException oe) {
            System.out.println("Caught an OSSException, which means your request made it to OSS, "
                + "but was rejected with an error response for some reason.");
            System.out.println("Error Message:" + oe.getErrorMessage());
            System.out.println("Error Code:" + oe.getErrorCode());
            System.out.println("Request ID:" + oe.getRequestId());
            System.out.println("Host ID:" + oe.getHostId());
        } catch (ClientException ce) {
            System.out.println("Caught an ClientException, which means the client encountered "
                + "a serious internal problem while trying to communicate with OSS, "
                + "such as not being able to access the network.");
            System.out.println("Error Message:" + ce.getMessage());
        } finally {
            if (ossClient != null) {
                ossClient.shutdown();
            }
        }
    }
}

```

Query the ACL of a bucket

The following code provides an example on how to query the ACL of a bucket:

```

import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.AccessControlList;

public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the name of the bucket. Example: examplebucket.
        String bucketName = "examplebucket";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);

        try {
            // Query the ACL of the bucket.
            AccessControlList acl = ossClient.getBucketAcl(bucketName);
            System.out.println(acl.toString());
        } catch (OSSException oe) {
            System.out.println("Caught an OSSException, which means your request made it to OSS, "
                + "but was rejected with an error response for some reason.");
            System.out.println("Error Message:" + oe.getErrorMessage());
            System.out.println("Error Code:" + oe.getErrorCode());
            System.out.println("Request ID:" + oe.getRequestId());
            System.out.println("Host ID:" + oe.getHostId());
        } catch (ClientException ce) {
            System.out.println("Caught an ClientException, which means the client encountered "
                + "a serious internal problem while trying to communicate with OSS, "
                + "such as not being able to access the network.");
            System.out.println("Error Message:" + ce.getMessage());
        } finally {
            if (ossClient != null) {
                ossClient.shutdown();
            }
        }
    }
}

```

References

- For more information about the complete sample code for bucket ACLs, visit [GitHub](#).
- For more information about the API operation that you can call to configure the ACL of a bucket, see [PutBucketAcl](#).

- For more information about the API operation that you can call to query the ACL of a bucket, see [GetBucketAcl](#).

2.5.7. View whether the hierarchical namespace feature is enabled

After the hierarchical namespace feature is enabled for a bucket, you can manage the directory in the bucket, such as creating a directory. This topic describes how to view whether the hierarchical namespace feature is enabled for a bucket.

The following code provides an example on how to view whether the hierarchical namespace feature is enabled for the `examplebucket` bucket.

```
// Set yourEndpoint to the endpoint of the region where the bucket is deployed. For example
// , if your bucket is in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com.
String endpoint = "yourEndpoint";
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log on to OSS because the account has permissions on all API operations. We recommend that you use your RAM user's credentials to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
String accessKeyId = "yourAccessKeyId";
String accessKeySecret = "yourAccessKeySecret";
// Specify the bucket name.
String bucketName = "examplebucket";
// Create an OSSClient instance.
OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);
// Query the bucket information.
BucketInfo info = ossClient.getBucketInfo(bucketName);
// View whether the value of HnsStatus is Enabled. When the value of HnsStatus is Enabled, the hierarchical namespace feature is enabled for the bucket.
System.out.println("HnsStatus:" + info.getBucket().getHnsStatus());
// Shut down the OSSClient instance.
ossClient.shutdown();
```

2.5.8. Delete a bucket

If you no longer use a bucket, delete the bucket to stop unexpected charges.

 **Warning** Deleted buckets cannot be restored. Exercise caution when you perform this operation.

Prerequisites

- All objects in the bucket are deleted.
 - For more information about how to manually delete a small number of objects, see [删除文件](#).
 - To delete a large number of objects, we recommend that you configure lifecycle rules to batch delete the objects. For more information about how to delete a large number of objects, see [Configure lifecycle rules](#).

 Notice

If versioning is enabled for the bucket that you want to delete, make sure that all versions of objects in the bucket are deleted. For more information about how to delete all versions of objects in a bucket, see [Configure versioning](#).

- Parts that are uploaded by multipart upload or resumable upload tasks in the bucket are deleted. For more information about how to delete parts in a bucket, see [Manage parts](#).
- All LiveChannels in the bucket are deleted. For more information about how to delete LiveChannels, see [DeleteLiveChannel](#).

Sample code

The following sample code shows how to delete a bucket named examplebucket:

```

import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify the
        // actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account
        // to access OSS because the account has permissions on all API operations. We recommend that
        // you use a RAM user to call API operations or perform routine O&M. To create a RAM user,
        // log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the name of the bucket. Example: examplebucket.
        String bucketName = "examplebucket";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);

        try {
            // Delete the bucket.
            ossClient.deleteBucket(bucketName);
        } catch (OSSException oe) {
            System.out.println("Caught an OSSException, which means your request made it to
OSS, "
                + "but was rejected with an error response for some reason.");
            System.out.println("Error Message:" + oe.getErrorMessage());
            System.out.println("Error Code:" + oe.getErrorCode());
            System.out.println("Request ID:" + oe.getRequestId());
            System.out.println("Host ID:" + oe.getHostId());
        } catch (ClientException ce) {
            System.out.println("Caught an ClientException, which means the client encountered "
                + "a serious internal problem while trying to communicate with OSS, "
                + "such as not being able to access the network.");
            System.out.println("Error Message:" + ce.getMessage());
        } finally {
            if (ossClient != null) {
                ossClient.shutdown();
            }
        }
    }
}

```

References

- For more information about the complete sample code that is used to delete a bucket, visit [GitHub](#).
- For more information about the API operation that you can call to delete a bucket, see [DeleteBucket](#).

2.5.9. Bucket tagging

Bucket tags allow you to classify and manage buckets. For example, you can list only buckets with specific tags when calling the ListBucket operation.

Usage notes

- Only the bucket owner and the users who are granted the `oss:PutBucketTagging` permission can configure tags for buckets. If other users attempt to configure tags for buckets, 403 Forbidden is returned with the `AccessDenied` error code.
- You can configure up to 20 tags for a bucket.
- The key and value of a tag must be UTF-8-encoded.
- The key of a tag can be up to 64 characters in length and cannot start with `http://`, `https://`, or `Aliyun`. You cannot leave the key of a tag empty.
- The value of a tag can be up to 128 characters in length and can be left empty.

Configure bucket tags

The following code provides an example on how to configure tags for a bucket named examplebucket:


```

import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.SetBucketTaggingRequest;
public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the name of the bucket. Example: examplebucket.
        String bucketName = "examplebucket";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);

        try {
            // Configure tags for the bucket.
            SetBucketTaggingRequest request = new SetBucketTaggingRequest(bucketName);
            // Specify the key and the value of the bucket tag. For example, set the key to owner and the value to John.
            request.setTag("owner", "John");
            request.setTag("location", "hangzhou");
            ossClient.setBucketTagging(request);
        } catch (OSSException oe) {
            System.out.println("Caught an OSSException, which means your request made it to OSS, "
                + "but was rejected with an error response for some reason.");
            System.out.println("Error Message:" + oe.getErrorMessage());
            System.out.println("Error Code:" + oe.getErrorCode());
            System.out.println("Request ID:" + oe.getRequestId());
            System.out.println("Host ID:" + oe.getHostId());
        } catch (ClientException ce) {
            System.out.println("Caught an ClientException, which means the client encountered "
                + "a serious internal problem while trying to communicate with OSS, "
                + "such as not being able to access the network.");
            System.out.println("Error Message:" + ce.getMessage());
        } finally {
            if (ossClient != null) {
                ossClient.shutdown();
            }
        }
    }
}

```

Obtain bucket tags

The following code provides an example on how to obtain the tags of a bucket named `examplebucket`:

```

import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.GenericRequest;
import com.aliyun.oss.model.TagSet;
import java.util.Map;
public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the name of the bucket. Example: examplebucket.
        String bucketName = "examplebucket";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);

        try {
            // Query the tags configured for the bucket.
            TagSet tagSet = ossClient.getBucketTagging(new GenericRequest(bucketName));
            Map<String, String> tags = tagSet.getAllTags();
            for (Map.Entry tag:tags.entrySet()) {
                System.out.println("key:"+tag.getKey()+" value:"+tag.getValue());
            }
        } catch (OSSException oe) {
            System.out.println("Caught an OSSException, which means your request made it to OSS, "
                + "but was rejected with an error response for some reason.");
            System.out.println("Error Message:" + oe.getErrorMessage());
            System.out.println("Error Code:" + oe.getErrorCode());
            System.out.println("Request ID:" + oe.getRequestId());
            System.out.println("Host ID:" + oe.getHostId());
        } catch (ClientException ce) {
            System.out.println("Caught an ClientException, which means the client encountered "
                + "a serious internal problem while trying to communicate with OSS, "
                + "such as not being able to access the network.");
            System.out.println("Error Message:" + ce.getMessage());
        } finally {
            if (ossClient != null) {
                ossClient.shutdown();
            }
        }
    }
}

```

List the buckets that have a specified tag

The following code provides an example on how to list buckets that have a specified tag:

```
import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.*;
public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);

        try {
            // List the buckets that have a specified tag.
            ListBucketsRequest listBucketsRequest = new ListBucketsRequest();
            // Specify the key and the value of the bucket tag. For example, set the key to owner and the value to John.
            listBucketsRequest.setTag("owner", "John");
            BucketList bucketList = ossClient.listBuckets(listBucketsRequest);
            for (Bucket o : bucketList.getBucketList()) {
                System.out.println("list result bucket: " + o.getName());
            }
        } catch (OSSException oe) {
            System.out.println("Caught an OSSException, which means your request made it to OSS, "
                + "but was rejected with an error response for some reason.");
            System.out.println("Error Message:" + oe.getErrorMessage());
            System.out.println("Error Code:" + oe.getErrorCode());
            System.out.println("Request ID:" + oe.getRequestId());
            System.out.println("Host ID:" + oe.getHostId());
        } catch (ClientException ce) {
            System.out.println("Caught an ClientException, which means the client encountered "
                + "a serious internal problem while trying to communicate with OSS, "
                + "such as not being able to access the network.");
            System.out.println("Error Message:" + ce.getMessage());
        } finally {
            if (ossClient != null) {
                ossClient.shutdown();
            }
        }
    }
}
```

Delete the tags of a bucket

The following code provides an example on how to delete the tags of the bucket named `examplebucket`.

 **Note** Before you delete the specified tags of a bucket, you must obtain the tags. Then, you can manually remove the tags and configure new tags for the bucket.

```
import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.*;

public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the name of the bucket. Example: examplebucket.
        String bucketName = "examplebucket";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);

        try {
            // Delete the tags of the bucket.
            ossClient.deleteBucketTagging(new GenericRequest(bucketName));
        } catch (OSSException oe) {
            System.out.println("Caught an OSSException, which means your request made it to OSS, "
                + "but was rejected with an error response for some reason.");
            System.out.println("Error Message:" + oe.getErrorMessage());
            System.out.println("Error Code:" + oe.getErrorCode());
            System.out.println("Request ID:" + oe.getRequestId());
            System.out.println("Host ID:" + oe.getHostId());
        } catch (ClientException ce) {
            System.out.println("Caught an ClientException, which means the client encountered "
                + "a serious internal problem while trying to communicate with OSS, "
                + "such as not being able to access the network.");
            System.out.println("Error Message:" + ce.getMessage());
        } finally {
            if (ossClient != null) {
                ossClient.shutdown();
            }
        }
    }
}
```

References

- For the complete sample code that is used to configure tags for a bucket, visit [GitHub](#).
- For more information about the API operation that you can call to configure tags for a bucket, see [PutBucketTags](#).
- For more information about the API operation that you can call to obtain the tags of a bucket, see

[GetBucketTags](#).

- For more information about the API operation that you can call to delete the tags of a bucket, see [DeleteBucketTags](#).

2.5.10. Bucket inventory

This topic describes how to create an inventory for a bucket and how to query, list, and delete the inventories configured for a bucket.

Create an inventory for a bucket

The following code provides an example on how to create an inventory for a bucket:

```
import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.*;
import java.util.ArrayList;
import java.util.List;
public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the name of the bucket. Example: examplebucket.
        String bucketName = "examplebucket";
        // Specify the name of the bucket in which you want to store the generated inventory lists.
        String destBucketName = "yourDestinationBucketName";
        // Specify the account ID granted by the bucket owner.
        String accountId = "yourDestinationBucketAccountId";
        // Specify the RAM role for which you want to configure inventories. The RAM role must have permissions to read the source bucket for which you want to configure the inventory and permissions to write data to the destination bucket in which you want to store the generated inventory lists.
        String roleArn = "yourDestinationBucketRoleArn";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);

        try {
            // Create an inventory.
            InventoryConfiguration inventoryConfiguration = new InventoryConfiguration();
            // Specify the name of the inventory.
            String inventoryId = "testid";
            inventoryConfiguration.setInventoryId(inventoryId);
            // Specify the object attributes to include in the inventory lists.
            List<String> fields = new ArrayList<String>();
```

```

        fields.add(InventoryOptionalFields.Size);
        fields.add(InventoryOptionalFields.LastModifiedDate);
        fields.add(InventoryOptionalFields.IsMultipartUploaded);
        fields.add(InventoryOptionalFields.StorageClass);
        fields.add(InventoryOptionalFields.ETag);
        fields.add(InventoryOptionalFields.EncryptionStatus);
        inventoryConfiguration.setOptionalFields(fields);
        // Specify whether to generate the inventory list on a daily basis or weekly basis. The following code provides an example on how to generate inventory list on a weekly basis. Weekly indicates that the inventory lists are generated once a week and Daily indicates that the inventory lists are generated once a day.
        inventoryConfiguration.setSchedule(new InventorySchedule().withFrequency(InventoryFrequency.Weekly));
        // Specify that the inventory lists include only the current version of objects. If you set the InventoryIncludedObjectVersions parameter to All, all versions of objects are included in the inventory lists. This configuration takes effect only when you enable versioning for the bucket.
        inventoryConfiguration.setIncludedObjectVersions(InventoryIncludedObjectVersions.Current);
        // Specify whether the inventory is enabled. Valid values: true and false. Set the value to true to enable the inventory. Set the value to false to disable the inventory.

        inventoryConfiguration.setEnabled(true);
        // Specify the rule used to filter the objects to include in the inventory list. The following code provides an example on how to filter the objects by a prefix.
        InventoryFilter inventoryFilter = new InventoryFilter().withPrefix("obj-prefix");
        inventoryConfiguration.setInventoryFilter(inventoryFilter);
        // Configure the destination bucket that you want to store the generated inventory lists.
        InventoryOSSBucketDestination ossInvDest = new InventoryOSSBucketDestination();
        // Specify the prefix of the path in which you want to store the generated inventory lists.
        ossInvDest.setPrefix("destination-prefix");
        // Specify the format of the inventory lists.
        ossInvDest.setFormat(InventoryFormat.CSV);
        // Specify the ID of the Alibaba Cloud account to which the destination bucket belongs.
        ossInvDest.setAccountId(accountId);
        // Specify the role ARN of the destination bucket.
        ossInvDest.setRoleArn(roleArn);
        // Specify the name of the destination bucket.
        ossInvDest.setBucket(destBucketName);
        // The following code provides an example on how to encrypt the inventory lists by using customer master keys (CMKs) hosted in KMS.
        // InventoryEncryption inventoryEncryption = new InventoryEncryption();
        // InventoryServerSideEncryptionKMS serverSideKmsEncryption = new InventoryServerSideEncryptionKMS().withKeyId("test-kms-id");
        // inventoryEncryption.setServerSideKmsEncryption(serverSideKmsEncryption);
        // ossInvDest.setEncryption(inventoryEncryption);
        // The following code provides an example on how to encrypt the inventory list on the OSS server.
        // InventoryEncryption inventoryEncryption = new InventoryEncryption();
        // inventoryEncryption.setServerSideOssEncryption(new InventoryServerSideEncryption

```



```

tionOSS());
    // ossInvDest.setEncryption(inventoryEncryption);
    // Specify the destination for the inventory results.
    InventoryDestination destination = new InventoryDestination();
    destination.setOssBucketDestination(ossInvDest);
    inventoryConfiguration.setDestination(destination);
    // Upload the inventory.
    ossClient.setBucketInventoryConfiguration(bucketName, inventoryConfiguration);
} catch (OSSEException oe) {
    System.out.println("Caught an OSSEException, which means your request made it to
OSS, "
        + "but was rejected with an error response for some reason.");
    System.out.println("Error Message:" + oe.getErrorMessage());
    System.out.println("Error Code:" + oe.getErrorCode());
    System.out.println("Request ID:" + oe.getRequestId());
    System.out.println("Host ID:" + oe.getHostId());
} catch (ClientException ce) {
    System.out.println("Caught an ClientException, which means the client encounter
ed "
        + "a serious internal problem while trying to communicate with OSS, "
        + "such as not being able to access the network.");
    System.out.println("Error Message:" + ce.getMessage());
} finally {
    if (ossClient != null) {
        ossClient.shutdown();
    }
}
}
}

```

Query an inventory configured for a bucket

The following code provides an example on how to query an inventory configured for a bucket:

```

import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSEException;
import com.aliyun.oss.model.*;
import java.util.List;
public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the name of the bucket. Example: examplebucket.
        String bucketName = "examplebucket";
        // Specify the inventory name.

```

```

String inventoryId = "yourInventoryConfigurationId";
// Create an OSSClient instance.
OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret
);
try {
    // Query the information about the inventory.
    GetBucketInventoryConfigurationRequest request = new GetBucketInventoryConfigur
ationRequest(bucketName, inventoryId);
    GetBucketInventoryConfigurationResult getResult = ossClient.getBucketInventoryC
onfiguration(request);
    // Display the information about the inventory.
    InventoryConfiguration config = getResult.getInventoryConfiguration();
    System.out.println("====Inventory configuration====");
    System.out.println("inventoryId:" + config.getInventoryId());
    System.out.println("isEnabled:" + config.isEnabled());
    System.out.println("includedVersions:" + config.getIncludedObjectVersions());
    System.out.println("schedule:" + config.getSchedule().getFrequency());
    if (config.getInventoryFilter().getPrefix() != null) {
        System.out.println("filter, prefix:" + config.getInventoryFilter().getPrefi
x());
    }
    List<String> fields = config.getOptionalFields();
    for (String field : fields) {
        System.out.println("field:" + field);
    }
    System.out.println("===bucket destination config===");
    InventoryOSSBucketDestination destin = config.getDestination().getOssBucketDest
ination();
    System.out.println("format:" + destin.getFormat());
    System.out.println("bucket:" + destin.getBucket());
    System.out.println("prefix:" + destin.getPrefix());
    System.out.println("accountId:" + destin.getAccountId());
    System.out.println("roleArn:" + destin.getRoleArn());
    if (destin.getEncryption() != null) {
        if (destin.getEncryption().getServerSideKmsEncryption() != null) {
            System.out.println("server-side kms encryption, key id:" + destin.getEn
ryption().getServerSideKmsEncryption().getKeyId());
        } else if (destin.getEncryption().getServerSideOssEncryption() != null) {
            System.out.println("server-side oss encryption.");
        }
    }
} catch (OSSException oe) {
    System.out.println("Caught an OSSException, which means your request made it to
OSS, "
        + "but was rejected with an error response for some reason.");
    System.out.println("Error Message:" + oe.getErrorMessage());
    System.out.println("Error Code:" + oe.getErrorCode());
    System.out.println("Request ID:" + oe.getRequestId());
    System.out.println("Host ID:" + oe.getHostId());
} catch (ClientException ce) {
    System.out.println("Caught an ClientException, which means the client encounter
ed "
        + "a serious internal problem while trying to communicate with OSS, "
        + "such as not being able to access the network.");
}

```

```

        System.out.println("Error Message:" + ce.getMessage());
    } finally {
        if (ossClient != null) {
            ossClient.shutdown();
        }
    }
}
}
}

```

List inventories configured for a bucket

 **Note** You can query up to 100 inventories in a single request. If you want to query more than 100 inventories, send multiple requests and use the token returned for each request as a parameter for the next request.

The following code provides an example on how to list inventories configured for a bucket:

```

package com.aliyun.oss.demo;
import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.*;
import java.util.List;

public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the name of the bucket. Example: examplebucket.
        String bucketName = "examplebucket";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);

        try {
            // List the inventories. By default, you can list up to 100 inventories. If more than 100 inventories are configured, the inventories are listed in pages. You can use the continuationToken parameter to list the inventories on the subsequent page.
            String continuationToken = null;
            while (true) {
                ListBucketInventoryConfigurationsRequest listRequest = new ListBucketInventoryConfigurationsRequest(bucketName, continuationToken);
                ListBucketInventoryConfigurationsResult result = ossClient.listBucketInventoryConfigurations(listRequest);
                System.out.println("====List bucket inventory configuration====");
                System.out.println("isTruncated:" + result.isTruncated());
                System.out.println("continuationToken:" + result.getContinuationToken());
            }
        }
    }
}

```

```

        System.out.println("nextContinuationToken:" + result.getNextContinuationToken());

        System.out.println("list size : " + result.getInventoryConfigurationList());
        if (result.getInventoryConfigurationList() != null && !result.getInventoryConfigurationList().isEmpty()) {
            for (InventoryConfiguration config : result.getInventoryConfigurationList()) {
                System.out.println("===Inventory configuration===");
                System.out.println("inventoryId:" + config.getInventoryId());
                System.out.println("isEnabled:" + config.isEnabled());
                System.out.println("includedVersions:" + config.getIncludedObjectVersions());

                System.out.println("schedule:" + config.getSchedule().getFrequency());

                if (config.getInventoryFilter().getPrefix() != null) {
                    System.out.println("filter, prefix:" + config.getInventoryFilter().getPrefix());
                }
                List<String> fields = config.getOptionalFields();
                for (String field : fields) {
                    System.out.println("field:" + field);
                }
                System.out.println("===bucket destination config===");
                InventoryOSSBucketDestination destin = config.getDestination().getOSSBucketDestination();
                System.out.println("format:" + destin.getFormat());
                System.out.println("bucket:" + destin.getBucket());
                System.out.println("prefix:" + destin.getPrefix());
                System.out.println("accountId:" + destin.getAccountId());
                System.out.println("roleArn:" + destin.getRoleArn());
                if (destin.getEncryption() != null) {
                    if (destin.getEncryption().getServerSideKmsEncryption() != null) {
                        System.out.println("server-side kms encryption key id:" + destin.getEncryption().getServerSideKmsEncryption().getKeyId());
                    } else if (destin.getEncryption().getServerSideOssEncryption() != null) {
                        System.out.println("server-side oss encryption.");
                    }
                }
            }
            if (result.isTruncated()) {
                continuationToken = result.getNextContinuationToken();
            } else {
                break;
            }
        }
    } catch (OSSEException oe) {
        System.out.println("Caught an OSSEException, which means your request made it to OSS, "
            + "but was rejected with an error response for some reason.");
        System.out.println("Error Message:" + oe.getErrorMessage());
        System.out.println("Error Code:" + oe.getErrorCode());
    }
}

```

```
        System.out.println("Request ID:" + oe.getRequestId());
        System.out.println("Host ID:" + oe.getHostId());
    } catch (ClientException ce) {
        System.out.println("Caught an ClientException, which means the client encounter  
ed "
            + "a serious internal problem while trying to communicate with OSS, "
            + "such as not being able to access the network.");
        System.out.println("Error Message:" + ce.getMessage());
    } finally {
        if (ossClient != null) {
            ossClient.shutdown();
        }
    }
}
```

Delete an inventory configured for a bucket

The following code provides an example on how to delete an inventory configured for a bucket:

```
import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the name of the bucket. Example: examplebucket.
        String bucketName = "examplebucket";
        // Specify the name of the bucket for which you want to delete the inventory.
        String inventoryId = "yourInventoryConfigurationId";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);
    };
    try {
        // Delete the inventory.
        ossClient.deleteBucketInventoryConfiguration(bucketName, inventoryId);
    } catch (OSSException oe) {
        System.out.println("Caught an OSSException, which means your request made it to OSS, "
            + "but was rejected with an error response for some reason.");
        System.out.println("Error Message:" + oe.getErrorMessage());
        System.out.println("Error Code:" + oe.getErrorCode());
        System.out.println("Request ID:" + oe.getRequestId());
        System.out.println("Host ID:" + oe.getHostId());
    } catch (ClientException ce) {
        System.out.println("Caught an ClientException, which means the client encountered "
            + "a serious internal problem while trying to communicate with OSS, "
            + "such as not being able to access the network.");
        System.out.println("Error Message:" + ce.getMessage());
    } finally {
        if (ossClient != null) {
            ossClient.shutdown();
        }
    }
}
```

References

- For more information about the complete sample code for bucket inventories, visit [GitHub](#).
- For more information about the API operation that is called to create an inventory, see [PutBucketInventory](#).

- For more information about the API operation that is called to query inventories configured for a bucket, see [GetBucketInventory](#).
- For more information about the API operation that is called to list multiple inventories configured for a bucket, see [ListBucketInventory](#).
- For more information about the API operation that is called to delete inventories configured for a bucket, see [DeleteBucketInventory](#).

2.5.11. Bucket policy

This topic describes how to configure, query, and delete policies for a bucket.

Configure a policy for a bucket

The following sample code shows how to configure a policy for a bucket:

```

import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify the
        // endpoint based on your business requirements.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account
        // to access OSS, because the account has permissions on all API operations. We recommend
        // that you use a RAM user to call API operations or perform routine O&M. To create a RAM
        // user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the name of the bucket. Example: examplebucket.
        String bucketName = "examplebucket";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);

        try {
            // Specify policyText.
            String policyText = "{\"Statement\": [{\"Effect\": \"Allow\", \"Action\": [\"oss:GetObject\", \"oss:ListObjects\"], \"Resource\": [\"acs:oss:*:*:/user1/*\"]}], \"Version\": \"1\"}";
            // Configure the policy.
            ossClient.setBucketPolicy(bucketName, policyText);
        } catch (OSSException oe) {
            System.out.println("Caught an OSSException, which means your request made it to
OSS, "
                + "but was rejected with an error response for some reason.");
            System.out.println("Error Message:" + oe.getErrorMessage());
            System.out.println("Error Code:" + oe.getErrorCode());
            System.out.println("Request ID:" + oe.getRequestId());
            System.out.println("Host ID:" + oe.getHostId());
        } catch (ClientException ce) {
            System.out.println("Caught an ClientException, which means the client encountered "
                + "a serious internal problem while trying to communicate with OSS, "
                + "such as not being able to access the network.");
            System.out.println("Error Message:" + ce.getMessage());
        } finally {
            if (ossClient != null) {
                ossClient.shutdown();
            }
        }
    }
}

```

Query the policies configured for a bucket

The following sample code shows how to query the policies configured for a bucket:


```

import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.GetBucketPolicyResult;
public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify the
        // endpoint based on your business requirements.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account
        // to access OSS, because the account has permissions on all API operations. We recommend
        // that you use a RAM user to call API operations or perform routine O&M. To create a RAM
        // user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the name of the bucket. Example: examplebucket.
        String bucketName = "examplebucket";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);

        try {
            // Query the bucket policies.
            GetBucketPolicyResult result = ossClient.getBucketPolicy(bucketName);
            System.out.println("policyText:" + result.getPolicyText());
        } catch (OSSException oe) {
            System.out.println("Caught an OSSException, which means your request made it to
OSS, "
                + "but was rejected with an error response for some reason.");
            System.out.println("Error Message:" + oe.getErrorMessage());
            System.out.println("Error Code:" + oe.getErrorCode());
            System.out.println("Request ID:" + oe.getRequestId());
            System.out.println("Host ID:" + oe.getHostId());
        } catch (ClientException ce) {
            System.out.println("Caught an ClientException, which means the client encountered "
                + "a serious internal problem while trying to communicate with OSS, "
                + "such as not being able to access the network.");
            System.out.println("Error Message:" + ce.getMessage());
        } finally {
            if (ossClient != null) {
                ossClient.shutdown();
            }
        }
    }
}

```

Delete the policies configured for a bucket

The following sample code shows how to delete the policies configured for a bucket:

```
import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify the
        // endpoint based on your business requirements.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account
        // to access OSS, because the account has permissions on all API operations. We recommend
        // that you use a RAM user to call API operations or perform routine O&M. To create a RAM
        // user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the name of the bucket. Example: examplebucket.
        String bucketName = "examplebucket";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);

        try {
            // Delete the policies configured for the bucket.
            ossClient.deleteBucketPolicy(bucketName);
        } catch (OSSException oe) {
            System.out.println("Caught an OSSException, which means your request made it to
OSS, "
                + "but was rejected with an error response for some reason.");
            System.out.println("Error Message:" + oe.getErrorMessage());
            System.out.println("Error Code:" + oe.getErrorCode());
            System.out.println("Request ID:" + oe.getRequestId());
            System.out.println("Host ID:" + oe.getHostId());
        } catch (ClientException ce) {
            System.out.println("Caught an ClientException, which means the client encountered "
                + "a serious internal problem while trying to communicate with OSS, "
                + "such as not being able to access the network.");
            System.out.println("Error Message:" + ce.getMessage());
        } finally {
            if (ossClient != null) {
                ossClient.shutdown();
            }
        }
    }
}
```

References

- For more information about the complete sample code for bucket policies, visit [GitHub](#).
- For more information about the API operation that you can call to configure a bucket policy for a bucket, see [PutBucketPolicy](#).
- For more information about the API operation that you can call to query the policies configured for a bucket, see [GetBucketPolicy](#).

- For more information about the API operation that you can call to delete the policies configured for a bucket, see [DeleteBucketPolicy](#).

2.5.12. Retention policies

You can configure a time-based retention policy for a bucket. The retention policy has a retention period that ranges from 1 day to 70 years. This topic describes how to create, query, and lock a retention policy.

Background information

Object Storage Service (OSS) supports the Write Once Read Many (WORM) feature, which prevents an object from being deleted or overwritten for a specified period of time.

If a retention policy is not locked within 24 hours after it is created, the retention policy becomes invalid. If the retention policy configured for a bucket is locked, you can upload objects to or read objects from the bucket. However, you cannot delete objects in the bucket or the retention policy within the retention period of the policy. The retention period of the policy can be extended but cannot be shortened. For more information about retention policies, see [Retention policies](#).

Create a retention policy

The following code provides an example on how to create a retention policy:

```
import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.InitiateBucketWormRequest;
import com.aliyun.oss.model.InitiateBucketWormResult;

public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the name of the bucket. Example: examplebucket.
        String bucketName = "examplebucket";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);

        try {
            // Create an InitiateBucketWormRequest object.
            InitiateBucketWormRequest initiateBucketWormRequest = new InitiateBucketWormRequest(bucketName);
            // Set the retention period to one day.
            initiateBucketWormRequest.setRetentionPeriodInDays(1);
            // Create a retention policy.
            InitiateBucketWormResult initiateBucketWormResult = ossClient.initiateBucketWorm(initiateBucketWormRequest);
        }
```

```
        // Query the ID of the retention policy.
        String wormId = initiateBucketWormResult.getWormId();
        System.out.println(wormId);
    } catch (OSSEException oe) {
        System.out.println("Caught an OSSEException, which means your request made it to
OSS, "
            + "but was rejected with an error response for some reason.");
        System.out.println("Error Message:" + oe.getErrorMessage());
        System.out.println("Error Code:" + oe.getErrorCode());
        System.out.println("Request ID:" + oe.getRequestId());
        System.out.println("Host ID:" + oe.getHostId());
    } catch (ClientException ce) {
        System.out.println("Caught an ClientException, which means the client encounter
ed "
            + "a serious internal problem while trying to communicate with OSS, "
            + "such as not being able to access the network.");
        System.out.println("Error Message:" + ce.getMessage());
    } finally {
        if (ossClient != null) {
            ossClient.shutdown();
        }
    }
}
```

Delete an unlocked retention policy

The following code provides an example on how to delete an unlocked retention policy:

```

import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the name of the bucket. Example: examplebucket.
        String bucketName = "examplebucket";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);

        try {
            // Delete the unlocked retention policy.
            ossClient.abortBucketWorm(bucketName);
        } catch (OSSException oe) {
            System.out.println("Caught an OSSException, which means your request made it to OSS, "
                + "but was rejected with an error response for some reason.");
            System.out.println("Error Message:" + oe.getErrorMessage());
            System.out.println("Error Code:" + oe.getErrorCode());
            System.out.println("Request ID:" + oe.getRequestId());
            System.out.println("Host ID:" + oe.getHostId());
        } catch (ClientException ce) {
            System.out.println("Caught an ClientException, which means the client encountered "
                + "a serious internal problem while trying to communicate with OSS, "
                + "such as not being able to access the network.");
            System.out.println("Error Message:" + ce.getMessage());
        } finally {
            if (ossClient != null) {
                ossClient.shutdown();
            }
        }
    }
}

```

Lock a retention policy

The following code provides an example on how to lock a retention policy:

```

import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the name of the bucket. Example: examplebucket.
        String bucketName = "examplebucket";
        String wormId = "yourWormId";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);

        try {
            // Lock the retention policy.
            ossClient.completeBucketWorm(bucketName, wormId);
        } catch (OSSException oe) {
            System.out.println("Caught an OSSException, which means your request made it to OSS, "
                + "but was rejected with an error response for some reason.");
            System.out.println("Error Message:" + oe.getErrorMessage());
            System.out.println("Error Code:" + oe.getErrorCode());
            System.out.println("Request ID:" + oe.getRequestId());
            System.out.println("Host ID:" + oe.getHostId());
        } catch (ClientException ce) {
            System.out.println("Caught an ClientException, which means the client encountered "
                + "a serious internal problem while trying to communicate with OSS, "
                + "such as not being able to access the network.");
            System.out.println("Error Message:" + ce.getMessage());
        } finally {
            if (ossClient != null) {
                ossClient.shutdown();
            }
        }
    }
}

```

Query a retention policy

The following code provides an example on how to query a retention policy:

```

import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;

```

```

import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.GetBucketWormResult;
public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the name of the bucket. Example: examplebucket.
        String bucketName = "examplebucket";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);

        try {
            // Query the retention policy.
            GetBucketWormResult getBucketWormResult = ossClient.getBucketWorm(bucketName);
            // Query the ID of the retention policy.
            System.out.println(getBucketWormResult.getWormId());
            // Query the status of the retention policy. InProgress indicates that the retention policy is not locked, while Locked indicates that the retention policy is locked.
            System.out.println(getBucketWormResult.getWormState());
            // Query the retention period of objects.
            System.out.println(getBucketWormResult.getRetentionPeriodInDays());
            // Query the time when the retention policy was created.
            System.out.println(getBucketWormResult.getCreationDate());
        } catch (OSSException oe) {
            System.out.println("Caught an OSSException, which means your request made it to OSS, "
                + "but was rejected with an error response for some reason.");
            System.out.println("Error Message:" + oe.getErrorMessage());
            System.out.println("Error Code:" + oe.getErrorCode());
            System.out.println("Request ID:" + oe.getRequestId());
            System.out.println("Host ID:" + oe.getHostId());
        } catch (ClientException ce) {
            System.out.println("Caught an ClientException, which means the client encountered "
                + "a serious internal problem while trying to communicate with OSS, "
                + "such as not being able to access the network.");
            System.out.println("Error Message:" + ce.getMessage());
        } finally {
            if (ossClient != null) {
                ossClient.shutdown();
            }
        }
    }
}

```

Extend the retention period of objects

The following code provides an example on how to extend the retention period of objects in a bucket for which a retention policy is locked:

```
import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the name of the bucket. Example: examplebucket.
        String bucketName = "examplebucket";
        String wormId = "yourWormId";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);

        try {
            // Extend the retention period of objects in the bucket for which a retention policy is locked.
            ossClient.extendBucketWorm(bucketName, wormId, 2);
        } catch (OSSException oe) {
            System.out.println("Caught an OSSException, which means your request made it to OSS, "
                + "but was rejected with an error response for some reason.");
            System.out.println("Error Message:" + oe.getErrorMessage());
            System.out.println("Error Code:" + oe.getErrorCode());
            System.out.println("Request ID:" + oe.getRequestId());
            System.out.println("Host ID:" + oe.getHostId());
        } catch (ClientException ce) {
            System.out.println("Caught an ClientException, which means the client encountered "
                + "a serious internal problem while trying to communicate with OSS, "
                + "such as not being able to access the network.");
            System.out.println("Error Message:" + ce.getMessage());
        } finally {
            if (ossClient != null) {
                ossClient.shutdown();
            }
        }
    }
}
```

References

- For more information about the complete sample code for a retention policy, visit [GitHub](#).

- For more information about the API operation that you can call to create a retention policy, see [InitiateBucketWorm](#).
- For more information about the API operation that you can call to delete an unlocked retention policy, see [AbortBucketWorm](#).
- For more information about the API operation that you can call to lock a retention policy, see [CompleteBucketWorm](#).
- For more information about the API operation that you can call to query a retention policy, see [GetBucketWorm](#).
- For more information about the API operation that you can call to extend the retention period of objects in a bucket whose retention policy is locked, see [ExtendBucketWorm](#).

2.5.13. Lifecycle

Object Storage Service (OSS) allows you to configure lifecycle rules to delete expired objects and parts or convert the storage class of expired objects to Infrequent Access (IA) or Archive to reduce your storage costs. This topic describes how to manage lifecycle rules for buckets.

Configure lifecycle rules

- Configure lifecycle rules that match objects by tag and prefix

The following sample code provides an example on how to configure a lifecycle rule that matches objects by prefix and tag for the bucket named examplebucket:

```
import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.common.utils.DateUtil;
import com.aliyun.oss.model.LifecycleRule;
import com.aliyun.oss.model.SetBucketLifecycleRequest;
import com.aliyun.oss.model.StorageClass;
import java.util.ArrayList;
import java.util.HashMap;
import java.util.List;
import java.util.Map;

public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify
        // the actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API oper-
        // ations. Using these credentials to perform operations in OSS is a high-risk operation. We
        // recommend that you use a RAM user to call API operations or perform routine O&M. To creat-
        // e a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the name of the bucket. Example: examplebucket.
        String bucketName = "examplebucket";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecr-
        et);

        try {
```

```

        // Create a request by using SetBucketLifecycleRequest.
        SetBucketLifecycleRequest request = new SetBucketLifecycleRequest(bucketName)
;
        // Specify the ID of the lifecycle rule.
        String ruleId0 = "rule0";
        // Specify the prefix that you want the lifecycle rule to match.
        String matchPrefix0 = "A0/";
        // Specify the tag that you want the lifecycle rule to match.
        Map<String, String> matchTags0 = new HashMap<String, String>();
        // Specify the key and value of the tag. In the example, the key is set to ow
ner and the value is set to John.
        matchTags0.put("owner", "John");
        String ruleId1 = "rule1";
        String matchPrefix1 = "A1/";
        Map<String, String> matchTags1 = new HashMap<String, String>();
        matchTags1.put("type", "document");
        String ruleId2 = "rule2";
        String matchPrefix2 = "A2/";
        String ruleId3 = "rule3";
        String matchPrefix3 = "A3/";
        String ruleId4 = "rule4";
        String matchPrefix4 = "A4/";
        String ruleId5 = "rule5";
        String matchPrefix5 = "A5/";
        String ruleId6 = "rule6";
        String matchPrefix6 = "A6/";
        // Set the expiration date to three days after the last modification time.
        LifecycleRule rule = new LifecycleRule(ruleId0, matchPrefix0, LifecycleRule.R
uleStatus.Enabled, 3);
        rule.setTags(matchTags0);
        request.AddLifecycleRule(rule);
        // Specify that the objects that are created before the specific date expire.

        rule = new LifecycleRule(ruleId1, matchPrefix1, LifecycleRule.RuleStatus.Enab
led);
        rule.setCreatedBeforeDate(DateUtil.parseIso8601Date("2022-10-12T00:00:00.000Z
"));
        rule.setTags(matchTags1);
        request.AddLifecycleRule(rule);
        // Set the expiration date of parts to three days later.
        rule = new LifecycleRule(ruleId2, matchPrefix2, LifecycleRule.RuleStatus.Enab
led);
        LifecycleRule.AbortMultipartUpload abortMultipartUpload = new LifecycleRule.A
bortMultipartUpload();
        abortMultipartUpload.setExpirationDays(3);
        rule.setAbortMultipartUpload(abortMultipartUpload);
        request.AddLifecycleRule(rule);
        // Specify that parts that are created before the specific date expire.
        rule = new LifecycleRule(ruleId3, matchPrefix3, LifecycleRule.RuleStatus.Enab
led);
        abortMultipartUpload = new LifecycleRule.AbortMultipartUpload();
        abortMultipartUpload.setCreatedBeforeDate(DateUtil.parseIso8601Date("2022-10-
12T00:00:00.000Z"));
        rule.setAbortMultipartUpload(abortMultipartUpload);

```

```

        request.AddLifecycleRule(rule);
        // Specify that the storage class of objects is converted to IA 10 days after
        they are last modified, and to Archive 30 days after they are last modified.
        rule = new LifecycleRule(ruleId4, matchPrefix4, LifecycleRule.RuleStatus.Enabled);

        List<LifecycleRule.StorageTransition> storageTransitions = new ArrayList<LifecycleRule.StorageTransition>();
        LifecycleRule.StorageTransition storageTransition = new LifecycleRule.StorageTransition();
        storageTransition.setStorageClass(StorageClass.IA);
        storageTransition.setExpirationDays(10);
        storageTransitions.add(storageTransition);
        storageTransition = new LifecycleRule.StorageTransition();
        storageTransition.setStorageClass(StorageClass.Archive);
        storageTransition.setExpirationDays(30);
        storageTransitions.add(storageTransition);
        rule.setStorageTransition(storageTransitions);
        request.AddLifecycleRule(rule);
        // Specify that the storage class of objects whose last modification time is
        before October 12, 2022 is converted to Archive.
        rule = new LifecycleRule(ruleId5, matchPrefix5, LifecycleRule.RuleStatus.Enabled);

        storageTransitions = new ArrayList<LifecycleRule.StorageTransition>();
        storageTransition = new LifecycleRule.StorageTransition();
        storageTransition.setCreatedBeforeDate(DateUtil.parseIso8601Date("2022-10-12T00:00:00.000Z"));
        storageTransition.setStorageClass(StorageClass.Archive);
        storageTransitions.add(storageTransition);
        rule.setStorageTransition(storageTransitions);
        request.AddLifecycleRule(rule);
        // Specify that rule6 is configured for versioned buckets.
        rule = new LifecycleRule(ruleId6, matchPrefix6, LifecycleRule.RuleStatus.Enabled);

        // Specify that the storage class of objects is converted to Archive 365 days
        after the objects are last modified.
        storageTransitions = new ArrayList<LifecycleRule.StorageTransition>();
        storageTransition = new LifecycleRule.StorageTransition();
        storageTransition.setStorageClass(StorageClass.Archive);
        storageTransition.setExpirationDays(365);
        storageTransitions.add(storageTransition);
        rule.setStorageTransition(storageTransitions);
        // Specify that delete markers are automatically removed when they expire.
        rule.setExpiredDeleteMarker(true);
        // Configure the lifecycle rule to convert the previous versions of objects to
        o the IA storage class 10 days after the objects are last modified.
        LifecycleRule.NoncurrentVersionStorageTransition noncurrentVersionStorageTransition =
            new LifecycleRule.NoncurrentVersionStorageTransition().withNoncurrent
            Days(10).withStorageClass(StorageClass.IA);
        // Configure the lifecycle rule to convert the previous versions of objects to
        o the Archive storage class 20 days after the objects are last modified.
        LifecycleRule.NoncurrentVersionStorageTransition noncurrentVersionStorageTransition2 =
            new LifecycleRule.NoncurrentVersionStorageTransition().withNoncurrent
            Days(20).withStorageClass(StorageClass.Archive);

```

```

        days(30).withStorageClass(StorageClass.Archive);
        // Specify that objects are deleted 30 days after they become previous versions.
        LifecycleRule.NoncurrentVersionExpiration noncurrentVersionExpiration = new LifecycleRule.NoncurrentVersionExpiration().withNoncurrentDays(30);
        List<LifecycleRule.NoncurrentVersionStorageTransition> noncurrentVersionStorageTransitions = new ArrayList<LifecycleRule.NoncurrentVersionStorageTransition>();
        noncurrentVersionStorageTransitions.add(noncurrentVersionStorageTransition2);
        rule.setStorageTransition(storageTransitions);
        rule.setNoncurrentVersionExpiration(noncurrentVersionExpiration);
        rule.setNoncurrentVersionStorageTransitions(noncurrentVersionStorageTransitions);

        request.AddLifecycleRule(rule);
        // Initiate a request to configure lifecycle rules.
        ossClient.setBucketLifecycle(request);
        // Query the lifecycle rules that are configured for the bucket.
        List<LifecycleRule> listRules = ossClient.getBucketLifecycle(bucketName);
        for(LifecycleRule rules : listRules){
            System.out.println("ruleId="+rules.getId()+" , matchPrefix="+rules.getPrefix());
        }
    } catch (OSSException oe) {
        System.out.println("Caught an OSSException, which means your request made it to OSS, "
            + "but was rejected with an error response for some reason.");
        System.out.println("Error Message:" + oe.getErrorMessage());
        System.out.println("Error Code:" + oe.getErrorCode());
        System.out.println("Request ID:" + oe.getRequestId());
        System.out.println("Host ID:" + oe.getHostId());
    } catch (ClientException ce) {
        System.out.println("Caught an ClientException, which means the client encountered "
            + "a serious internal problem while trying to communicate with OSS, "
            + "such as not being able to access the network.");
        System.out.println("Error Message:" + ce.getMessage());
    } finally {
        if (ossClient != null) {
            ossClient.shutdown();
        }
    }
}
}

```

- Configure a lifecycle rule for all objects in a bucket

The following code provides an example on how to configure a lifecycle rule for all objects in the bucket named examplebucket:

```

import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.LifecycleRule;
import com.aliyun.oss.model.SetBucketLifecycleRequest;
import java.util.List;
public class Demo {

```

```

public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify
        // the actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API oper-
        // ations. Using these credentials to perform operations in OSS is a high-risk operation. We
        // recommend that you use a RAM user to call API operations or perform routine O&M. To creat-
        // e a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the name of the bucket. Example: examplebucket.
        String bucketName = "examplebucket";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecr-
        et);

        try {
            // Create a request by using SetBucketLifecycleRequest.
            SetBucketLifecycleRequest request = new SetBucketLifecycleRequest(bucketName)
            ;

            // Specify the ID of the lifecycle rule.
            String ruleId0 = "rule0";
            // If the prefix that is used to match objects is left empty, the lifecycle r-
            // ule takes effect on all objects in the bucket.
            String matchPrefix0 = null;
            // Set the expiration date to three days after the last modification time.
            LifecycleRule rule = new LifecycleRule(ruleId0, matchPrefix0, LifecycleRule.R-
            uleStatus.Enabled, 3);
            request.AddLifecycleRule(rule);
            // Initiate a request to configure lifecycle rules.
            ossClient.setBucketLifecycle(request);
            // Query the lifecycle rules that are configured for the bucket.
            List<LifecycleRule> listRules = ossClient.getBucketLifecycle(bucketName);
            for(LifecycleRule rules : listRules){
                System.out.println("ruleId="+rules.getId()+"", matchPrefix="+rules.getPref-
                ix()+"", expirationDays="+rules.getExpirationDays());
            }
        } catch (OSSEException oe) {
            System.out.println("Caught an OSSEException, which means your request made it
            to OSS, "

                + "but was rejected with an error response for some reason.");
            System.out.println("Error Message:" + oe.getErrorMessage());
            System.out.println("Error Code:" + oe.getErrorCode());
            System.out.println("Request ID:" + oe.getRequestId());
            System.out.println("Host ID:" + oe.getHostId());
        } catch (ClientException ce) {
            System.out.println("Caught an ClientException, which means the client encount-
            ered "

                + "a serious internal problem while trying to communicate with OSS, "
                + "such as not being able to access the network.");
            System.out.println("Error Message:" + ce.getMessage());
        } finally {
            if (ossClient != null) {
                ossClient.shutdown();
            }
        }
    }
}

```

```

    }
}
}

```

Query lifecycle rules

The following code provides an example on how to query the lifecycle rules configured for the bucket named `examplebucket`:

```

import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.LifecycleRule;
import java.util.List;
public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify the actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the name of the bucket. Example: examplebucket.
        String bucketName = "examplebucket";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);

        try {
            // Query the lifecycle rule.
            List<LifecycleRule> rules = ossClient.getBucketLifecycle(bucketName);
            // Query the lifecycle rules that are configured for the bucket.
            for (LifecycleRule r : rules) {
                System.out.println("=====");
                // Query the rule ID.
                System.out.println("rule id: " + r.getId());
                // View the status of the rule.
                System.out.println("rule status: " + r.getStatus());
                // View the prefix configured in the lifecycle rule.
                System.out.println("rule prefix: " + r.getPrefix());
                // View the tag configured in the lifecycle rule.
                if (r.hasTags()) {
                    System.out.println("rule tagging: " + r.getTags().toString());
                }
                // View the expiration days configured in the lifecycle rule.
                if (r.hasExpirationDays()) {
                    System.out.println("rule expiration days: " + r.getExpirationDays());
                }
                // View the expiration date configured in the lifecycle rule.
                if (r.hasCreatedBeforeDate()) {
                    System.out.println("rule expiration create before days: " + r.getCreate

```

```

dBeforeDate());
    }
    // View the rule for expired parts.
    if(r.hasAbortMultipartUpload()) {
        if(r.getAbortMultipartUpload().hasExpirationDays()) {
            System.out.println("rule abort uppart days: " + r.getAbortMultipart
Upload().getExpirationDays());
        }
        if (r.getAbortMultipartUpload().hasCreatedBeforeDate()) {
            System.out.println("rule abort uppart create before date: " + r.get
AbortMultipartUpload().getCreatedBeforeDate());
        }
    }
    // View the rule of storage class conversion.
    if (r.getStorageTransition().size() > 0) {
        for (LifecycleRule.StorageTransition transition : r.getStorageTransitio
n()) {
            if (transition.hasExpirationDays()) {
                System.out.println("rule storage trans days: " + transition.get
ExpirationDays() +
                    " trans storage class: " + transition.getStorageClass()
);
            }
            if (transition.hasCreatedBeforeDate()) {
                System.out.println("rule storage trans before create date: " +
transition.getCreatedBeforeDate());
            }
        }
    }
    // View the lifecycle rule to check whether expired delete markers are auto
matically removed.
    if (r.hasExpiredDeleteMarker()) {
        System.out.println("rule expired delete marker: " + r.getExpiredDeleteM
arker());
    }
    // View the rule to convert the storage classes of objects of previous vers
ions.
    if (r.hasNoncurrentVersionStorageTransitions()) {
        for (LifecycleRule.NoncurrentVersionStorageTransition transition : r.ge
tNoncurrentVersionStorageTransitions()) {
            System.out.println("rule noncurrent versions trans days:" + transit
ion.getNoncurrentDays() +
                " trans storage class: " + transition.getStorageClass());
        }
    }
    // View the expiration rule for objects of previous versions.
    if (r.hasNoncurrentVersionExpiration()) {
        System.out.println("rule noncurrent versions expiration days:" + r.getN
oncurrentVersionExpiration().getNoncurrentDays());
    }
}
} catch (OSSEException oe) {
    System.out.println("Caught an OSSEException, which means your request made it to
OSS, "

```

```
        + "but was rejected with an error response for some reason.");
        System.out.println("Error Message:" + oe.getErrorMessage());
        System.out.println("Error Code:" + oe.getErrorCode());
        System.out.println("Request ID:" + oe.getRequestId());
        System.out.println("Host ID:" + oe.getHostId());
    } catch (ClientException ce) {
        System.out.println("Caught an ClientException, which means the client encounter  
ed "
            + "a serious internal problem while trying to communicate with OSS, "
            + "such as not being able to access the network.");
        System.out.println("Error Message:" + ce.getMessage());
    } finally {
        if (ossClient != null) {
            ossClient.shutdown();
        }
    }
}
```

Delete lifecycle rules

The following code provides an example on how to delete lifecycle rules configured for the bucket named `examplebucket`:


```
import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify the actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the name of the bucket. Example: examplebucket.
        String bucketName = "examplebucket";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);

        try {
            ossClient.deleteBucketLifecycle(bucketName);
        } catch (OSSException oe) {
            System.out.println("Caught an OSSException, which means your request made it to OSS, "
                + "but was rejected with an error response for some reason.");
            System.out.println("Error Message:" + oe.getErrorMessage());
            System.out.println("Error Code:" + oe.getErrorCode());
            System.out.println("Request ID:" + oe.getRequestId());
            System.out.println("Host ID:" + oe.getHostId());
        } catch (ClientException ce) {
            System.out.println("Caught an ClientException, which means the client encountered "
                + "a serious internal problem while trying to communicate with OSS, "
                + "such as not being able to access the network.");
            System.out.println("Error Message:" + ce.getMessage());
        } finally {
            if (ossClient != null) {
                ossClient.shutdown();
            }
        }
    }
}
```

References

- For more information about the complete sample code for lifecycle rules, visit [GitHub](#).
- For more information about the API operation that you can call to configure lifecycle rules, see [Put Bucket Lifecycle](#).
- For more information about the API operation that you can call to query lifecycle rules, see [Get Bucket Lifecycle](#).
- For more information about the API operation that you can call to delete lifecycle rules, see [Delete Bucket Lifecycle](#).

[DeleteBucketLifecycle](#).

2.5.14. Enable pay-by-requester

When pay-by-requester is enabled for a bucket in Object Storage Service (OSS), the requester is charged the requests and traffic fees instead of the bucket owner. The bucket owner is charged only the storage fees. You can enable pay-by-requester for a bucket to share data in the bucket without paying for the request and traffic fees incurred by the access to your bucket.

Enable pay-by-requester

The following code provides an example on how to enable pay-by-requester for a bucket:

```

import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.*;
public class Demo {
    public static void main(String[] args) {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify your actual endpoint. For more information about the endpoints of other regions, see Regions and endpoints.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the name of the bucket. Example: examplebucket.
        String bucketName = "examplebucket";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);

        try {
            // Enable pay-by-requester for the bucket.
            Payer payer = Payer.Requester;
            ossClient.setBucketRequestPayment(bucketName, payer);
        } catch (OSSException oe) {
            System.out.println("Caught an OSSException, which means your request made it to OSS, "
                + "but was rejected with an error response for some reason.");
            System.out.println("Error Message:" + oe.getErrorMessage());
            System.out.println("Error Code:" + oe.getErrorCode());
            System.out.println("Request ID:" + oe.getRequestId());
            System.out.println("Host ID:" + oe.getHostId());
        } catch (Throwable ce) {
            System.out.println("Caught an ClientException, which means the client encountered "
                + "a serious internal problem while trying to communicate with OSS, "
                + "such as not being able to access the network.");
            System.out.println("Error Message:" + ce.getMessage());
        } finally {
            // Shut down the OSSClient instance.
            if (ossClient != null) {
                ossClient.shutdown();
            }
        }
    }
}

```

Queries the pay-by-requester configurations of a bucket

The following code provides an example on how to query the pay-by-requester configurations of a bucket:

```

import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.*;
public class Demo {
    public static void main(String[] args) {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the name of the bucket. Example: examplebucket.
        String bucketName = "examplebucket";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);

        try {
            // Query the pay-by-requester configurations of the bucket.
            GetBucketRequestPaymentResult result = ossClient.getBucketRequestPayment(bucketName);
            System.out.println("Payer:" + result.getPayer());
        } catch (OSSException oe) {
            System.out.println("Caught an OSSException, which means your request made it to OSS, "
                + "but was rejected with an error response for some reason.");
            System.out.println("Error Message:" + oe.getErrorMessage());
            System.out.println("Error Code:" + oe.getErrorCode());
            System.out.println("Request ID:" + oe.getRequestId());
            System.out.println("Host ID:" + oe.getHostId());
        } catch (Throwable ce) {
            System.out.println("Caught an ClientException, which means the client encountered "
                + "a serious internal problem while trying to communicate with OSS, "
                + "such as not being able to access the network.");
            System.out.println("Error Message:" + ce.getMessage());
        } finally {
            // Shut down the OSSClient instance.
            if (ossClient != null) {
                ossClient.shutdown();
            }
        }
    }
}

```

Specify that third parties are charged for access to objects

If you specify that third parties are charged for access to objects in a bucket, requesters must include the `x-oss-request-payer:requester` header in the HTTP requests to perform operations on your objects. If this header is not included, an error is returned.

The following code provides an example on how to specify that third parties are charged when they access objects by calling the PutObject, GetObject, and DeleteObject operations. You can use the method to specify that third parties are charged when they perform read and write operations on objects by calling other API operations in the similar way.

The following code provides an example on how to specify that third parties are charged when they access objects:

```
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.*;
import java.io.ByteArrayInputStream;
public class Demo {
    public static void main(String[] args) {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify your
        // actual endpoint. For more information about the endpoints of other regions, see Regions
        // and endpoints.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account
        // to access OSS because the account has permissions on all API operations. We recommend that
        // you use a RAM user to call API operations or perform routine O&M. To create a RAM user,
        // log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the name of the bucket. Example: examplebucket.
        String bucketName = "examplebucket";
        // Specify the full path of the object. Example: exampledir/exampleobject.txt. The
        // full path of the object cannot contain the bucket name.
        String objectName = "exampledir/exampleobject.txt";
        Payer payer = Payer.Requester;
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);

        try {
            // Specify the payer when a third party calls the PutObject operation to access
            // objects.
            String content = "hello";
            PutObjectRequest putObjectRequest = new PutObjectRequest(bucketName, objectName,
                new ByteArrayInputStream(content.getBytes()));
            putObjectRequest.setRequestPayer(payer);
            ossClient.putObject(putObjectRequest);
            // Specify the payer when a third party calls the GetObject operation to access
            // objects.
            GetObjectRequest getObjectRequest = new GetObjectRequest(bucketName, objectName);
            getObjectRequest.setRequestPayer(payer);
            OSSObject ossObject = ossClient.getObject(getObjectRequest);
            ossObject.close();
            // Specify the payer when a third party calls the DeleteObject operation to access
            // objects.
            GenericRequest genericRequest = new GenericRequest(bucketName, objectName);
            genericRequest.setRequestPayer(payer);
            ossClient.deleteObject(genericRequest);
        } catch (OSSException oe) {
```

```

    } catch (OSSException oe) {
        System.out.println("Caught an OSSException, which means your request made it to
OSS, "
            + "but was rejected with an error response for some reason.");
        System.out.println("Error Message:" + oe.getErrorMessage());
        System.out.println("Error Code:" + oe.getErrorCode());
        System.out.println("Request ID:" + oe.getRequestId());
        System.out.println("Host ID:" + oe.getHostId());
    } catch (Throwable ce) {
        System.out.println("Caught an ClientException, which means the client encounter
ed "
            + "a serious internal problem while trying to communicate with OSS, "
            + "such as not being able to access the network.");
        System.out.println("Error Message:" + ce.getMessage());
    } finally {
        // Shut down the OSSClient instance.
        if (ossClient != null) {
            ossClient.shutdown();
        }
    }
}
}
}

```

References

- For more information about the complete sample code that is used to enable pay-by-requester for a bucket, visit [GitHub](#).
- For more information about the API operation that you can call to enable pay-by-requester for a bucket, see [PutBucketRequestPayment](#).
- For more information about the API operation that you can call to query the pay-by-requester configurations of a bucket, see [GetBucketRequestPayment](#).

2.5.15. Hotlink protection

You can configure hotlink protection for your Object Storage Service (OSS) bucket to prevent your resources in the bucket from unauthorized access.

Background information

To prevent your data stored in Object Storage Service (OSS) from unauthorized access, you can configure a Referrer whitelist for your bucket by specifying the following parameters:

- Referrer Whitelist: specifies that only specified domain names are allowed to access your resources.
- Allow Empty Referrer: determines whether requests that contain an empty Referrer field are allowed. If you specify that an empty Referrer field is not allowed, only HTTP and HTTPS requests that contain an allowed Referrer field can access your OSS resources.

For more information about Referrers, see [Hotlink protection](#).

Configure a Referrer whitelist for a bucket

The following code provides an example on how to configure a Referrer whitelist for a bucket:

```

import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;

```

```

import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.BucketReferer;
import java.util.ArrayList;
import java.util.List;

public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the bucket name. Example: examplebucket.
        String bucketName = "examplebucket";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);

        try {
            List<String> refererList = new ArrayList<String>();
            // Add Referers to the Referer whitelist. You can use asterisks (*) and question marks (?) as wildcards in Referers.
            refererList.add("http://www.aliyun.com");
            refererList.add("http://www.*.com");
            refererList.add("http://www?.aliyuncs.com");
            // Configure the Referer whitelist of the bucket. If this parameter is set to true, empty Referer fields are allowed. If this parameter is set to false, empty Referer fields are not allowed.
            BucketReferer br = new BucketReferer(true, refererList);
            ossClient.setBucketReferer(bucketName, br);
        } catch (OSSException oe) {
            System.out.println("Caught an OSSException, which means your request made it to OSS, "
                + "but was rejected with an error response for some reason.");
            System.out.println("Error Message:" + oe.getErrorMessage());
            System.out.println("Error Code:" + oe.getErrorCode());
            System.out.println("Request ID:" + oe.getRequestId());
            System.out.println("Host ID:" + oe.getHostId());
        } catch (ClientException ce) {
            System.out.println("Caught an ClientException, which means the client encountered "
                + "a serious internal problem while trying to communicate with OSS, "
                + "such as not being able to access the network.");
            System.out.println("Error Message:" + ce.getMessage());
        } finally {
            if (ossClient != null) {
                ossClient.shutdown();
            }
        }
    }
}

```

```
,
```

Query a Referer whitelist of a bucket

The following code provides an example on how to query a Referer whitelist of a bucket:


```

import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.BucketReferer;
import java.util.List;
public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the bucket name. Example: examplebucket.
        String bucketName = "examplebucket";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);

        try {
            // Query the Referer whitelist of the bucket.
            BucketReferer br = ossClient.getBucketReferer(bucketName);
            List<String> refererList = br.getRefererList();
            for (String referer : refererList) {
                System.out.println(referer);
            }
        } catch (OSSException oe) {
            System.out.println("Caught an OSSException, which means your request made it to OSS, "
                + "but was rejected with an error response for some reason.");
            System.out.println("Error Message:" + oe.getErrorMessage());
            System.out.println("Error Code:" + oe.getErrorCode());
            System.out.println("Request ID:" + oe.getRequestId());
            System.out.println("Host ID:" + oe.getHostId());
        } catch (ClientException ce) {
            System.out.println("Caught an ClientException, which means the client encountered "
                + "a serious internal problem while trying to communicate with OSS, "
                + "such as not being able to access the network.");
            System.out.println("Error Message:" + ce.getMessage());
        } finally {
            if (ossClient != null) {
                ossClient.shutdown();
            }
        }
    }
}

```

Clear a Referer whitelist of a bucket

The following code provides an example on how to clear a Referer whitelist of a bucket:

```
import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.BucketReferer;
public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the bucket name. Example: examplebucket.
        String bucketName = "examplebucket";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);

        try {
            // The Referer whitelist of a bucket cannot be directly cleared. You must create a Referer whitelist that allows empty Referer fields to overwrite the existing Referer whitelist.
            BucketReferer br = new BucketReferer();
            ossClient.setBucketReferer(bucketName, br);
        } catch (OSSException oe) {
            System.out.println("Caught an OSSException, which means your request made it to OSS, "
                + "but was rejected with an error response for some reason.");
            System.out.println("Error Message:" + oe.getErrorMessage());
            System.out.println("Error Code:" + oe.getErrorCode());
            System.out.println("Request ID:" + oe.getRequestId());
            System.out.println("Host ID:" + oe.getHostId());
        } catch (ClientException ce) {
            System.out.println("Caught an ClientException, which means the client encountered "
                + "a serious internal problem while trying to communicate with OSS, "
                + "such as not being able to access the network.");
            System.out.println("Error Message:" + ce.getMessage());
        } finally {
            if (ossClient != null) {
                ossClient.shutdown();
            }
        }
    }
}
```

References

- For the complete sample code that is used to configure a Referer whitelist, visit [GitHub](#).
- For more information about the API operation that you can call to configure a Referer whitelist for a bucket, see [PutBucketReferer](#).
- For more information about the API operation that you can call to query a Referer whitelist of a bucket, see [GetBucketReferer](#).

2.5.16. Logging

A large number of logs are generated when Object Storage Service (OSS) resources are accessed. After you enable and configure logging for a bucket, OSS generates logs every hour in accordance with predefined naming conventions and then stores the logs as objects in a specified bucket.

Enable logging for a bucket

The following code provides an example on how to enable logging for a bucket:

```
import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.SetBucketLoggingRequest;

public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in Object Storage Service (OSS) is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the name of the bucket for which you want to enable logging. Example: examplebucket.
        String bucketName = "examplebucket";
        // Specify the destination bucket used to store log objects. The source bucket and the destination bucket can be the same bucket or different buckets.
        String targetBucketName = "yourTargetBucketName";
        // Set the directory in which log objects are stored to log/. If you specify this directory, log objects are stored in the specified directory of the destination bucket. If you do not specify this directory, log objects are stored in the root directory of the destination bucket.
        String targetPrefix = "log/";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);

        try {
            SetBucketLoggingRequest request = new SetBucketLoggingRequest(bucketName);
            request.setTargetBucket(targetBucketName);
            request.setTargetPrefix(targetPrefix);
            ossClient.setBucketLogging(request);
        } catch (OSSException oe) {
            System.out.println("Caught an OSSException, which means your request made it to OSS. "
                + oe.getMessage());
        }
    }
}
```

```
OSS, "
        + "but was rejected with an error response for some reason.");
    System.out.println("Error Message:" + oe.getErrorMessage());
    System.out.println("Error Code:" + oe.getErrorCode());
    System.out.println("Request ID:" + oe.getRequestId());
    System.out.println("Host ID:" + oe.getHostId());
} catch (ClientException ce) {
    System.out.println("Caught an ClientException, which means the client encounter
ed "
        + "a serious internal problem while trying to communicate with OSS, "
        + "such as not being able to access the network.");
    System.out.println("Error Message:" + ce.getMessage());
} finally {
    if (ossClient != null) {
        ossClient.shutdown();
    }
}
}
```

View the logging configurations of a bucket

The following code provides an example on how to view the logging configurations of a bucket:

```

import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.BucketLoggingResult;
public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the bucket name. Example: examplebucket.
        String bucketName = "examplebucket";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);
        try {
            BucketLoggingResult result = ossClient.getBucketLogging(bucketName);
            System.out.println(result.getTargetBucket());
            System.out.println(result.getTargetPrefix());
        } catch (OSSException oe) {
            System.out.println("Caught an OSSException, which means your request made it to OSS, "
                + "but was rejected with an error response for some reason.");
            System.out.println("Error Message:" + oe.getErrorMessage());
            System.out.println("Error Code:" + oe.getErrorCode());
            System.out.println("Request ID:" + oe.getRequestId());
            System.out.println("Host ID:" + oe.getHostId());
        } catch (ClientException ce) {
            System.out.println("Caught an ClientException, which means the client encountered "
                + "a serious internal problem while trying to communicate with OSS, "
                + "such as not being able to access the network.");
            System.out.println("Error Message:" + ce.getMessage());
        } finally {
            if (ossClient != null) {
                ossClient.shutdown();
            }
        }
    }
}

```

Disable logging for a bucket

The following code provides an example on how to disable logging for a bucket:

```

import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.SetBucketLoggingRequest;
public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the bucket name. Example: examplebucket.
        String bucketName = "examplebucket";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);
        try {
            SetBucketLoggingRequest request = new SetBucketLoggingRequest(bucketName);
            request.setTargetBucket(null);
            request.setTargetPrefix(null);
            ossClient.setBucketLogging(request);
        } catch (OSSException oe) {
            System.out.println("Caught an OSSException, which means your request made it to OSS, "
                + "but was rejected with an error response for some reason.");
            System.out.println("Error Message:" + oe.getErrorMessage());
            System.out.println("Error Code:" + oe.getErrorCode());
            System.out.println("Request ID:" + oe.getRequestId());
            System.out.println("Host ID:" + oe.getHostId());
        } catch (ClientException ce) {
            System.out.println("Caught an ClientException, which means the client encountered "
                + "a serious internal problem while trying to communicate with OSS, "
                + "such as not being able to access the network.");
            System.out.println("Error Message:" + ce.getMessage());
        } finally {
            if (ossClient != null) {
                ossClient.shutdown();
            }
        }
    }
}

```

References

- For the complete sample code that is used to configure logging, visit [GitHub](#).
- For more information about the API operation that you can call to enable logging for a bucket, see

[PutBucketLogging](#).

- For more information about the API operation that you can call to view the logging configurations of a bucket, see [GetBucketLogging](#).
- For more information about the API operation that you can call to disable logging for a bucket, see [DeleteBucketLogging](#).

2.5.17. Static website hosting (mirroring-based back-to-origin)

You can enable static website hosting for buckets and configure mirroring-based back-to-origin rules. After you host a static website on a bucket, you can access the bucket to visit the website. You are automatically redirected to the specified index page or error page. After mirroring-based back-to-origin rules are configured and take effect, you can use mirroring-based back-to-origin to seamlessly migrate data to Object Storage Service (OSS).

Static website hosting

- The following code provides an example on how to configure static website hosting:

```

import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.SetBucketWebsiteRequest;
public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify
        // your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API oper
        // ations. Using these credentials to perform operations in OSS is a high-risk operation. We
        // recommend that you use a RAM user to call API operations or perform routine O&M. To creat
        // e a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the bucket name. Example: examplebucket.
        String bucketName = "examplebucket";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecr
et);
        try {
            // Specify the bucket name.
            SetBucketWebsiteRequest request = new SetBucketWebsiteRequest(bucketName);
            // Specify the default homepage of the static website hosted on the bucket.
            request.setIndexDocument("index.html");
            // Specify the default 404 page of the static website hosted on the bucket.
            request.setErrorDocument("error.html");
            ossClient.setBucketWebsite(request);
        } catch (OSSException oe) {
            System.out.println("Caught an OSSException, which means your request made it
to OSS, "
                + "but was rejected with an error response for some reason.");
            System.out.println("Error Message:" + oe.getErrorMessage());
            System.out.println("Error Code:" + oe.getErrorCode());
            System.out.println("Request ID:" + oe.getRequestId());
            System.out.println("Host ID:" + oe.getHostId());
        } catch (ClientException ce) {
            System.out.println("Caught an ClientException, which means the client encount
ered "
                + "a serious internal problem while trying to communicate with OSS, "
                + "such as not being able to access the network.");
            System.out.println("Error Message:" + ce.getMessage());
        } finally {
            if (ossClient != null) {
                ossClient.shutdown();
            }
        }
    }
}

```

Configure static website hosting

- The following code provides an example on how to query static website hosting configurations:


```

import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.BucketWebsiteResult;
public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify
        // your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API oper-
        // ations. Using these credentials to perform operations in OSS is a high-risk operation. We
        // recommend that you use a RAM user to call API operations or perform routine O&M. To creat-
        // e a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the bucket name. Example: examplebucket.
        String bucketName = "examplebucket";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecr-
        et);
        try {
            // Specify the bucket name.
            BucketWebsiteResult result = ossClient.getBucketWebsite(bucketName);
            // Query the default homepage and default 404 page of the static website host-
            // ed on the bucket.
            System.out.println(result.getIndexDocument());
            System.out.println(result.getErrorDocument());
        } catch (OSSException oe) {
            System.out.println("Caught an OSSException, which means your request made it
            to OSS, "
                + "but was rejected with an error response for some reason.");
            System.out.println("Error Message:" + oe.getErrorMessage());
            System.out.println("Error Code:" + oe.getErrorCode());
            System.out.println("Request ID:" + oe.getRequestId());
            System.out.println("Host ID:" + oe.getHostId());
        } catch (ClientException ce) {
            System.out.println("Caught an ClientException, which means the client encount-
            ered "
                + "a serious internal problem while trying to communicate with OSS, "
                + "such as not being able to access the network.");
            System.out.println("Error Message:" + ce.getMessage());
        } finally {
            if (ossClient != null) {
                ossClient.shutdown();
            }
        }
    }
}

```

Query static website hosting configurations

- The following code provides an example on how to delete static website hosting configurations:

```

import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify
        // your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API oper-
        // ations. Using these credentials to perform operations in OSS is a high-risk operation. We
        // recommend that you use a RAM user to call API operations or perform routine O&M. To creat-
        // e a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the bucket name. Example: examplebucket.
        String bucketName = "examplebucket";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySec-
        et);
        try {
            // Specify the bucket name.
            ossClient.deleteBucketWebsite(bucketName);
        } catch (OSSException oe) {
            System.out.println("Caught an OSSException, which means your request made it
            to OSS, "
                + "but was rejected with an error response for some reason.");
            System.out.println("Error Message:" + oe.getErrorMessage());
            System.out.println("Error Code:" + oe.getErrorCode());
            System.out.println("Request ID:" + oe.getRequestId());
            System.out.println("Host ID:" + oe.getHostId());
        } catch (ClientException ce) {
            System.out.println("Caught an ClientException, which means the client encount-
            ered "
                + "a serious internal problem while trying to communicate with OSS, "
                + "such as not being able to access the network.");
            System.out.println("Error Message:" + ce.getMessage());
        } finally {
            if (ossClient != null) {
                ossClient.shutdown();
            }
        }
    }
}

```

Delete static website hosting configurations

Mirroring-based back-to-origin

Mirroring-based back-to-origin is used to seamlessly migrate data to OSS. For example, you can migrate services from a self-managed origin or from another cloud service to OSS without service interruption. You can use mirroring-based back-to-origin rules during migration to obtain the data that is not migrated to OSS. This ensures service continuity.

- If a requester attempts to access an object in the specified bucket and the object does not exist, you can specify the URL of the object in the origin and back-to-origin conditions to allow the requester to obtain the object from the origin. For example, a bucket named `examplebucket` resides in the China (Hangzhou) region. When a requester attempts to access an object in the `examplefolder` directory of the root directory of the bucket but the object does not exist, the requester is redirected to `https://www.example.com/` to access the required object that is stored in the `examplefolder` directory of the origin.

The following code provides an example on how to configure mirroring-based back-to-origin rules in the preceding scenario:

```
import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.RoutingRule;
import com.aliyun.oss.model.SetBucketWebsiteRequest;
import java.util.ArrayList;
import java.util.HashMap;
import java.util.List;
import java.util.Map;
import java.util.Map;

public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify
        // your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API oper-
        // ations. Using these credentials to perform operations in OSS is a high-risk operation. We
        // recommend that you use a RAM user to call API operations or perform routine O&M. To creat-
        // e a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the bucket name. Example: examplebucket.
        String bucketName = "examplebucket";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecr-
        et);

        try {
            SetBucketWebsiteRequest request = new SetBucketWebsiteRequest(bucketName);
            // Specify the operation to perform if the default homepage is specified, the
            // name of the accessed object does not end with a forward slash (/), and the object does no-
            // t exist.
            //request.setSubDirType(null);
            // Specify whether to redirect the requester to the default homepage in the s-
            // ubdirectory when the subdirectory is accessed.
            //request.setSupportSubDir(false);
            List<RoutingRule> routingRules = new ArrayList<RoutingRule>();
            RoutingRule rule = new RoutingRule();
            rule.setNumber(1);
            // Specify the prefix of object names. Only the objects whose names contain t-
            // he specified prefix match the rule.
            rule.getCondition().setKeyPrefixEquals("examplebucket");
            // Specify the HTTP status code. The rule is matched only when the specified
            // object is accessed and HTTP status code 404 is returned.
            rule.getCondition().setHttpErrorCodeReturnedEquals(404);
```

```

        rule.getCondition().setHttpErrorCodeReturnedEquals(404);
        // Specify the redirect type.
        rule.getRedirect().setRedirectType(RoutingRule.RedirectType.Mirror);
        // Specify the origin URL for the mirroring-based back-to-origin rule. Example: https://www.example.com/.
        rule.getRedirect().setMirrorURL("<yourMirrorURL>");
        //rule.getRedirect().setMirrorRole("AliyunOSSMirrorDefaultRole");
        // Specify whether to include parameters of the original request in the redirect request when the system uses the redirection rule or the mirroring-based back-to-origin rule.

        rule.getRedirect().setPassQueryString(true);
        // This parameter is used in the same manner as the PassQueryString parameter and is assigned a higher priority level than the PassQueryString parameter. This parameter takes effect only when RedirectType is set to Mirror.
        rule.getRedirect().setMirrorPassQueryString(true);
        // Specify the HTTP status code in the response. This parameter takes effect only when RedirectType is set to External or AliCDN.
        //rule.getRedirect().setHttpRedirectCode(302);
        // Specify the domain name that is used for redirection, which must comply with the naming conventions for domains.
        //rule.getRedirect().setHostName("oss.aliyuncs.com");
        // Specify the protocol that is used for redirection. This parameter takes effect only when RedirectType is set to External or AliCDN.
        //rule.getRedirect().setProtocol(RoutingRule.Protocol.Https);
        // Specify the string that is used to replace the object name when the request is redirected. This parameter can be set to a variable.
        //rule.getRedirect().setReplaceKeyWith("${key}.jpg");
        // If this parameter is set to true, the prefix of the object name is replaced with the value specified by ReplaceKeyPrefixWith.
        rule.getRedirect().setEnableReplacePrefix(true);
        // Specify the string that is used to replace the prefix of the object name when the request is redirected.
        rule.getRedirect().setReplaceKeyPrefixWith("examplebucket");
        // Specify whether to check the MD5 hash of the response body that is returned by the origin. This parameter takes effect only when RedirectType is set to Mirror.
        rule.getRedirect().setMirrorCheckMd5(true);
        RoutingRule.MirrorHeaders mirrorHeaders = new RoutingRule.MirrorHeaders();
        // Specify whether to pass through all request headers to the origin. This parameter takes effect only when RedirectType is set to Mirror.
        mirrorHeaders.setPassAll(false);
        List passes = new ArrayList<String>();
        passes.add("cache-control");
        // Specify the headers to pass through to the origin. This parameter takes effect only when RedirectType is set to Mirror.
        mirrorHeaders.setPass(passes);
        List removes = new ArrayList<String>();
        removes.add("content-type");
        // Specify the headers that are not allowed to pass through to the origin. This parameter takes effect only when RedirectType is set to Mirror.
        mirrorHeaders.setRemove(removes);
        List sets = new ArrayList<Map<String, String>>();
        Map header1 = new HashMap<String, String>();
        header1.put("Key", "key1");
        header1.put("Value", "value1");
        Map header2 = new HashMap<String, String>();

```

```

        header2.put("Key", "key2");
        header2.put("Value", "value2");
        sets.add(header1);
        sets.add(header2);
        // Specify the headers that are passed to the origin. The specified headers are
        // passed to the origin regardless of whether they are included in the request.
        mirrorHeaders.setSet(sets);
        // Specify the headers included in the request when you use mirroring-based back-
        // to-origin. This parameter takes effect only when RedirectType is set to Mirror.
        rule.getRedirect().setMirrorHeaders(mirrorHeaders);
        routingRules.add(rule);
        request.setRoutingRules(routingRules);
        ossClient.setBucketWebsite(request);
    } catch (OSSException oe) {
        System.out.println("Caught an OSSException, which means your request made it
        to OSS, "
            + "but was rejected with an error response for some reason.");
        System.out.println("Error Message:" + oe.getErrorMessage());
        System.out.println("Error Code:" + oe.getErrorCode());
        System.out.println("Request ID:" + oe.getRequestId());
        System.out.println("Host ID:" + oe.getHostId());
    } catch (ClientException ce) {
        System.out.println("Caught an ClientException, which means the client encount
        ered "
            + "a serious internal problem while trying to communicate with OSS, "
            + "such as not being able to access the network.");
        System.out.println("Error Message:" + ce.getMessage());
    } finally {
        if (ossClient != null) {
            ossClient.shutdown();
        }
    }
}
}
}

```

Configure mirroring-based back-to-origin

- The following code provides an example on how to query mirroring-based back-to-origin configurations:

```

import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.BucketWebsiteResult;
public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify
        // your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API oper
        // ations. Using these credentials to perform operations in OSS is a high-risk operation. We
        // recommend that you use a RAM user to call API operations or perform routine O&M. To creat
        // e a RAM user, log on to the RAM console.
    }
}

```

```

        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the bucket name. Example: examplebucket.
        String bucketName = "examplebucket";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);

        try {
            BucketWebsiteResult result = ossClient.getBucketWebsite(bucketName);
            result.getSubDirType();
            // Query the sequence number that is used to match and run redirection rules
            // or mirroring-based back-to-origin rules.
            System.out.println(result.getRoutingRules().get(0).getNumber());
            // Query the prefix that is used for rule matching.
            System.out.println(result.getRoutingRules().get(0).getCondition().getKeyPrefixEquals());
            // Query the HTTP status code.
            System.out.println(result.getRoutingRules().get(0).getCondition().getHttpErrorReturnedEquals());
            // Query the suffix that is used for rule matching.
            System.out.println(result.getRoutingRules().get(0).getCondition().getKeySuffixEquals());
            // Query the redirect type.
            System.out.println(result.getRoutingRules().get(0).getRedirect().getRedirectType());
            // Query the parameters in the request.
            System.out.println(result.getRoutingRules().get(0).getRedirect().getMirrorPassQueryString());
            // Query the origin URL for mirroring-based back-to-origin.
            System.out.println(result.getRoutingRules().get(0).getRedirect().getMirrorURL());
            // Query the HTTP status code in the response.
            System.out.println(result.getRoutingRules().get(0).getRedirect().getHttpRedirectCode());
            // Query the headers that can be passed through.
            System.out.println(result.getRoutingRules().get(0).getRedirect().getMirrorHeaders().getPass().get(0));
            // Query the headers that cannot be passed through.
            System.out.println(result.getRoutingRules().get(0).getRedirect().getMirrorHeaders().getRemove().get(0));
            // Query the protocol that is used for redirection.
            System.out.println(result.getRoutingRules().get(0).getRedirect().getProtocol());
            // Query the domain name to which the request is redirected.
            System.out.println(result.getRoutingRules().get(0).getRedirect().getHostName());
            // Specify the string that is used to replace the prefix of the object name when the request is redirected. If the prefix of an object is empty, this string precedes the object name.
            System.out.println(result.getRoutingRules().get(0).getRedirect().getReplaceKeyPrefixWith());
            // Specify the string that is used to replace the object name when the request is redirected. This parameter can be set to a variable.
            System.out.println(result.getRoutingRules().get(0).getRedirect().getReplaceKeyWith());
        } catch (OSSException oe) {
            System.out.println("An exception occurred: " + oe.getMessage());
            oe.printStackTrace();
        } catch (IOException ioe) {
            System.out.println("An IOException occurred: " + ioe.getMessage());
            ioe.printStackTrace();
        }
    }
}

```

```
    yWith());  
        // Query the HTTP status code in the response.  
        System.out.println(result.getRoutingRules().get(0).getRedirect().getHttpRedirectCode());  
    } catch (OSSEException oe) {  
        System.out.println("Caught an OSSEException, which means your request made it  
to OSS, "  
            + "but was rejected with an error response for some reason.");  
        System.out.println("Error Message:" + oe.getErrorMessage());  
        System.out.println("Error Code:" + oe.getErrorCode());  
        System.out.println("Request ID:" + oe.getRequestId());  
        System.out.println("Host ID:" + oe.getHostId());  
    } catch (ClientException ce) {  
        System.out.println("Caught an ClientException, which means the client encount  
ered "  
            + "a serious internal problem while trying to communicate with OSS, "  
            + "such as not being able to access the network.");  
        System.out.println("Error Message:" + ce.getMessage());  
    } finally {  
        if (ossClient != null) {  
            ossClient.shutdown();  
        }  
    }  
}
```

Query mirroring-based back-to-origin configurations

- The following code provides an example on how to delete mirroring-based back-to-origin configurations:

```

import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify
        // your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API oper-
        // ations. Using these credentials to perform operations in OSS is a high-risk operation. We
        // recommend that you use a RAM user to call API operations or perform routine O&M. To creat-
        // e a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the bucket name. Example: examplebucket.
        String bucketName = "examplebucket";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecr-
        et);
        try {
            // Specify the bucket name.
            ossClient.deleteBucketWebsite(bucketName);
        } catch (OSSException oe) {
            System.out.println("Caught an OSSException, which means your request made it
            to OSS, "
                + "but was rejected with an error response for some reason.");
            System.out.println("Error Message:" + oe.getErrorMessage());
            System.out.println("Error Code:" + oe.getErrorCode());
            System.out.println("Request ID:" + oe.getRequestId());
            System.out.println("Host ID:" + oe.getHostId());
        } catch (ClientException ce) {
            System.out.println("Caught an ClientException, which means the client encount-
            ered "
                + "a serious internal problem while trying to communicate with OSS, "
                + "such as not being able to access the network.");
            System.out.println("Error Message:" + ce.getMessage());
        } finally {
            if (ossClient != null) {
                ossClient.shutdown();
            }
        }
    }
}

```

Delete mirroring-based back-to-origin configurations

References

- For the complete sample code that is used to configure static website hosting and mirroring-based back-to-origin, visit [GitHub](#).
- For more information about the API operation that you can call to configure static website hosting or mirroring-based back-to-origin, see [PutBucketWebsite](#).

- For more information about the API operation that you can call to query static website hosting or mirroring-based back-to-origin configurations, see [GetBucketWebsite](#).
- For more information about the API operation that you can call to delete static website hosting or mirroring-based back-to-origin configurations, see [DeleteBucketWebsite](#).

2.5.18. Cross-region replication

Cross-region replication (CRR) can automatically replicate objects across buckets in different Object Storage Service (OSS) regions in asynchronous mode. CRR can also synchronize operations, for example, create, overwrite, and delete operations on objects, from a source bucket to a destination bucket. This feature can meet your requirements for geo-disaster recovery and data replication.

Enable CRR

The following code provides an example on how to enable CRR:

```
// Specify the endpoint of the region in which the source bucket is located. The endpoint of the China (Hangzhou) region is used in this example. Specify the endpoint based on your business requirements.
String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
String accessKeyId = "yourAccessKeyId";
String accessKeySecret = "yourAccessKeySecret";
// Specify the bucket name. For more information about the naming conventions for buckets, see Bucket naming conventions.
String bucketName = "yourBucketName";
// Create an OSSClient instance.
OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);
AddBucketReplicationRequest request = new AddBucketReplicationRequest(bucketName);
// Specify the destination bucket to which the data is replicated.
request.setTargetBucketName("yourTargetBucketName");
// Specify the endpoint of the region in which the destination bucket is located.
request.setTargetBucketLocation("https://oss-cn-beijing.aliyuncs.com");
// Specify whether to synchronize historical data. By default, the historical data is synchronized. In this example, this parameter is set to False, which indicates that the historical data is not synchronized.
request.setEnableHistoricalObjectReplication(false);
// Specify the name of the role that you authorize OSS to use to replicate data. If you want to use SSE-KMS to encrypt the objects that are replicated to the destination bucket, you must configure this parameter.
//request.setSyncRole("yourRole");
// Specify whether to replicate objects encrypted by using SSE-KMS.
//request.setSseKmsEncryptedObjectsStatus("Enabled");
// Specify the customer master key (CMK) ID used in SSE-KMS. If Status is set to Enabled, you must specify this parameter.
//request.setReplicaKmsKeyID("3542abdd-5821-4fb5-a425-90adca***");
//List prefixes = new ArrayList();
//prefixes.add("image/");
//prefixes.add("video");
//prefixes.add("a");
//prefixes.add("A");
// Specify the prefix that is used to determine the object that you want to replicate. After you specify the prefix, only objects whose names contain the prefix are replicated to the destination bucket.
//request.setObjectPrefixList(prefixes);
//List actions = new ArrayList();
//actions.add(AddBucketReplicationRequest.ReplicationAction.ALL);
// Specify the operations that can be synchronized to the destination bucket. By default, the value is ALL, which indicates that all operations on the source bucket are synchronized to the destination bucket.
//request.setReplicationActionList(actions);
ossClient.addBucketReplication(request);
// Shut down the OSSClient instance.
ossClient.shutdown();
```

Query CRR configurations

The following code provides an example on how to query CRR configurations:

```
// In this example, the endpoint of the China (Hangzhou) region is used. Specify the endpoint based on your business requirements.
String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
String accessKeyId = "yourAccessKeyId";
String accessKeySecret = "yourAccessKeySecret";
// Specify the bucket name.
String bucketName = "yourBucketName";
// Create an OSSClient instance.
OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);
// Query CRR configurations.
List<ReplicationRule> rules = ossClient.getBucketReplication(bucketName);
for (ReplicationRule rule : rules) {
    System.out.println(rule.getReplicationRuleID());
    System.out.println(rule.getTargetBucketLocation());
    System.out.println(rule.getTargetBucketName());
}
// Shut down the OSSClient instance.
ossClient.shutdown();
```

Query the regions in which CRR can be performed

The following code provides an example on how to query the regions in which CRR can be performed:

```
// In this example, the endpoint of the China (Hangzhou) region is used. Specify the endpoint based on your business requirements.
String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
String accessKeyId = "yourAccessKeyId";
String accessKeySecret = "yourAccessKeySecret";
// Specify the bucket name.
String bucketName = "yourBucketName";
// Create an OSSClient instance.
OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);
// Query the regions in which CRR can be performed.
List<String> locations = ossClient.getBucketReplicationLocation(bucketName);
for (String loc : locations) {
    System.out.println(loc);
}
// Shut down the OSSClient instance.
ossClient.shutdown();
```

Query the progress of a CRR task

You can query the progress of historical data synchronization and the progress of real-time data synchronization.

- Percent signs (%) indicate the progress of historical data synchronization. This only works for buckets for which historical data synchronization has been enabled.
- Time points (the point of the latest time the data is written) indicate the progress of real-time data synchronization. If a time point is displayed, the data created before this time point has been synchronized.

The following code provides an example on how to query the progress of a CRR task:

```
// In this example, the endpoint of the China (Hangzhou) region is used. Specify the endpoint based on your business requirements.
String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
String accessKeyId = "yourAccessKeyId";
String accessKeySecret = "yourAccessKeySecret";
// Specify the bucket name.
String bucketName = "yourBucketName";
// Create an OSSClient instance.
OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);
BucketReplicationProgress process = ossClient.getBucketReplicationProgress(bucketName, "yourRuleId");
System.out.println(process.getReplicationRuleID());
System.out.println(process.isEnableHistoricalObjectReplication());
// View the progress of historical data synchronization.
System.out.println(process.getHistoricalObjectProgress());
// View the progress of real-time data synchronization.
System.out.println(process.getNewObjectProgress());
// Shut down the OSSClient instance.
ossClient.shutdown();
```

Disable CRR

The following code provides an example on how to disable CRR:

```
// In this example, the endpoint of the China (Hangzhou) region is used. Specify the endpoint based on your business requirements.
String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
String accessKeyId = "yourAccessKeyId";
String accessKeySecret = "yourAccessKeySecret";
// Specify the bucket name.
String bucketName = "yourBucketName";
// Create an OSSClient instance.
OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);
// Disable CRR. After CRR is disabled, the files that are synchronized to the destination bucket still exist. However, all changes to the files in the source bucket are no longer synchronized to the destination bucket.
ossClient.deleteBucketReplication(bucketName, "yourRuleId");
// Shut down the OSSClient instance.
ossClient.shutdown();
```

References

- For more information about the API operation that you can call to enable CRR, see [PutBucketReplication](#).
- For more information about the API operation that you can call to query CRR configurations, see [GetBucketReplication](#).
- For more information about the API operation that you can call to query the regions in which CRR can be performed, see [GetBucketReplicationLocation](#).
- For more information about the API operation that you can call to query the progress of a CRR task, see [GetBucketReplicationProgress](#).
- For more information about the API operation that you can call to disable CRR, see [DeleteBucketReplication](#).

2.5.19. CORS

Cross-origin resource sharing (CORS) is a standard cross-origin solution provided by HTML5 to allow web application servers to control cross-origin access and secure cross-origin data transmission.

Configure CORS rules

The following code provides an example on how to configure CORS rules for a specific bucket:

```
import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.SetBucketCORSRequest;
import java.util.ArrayList;
public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify your actual endpoint.
```

```

String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access Object Storage Service (OSS) because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
String accessKeyId = "yourAccessKeyId";
String accessKeySecret = "yourAccessKeySecret";
// Specify the bucket name. Example: examplebucket.
String bucketName = "examplebucket";
// Create an OSSClient instance.
OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);

try {
    SetBucketCORSRequest request = new SetBucketCORSRequest(bucketName);
    // Create a container for the CORS rules. You can configure up to 10 CORS rules for each bucket.
    ArrayList<SetBucketCORSRequest.CORSRule> putCorsRules = new ArrayList<SetBucketCORSRequest.CORSRule>();
    SetBucketCORSRequest.CORSRule corRule = new SetBucketCORSRequest.CORSRule();
    ArrayList<String> allowedOrigin = new ArrayList<String>();
    // Specify the origins from which cross-origin requests are allowed.
    allowedOrigin.add("http://example.com");
    ArrayList<String> allowedMethod = new ArrayList<String>();
    // Specify the HTTP methods that cross-origin requests are allowed to use, including PUT, POST, GET, DELETE, and HEAD.
    allowedMethod.add("GET");
    ArrayList<String> allowedHeader = new ArrayList<String>();
    // Specify whether the headers that are specified by Access-Control-Request-Headers in the OPTIONS preflight request are allowed.
    allowedHeader.add("x-oss-test");
    ArrayList<String> exposedHeader = new ArrayList<String>();
    // Specify the response headers for allowed access requests from applications.
    exposedHeader.add("x-oss-test1");
    // You can use only one asterisk (*) as the wildcard character for AllowedOrigins and AllowedMethods in a CORS rule. The wildcard character (*) specifies that all origins or operations are allowed.
    corRule.setAllowedMethods(allowedMethod);
    corRule.setAllowedOrigins(allowedOrigin);
    // AllowedHeaders and ExposeHeaders do not support wildcard characters.
    corRule.setAllowedHeaders(allowedHeader);
    corRule.setExposeHeaders(exposedHeader);
    // Specify the period of time in which the browser can cache the response to an OPTIONS preflight request for specific resources. Unit: seconds.
    corRule.setMaxAgeSeconds(10);
    // You can configure up to 10 CORS rules for the bucket.
    putCorsRules.add(corRule);
    // The existing rules are overwritten.
    request.setCorsRules(putCorsRules);
    // Specify whether to return the Vary: Origin header. If you set this parameter to TRUE, the Vary: Origin header is returned regardless of whether the request is a cross-origin request or whether the cross-origin request is successful. If you set this parameter to False, the Vary: Origin header is not returned.
    // request.setResponseVary(Boolean.TRUE);
    ossClient.setBucketCORS(request);
}

```

```

    } catch (OSSException oe) {
        System.out.println("Caught an OSSException, which means your request made it to
OSS, "
            + "but was rejected with an error response for some reason.");
        System.out.println("Error Message:" + oe.getErrorMessage());
        System.out.println("Error Code:" + oe.getErrorCode());
        System.out.println("Request ID:" + oe.getRequestId());
        System.out.println("Host ID:" + oe.getHostId());
    } catch (ClientException ce) {
        System.out.println("Caught an ClientException, which means the client encounter
ed "
            + "a serious internal problem while trying to communicate with OSS, "
            + "such as not being able to access the network.");
        System.out.println("Error Message:" + ce.getMessage());
    } finally {
        if (ossClient != null) {
            ossClient.shutdown();
        }
    }
}
}

```

Query CORS rules

The following code provides an example on how to query CORS rules that are configured for a specific bucket:

```

import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.SetBucketCORSRequest;
import java.util.ArrayList;

public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify yo
ur actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud accou
nt to access OSS because the account has permissions on all API operations. We recommend th
at you use a RAM user to call API operations or perform routine O&M. To create a RAM user,
log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the bucket name. Example: examplebucket.
        String bucketName = "examplebucket";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret
);
        try {
            ArrayList<SetBucketCORSRequest.CORSRule> corsRules;
            // Query the CORS rules.
            corsRules = (ArrayList<SetBucketCORSRequest.CORSRule>) ossClient.getBucketCORS
Rules(bucketName);

```

```

        for (SetBucketCORSRequest.CORSRule rule : corsRules) {
            for (String allowedOrigin1 : rule.getAllowedOrigins()) {
                // Query the origins from which cross-origin requests are allowed.
                System.out.println(allowedOrigin1);
            }
            for (String allowedMethod1 : rule.getAllowedMethods()) {
                // Query the HTTP methods that cross-origin requests are allowed to use
                System.out.println(allowedMethod1);
            }
            if (rule.getAllowedHeaders().size() > 0){
                for (String allowedHeader1 : rule.getAllowedHeaders()) {
                    // Query the headers that can be included in the responses to allow
ed cross-origin requests.
                    System.out.println(allowedHeader1);
                }
            }
            if (rule.getExposeHeaders().size() > 0) {
                for (String exposeHeader : rule.getExposeHeaders()) {
                    // Query the response headers for allowed access requests from appl
ications.
                    System.out.println(exposeHeader);
                }
            }
            if ( null != rule.getMaxAgeSeconds()) {
                System.out.println(rule.getMaxAgeSeconds());
            }
        }
    } catch (OSSEException oe) {
        System.out.println("Caught an OSSEException, which means your request made it to
OSS, "
            + "but was rejected with an error response for some reason.");
        System.out.println("Error Message:" + oe.getErrorMessage());
        System.out.println("Error Code:" + oe.getErrorCode());
        System.out.println("Request ID:" + oe.getRequestId());
        System.out.println("Host ID:" + oe.getHostId());
    } catch (ClientException ce) {
        System.out.println("Caught an ClientException, which means the client encounter
ed "
            + "a serious internal problem while trying to communicate with OSS, "
            + "such as not being able to access the network.");
        System.out.println("Error Message:" + ce.getMessage());
    } finally {
        if (ossClient != null) {
            ossClient.shutdown();
        }
    }
}
}

```

Delete CORS rules

The following code provides an example on how to delete the CORS rules that are configured for a specific bucket:

```
import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the bucket name. Example: examplebucket.
        String bucketName = "examplebucket";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);

        try {
            // Delete the CORS rules that are configured for the bucket.
            ossClient.deleteBucketCORSRules(bucketName);
        } catch (OSSException oe) {
            System.out.println("Caught an OSSException, which means your request made it to OSS, "
                + "but was rejected with an error response for some reason.");
            System.out.println("Error Message:" + oe.getErrorMessage());
            System.out.println("Error Code:" + oe.getErrorCode());
            System.out.println("Request ID:" + oe.getRequestId());
            System.out.println("Host ID:" + oe.getHostId());
        } catch (ClientException ce) {
            System.out.println("Caught an ClientException, which means the client encountered "
                + "a serious internal problem while trying to communicate with OSS, "
                + "such as not being able to access the network.");
            System.out.println("Error Message:" + ce.getMessage());
        } finally {
            if (ossClient != null) {
                ossClient.shutdown();
            }
        }
    }
}
```

References

- For the complete sample code that is used to manage CORS rules, visit [GitHub](#).
- For more information about the API operation that you can call to configure CORS rules, see [Put Bucket Cors](#).

- For more information about the API operation that you can call to query CORS rules, see [GetBucketCors](#).
- For more information about the API operation that you can call to delete CORS rules, see [DeleteBucketCors](#).

2.5.20. Transfer acceleration

The transfer acceleration feature allows users worldwide to access objects stored in Object Storage Service (OSS) more quickly. This feature is applicable to scenarios where data must be transferred over long geographical distances. This feature can also be used to download or upload large objects that are gigabytes or terabytes in size.

Enable transfer acceleration

The following code provides an example on how to enable transfer acceleration for a bucket named `examplebucket`:

```

import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify the
        // endpoint based on your business requirements.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account
        // to access OSS because the account has permissions on all API operations. We recommend
        // that you use a RAM user to call API operations or perform routine O&M. To create a RAM
        // user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the name of the bucket. Example: examplebucket.
        String bucketName = "examplebucket";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);

        try {
            // Configure transfer acceleration for the bucket.
            // If enabled is set to true, transfer acceleration is enabled. If enabled is set
            // to false, transfer acceleration is disabled.
            boolean enabled = true;
            ossClient.setBucketTransferAcceleration(bucketName, enabled);
        } catch (OSSException oe) {
            System.out.println("Caught an OSSException, which means your request made it to
            OSS, "
                + "but was rejected with an error response for some reason.");
            System.out.println("Error Message:" + oe.getErrorMessage());
            System.out.println("Error Code:" + oe.getErrorCode());
            System.out.println("Request ID:" + oe.getRequestId());
            System.out.println("Host ID:" + oe.getHostId());
        } catch (ClientException ce) {
            System.out.println("Caught an ClientException, which means the client encountered
            "
                + "a serious internal problem while trying to communicate with OSS, "
                + "such as not being able to access the network.");
            System.out.println("Error Message:" + ce.getMessage());
        } finally {
            if (ossClient != null) {
                ossClient.shutdown();
            }
        }
    }
}

```

Query the transfer acceleration state of a bucket

The following code provides an example on how to query the transfer acceleration state of a bucket named examplebucket:

```

import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.TransferAcceleration;
public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify the
        // endpoint based on your business requirements.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account
        // to access OSS because the account has permissions on all API operations. We recommend that
        // you use a RAM user to call API operations or perform routine O&M. To create a RAM user,
        // log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the name of the bucket. Example: examplebucket.
        String bucketName = "examplebucket";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);

        try {
            // Query the transfer acceleration state of the bucket.
            // If the returned value is true, the transfer acceleration feature is enabled
            // for the bucket. If the returned value is false, the transfer acceleration feature is disabled
            // for the bucket.
            TransferAcceleration result = ossClient.getBucketTransferAcceleration(bucketName);

            System.out.println("Is transfer acceleration enabled:" + result.isEnabled());
        } catch (OSSException oe) {
            System.out.println("Caught an OSSException, which means your request made it to
            OSS, "
                + "but was rejected with an error response for some reason.");
            System.out.println("Error Message:" + oe.getErrorMessage());
            System.out.println("Error Code:" + oe.getErrorCode());
            System.out.println("Request ID:" + oe.getRequestId());
            System.out.println("Host ID:" + oe.getHostId());
        } catch (ClientException ce) {
            System.out.println("Caught an ClientException, which means the client encountered "
                + "a serious internal problem while trying to communicate with OSS, "
                + "such as not being able to access the network.");
            System.out.println("Error Message:" + ce.getMessage());
        } finally {
            if (ossClient != null) {
                ossClient.shutdown();
            }
        }
    }
}

```

References

- For more information about the complete sample code that is used to enable transfer acceleration for a bucket, visit [GitHub](#).
- For more information about the API operation that is called to enable transfer acceleration for a bucket, see [PutBucketTransferAcceleration](#).
- For more information about the API operation that is called to query the transfer acceleration state of a bucket, see [GetBucketTransferAcceleration](#).

2.6. Upload objects

2.6.1. Overview

This topic is the overview of object upload in OSS Java SDK.

In OSS, an object is the basic unit for data operations. OSS Java SDK provides the following file upload methods:

- **Simple upload**: supports the upload of a file up to 5 GB in size. It includes streaming upload and file upload.
- **Form upload**: supports the upload of a file up to 5 GB in size.
- **Append upload**: supports the upload of a file up to 5 GB in size.
- **Resumable upload**: supports the upload of a file up to 48.8 TB in size. It applies to the upload of large files. It supports concurrent upload and you can define the size of each part.
- **Multipart upload**: supports the upload of a file up to 48.8 TB in size. It applies to the upload of large files.

 **Note** For more information about the usage scenarios of each upload method, see “Upload files” in OSS Developer Guide.

During the file upload, you can set **Object Meta**, and view the upload progress in the **upload progress bar**. After you complete the file upload, you can perform **upload callback**.

2.6.2. Simple upload

To perform simple upload, you can call `PutObject` to upload a single object. Simple upload allows you to perform streaming upload and object upload. For streaming upload, `InputStream` is used to upload data to Object Storage Service (OSS). For object upload, local files are directly uploaded to OSS. This topic describes how to use streaming upload and object upload to upload objects.

Streaming upload

You can use streaming upload to upload a data stream to an OSS object.

 **Note** If an object with the same name already exists, the uploaded data overwrites the existing content of the object. Otherwise, a new object is created.

- Upload a string

The following code provides an example on how to upload a string to the `exampleobject.txt` object in the `exampledir` directory of the `examplebucket` bucket:

```
import com.aliyun.oss.ClientException;
```

```

import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.PutObjectRequest;
import java.io.ByteArrayInputStream;
public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify
the actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API oper
ations. Using these credentials to perform operations in OSS is a high-risk operation. We
recommend that you use a RAM user to call API operations or perform routine O&M. To creat
e a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the name of the bucket. Example: examplebucket.
        String bucketName = "examplebucket";
        // Specify the full path of the object. The path cannot contain the bucket name.
Example: exampledir/exampleobject.txt.
        String objectName = "exampledir/exampleobject.txt";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecr
et);
        try {
            // Specify the string that you want to upload.
            String content = "Hello OSS";
            // Create a PutObjectRequest object.
            PutObjectRequest putObjectRequest = new PutObjectRequest(bucketName, objectNa
me, new ByteArrayInputStream(content.getBytes()));
            // (Optional) Specify the storage class and the access control list (ACL) of
the object by using the following sample code:
            // ObjectMetadata metadata = new ObjectMetadata();
            // metadata.setHeader(OSSHeaders.OSS_STORAGE_CLASS, StorageClass.Standard.toS
tring());
            // metadata.setObjectAcl(CannedAccessControlList.Private);
            // putObjectRequest.setMetadata(metadata);
            // Upload the string.
            ossClient.putObject(putObjectRequest);
        } catch (OSSException oe) {
            System.out.println("Caught an OSSException, which means your request made it
to OSS, "
                + "but was rejected with an error response for some reason.");
            System.out.println("Error Message:" + oe.getErrorMessage());
            System.out.println("Error Code:" + oe.getErrorCode());
            System.out.println("Request ID:" + oe.getRequestId());
            System.out.println("Host ID:" + oe.getHostId());
        } catch (ClientException ce) {
            System.out.println("Caught an ClientException, which means the client encount
ered "
                + "a serious internal problem while trying to communicate with OSS, "
                + "such as not being able to access the network.");
            System.out.println("Error Message:" + ce.getMessage());
        } finally {

```

```
        if (ossClient != null) {  
            ossClient.shutdown();  
        }  
    }  
}
```

- Upload a byte array

The following code provides an example on how to upload a byte array to the `exampleobject.txt` object in the `exampledir` directory of the `examplebucket` bucket:

```

import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import java.io.ByteArrayInputStream;
public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify
        the actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API oper
        ations. Using these credentials to perform operations in OSS is a high-risk operation. We
        recommend that you use a RAM user to call API operations or perform routine O&M. To creat
        e a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the name of the bucket. Example: examplebucket.
        String bucketName = "examplebucket";
        // Specify the full path of the object. The path cannot contain the bucket name.
        Example: exampledir/exampleobject.txt.
        String objectName = "exampledir/exampleobject.txt";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecr
        et);
        try {
            // Specify a byte array.
            byte[] content = "Hello OSS".getBytes();
            // Create a PutObject request.
            ossClient.putObject(bucketName, objectName, new ByteArrayInputStream(content)
        );
        } catch (OSSException oe) {
            System.out.println("Caught an OSSException, which means your request made it
            to OSS, "
                + "but was rejected with an error response for some reason.");
            System.out.println("Error Message:" + oe.getErrorMessage());
            System.out.println("Error Code:" + oe.getErrorCode());
            System.out.println("Request ID:" + oe.getRequestId());
            System.out.println("Host ID:" + oe.getHostId());
        } catch (ClientException ce) {
            System.out.println("Caught an ClientException, which means the client encount
            ered "
                + "a serious internal problem while trying to communicate with OSS, "
                + "such as not being able to access the network.");
            System.out.println("Error Message:" + ce.getMessage());
        } finally {
            if (ossClient != null) {
                ossClient.shutdown();
            }
        }
    }
}

```

- Upload a network stream

The following code provides an example on how to upload a network stream to the `exampleobject.txt` object in the `exampledir` directory of the `examplebucket` bucket:

```

import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import java.io.InputStream;
import java.net.URL;
public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify
        the actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API oper
        ations. Using these credentials to perform operations in OSS is a high-risk operation. We
        recommend that you use a RAM user to call API operations or perform routine O&M. To creat
        e a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the name of the bucket. Example: examplebucket.
        String bucketName = "examplebucket";
        // Specify the full path of the object. The path cannot contain the bucket name.
        Example: exampledir/exampleobject.txt.
        String objectName = "exampledir/exampleobject.txt";
        // Specify the address of the network stream.
        String url = "https://www.aliyun.com/";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecr
        et);
        try {
            InputStream inputStream = new URL(url).openStream();
            // Create a PutObject request.
            ossClient.putObject(bucketName, objectName, inputStream);
        } catch (OSSException oe) {
            System.out.println("Caught an OSSException, which means your request made it
            to OSS, "
                + "but was rejected with an error response for some reason.");
            System.out.println("Error Message:" + oe.getErrorMessage());
            System.out.println("Error Code:" + oe.getErrorCode());
            System.out.println("Request ID:" + oe.getRequestId());
            System.out.println("Host ID:" + oe.getHostId());
        } catch (ClientException ce) {
            System.out.println("Caught an ClientException, which means the client encount
            ered "
                + "a serious internal problem while trying to communicate with OSS, "
                + "such as not being able to access the network.");
            System.out.println("Error Message:" + ce.getMessage());
        } finally {
            if (ossClient != null) {
                ossClient.shutdown();
            }
        }
    }
}

```

- Upload a file stream

The following code provides an example on how to upload a file stream to the `exampleobject.txt` object in the `exampledir` directory of the `examplebucket` bucket:

```
import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import java.io.FileInputStream;
import java.io.InputStream;
public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify
        // the actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API oper-
        // ations. Using these credentials to perform operations in OSS is a high-risk operation. We
        // recommend that you use a RAM user to call API operations or perform routine O&M. To creat
        // e a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the name of the bucket. Example: examplebucket.
        String bucketName = "examplebucket";
        // Specify the full path of the object. The path cannot contain the bucket name.
        // Example: exampledir/exampleobject.txt.
        String objectName = "exampledir/exampleobject.txt";
        // Specify the full path of the local file. Example: D:\\localpath\\examplefile.t
        // xt.
        // By default, if the path of the local file is not specified, the local file is
        // uploaded from the path of the project to which the sample program belongs.
        String filePath = "D:\\localpath\\examplefile.txt";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecr
        et);
        try {
            InputStream inputStream = new FileInputStream(filePath);
            // Create a PutObject request.
            ossClient.putObject(bucketName, objectName, inputStream);
        } catch (OSSException oe) {
            System.out.println("Caught an OSSException, which means your request made it
            to OSS, "
                + "but was rejected with an error response for some reason.");
            System.out.println("Error Message:" + oe.getErrorMessage());
            System.out.println("Error Code:" + oe.getErrorCode());
            System.out.println("Request ID:" + oe.getRequestId());
            System.out.println("Host ID:" + oe.getHostId());
        } catch (ClientException ce) {
            System.out.println("Caught an ClientException, which means the client encount
            ered "
                + "a serious internal problem while trying to communicate with OSS, "
                + "such as not being able to access the network.");
            System.out.println("Error Message:" + ce.getMessage());
        } finally {
            if (ossClient != null) {
                ossClient.shutdown();
            }
        }
    }
}
```

```

    }
}
}
}

```

Upload an object

You can use object upload to upload a local file to OSS.

The following code provides an example on how to upload a local file named `examplefile.txt` to the `exampledir` directory of the `examplebucket` bucket. After the local file is uploaded, the object in the `exampledir` directory is named `exampleobject.txt`.

```

import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.PutObjectRequest;
import java.io.File;

public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify the actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the name of the bucket. Example: examplebucket.
        String bucketName = "examplebucket";
        // Specify the full path of the object. The path cannot contain the bucket name. Example: exampledir/exampleobject.txt.
        String objectName = "exampledir/exampleobject.txt";
        // Specify the full path of the local file. Example: D:\\localpath\\examplefile.txt.
        // By default, if you do not specify the path of the local file, the file is uploaded from the local path of the project to which the sample program belongs.
        String filePath = "D:\\localpath\\examplefile.txt";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);

        try {
            // Create a PutObjectRequest object.
            PutObjectRequest putObjectRequest = new PutObjectRequest(bucketName, objectName, new File(filePath));
            // (Optional) Specify the storage class and the ACL of the object by using the following sample code:
            // ObjectMetadata metadata = new ObjectMetadata();
            // metadata.setHeader(OSSHeaders.OSS_STORAGE_CLASS, StorageClass.Standard.toString());
            // metadata.setObjectAcl(CannedAccessControlList.Private);
            // putObjectRequest.setMetadata(metadata);
            // Upload the object.

```

```

        ossClient.putObject(putObjectRequest);
    } catch (OSSException oe) {
        System.out.println("Caught an OSSException, which means your request made it to
OSS, "
            + "but was rejected with an error response for some reason.");
        System.out.println("Error Message:" + oe.getErrorMessage());
        System.out.println("Error Code:" + oe.getErrorCode());
        System.out.println("Request ID:" + oe.getRequestId());
        System.out.println("Host ID:" + oe.getHostId());
    } catch (ClientException ce) {
        System.out.println("Caught an ClientException, which means the client encounter
ed "
            + "a serious internal problem while trying to communicate with OSS, "
            + "such as not being able to access the network.");
        System.out.println("Error Message:" + ce.getMessage());
    } finally {
        if (ossClient != null) {
            ossClient.shutdown();
        }
    }
}
}
}

```

FAQ

- [How do I obtain the URL of an uploaded object?](#)
- [How do I upload and download folders to and from OSS?](#)

References

- For more information about the complete sample code for simple upload, visit [GitHub](#).
- For more information about the API operation that you can call to perform simple upload, see [Put Object](#).

2.6.3. Form upload

You can use an HTML form to upload an object up to 5 GB in size to a specified bucket. This upload method is called form upload. This topic describes how to perform form upload in Object Storage Service (OSS).

Sample code

The following code provides an example on how to perform form upload:

```

package com.aliyun.oss;
import com.aliyun.oss.common.utils.BinaryUtil;
import com.aliyun.oss.internal.OSSUtils;
import com.aliyun.oss.model.Callback;
import org.apache.commons.codec.binary.Base64;
import javax.activation.MimetypesFileTypeMap;
import java.io.*;
import java.net.HttpURLConnection;
import java.net.URL;
import java.net.URLEncoder;

```

```

import java.util.Iterator;
import java.util.LinkedHashMap;
import java.util.Map;
public class PostObjectSample {
    // Specify the full path of the local file that you want to upload.
    private String localFilePath = "yourLocalFile";
    // In this example, the endpoint of the China (Hangzhou) region is used. Specify the actual endpoint.
    private String endpoint = "http://oss-cn-hangzhou.aliyuncs.com";
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS, because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
    private String accessKeyId = "yourAccessKeyId";
    private String accessKeySecret = "yourAccessKeySecret";
    // Specify the name of the bucket.
    private String bucketName = "yourBucketName";
    // Specify the full path of the object. The path cannot contain the bucket name.
    private String objectName = "yourObjectName";
    // Specify the URL of the server to which you want to send the callback request. Example: http://oss-demo.oss-cn-hangzhou.aliyuncs.com:23450 or http://127.0.0.1:9090.
    private String callbackServerUrl = "yourCallbackServerUrl";
    // Specify the Host header in the callback request. Example: oss-cn-hangzhou.aliyuncs.com.
    private String callbackServerHost = "yourCallbackServerHost";
    // Specify the MD5 hash. The MD5 hash is calculated based on the entire body.
    private static String contentMD5 = "yourContentMD5";
    /**
     * Perform form upload.
     * @throws Exception
     */
    private void PostObject() throws Exception {
        // Add the bucket name to the URL. The URL format is http://yourBucketName.oss-cn-hangzhou.aliyuncs.com.
        String urlStr = endpoint.replace("http://", "http://" + bucketName + ".");
        // Specify Map for the form.
        Map<String, String> formFields = new LinkedHashMap<String, String>();
        // Specify the object name.
        formFields.put("key", this.objectName);
        // Specify Content-Disposition.
        formFields.put("Content-Disposition", "attachment;filename="
            + localFilePath);
        // Specify callback parameters. For more information about callbacks, see Callback.

        // Callback callback = new Callback();
        // Specify the URL of the server to which you want to send the callback request. Example: http://oss-demo.oss-cn-hangzhou.aliyuncs.com:23450 or http://127.0.0.1:9090.
        // callback.setCallbackUrl(callbackServerUrl);
        // Specify the Host header in the callback request. Example: oss-cn-hangzhou.aliyuncs.com.
        // callback.setCallbackHost(callbackServerHost);
        // Specify the value of the body field included in the callback request.
        // callback.setCallbackBody("{\"\\\"mime\\\"type\\\"\": \"${mime\\\"Type}\\\"\", \"\\\"size\\\"\": \"${size}\\\"}\"");
        // Specify Content-Type in the callback request.
        // callback.setCallbackBodyType(Callback.CallbackBodyType.JSON);

```

```

        // callback.setCallbackBodyType(callback.setCallbackBodyType.OSS);
        // Specify the custom parameters used to send the callback request. Each custom parameter consists of a key-value pair. The key must start with x: and be in lower case.
        // callback.addCallbackVar("x:var1", "value1");
        // callback.addCallbackVar("x:var2", "value2");
        // Specify the callback parameter in Map for the form.
        // setCallBack(formFields, callback);
        // Specify OSSAccessKeyId.
        formFields.put("OSSAccessKeyId", accessKeyId);
        String policy = "{\n\"expiration\": \"2120-01-01T12:00:00.000Z\",\n\"conditions\": [[\ncontent-length-range\", 0, 104857600]]}";
        String encodePolicy = new String(Base64.encodeBase64(policy.getBytes()));
        // Specify the policy.
        formFields.put("policy", encodePolicy);
        // Generate a signature.
        String signaturecom = com.aliyun.oss.common.auth.ServiceSignature.create().computeSignature(accessKeySecret, encodePolicy);
        // Add the signature to the policy.
        formFields.put("Signature", signaturecom);
        String ret = formUpload(urlStr, formFields, localFilePath);
        System.out.println("Post Object [" + this.objectName + "] to bucket [" + bucketName + "]);");
        System.out.println("post reponse:" + ret);
    }

    private static String formUpload(String urlStr, Map<String, String> formFields, String localFile)
        throws Exception {
        String res = "";
        HttpURLConnection conn = null;
        String boundary = "9431149156168";
        try {
            URL url = new URL(urlStr);
            conn = (HttpURLConnection) url.openConnection();
            conn.setConnectTimeout(5000);
            conn.setReadTimeout(30000);
            conn.setDoOutput(true);
            conn.setDoInput(true);
            conn.setRequestMethod("POST");
            conn.setRequestProperty("User-Agent",
                "Mozilla/5.0 (Windows; U; Windows NT 6.1; zh-CN; rv:1.9.2.6)");
            // Specify the MD5 hash. The MD5 hash is calculated based on the entire body. You can configure MD5 encryption settings to enable MD5 verification.
            // conn.setRequestProperty("Content-MD5", contentMD5);
            conn.setRequestProperty("Content-Type",
                "multipart/form-data; boundary=" + boundary);
            OutputStream out = new DataOutputStream(conn.getOutputStream());
            // Recursively read all Map data for the form and write the data to the output stream.
            if (formFields != null) {
                StringBuffer strBuf = new StringBuffer();
                Iterator<Map.Entry<String, String>> iter = formFields.entrySet().iterator();

                int i = 0;
                while (iter.hasNext()) {
                    Map.Entry<String, String> entry = iter.next();

```

```

        String inputName = entry.getKey();
        String inputValue = entry.getValue();
        if (inputValue == null) {
            continue;
        }
        if (i == 0) {
            strBuf.append("--").append(boundary).append("\r\n");
            strBuf.append("Content-Disposition: form-data; name=\""
                + inputName + "\"\r\n\r\n");
            strBuf.append(inputValue);
        } else {
            strBuf.append("\r\n").append("--").append(boundary).append("\r\n");
            strBuf.append("Content-Disposition: form-data; name=\""
                + inputName + "\"\r\n\r\n");
            strBuf.append(inputValue);
        }
        i++;
    }
    out.write(strBuf.toString().getBytes());
}
// Read the file and write it to the output stream.
File file = new File(localFile);
String filename = file.getName();
String contentType = new MimetypesFileTypeMap().getContentType(file);
if (contentType == null || contentType.equals("")) {
    contentType = "application/octet-stream";
}
StringBuffer strBuf = new StringBuffer();
strBuf.append("\r\n").append("--").append(boundary)
    .append("\r\n");
strBuf.append("Content-Disposition: form-data; name=\"file\"; "
    + "filename=\"" + filename + "\"\r\n");
strBuf.append("Content-Type: " + contentType + "\r\n\r\n");
out.write(strBuf.toString().getBytes());
DataInputStream in = new DataInputStream(new FileInputStream(file));
int bytes = 0;
byte[] bufferOut = new byte[1024];
while ((bytes = in.read(bufferOut)) != -1) {
    out.write(bufferOut, 0, bytes);
}
in.close();
byte[] endData = ("\r\n--" + boundary + "--\r\n").getBytes();
out.write(endData);
out.flush();
out.close();
// Read returned data.
strBuf = new StringBuffer();
BufferedReader reader = new BufferedReader(new InputStreamReader(conn.getInputS
tream()));
String line = null;
while ((line = reader.readLine()) != null) {
    strBuf.append(line).append("\n");
}
res = strBuf.toString();

```



```
        reader.close();
        reader = null;
    } catch (ClientException e) {
//        System.err.println("Send post request exception: " + e);
        System.err.println(e.getErrorCode()+" msg="+e.getMessage());
        throw e;
    } finally {
        if (conn != null) {
            conn.disconnect();
            conn = null;
        }
    }
    return res;
}

private static void setCallBack(Map<String, String> formFields, Callback callback) {
    if (callback != null) {
        String jsonCb = OSSUtils.jsonizeCallback(callback);
        String base64Cb = BinaryUtil.toBase64String(jsonCb.getBytes());
        formFields.put("callback", base64Cb);
        if (callback.hasCallbackVar()) {
            Map<String, String> varMap = callback.getCallbackVar();
            for (Map.Entry<String, String> entry : varMap.entrySet()) {
                formFields.put(entry.getKey(), entry.getValue());
            }
        }
    }
}

public static void main(String[] args) throws Exception {
    PostObjectSample ossPostObject = new PostObjectSample();
    ossPostObject.PostObject();
}
}
```

References

For more information about the API operation that you can call to perform form upload, see [PostObject](#).

2.6.4. Append upload

You can call the AppendObject operation to append content to existing appendable objects.

Usage notes

- If the object to which you want to append content does not exist, an appendable object is created when you call the AppendObject operation.
- When you call the AppendObject operation to append content to an object, you may encounter a few situations: If the object is an appendable object, and the specified position from which the append operation starts is equal to the current object length, the content is appended to the end of the object. If the specified position from which the append operation starts is not equal to the current object length, a `PositionNotEqualToLength` exception is thrown. If the object is not an appendable object, an `ObjectNotAppendable` exception is thrown.
- The CopyObject operation cannot be performed on appendable objects.

Sample code

The following code provides an example on how to perform an append upload operation:

```
import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.AppendObjectRequest;
import com.aliyun.oss.model.AppendObjectResult;
import com.aliyun.oss.model.ObjectMetadata;
import java.io.ByteArrayInputStream;

public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify the actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the name of the bucket. Example: examplebucket.
        String bucketName = "examplebucket";
        // Specify the full path of the object. The full path cannot contain the bucket name. Example: exampledir/exampleobject.txt.
        String objectName = "exampledir/exampleobject.txt";
        String content1 = "Hello OSS A \n";
        String content2 = "Hello OSS B \n";
        String content3 = "Hello OSS C \n";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);

        try {
            ObjectMetadata meta = new ObjectMetadata();
            // Specify the type of content that you want to upload.
            meta.setContentType("text/plain");
            // Specify the caching behavior of the web page for the object.
            //meta.setCacheControl("no-cache");
            // Specify the name of the object when the object is downloaded.
            //meta.setContentDisposition("attachment;filename=oss_download.txt");
            // Specify the encoding format for the content of the object.
            //meta.setContentEncoding(OSSConstants.DEFAULT_CHARSET_NAME);
            // Specify the request header that is used to check whether the content of the received message is the same as the content of the sent message.
            //meta.setContentMD5("ohhngLBJfKkPSBOleNaUA==");
            // Specify the expiration time.
            //try {
            //    meta.setExpirationTime(DateUtil.parseRfc822Date("Wed, 08 Jul 2022 16:57:01 GMT"));
            //} catch (ParseException e) {
            //    e.printStackTrace();
            //}
            // Specify the server-side encryption method. In this example, the method is sse-kms.
```

```

        // Specify the server-side encryption method. In this example, the method is set
        // to server-side encryption by using OSS-managed keys (SSE-OSS).
        //meta.setServerSideEncryption(ObjectMetadata.AES_256_SERVER_SIDE_ENCRYPTION);
        // Specify the access control list (ACL) of the object. In this example, the ACL
        // of the object is set to private.
        //meta.setObjectAcl(CannedAccessControlList.Private);
        // Specify the storage class of the object.
        //meta.setHeader(OSSHeaders.OSS_STORAGE_CLASS, StorageClass.Standard);
        // You can add parameters whose names are prefixed with x-oss-meta-* when you call
        // the AppendObject operation to create an appendable object. These parameters cannot be included
        // in the requests when you append content to an existing appendable object. Parameters
        // whose names are prefixed with x-oss-meta-* are considered as the metadata of the object.
        //meta.setHeader("x-oss-meta-author", "Alice");
        // Configure multiple parameters by using AppendObjectRequest.
        AppendObjectRequest appendObjectRequest = new AppendObjectRequest(bucketName, objectName,
        new ByteArrayInputStream(content1.getBytes()), meta);
        // Configure a single parameter by using AppendObjectRequest.
        // Specify the bucket name.
        //appendObjectRequest.setBucketName(bucketName);
        // Specify the object name.
        //appendObjectRequest.setKey(objectName);
        // Specify the content that you want to append. You can choose InputStream or File. In this example,
        // the content is set to InputStream.
        //appendObjectRequest.setInputStream(new ByteArrayInputStream(content1.getBytes()));
        // Specify the content that you want to append. You can choose InputStream or File. In this example,
        // the content is set to File.
        //appendObjectRequest.setFile(new File("D:\\localpath\\examplefile.txt"));
        // Specify the metadata of the object. You can specify the metadata of an object only when you perform
        // the first append operation on the object.
        //appendObjectRequest.setMetadata(meta);
        // Perform the first append operation.
        // Specify the position from which the append operation starts.
        appendObjectRequest.setPosition(0L);
        AppendObjectResult appendObjectResult = ossClient.appendObject(appendObjectRequest);
        // Calculate the CRC-64 value of the object. The value is calculated based on the ECMA-182 specification.
        System.out.println(appendObjectResult.getObjectCRC());
        // Perform the second append operation.
        // NextPosition specifies the position from which the next append operation starts, which is the length of the object.
        appendObjectRequest.setPosition(appendObjectResult.getNextPosition());
        appendObjectRequest.setInputStream(new ByteArrayInputStream(content2.getBytes()));
        appendObjectResult = ossClient.appendObject(appendObjectRequest);
        // Perform the third append operation.
        appendObjectRequest.setPosition(appendObjectResult.getNextPosition());
        appendObjectRequest.setInputStream(new ByteArrayInputStream(content3.getBytes()));
        appendObjectResult = ossClient.appendObject(appendObjectRequest);
    } catch (OSSException oe) {
        System.out.println("Caught an OSSException, which means your request made it to OSS, "
        + "but was rejected with an error response for some reason.");
    }
}

```

```

        // oss has rejected with an error response for some reason.
        System.out.println("Error Message:" + oe.getErrorMessage());
        System.out.println("Error Code:" + oe.getErrorCode());
        System.out.println("Request ID:" + oe.getRequestId());
        System.out.println("Host ID:" + oe.getHostId());
    } catch (ClientException ce) {
        System.out.println("Caught an ClientException, which means the client encounter
ed "
            + "a serious internal problem while trying to communicate with OSS, "
            + "such as not being able to access the network.");
        System.out.println("Error Message:" + ce.getMessage());
    } finally {
        if (ossClient != null) {
            ossClient.shutdown();
        }
    }
}
}

```

References

- For more information about the complete sample code that is used to perform append upload, visit [GitHub](#).
- For more information about the API operation that you can call to perform append upload, see [AppendObject](#).

2.6.5. Resumable upload

Before you upload an object to Object Storage Service (OSS) by using resumable upload, you can specify a directory for the checkpoint file that stores resumable upload records. If an object fails to upload due to a network exception or program error, the upload task is resumed from the position recorded in the checkpoint file to upload the remaining data.

Examples

The following code provides an example on how to perform resumable upload by using multipart upload:

```

import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.*;
public class Demo {
    public static void main(String[] args) {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Create an OSSClient instance.

```

```

        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret
    );

    try {
        ObjectMetadata meta = new ObjectMetadata();
        // Specify the type of content that you want to upload.
        meta.setContentType("text/plain");
        // Specify the access control list (ACL) of the object to upload.
        // meta.setObjectAcl(CannedAccessControlList.Private);
        // Configure parameters by using UploadFileRequest.
        // Specify the bucket name and the full path of the object, and the full path c
annot contain the bucket name. Examples: examplebucket for the bucket name and exampledir/e
xampleobject.txt for the path of the object.
        UploadFileRequest uploadFileRequest = new UploadFileRequest("examplebucket","ex
ampledir/exampleobject.txt");
        // Configure a single parameter by using UploadFileRequest.
        // Specify the full path of the local file. Example: D:\\localpath\\examplefile
.txt. By default, if you do not specify the full path of the local file, the local file is
uploaded from the path of the project to which the sample program belongs.
        uploadFileRequest.setUploadFile("D:\\localpath\\examplefile.txt");
        // Specify the number of threads that can be concurrently processed. Default va
lue: 1.
        uploadFileRequest.setTaskNum(5);
        // Specify the size of each part. Unit: bytes. Valid values: 100 KB to 5 GB. De
fault value: 100 KB.
        uploadFileRequest.setPartSize(1 * 1024 * 1024);
        // Specify whether to enable resumable upload. By default, resumable upload is
disabled.
        uploadFileRequest.setEnableCheckpoint(true);
        // Specify the checkpoint file that records the upload result of each part. Thi
s file stores information about upload progress. If a part fails to upload, the task can be
continued based on the progress recorded in the checkpoint file. After the object is upload
ed, the checkpoint file is deleted.
        // By default, if you do not specify this parameter, this checkpoint file share
s the same directory as the object you want to upload. The directory is named ${uploadFile}
.ucp.
        uploadFileRequest.setCheckpointFile("yourCheckpointFile");
        // Configure object metadata.
        uploadFileRequest.setObjectMetadata(meta);
        // Configure upload callback. The parameter type is Callback.
        //uploadFileRequest.setCallback("yourCallbackEvent");
        // Start resumable upload.
        ossClient.uploadFile(uploadFileRequest);
    } catch (OSSException oe) {
        System.out.println("Caught an OSSException, which means your request made it to
OSS, "

            + "but was rejected with an error response for some reason.");
        System.out.println("Error Message:" + oe.getErrorMessage());
        System.out.println("Error Code:" + oe.getErrorCode());
        System.out.println("Request ID:" + oe.getRequestId());
        System.out.println("Host ID:" + oe.getHostId());
    } catch (Throwable ce) {
        System.out.println("Caught an ClientException, which means the client encounter
ed "

            + "a serious internal problem while trying to communicate with OSS, "

```

```
        + "such as not being able to access the network.");
        System.out.println("Error Message:" + ce.getMessage());
    } finally {
        // Shut down the OSSClient instance.
        if (ossClient != null) {
            ossClient.shutdown();
        }
    }
}
```

References

For more information about the complete sample code for resumable upload, visit [GitHub](#).

2.6.6. Multipart upload

By using the multipart upload feature provided by OSS, you can split a large object into multiple parts and upload them separately. After all parts are uploaded, call the CompleteMultipartUpload operation to combine these parts into a single object to implement resumable upload.

Process

To upload an object by using multipart upload, perform the following steps:

1. Initiate a multipart upload task.

Call the `ossClient.initiateMultipartUpload` method to obtain an upload ID that is unique in OSS.

2. Upload parts.

Call the `ossClient.uploadPart` method to upload the parts.

Note

- For parts that are uploaded by a multipart upload task with a specific upload ID, part numbers identify their relative positions in an object. If you upload a part with the same part number as an existing part, the existing part is overwritten by the uploaded part.
- OSS includes the MD5 hash of part data in the ETag header and returns the MD5 hash to the user.
- OSS calculates the MD5 hash of uploaded data and compares the MD5 hash with the MD5 hash calculated by the SDK. If the two hashes are different, the `InvalidDigest` error code is returned.

3. Complete the multipart upload task.

After all parts are uploaded, call the `ossClient.completeMultipartUpload` method to combine these parts into a complete object.

For more information about multipart upload and its applicable scenarios, see [Multipart upload](#).

Complete sample code

The following code provides a complete example that describes the process of multipart upload:

```
import com.aliyun.oss.ClientException;
```

```

import com.aliyun.oss.OSSException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.*;
import java.io.File;
import java.io.FileInputStream;
import java.io.InputStream;
import java.util.ArrayList;
import java.util.List;

public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the name of the bucket. Example: examplebucket.
        String bucketName = "examplebucket";
        // Specify the full path of the object. Example: exampledir/exampleobject.txt. The full path of the object cannot contain the bucket name.
        String objectName = "exampledir/exampleobject.txt";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);

        try {
            // Create an InitiateMultipartUploadRequest object.
            InitiateMultipartUploadRequest request = new InitiateMultipartUploadRequest(bucketName, objectName);
            // The following code provides an example on how to specify the request headers when you initiate a multipart upload task.
            // ObjectMetadata metadata = new ObjectMetadata();
            // metadata.setHeader(OSSHeaders.OSS_STORAGE_CLASS, StorageClass.Standard.toString());

            // Specify the caching behavior of the web page for the object.
            // metadata.setCacheControl("no-cache");
            // Specify the name of the object when the object is downloaded.
            // metadata.setContentDisposition("attachment;filename=oss_MultipartUpload.txt");

            // Specify the encoding format for the content of the object.
            // metadata.setContentEncoding(OSSConstants.DEFAULT_CHARSET_NAME);
            // Specify whether existing objects are overwritten by objects with the same names when the multipart upload task is initiated. In this example, this parameter is set to true, which indicates that the object with the same name cannot be overwritten.
            // metadata.setHeader("x-oss-forbid-overwrite", "true");
            // Specify the server-side encryption method that is used to encrypt each part of the object to upload.
            // metadata.setHeader(OSSHeaders.OSS_SERVER_SIDE_ENCRYPTION, ObjectMetadata.KMS_SERVER_SIDE_ENCRYPTION);
            // Specify the encryption algorithm that is used to encrypt the object. If you do not configure this parameter, objects are encrypted by using AES-256.

```

```

        // metadata.setHeader(OSSHeaders.OSS_SERVER_SIDE_DATA_ENCRYPTION, ObjectMetadata.
a.KMS_SERVER_SIDE_ENCRYPTION);
        // Specify the ID of the customer master key (CMK) that is managed by Key Manag
ement Service (KMS).
        // metadata.setHeader(OSSHeaders.OSS_SERVER_SIDE_ENCRYPTION_KEY_ID, "9468da86-3
509-4f8d-a61e-6eableac****");
        // Specify the storage class of the object.
        // metadata.setHeader(OSSHeaders.OSS_STORAGE_CLASS, StorageClass.Standard);
        // Configure tagging for the object. You can specify multiple tags for the obje
ct at the same time.
        // metadata.setHeader(OSSHeaders.OSS_TAGGING, "a:1");
        // request.setObjectMetadata(metadata);
        // Initiate a multipart copy task.
        InitiateMultipartUploadResult upresult = ossClient.initiateMultipartUpload(requ
est);

        // Obtain the upload ID, which uniquely identifies the multipart upload task. Y
ou can use the upload ID to cancel or query the multipart upload task.
        String uploadId = upresult.getUploadId();
        // partETags is the set of PartETags. A PartETag consists of the part number an
d ETag of an uploaded part
        List<PartETag> partETags = new ArrayList<PartETag>();
        // The size of each part, which is used to calculate the number of parts of the
object. Unit: bytes.
        final long partSize = 1 * 1024 * 1024L;    // Set the part size to 1 MB.
        // Specify the full path of the local file that you want to upload. By default,
if you do not specify the full path of the local file, the local file is uploaded from the
path of the project to which the sample program belongs.
        final File sampleFile = new File("D:\\localpath\\examplefile.txt");
        long fileLength = sampleFile.length();
        int partCount = (int) (fileLength / partSize);
        if (fileLength % partSize != 0) {
            partCount++;
        }
        // Upload each part until all parts are uploaded.
        for (int i = 0; i < partCount; i++) {
            long startPos = i * partSize;
            long curPartSize = (i + 1 == partCount) ? (fileLength - startPos) : partSiz
e;

            InputStream instream = new FileInputStream(sampleFile);
            // Skip the parts that have been uploaded.
            instream.skip(startPos);
            UploadPartRequest uploadPartRequest = new UploadPartRequest();
            uploadPartRequest.setBucketName(bucketName);
            uploadPartRequest.setKey(objectName);
            uploadPartRequest.setUploadId(uploadId);
            uploadPartRequest.setInputStream(instream);
            // Configure the size available for each part. Each part except for the las
t part must be larger than 100 KB in size.
            uploadPartRequest.setPartSize(curPartSize);
            // Set part numbers. Each part has a part number. The number ranges from 1
to 10000. If the specified number is beyond the range, OSS returns an InvalidArgument error
code.
            uploadPartRequest.setPartNumber(i + 1);
            // Parts are not necessarily uploaded in sequence. They can be uploaded fro

```



```

m different OSS clients. OSS sorts the parts based on their part numbers and combines them
into a complete object.
        UploadPartResult uploadPartResult = ossClient.uploadPart(uploadPartRequest)
;
        // Each time a part is uploaded, OSS returns a result that contains a PartE
Tag. The PartETags are stored in partETags.
        partETags.add(uploadPartResult.getPartETag());
    }
    // Create a CompleteMultipartUploadRequest object.
    // When the multipart upload task is completed, you must provide all valid Part
ETags. After receiving the PartETags, OSS verifies the validity of all parts one by one. Af
ter all parts are verified, OSS combines these parts into a complete object.
    CompleteMultipartUploadRequest completeMultipartUploadRequest =
        new CompleteMultipartUploadRequest(bucketName, objectName, uploadId, pa
rtETags);
    // Optional. The following code provides an example on how to set the access co
ntrol list (ACL) of the object.
    // completeMultipartUploadRequest.setObjectACL(CannedAccessControlList.Private)
;
    // Specifies whether to list all parts that are uploaded by using the current u
pload ID. If you want to combine the parts by listing the parts in the server side, you hav
e the option to leave partETags contained in CompleteMultipartUploadRequest empty.
    // Map<String, String> headers = new HashMap<String, String>();
    // If x-oss-complete-all:yes is specified in the request, OSS lists all parts t
hat are uploaded by using the current upload ID, sorts the parts by part number, and then p
erforms the CompleteMultipartUpload operation.
    // If you configure x-oss-complete-all:yes in the request, the request body can
not be specified. Otherwise, an error occurs.
    // headers.put("x-oss-complete-all", "yes");
    // completeMultipartUploadRequest.setHeaders(headers);
    // Complete the multipart upload task.
    CompleteMultipartUploadResult completeMultipartUploadResult = ossClient.complet
eMultipartUpload(completeMultipartUploadRequest);
    System.out.println(completeMultipartUploadResult.getETag());
} catch (OSSException oe) {
    System.out.println("Caught an OSSException, which means your request made it to
OSS, "
        + "but was rejected with an error response for some reason.");
    System.out.println("Error Message:" + oe.getErrorMessage());
    System.out.println("Error Code:" + oe.getErrorCode());
    System.out.println("Request ID:" + oe.getRequestId());
    System.out.println("Host ID:" + oe.getHostId());
} catch (ClientException ce) {
    System.out.println("Caught an ClientException, which means the client encounter
ed "
        + "a serious internal problem while trying to communicate with OSS, "
        + "such as not being able to access the network.");
    System.out.println("Error Message:" + ce.getMessage());
} finally {
    if (ossClient != null) {
        ossClient.shutdown();
    }
}
}

```

```
}
```

Cancel a multipart upload task

You can call the `ossClient.abortMultipartUpload` method to cancel a multipart upload task. If you cancel a multipart upload task, you cannot use the upload ID to upload parts. The uploaded parts are deleted.

The following code provides an example on how to cancel a multipart upload task:

```

import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.*;
public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the name of the bucket. Example: examplebucket.
        String bucketName = "examplebucket";
        // Specify the full path of the object. Example: exampledir/exampleobject.txt. The full path of the object cannot contain the bucket name.
        String objectName = "exampledir/exampleobject.txt";
        // Specify the upload ID. Example: 0004B999EF518A1FE585B0C9360D****. The upload ID is obtained from the response of InitiateMultipartUpload.
        String uploadId = "0004B999EF518A1FE585B0C9360D****";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);
        try {
            // Cancel the multipart upload task.
            AbortMultipartUploadRequest abortMultipartUploadRequest =
                new AbortMultipartUploadRequest(bucketName, objectName, uploadId);
            ossClient.abortMultipartUpload(abortMultipartUploadRequest);
        } catch (OSSException oe) {
            System.out.println("Caught an OSSException, which means your request made it to OSS, "
                + "but was rejected with an error response for some reason.");
            System.out.println("Error Message:" + oe.getErrorMessage());
            System.out.println("Error Code:" + oe.getErrorCode());
            System.out.println("Request ID:" + oe.getRequestId());
            System.out.println("Host ID:" + oe.getHostId());
        } catch (ClientException ce) {
            System.out.println("Caught an ClientException, which means the client encountered "
                + "a serious internal problem while trying to communicate with OSS, "
                + "such as not being able to access the network.");
            System.out.println("Error Message:" + ce.getMessage());
        } finally {
            if (ossClient != null) {
                ossClient.shutdown();
            }
        }
    }
}

```

List uploaded parts

You can call the `ossClient.listParts` method to list all parts that are uploaded by using a specific upload ID.

- List uploaded parts by using simple list

The following code provides an example on how to use simple list to list uploaded parts:

```
import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.*;
public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify
        // your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud acc
        // ount to access OSS because the account has permissions on all API operations. We recommen
        // d that you use a RAM user to call API operations or perform routine O&M. To create a RAM
        // user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the name of the bucket. Example: examplebucket.
        String bucketName = "examplebucket";
        // Specify the full path of the object. Example: exampledir/exampleobject.txt. Th
        // e full path of the object cannot contain the bucket name.
        String objectName = "exampledir/exampleobject.txt";
        // Specify the upload ID. Example: 0004B999EF518A1FE585B0C9360D****. The upload I
        // D is obtained from the response of InitiateMultipartUpload.
        String uploadId = "0004B999EF518A1FE585B0C9360D****";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecr
        et);
        try {
            // List the uploaded parts.
            ListPartsRequest listPartsRequest = new ListPartsRequest(bucketName, objectNa
            me, uploadId);
            // Set uploadId.
            //listPartsRequest.setUploadId(uploadId);
            // Set the maximum number of parts that can be displayed on each page to 100.
            // By default, 1,000 parts are listed.
            listPartsRequest.setMaxParts(100);
            // Specify the start position of the list. Only the parts whose part numbers
            // are greater than the value of this parameter are listed.
            listPartsRequest.setPartNumberMarker(2);
            PartListing partListing = ossClient.listParts(listPartsRequest);
            for (PartSummary part : partListing.getParts()) {
                // Obtain the part numbers.
                part.getPartNumber();
                // Obtain the part size of each part.
                part.getSize();
                // Obtain the ETag of each part.
                part.getETag();
            }
        } catch (OSSException oe) {
            System.out.println("An exception occurred: " + oe.getMessage());
            oe.printStackTrace();
        } catch (ClientException ce) {
            System.out.println("An exception occurred: " + ce.getMessage());
            ce.printStackTrace();
        }
    }
}
```

```

        part.getETag();
        // Obtain the last modified time of each part.
        part.getLastModified();
    }
} catch (OSSEException oe) {
    System.out.println("Caught an OSSEException, which means your request made it
to OSS, "
        + "but was rejected with an error response for some reason.");
    System.out.println("Error Message:" + oe.getErrorMessage());
    System.out.println("Error Code:" + oe.getErrorCode());
    System.out.println("Request ID:" + oe.getRequestId());
    System.out.println("Host ID:" + oe.getHostId());
} catch (ClientException ce) {
    System.out.println("Caught an ClientException, which means the client encount
ered "
        + "a serious internal problem while trying to communicate with OSS, "
        + "such as not being able to access the network.");
    System.out.println("Error Message:" + ce.getMessage());
} finally {
    if (ossClient != null) {
        ossClient.shutdown();
    }
}
}
}

```

- List all uploaded parts

By default, listParts can list up to 1,000 parts at a time. To list more than 1,000 uploaded parts, use the following code:

```

import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSEException;
import com.aliyun.oss.model.*;
public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify
your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud acc
ount to access OSS because the account has permissions on all API operations. We recommen
d that you use a RAM user to call API operations or perform routine O&M. To create a RAM
user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the name of the bucket. Example: examplebucket.
        String bucketName = "examplebucket";
        // Specify the full path of the object. Example: exampledir/exampleobject.txt. Th
e full path of the object cannot contain the bucket name.
        String objectName = "exampledir/exampleobject.txt";
        // Specify the upload ID. Example: 0004B999EF518A1FE585B0C9360D****. The upload I
D is obtained from the response of InitiateMultipartUpload.
        String uploadId = "0004B999EF518A1FE585B0C9360D****";
    }
}

```

```

// Create an OSSClient instance.
OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);

try {
    // List all uploaded parts.
    PartListing partListing;
    ListPartsRequest listPartsRequest = new ListPartsRequest(bucketName, objectName, uploadId);

    do {
        partListing = ossClient.listParts(listPartsRequest);
        for (PartSummary part : partListing.getParts()) {
            // Obtain the part numbers.
            part.getPartNumber();
            // Obtain the part size of each part.
            part.getSize();
            // Obtain the ETag of each part.
            part.getETag();
            // Obtain the last modified time of each part.
            part.getLastModified();
        }

        // Specify the start position of the list. Only the parts whose part numbers are greater than the value of this parameter are listed.
        listPartsRequest.setPartNumberMarker(partListing.getNextPartNumberMarker());

    } while (partListing.isTruncated());
} catch (OSSException oe) {
    System.out.println("Caught an OSSException, which means your request made it to OSS, "
        + "but was rejected with an error response for some reason.");
    System.out.println("Error Message:" + oe.getErrorMessage());
    System.out.println("Error Code:" + oe.getErrorCode());
    System.out.println("Request ID:" + oe.getRequestId());
    System.out.println("Host ID:" + oe.getHostId());
} catch (ClientException ce) {
    System.out.println("Caught an ClientException, which means the client encountered "
        + "a serious internal problem while trying to communicate with OSS, "
        + "such as not being able to access the network.");
    System.out.println("Error Message:" + ce.getMessage());
} finally {
    if (ossClient != null) {
        ossClient.shutdown();
    }
}
}

```

- List all uploaded parts by page

The following code provides an example on how to list all uploaded parts by page and specify the maximum number of parts to list per page:

```

import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;

```

```

import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.*;
public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify
        // your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud acc
        // ount to access OSS because the account has permissions on all API operations. We recommen
        // d that you use a RAM user to call API operations or perform routine O&M. To create a RAM
        // user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the name of the bucket. Example: examplebucket.
        String bucketName = "examplebucket";
        // Specify the full path of the object. Example: exampledir/exampleobject.txt. Th
        // e full path of the object cannot contain the bucket name.
        String objectName = "exampledir/exampleobject.txt";
        // Specify the upload ID. Example: 0004B999EF518A1FE585B0C9360D****. The upload I
        // D is obtained from the response of InitiateMultipartUpload.
        String uploadId = "0004B999EF518A1FE585B0C9360D****";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecr
        et);

        try {
            // List all uploaded parts by page.
            PartListing partListing;
            ListPartsRequest listPartsRequest = new ListPartsRequest(bucketName, objectNa
            me, uploadId);
            // Set the maximum number of parts to list per page to 100.
            listPartsRequest.setMaxParts(100);
            do {
                partListing = ossClient.listParts(listPartsRequest);
                for (PartSummary part : partListing.getParts()) {
                    // Obtain the part numbers.
                    part.getPartNumber();
                    // Obtain the part size of each part.
                    part.getSize();
                    // Obtain the ETag of each part.
                    part.getETag();
                    // Obtain the last modified time of each part.
                    part.getLastModified();
                }
                listPartsRequest.setPartNumberMarker(partListing.getNextPartNumberMarker(
            ));
            } while (partListing.isTruncated());
        } catch (OSSException oe) {
            System.out.println("Caught an OSSException, which means your request made it
            to OSS, "
                + "but was rejected with an error response for some reason.");
            System.out.println("Error Message:" + oe.getErrorMessage());
            System.out.println("Error Code:" + oe.getErrorCode());
            System.out.println("Request ID:" + oe.getRequestId());
            System.out.println("Host ID:" + oe.getHostId());
        } catch (ClientException ce) {

```

```

    } catch (ClientException ce) {
        System.out.println("Caught an ClientException, which means the client encount
ered "
            + "a serious internal problem while trying to communicate with OSS, "
            + "such as not being able to access the network.");
        System.out.println("Error Message:" + ce.getMessage());
    } finally {
        if (ossClient != null) {
            ossClient.shutdown();
        }
    }
}
}
}

```

List multipart upload tasks

You can call the `ossClient.listMultipartUploads` method to list all ongoing multipart upload tasks. Ongoing multipart upload tasks refer to tasks that are initiated but incomplete and tasks that are canceled. The following table describes the parameters that you can configure to list multipart upload tasks.

Parameter	Description	Method
prefix	The prefix that the names of returned objects must contain. If you use a prefix for a query, the returned object names contain the prefix.	<code>ListMultipartUploadsRequest.setPrefix(String prefix)</code>
delimiter	The character used to group object names. The substring between the specified prefix and the occurrence of the first delimiter. A substring of object names starts with the prefix and ends with the next occurrence of the specified delimiter.	<code>ListMultipartUploadsRequest.setDelimiter(String delimiter)</code>
maxUploads	The maximum number of multipart upload tasks to list in this query. The default value is 1000. The maximum value is 1000.	<code>ListMultipartUploadsRequest.setMaxUploads(Integer maxUploads)</code>
keyMarker	Specifies the name of the object after which the listing of multipart upload tasks begins. All multipart upload tasks in which objects whose names start with a letter that is alphabetically after the <code>keyMarker</code> parameter value are listed. You can use this parameter with the <code>uploadIdMarker</code> parameter to specify the start point to list the returned results.	<code>ListMultipartUploadsRequest.setKeyMarker(String keyMarker)</code>

Parameter	Description	Method
uploadIdMarker	UploadIdMarker is used with the keyMarker parameter to specify the start position of the list. If the keyMarker parameter is not configured, the uploadIdMarker parameter does not take effect. If the keyMarker parameter is specified, the result includes: - All objects whose names start with a letter that is alphabetically after the keyMarker parameter value in alphabetical order. - All objects whose names start with a letter that is the same as the keyMarker parameter value in alphabetical order and of which the uploadId value is greater than the uploadIdMarker value.	ListMultipartUploadsRequest.setUploadIdMarker(String uploadIdMarker)

- List multipart upload tasks by using simple list

The following code provides an example on how to use simple list to list multipart upload tasks:

```
import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.*;
public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify
        // your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud acc
        // ount to access OSS because the account has permissions on all API operations. We recommen
        // d that you use a RAM user to call API operations or perform routine O&M. To create a RAM
        // user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the name of the bucket. Example: examplebucket.
        String bucketName = "examplebucket";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecr
        et);
        try {
            // List multipart upload tasks. By default, 1,000 parts are listed.
            ListMultipartUploadsRequest listMultipartUploadsRequest = new ListMultipartUp
            loadsRequest(bucketName);
            MultipartUploadListing multipartUploadListing = ossClient.listMultipartUpload
            s(listMultipartUploadsRequest);
            for (MultipartUpload multipartUpload : multipartUploadListing.getMultipartUp
            loads()) {
                // Obtain the upload ID of each multipart upload task.
                multipartUpload.getUploadId();
                // Obtain the object names.
                multipartUpload.getKey();
                // Obtain the point in time when each multipart upload task is initiated.
            }
        }
    }
}
```

```

        multipartUpload.getInitiated();
    }
} catch (OSSEException oe) {
    System.out.println("Caught an OSSEException, which means your request made it
to OSS, "
        + "but was rejected with an error response for some reason.");
    System.out.println("Error Message:" + oe.getErrorMessage());
    System.out.println("Error Code:" + oe.getErrorCode());
    System.out.println("Request ID:" + oe.getRequestId());
    System.out.println("Host ID:" + oe.getHostId());
} catch (ClientException ce) {
    System.out.println("Caught an ClientException, which means the client encount
ered "
        + "a serious internal problem while trying to communicate with OSS, "
        + "such as not being able to access the network.");
    System.out.println("Error Message:" + ce.getMessage());
} finally {
    if (ossClient != null) {
        ossClient.shutdown();
    }
}
}
}
}

```

If the value of the `isTruncated` field in the returned result is false, values of `nextKeyMarker` and `nextUploadIdMarker` are returned and used as the start position of the next reading. If the response does not contain all multipart upload tasks, list multipart upload tasks by page.

- List all multipart upload tasks

By default, `listMultipartUploads` can list up to 1,000 parts at a time. The following code provides an example on how to list all multipart upload tasks when the number of parts is larger than 1,000:

```

import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSEException;
import com.aliyun.oss.model.*;
public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify
        your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud acc
        ount to access OSS because the account has permissions on all API operations. We recommen
        d that you use a RAM user to call API operations or perform routine O&M. To create a RAM
        user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the name of the bucket. Example: examplebucket.
        String bucketName = "examplebucket";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecr
        et);
    }
}

```

```

        try {
            // List multipart upload tasks.
            MultipartUploadListing multipartUploadListing;
            ListMultipartUploadsRequest listMultipartUploadsRequest = new ListMultipartUp
loadsRequest(bucketName);
            do {
                multipartUploadListing = ossClient.listMultipartUploads(listMultipartUplo
adsRequest);
                for (MultipartUpload multipartUpload : multipartUploadListing.getMultipart
Uploads()) {
                    // Obtain the upload ID of each multipart upload task.
                    multipartUpload.getUploadId();
                    // Obtain the object names.
                    multipartUpload.getKey();
                    // Obtain the point in time when each multipart upload task is initia
ted.
                    multipartUpload.getInitiated();
                }
                listMultipartUploadsRequest.setKeyMarker(multipartUploadListing.getNextKe
yMarker());
                listMultipartUploadsRequest.setUploadIdMarker(multipartUploadListing.getN
extUploadIdMarker());
            } while (multipartUploadListing.isTruncated());
        } catch (OSSException oe) {
            System.out.println("Caught an OSSException, which means your request made it
to OSS, "
                + "but was rejected with an error response for some reason.");
            System.out.println("Error Message:" + oe.getErrorMessage());
            System.out.println("Error Code:" + oe.getErrorCode());
            System.out.println("Request ID:" + oe.getRequestId());
            System.out.println("Host ID:" + oe.getHostId());
        } catch (ClientException ce) {
            System.out.println("Caught an ClientException, which means the client encount
ered "
                + "a serious internal problem while trying to communicate with OSS, "
                + "such as not being able to access the network.");
            System.out.println("Error Message:" + ce.getMessage());
        } finally {
            if (ossClient != null) {
                ossClient.shutdown();
            }
        }
    }
}

```

- List multipart upload tasks on one or more pages

The following code provides an example on how to list all multipart upload tasks by page:

```

import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.*;
public class Demo {

```

```

    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify
        // your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud acc
        // ount to access OSS because the account has permissions on all API operations. We recommen
        // d that you use a RAM user to call API operations or perform routine O&M. To create a RAM
        // user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the name of the bucket. Example: examplebucket.
        String bucketName = "examplebucket";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecr
        et);

        try {
            // List multipart upload tasks.
            MultipartUploadListing multipartUploadListing;
            ListMultipartUploadsRequest listMultipartUploadsRequest = new ListMultipartUp
            loadsRequest(bucketName);
            // Specify the number of multipart upload tasks to list on each page.
            listMultipartUploadsRequest.setMaxUploads(50);
            do {
                multipartUploadListing = ossClient.listMultipartUploads(listMultipartUplo
                adsRequest);
                for (MultipartUpload multipartUpload : multipartUploadListing.getMultipart
                Uploads()) {
                    // Obtain the upload ID of each multipart upload task.
                    multipartUpload.getUploadId();
                    // Obtain the object names.
                    multipartUpload.getKey();
                    // Obtain the point in time when each multipart upload task is initia
                    ted.
                    multipartUpload.getInitiated();
                }
                listMultipartUploadsRequest.setKeyMarker(multipartUploadListing.getNextKe
                yMarker());
                listMultipartUploadsRequest.setUploadIdMarker(multipartUploadListing.getN
                extUploadIdMarker());
            } while (multipartUploadListing.isTruncated());
        } catch (OSSException oe) {
            System.out.println("Caught an OSSException, which means your request made it
            to OSS, "
                + "but was rejected with an error response for some reason.");
            System.out.println("Error Message:" + oe.getErrorMessage());
            System.out.println("Error Code:" + oe.getErrorCode());
            System.out.println("Request ID:" + oe.getRequestId());
            System.out.println("Host ID:" + oe.getHostId());
        } catch (ClientException ce) {
            System.out.println("Caught an ClientException, which means the client encount
            ered "
                + "a serious internal problem while trying to communicate with OSS, "
                + "such as not being able to access the network.");
            System.out.println("Error Message:" + ce.getMessage());
        }
    }

```

```
        } finally {
            if (ossClient != null) {
                ossClient.shutdown();
            }
        }
    }
}
```

References

- For more information about the complete sample code for multipart upload, visit [GitHub](#).
- The following API operations are required to perform multipart upload:
 - The API operation that you can call to initiate a multipart upload task. For more information, see [InitiateMultipartUpload](#).
 - The API operation that you can call to upload data by part. For more information, see [UploadPart](#).
 - The API operation that you can call to complete a multipart upload task. For more information, see [CompleteMultipartUpload](#).
- For more information about the API operation that you can call to cancel a multipart upload task, see [AbortMultipartUpload](#).
- For more information about the API operation that you can call to list uploaded parts, see [ListParts](#).
- For more information about the API operation that you can call to list ongoing multipart upload tasks, see [ListMultipartUploads](#). Ongoing multipart upload tasks refer to tasks that have been initiated but are not completed and tasks that are canceled.

2.6.7. Progress bars

You can use a progress bar to indicate the progress of an object that is being uploaded or downloaded. This topic describes how to use a progress bar when you use the `ossClient.putObject` method to upload an object.

Sample code

The following code provides an example on how to use a progress bar when you use the `ossClient.putObject` method to upload an object.

 **Note** The following methods support the progress bar feature: `ossClient.putObject`, `ossClient.getObject`, `ossClient.uploadPart`, `ossClient.uploadFile`, and `ossClient.downloadFile`. The procedure to apply progress bars to these methods is the same as that of the `ossClient.putObject` method.

```
import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.event.ProgressEvent;
import com.aliyun.oss.event.ProgressEventType;
import com.aliyun.oss.event.ProgressListener;
import com.aliyun.oss.model.PutObjectRequest;
import java.io.File;

public class PutObjectProgressListener implements ProgressListener {
```

```

private long bytesWritten = 0;
private long totalBytes = -1;
private boolean succeed = false;
public static void main(String[] args) throws Exception {
    // In this example, the endpoint of the China (Hangzhou) region is used. Specify the actual endpoint.
    String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
    // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
    String accessKeyId = "yourAccessKeyId";
    String accessKeySecret = "yourAccessKeySecret";
    // Specify the name of the bucket. Example: examplebucket.
    String bucketName = "examplebucket";
    // Specify the full path of the object. Example: exampledir/exampleobject.txt. The full path of the object cannot contain the bucket name.
    String objectName = "exampledir/exampleobject.txt";
    String pathName = "yourLocalFile";
    // Create an OSSClient instance.
    OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);

    try {
        // Specify the parameter of the progress bar when you upload the object.
        ossClient.putObject(new PutObjectRequest(bucketName, objectName, new File(pathName)),
            <PutObjectRequest>withProgressListener(new PutObjectProgressListener()));
    } catch (OSSException oe) {
        System.out.println("Caught an OSSException, which means your request made it to OSS, "
            + "but was rejected with an error response for some reason.");
        System.out.println("Error Message:" + oe.getErrorMessage());
        System.out.println("Error Code:" + oe.getErrorCode());
        System.out.println("Request ID:" + oe.getRequestId());
        System.out.println("Host ID:" + oe.getHostId());
    } catch (ClientException ce) {
        System.out.println("Caught an ClientException, which means the client encountered "
            + "a serious internal problem while trying to communicate with OSS, "
            + "such as not being able to access the network.");
        System.out.println("Error Message:" + ce.getMessage());
    } finally {
        if (ossClient != null) {
            ossClient.shutdown();
        }
    }
}

public boolean isSucceed() {
    return succeed;
}

@Override
public void progressChanged(ProgressEvent progressEvent) {
    long bytes = progressEvent.getBytes();
    ProgressEventTime currentTime = progressEvent.getCurrentTime();

```

```

        ProgressEventType eventType = progressEvent.getEventType();
        switch (eventType) {
            case TRANSFER_STARTED_EVENT:
                System.out.println("Start to upload.....");
                break;
            case REQUEST_CONTENT_LENGTH_EVENT:
                this.totalBytes = bytes;
                System.out.println(this.totalBytes + " bytes in total will be uploaded to OSS");
                break;
            case REQUEST_BYTE_TRANSFER_EVENT:
                this.bytesWritten += bytes;
                if (this.totalBytes != -1) {
                    int percent = (int)(this.bytesWritten * 100.0 / this.totalBytes);
                    System.out.println(bytes + " bytes have been written at this time, upload progress: " + percent + "%(" + this.bytesWritten + "/" + this.totalBytes + ")");
                } else {
                    System.out.println(bytes + " bytes have been written at this time, upload ratio: unknown" + "(" + this.bytesWritten + "/...");
                }
                break;
            case TRANSFER_COMPLETED_EVENT:
                this.succeed = true;
                System.out.println("Succeed to upload, " + this.bytesWritten + " bytes have been transferred in total");
                break;
            case TRANSFER_FAILED_EVENT:
                System.out.println("Failed to upload, " + this.bytesWritten + " bytes have been transferred");
                break;
            default:
                break;
        }
    }
}

```

References

For the complete sample code of progress bars in object upload, visit [GitHub](#).

2.6.8. Upload callback

After an object is uploaded, Object Storage Service (OSS) can send a callback request to the application server. To implement callback, you need only to send a request that contains relevant callback parameters to OSS.

Examples

The following code provides an example on how to configure upload callback:

```

import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.*;

```

```

import java.io.ByteArrayInputStream;
public class Demo {
    public static void main(String[] args) {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify the actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the name of the bucket. Example: examplebucket.
        String bucketName = "examplebucket";
        // Specify the full path of the object. Example: exampledir/exampleobject.txt. The full path of the object cannot contain the bucket name.
        String objectName = "exampledir/exampleobject.txt";
        // Specify the address of the server to which the callback request is sent. Example: https://example.com:23450 or https://127.0.0.1:9090.
        String callbackUrl = "yourCallbackServerUrl";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);

        try {
            String content = "Hello OSS";
            PutObjectRequest putObjectRequest = new PutObjectRequest(bucketName, objectName, new ByteArrayInputStream(content.getBytes()));
            // Upload callback parameters.
            Callback callback = new Callback();
            callback.setCallbackUrl(callbackUrl);
            // Optional. Set the value of the Host field included in the callback request header.
            // callback.setCallbackHost("yourCallbackHost");
            // Set the value of the body field included in the callback request.
            callback.setCallbackBody("{\"mime\\\": \"${mimeType}\", \"size\\\": \"${size}\"}");

            // Set Content-Type for the callback request.
            callback.setCallbackBodyType(Callback.CallbackBodyType.JSON);
            // Configure custom parameters for the callback request. Each custom parameter consists of a key and a value. The key must start with x:.
            callback.addCallbackVar("x:var1", "value1");
            callback.addCallbackVar("x:var2", "value2");
            putObjectRequest.setCallback(callback);
            PutObjectResult putObjectResult = ossClient.putObject(putObjectRequest);
            // Read the message content returned from upload callback.
            byte[] buffer = new byte[1024];
            putObjectResult.getResponse().getContent().read(buffer);
            // You must close the stream after the data is read. Otherwise, connection leaks may occur. Consequently, no connections are available and an exception occurs.
            putObjectResult.getResponse().getContent().close();
        } catch (OSSException oe) {
            System.out.println("Caught an OSSException, which means your request made it to OSS, "
                + "but was rejected with an error response for some reason.");
        }
    }
}

```



```
        System.out.println("Error Message:" + oe.getMessage());
        System.out.println("Error Code:" + oe.getErrorCode());
        System.out.println("Request ID:" + oe.getRequestId());
        System.out.println("Host ID:" + oe.getHostId());
    } catch (Throwable ce) {
        System.out.println("Caught an ClientException, which means the client encounter  
ed "
            + "a serious internal problem while trying to communicate with OSS, "  
            + "such as not being able to access the network.");
        System.out.println("Error Message:" + ce.getMessage());
    } finally {
        if (ossClient != null) {
            ossClient.shutdown();
        }
    }
}
```

References

- For more information about the complete sample code for upload callback, visit [GitHub](#).
- For more information about the API operation that is called to configure upload callback, see [Callback](#).

2.7. Download objects

2.7.1. Overview

This topic is the overview of object download in OSS Java SDK.

OSS Java SDK provides the following download methods:

- [Streaming download](#)
- [Download to your local file](#)
- [Range download](#)
- [Resumable download](#)
- [Conditional download](#)

You can view the download progress shown in the [download progress bar](#).

2.7.2. Streaming download

If you want to download a large object or it takes a long time to download an object at a time, you can use streaming download to download the object incrementally until the entire object is downloaded.

Examples

The following code provides an example on how to use streaming download to download an object named exampleobject.txt from a bucket named examplebucket:

```
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
```

```

import com.aliyun.oss.OSSClient;
import com.aliyun.oss.model.*;
import java.io.BufferedReader;
import java.io.InputStreamReader;
public class Demo {
    public static void main(String[] args) {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify the
        // actual endpoint. For more information about the endpoints of other regions, see Regions and endpoints.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account
        // to access Object Storage Service (OSS) because the account has permissions on all API operations.
        // We recommend that you use a RAM user to call API operations or perform routine O&M.
        // To create a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the name of the bucket. Example: examplebucket.
        String bucketName = "examplebucket";
        // Specify the full path of the object. Example: exampledir/exampleobject.txt. The
        // full path of the object cannot contain the bucket name.
        String objectName = "exampledir/exampleobject.txt";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);

        try {
            // ossObject specifies the bucket name, object name, object metadata, and an input stream.
            OSSObject ossObject = ossClient.getObject(bucketName, objectName);

            // Read the object content.
            System.out.println("Object content:");
            BufferedReader reader = new BufferedReader(new InputStreamReader(ossObject.getObjectContent()));
            while (true) {
                String line = reader.readLine();
                if (line == null) break;
                System.out.println("\n" + line);
            }
            // You must close the stream after the data is read. Otherwise, connection leaks may occur.
            // Consequently, no connections are available and an exception occurs.
            reader.close();
            // If you do not close the ossObject object after it is used, connection leaks may occur.
            // Consequently, no connections are available and an exception occurs.
            ossObject.close();
        } catch (OSSException oe) {
            System.out.println("Caught an OSSException, which means your request made it to OSS, "
                + "but was rejected with an error response for some reason.");
            System.out.println("Error Message:" + oe.getErrorMessage());
            System.out.println("Error Code:" + oe.getErrorCode());
            System.out.println("Request ID:" + oe.getRequestId());
            System.out.println("Host ID:" + oe.getHostId());
        } catch (Throwable ce) {
            System.out.println("Caught an ClientException, which means the client encountered "

```

```
        + "a serious internal problem while trying to communicate with OSS, "
        + "such as not being able to access the network.");
    System.out.println("Error Message:" + ce.getMessage());
} finally {
    if (ossClient != null) {
        ossClient.shutdown();
    }
}
}
```

References

- For more information about the complete sample code for streaming download, visit [GitHub](#).
- For more information about the API operation that is called to perform streaming download, see [GetObject](#).

2.7.3. Download objects to local computers

This topic describes how to download an object from a bucket to your computer.

Sample code

The following code provides an example on how to download `exampleobject.txt` from the `testfolder` directory of the `examplebucket` bucket to your local computer and save it to `D:\localpath` as `examplefile.txt`.

```
package com.aliyun.oss.demo;
import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.GetObjectRequest;
import java.io.File;
public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify the actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the name of the bucket. Example: examplebucket.
        String bucketName = "examplebucket";
        // Specify the full path of the object. The full path of the object cannot contain the bucket name. Example: testfolder/exampleobject.txt.
        String objectName = "testfolder/exampleobject.txt";
        String pathName = "D:\\localpath\\examplefile.txt";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret
```

```

);
    try {
        // Download the object to the specified path on your computer. If a file with the
        // same name already exists, the downloaded object overwrites the file. Otherwise, a new file
        // is created.
        // If the path is not specified, the downloaded object is saved to the path of
        // the project to which the sample program belongs.
        ossClient.getObject(new GetObjectRequest(bucketName, objectName), new File(path
        Name));
    } catch (OSSException oe) {
        System.out.println("Caught an OSSException, which means your request made it to
        OSS, "
            + "but was rejected with an error response for some reason.");
        System.out.println("Error Message:" + oe.getErrorMessage());
        System.out.println("Error Code:" + oe.getErrorCode());
        System.out.println("Request ID:" + oe.getRequestId());
        System.out.println("Host ID:" + oe.getHostId());
    } catch (ClientException ce) {
        System.out.println("Caught an ClientException, which means the client encounter
        ed "
            + "a serious internal problem while trying to communicate with OSS, "
            + "such as not being able to access the network.");
        System.out.println("Error Message:" + ce.getMessage());
    } finally {
        if (ossClient != null) {
            ossClient.shutdown();
        }
    }
}
}
}

```

References

- For more information about the complete sample code that is used to download objects to your computer, visit [GitHub](#).
- For more information about the API operation that you can call to download objects to your computer, see [GetObject](#).

2.7.4. Range download

You can use range download to download parts of the content of an object.

Specify a valid range to download data

The following code provides an example on how to specify a valid range to download data:

```

import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.GetObjectRequest;
import com.aliyun.oss.model.OSSObject;
import java.io.InputStream;

public class Demo {

```

```

public static void main(String[] args) throws Exception {
    // In this example, the endpoint of the China (Hangzhou) region is used. Specify the
    // actual endpoint. For more information about the endpoints of other regions, see Regions and endpoints.
    String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account
    // to access Object Storage Service (OSS) because the account has permissions on all API operations.
    // We recommend that you use a RAM user to call API operations or perform routine O&M.
    // To create a RAM user, log on to the RAM console.
    String accessKeyId = "yourAccessKeyId";
    String accessKeySecret = "yourAccessKeySecret";
    // Specify the name of the bucket. Example: examplebucket.
    String bucketName = "examplebucket";
    // Specify the full path of the object. Example: exampledir/exampleobject.txt. The
    // full path of the object cannot contain the bucket name.
    String objectName = "exampledir/exampleobject.txt";
    // Create an OSSClient instance.
    OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);

    InputStream in = null;
    try {
        GetObjectRequest getObjectRequest = new GetObjectRequest(bucketName, objectName);

        // For an object whose size is 1,000 bytes, the valid range is from byte 0 to byte 999.
        // Query the data that is within the range from byte 0 to byte 999, which includes
        // a total of 1,000 bytes. If the specified range is invalid, for example, the specified
        // range includes a negative number, or the specified value is larger than the object size,
        // the entire object is downloaded.
        getObjectRequest.setRange(0, 999);
        // Start range download.
        OSSObject ossObject = ossClient.getObject(getObjectRequest);
        // Read data.
        byte[] buf = new byte[1024];
        in = ossObject.getObjectContent();
        for (int n = 0; n != -1; ) {
            n = in.read(buf, 0, buf.length);
        }
    } catch (OSSException oe) {
        System.out.println("Caught an OSSException, which means your request made it to
        OSS, "
            + "but was rejected with an error response for some reason.");
        System.out.println("Error Message:" + oe.getErrorMessage());
        System.out.println("Error Code:" + oe.getErrorCode());
        System.out.println("Request ID:" + oe.getRequestId());
        System.out.println("Host ID:" + oe.getHostId());
    } catch (ClientException ce) {
        System.out.println("Caught an ClientException, which means the client encountered
        "
            + "a serious internal problem while trying to communicate with OSS, "
            + "such as not being able to access the network.");
        System.out.println("Error Message:" + ce.getMessage());
    } finally {
        if (ossClient != null) {

```

```
        ossClient.shutdown();
    }
    // You must close the stream after the data is read. Otherwise, connection leak
    s may occur. Consequently, no connections are available and an exception occurs.
    if (in != null) {
        in.close();
    }
}
}
```

Streaming download may not read all data at a time. To read 64 KB of data from OSS in streaming mode, you can use the following method to read the data multiple times until 64 KB of data or the entire object is read. For more information, visit [InputStream.read](#).

```
byte[] buf = new byte[1024];
InputStream in = ossObject.getObjectContent();
for (int n = 0; n != -1; ) {
    n = in.read(buf, 0, buf.length);
}
in.close();
```

An invalid range is specified

For an object whose size is 1,000 bytes, the valid range is from byte 0 to byte 999. If the specified range is not within the range from byte 0 to byte 999, the range does not take effect. Then, OSS returns HTTP status code 200 and the data of the entire object. The following list provides examples of invalid requests and the returned results:

- If you set Range: bytes to 500-2000, the value at the end of the range is invalid. Then, OSS returns HTTP status code 200 and the data of the entire object.
- If you set Range: bytes to 1000-2000, the value at the start of the range is invalid. Then, OSS returns HTTP status code 200 and the data of the entire object.

Specify compatible behaviors to download data by range

If you add x-oss-range-behavior:standard to the request header, the download behavior is modified when the specified range is not within the valid range. Example: Use range download to download an object whose size is 1,000 bytes.

- If you set Range: bytes to 500-2000, the value at the end of the range is invalid. Then, OSS returns HTTP status code 206 and the data that is within the range from byte 500 to byte 999.
- If you set Range: bytes to 1000-2000, the value at the start of the range is invalid. Then, OSS returns HTTP status code 416 and error code InvalidRange.

The following code provides an example on how to specify compatible behaviors to download data by range:

```
import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.GetObjectRequest;
import com.aliyun.oss.model.OSSObject;
```

```

public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify the
        // actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account
        // to access OSS because the account has permissions on all API operations. We recommend
        // that you use a RAM user to call API operations or perform routine O&M. To create a RAM
        // user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the name of the bucket. Example: examplebucket.
        String bucketName = "examplebucket";
        // Specify the full path of the object. Example: exampledir/exampleobject.txt. The
        // full path of the object cannot contain the bucket name.
        String objectName = "exampledir/exampleobject.txt";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);

        try {
            // Specify the compatible behavior.
            // When the value at the end of the range is invalid, OSS returns the data that
            // is within the range from byte 500 to byte 999. HTTP status code 206 is also returned.
            GetObjectRequest getObjectRequest = new GetObjectRequest(bucketName, objectName);

            getObjectRequest.setRange(500, 2000);
            getObjectRequest.addHeader("x-oss-range-behavior", "standard");
            OSSObject ossObject = ossClient.getObject(getObjectRequest);
            ossObject.close();
            System.out.println("standard get " + "500~2000 " + "statusCode:" + ossObject.getResponse().getStatusCode());
            System.out.println("standard get " + "500~2000 " + "contentLength:" + ossObject.getResponse().getContentLength());
            // When the value at the start of the range is invalid, exceptions are returned
            // for the following code. OSS returns HTTP status code 416 and the InvalidRange error code.
            getObjectRequest = new GetObjectRequest(bucketName, objectName);
            getObjectRequest.setRange(1000, 2000);
            getObjectRequest.addHeader("x-oss-range-behavior", "standard");
            ossClient.getObject(getObjectRequest);
        } catch (OSSException oe) {
            System.out.println("Caught an OSSException, which means your request made it to
            OSS, "
                + "but was rejected with an error response for some reason.");
            System.out.println("Error Message:" + oe.getErrorMessage());
            System.out.println("Error Code:" + oe.getErrorCode());
            System.out.println("Request ID:" + oe.getRequestId());
            System.out.println("Host ID:" + oe.getHostId());
        } catch (ClientException ce) {
            System.out.println("Caught an ClientException, which means the client encountered
            "
                + "a serious internal problem while trying to communicate with OSS, "
                + "such as not being able to access the network.");
            System.out.println("Error Message:" + ce.getMessage());
        } finally {

```

```

        if (ossClient != null) {
            ossClient.shutdown();
        }
    }
}

```

References

For more information about the API operation that is called to perform range download, see [GetObject](#).

2.7.5. Resumable download

You may fail to download a large object if the network is unstable or other exceptions occur. In some cases, you may still fail to download the object even after multiple attempts. To resolve this issue, Object Storage Service (OSS) SDK for Java provides the resumable download feature. In resumable download, the object that you want to download is split into multiple parts and each part is separately downloaded. After all parts are downloaded, these parts are combined into a complete object.

Examples

The following code provides an example on how to implement resumable download:

```

import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.*;

public class Demo {
    public static void main(String[] args) {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify the actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the name of the bucket. Example: examplebucket.
        String bucketName = "examplebucket";
        // Specify the full path of the object. Example: exampledir/exampleobject.txt. The full path of the object cannot contain the bucket name.
        String objectName = "exampledir/exampleobject.txt";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);

        try {
            // Perform resumable download in which 10 parts can be concurrently downloaded.

            DownloadFileRequest downloadFileRequest = new DownloadFileRequest(bucketName, objectName);
            // Specify the full path of the local file. Example: D:\\localpath\\examplefile.txt.
            downloadFileRequest.setDownloadFile("D:\\localpath\\examplefile.txt");

```



```

downloadFileRequest.setObjectMetadata(ETag("examplefile.txt"));
// Specify the size of each part. Unit: bytes. Valid values: 100 KB to 5 GB. De
fault value: 100 KB.
downloadFileRequest.setPartSize(1 * 1024 * 1024);
// Specify the number of threads for the resumable download task. Default value
: 1.
downloadFileRequest.setTaskNum(10);
// Specify whether to enable resumable download. By default, resumable upload i
s disabled.
downloadFileRequest.setEnableCheckpoint(true);
// Specify the full path of the checkpoint file. Example: D:\\localpath\\exampl
efile.txt.dcp.
// After the checkpoint file is generated because the object download is interr
upted, you must set the full path of the checkpoint file only when you want to continue the
download of the object. After the object is downloaded, the checkpoint file is deleted.
//downloadFileRequest.setCheckpointFile("D:\\localpath\\examplefile.txt.dcp");
// Download the object.
DownloadFileResult downloadRes = ossClient.downloadFile(downloadFileRequest);
// After the object is downloaded, the object metadata is returned.
ObjectMetadata objectMetadata = downloadRes.getObjectMetadata();
System.out.println(objectMetadata.getETag());
System.out.println(objectMetadata.getLastModified());
System.out.println(objectMetadata.getUserMetadata().get("meta"));
} catch (OSSException oe) {
    System.out.println("Caught an OSSException, which means your request made it to
OSS, "
        + "but was rejected with an error response for some reason.");
    System.out.println("Error Message:" + oe.getErrorMessage());
    System.out.println("Error Code:" + oe.getErrorCode());
    System.out.println("Request ID:" + oe.getRequestId());
    System.out.println("Host ID:" + oe.getHostId());
} catch (Throwable ce) {
    System.out.println("Caught an ClientException, which means the client encounter
ed "
        + "a serious internal problem while trying to communicate with OSS, "
        + "such as not being able to access the network.");
    System.out.println("Error Message:" + ce.getMessage());
} finally {
    if (ossClient != null) {
        ossClient.shutdown();
    }
}
}
}

```

References

- For more information about the complete sample code for resumable download, visit [GitHub](#).
- For more information about the API operation that you can call to perform resumable download, see [Get Object](#).

2.7.6. Conditional download

You can specify one or more conditions to download objects. If the specified conditions are met, the object is downloaded. If the specified conditions are not met, an error is returned and the object is not downloaded.

Conditions

The following table describes the available object download conditions.

Note

- Both If-Modified-Since and If-Unmodified-Since can exist at the same time as object download conditions. Both If-Match and If-None-Match can exist at the same time as object download conditions.
- You can obtain the ETag by using `ossClient.getObjectMeta`.

Parameter	Description
If-Modified-Since	If the specified time is earlier than the time the object is modified, the object can be downloaded. Otherwise, an error (304 Not modified) is returned.
If-Unmodified-Since	If the specified time is later than or equal to the time the object is modified, the object can be downloaded. Otherwise, an error (412 Precondition failed) is returned.
If-Match	If the specified ETag matches that of an object, the object can be downloaded. Otherwise, an error (412 Precondition failed) is returned.
If-None-Match	If the specified ETag does not match that of an object, the object can be downloaded. Otherwise, an error (304 Not modified) is returned.

Sample code

The following code provides an example on how to use `ossClient.getObject` with download conditions to download an object. You can use `ossClient.downloadFile` with download conditions to download an object in a way that is similar to `ossClient.getObject`.

```
import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.GetObjectRequest;
import java.io.File;
import java.util.Date;
public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify the actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
```

```

String accessKeySecret = "yourAccessKeySecret";
// Specify the name of the bucket. Example: examplebucket.
String bucketName = "examplebucket";
// Specify the full path of the object. The full path of the object cannot contain
the bucket name. Example: testfolder/exampleobject.txt.
String objectName = "testfolder/exampleobject.txt";
String pathName = "D:\\localpath\\examplefile.txt";
// Create an OSSClient instance.
OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);

try {
    GetObjectRequest request = new GetObjectRequest(bucketName, objectName);
    // Configure download conditions.
    request.setModifiedSinceConstraint(new Date());
    // Download an object to your local computer.
    ossClient.getObject(request, new File(pathName));
} catch (OSSException oe) {
    System.out.println("Caught an OSSException, which means your request made it to
OSS, "
        + "but was rejected with an error response for some reason.");
    System.out.println("Error Message:" + oe.getErrorMessage());
    System.out.println("Error Code:" + oe.getErrorCode());
    System.out.println("Request ID:" + oe.getRequestId());
    System.out.println("Host ID:" + oe.getHostId());
} catch (ClientException ce) {
    System.out.println("Caught an ClientException, which means the client encounter
ed "
        + "a serious internal problem while trying to communicate with OSS, "
        + "such as not being able to access the network.");
    System.out.println("Error Message:" + ce.getMessage());
} finally {
    if (ossClient != null) {
        ossClient.shutdown();
    }
}
}

```

References

For more information about the API operation that you can call to perform conditional download, see [GetObject](#).

2.7.7. Progress bars

You can use a progress bar to indicate the progress of an object that is being uploaded or downloaded. This topic describes how to use a progress bar when you use the `ossClient.getObject` method to download an object.

Sample code

The following code provides an example on how to use a progress bar when you use the `ossClient.getObject` method to download an object.

 **Note** The following methods support the progress bar feature: `ossClient.putObject`, `ossClient.getObject`, `ossClient.uploadPart`, `ossClient.uploadFile`, and `ossClient.downloadFile`. The procedure to apply progress bars to these methods is the same as that of the `ossClient.putObject` method.

```
import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.event.ProgressEvent;
import com.aliyun.oss.event.ProgressEventType;
import com.aliyun.oss.event.ProgressListener;
import com.aliyun.oss.model.GetObjectRequest;
import java.io.File;

public class GetObjectProgressListener implements ProgressListener {
    private long bytesRead = 0;
    private long totalBytes = -1;
    private boolean succeed = false;
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify the actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the name of the bucket. Example: examplebucket.
        String bucketName = "examplebucket";
        // Specify the full path of the object. The full path of the object cannot contain the bucket name. Example: testfolder/exampleobject.txt.
        String objectName = "testfolder/exampleobject.txt";
        String pathName = "D:\\\\localpath\\\\examplefile.txt";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);

        try {
            // Specify the parameter of the progress bar while downloading the object.
            ossClient.getObject(new GetObjectRequest(bucketName, objectName).
                <GetObjectRequest>withProgressListener(new GetObjectProgressListener()),
                new File(pathName));
        } catch (OSSException oe) {
            System.out.println("Caught an OSSException, which means your request made it to OSS, "
                + "but was rejected with an error response for some reason.");
            System.out.println("Error Message:" + oe.getErrorMessage());
            System.out.println("Error Code:" + oe.getErrorCode());
            System.out.println("Request ID:" + oe.getRequestId());
            System.out.println("Host ID:" + oe.getHostId());
        } catch (ClientException ce) {
            System.out.println("Caught an ClientException, which means the client encounter
```

```

ed "
            + "a serious internal problem while trying to communicate with OSS, "
            + "such as not being able to access the network.");
        System.out.println("Error Message:" + ce.getMessage());
    } finally {
        if (ossClient != null) {
            ossClient.shutdown();
        }
    }
}
@Override
public void progressChanged(ProgressEvent progressEvent) {
    long bytes = progressEvent.getBytes();
    ProgressEventType eventType = progressEvent.getEventType();
    switch (eventType) {
        case TRANSFER_STARTED_EVENT:
            System.out.println("Start to download.....");
            break;
        case RESPONSE_CONTENT_LENGTH_EVENT:
            this.totalBytes = bytes;
            System.out.println(this.totalBytes + " bytes in total will be downloaded to
a local file");
            break;
        case RESPONSE_BYTE_TRANSFER_EVENT:
            this.bytesRead += bytes;
            if (this.totalBytes != -1) {
                int percent = (int)(this.bytesRead * 100.0 / this.totalBytes);
                System.out.println(bytes + " bytes have been read at this time, downloa
d progress: " +
                    percent + "%(" + this.bytesRead + "/" + this.totalBytes + ")");
            } else {
                System.out.println(bytes + " bytes have been read at this time, downloa
d ratio: unknown" +
                    "(" + this.bytesRead + "/...)");
            }
            break;
        case TRANSFER_COMPLETED_EVENT:
            this.succeed = true;
            System.out.println("Succeed to download, " + this.bytesRead + " bytes have
been transferred in total");
            break;
        case TRANSFER_FAILED_EVENT:
            System.out.println("Failed to download, " + this.bytesRead + " bytes have b
een transferred");
            break;
        default:
            break;
    }
}
public boolean isSucceed() {
    return succeed;
}
}

```

References

For the complete sample code of progress bars in object download, visit [GitHub](#).

2.8. Manage objects

2.8.1. Overview

This topic is the overview of object management in OSS Java SDK.

You can perform the following operations on objects in a bucket with APIs:

- [Determine whether an object exists](#)
- [Manage ACL for an object](#)
- [Manage Object Meta](#)
- [Convert the storage classes of objects](#)
- [List objects](#)
- [Query objects](#)
- [Delete an object](#)
- [Copy an object](#)
- [Restore an archive object](#)
- [Manage a symbolic link](#)

2.8.2. Determine whether an object exists

This topic describes how to determine whether an object exists.

Sample code

The following code provides an example on how to check whether an object named `exampleobject.txt` exists in a bucket named `examplebucket`:

```
import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;

public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify the actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the name of the bucket. Example: examplebucket.
        String bucketName = "examplebucket";
        // Specify the full path of the object. The full path of the object cannot contain the bucket name. Example: exampleobject.txt.
```

```

String objectName = "exampleobject.txt";
// Create an OSSClient instance.
OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret
);

try {
    // Check whether the object exists. A value of true returned indicates the spec
ified object exists. Otherwise, the specified object or bucket does not exist.
    // Configure whether to perform redirection-based or mirroring-based back-to-or
igin. The default value of isINoss is true, which indicates that redirection-based or mirro
ring-based back-to-origin is not performed. If you set isINoss to false, redirection-based
or mirroring-based back-to-origin is performed.
    //boolean isINoss = true;
    boolean found = ossClient.doesObjectExist(bucketName, objectName);
    //boolean found = ossClient.doesObjectExist(bucketName, objectName, isINoss);
    System.out.println(found);
} catch (OSSException oe) {
    System.out.println("Caught an OSSException, which means your request made it to
OSS, "
        + "but was rejected with an error response for some reason.");
    System.out.println("Error Message:" + oe.getErrorMessage());
    System.out.println("Error Code:" + oe.getErrorCode());
    System.out.println("Request ID:" + oe.getRequestId());
    System.out.println("Host ID:" + oe.getHostId());
} catch (ClientException ce) {
    System.out.println("Caught an ClientException, which means the client encounter
ed "
        + "a serious internal problem while trying to communicate with OSS, "
        + "such as not being able to access the network.");
    System.out.println("Error Message:" + ce.getMessage());
} finally {
    if (ossClient != null) {
        ossClient.shutdown();
    }
}
}

```

References

For more information about the complete sample code that is used to determine whether an object exists, visit [GitHub](#).

2.8.3. Manage object ACLs

Object Storage Service (OSS) allows you to manage the access control lists (ACLs) of both buckets and objects. You can configure the ACL of an object when you upload the object or modify the ACL of an uploaded object.

Object ACLs

The following table describes the ACLs that you can configure for an object.

 **Note** The ACL of an object takes precedence over the ACL of the bucket in which the object is stored. For example, if the ACL of an object in a private bucket is set to public read, all users, including anonymous users, can read the object regardless of the ACL of the bucket.

Permission	Description	Value
Inherited from bucket	The ACL of the object is the same as that of the bucket in which the object is stored. This is the default ACL of an object.	<code>CannedAccessControlList.Default</code>
Private	Only the object owner can read and write the object. Other users cannot access the object.	<code>CannedAccessControlList.Private</code>
Public read	Only the object owner can write the object. Other users, including anonymous users, can only read the object.  Warning All users can access the object over the Internet. This may result in unexpected access to the data in your bucket and unexpectedly high fees. Exercise caution when you set the object ACL to public read.	<code>CannedAccessControlList.PublicRead</code>
Public read/write	Any users, including anonymous users, can read and write the object.  Warning When you set the object ACL to this value, all users can access the object over the Internet and write data to the object. This may result in unexpected access to the data in your bucket and unexpectedly high fees. If a user uploads prohibited data or information, your legitimate interests and rights may be infringed. Therefore, we recommend that you do not set the object ACL to public read/write except in special cases.	<code>CannedAccessControlList.PublicReadWrite</code>

Configure the ACL of an object

The following code provides an example on how to configure the ACL of a specified object:


```

import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.CannedAccessControlList;
public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the bucket name. Example: examplebucket.
        String bucketName = "examplebucket";
        // Specify the full path of the object. The full path of the object cannot contain the bucket name. Example: testfolder/exampleobject.txt.
        String objectName = "testfolder/exampleobject.txt";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);
        try {
            // Set the ACL of the object to public read.
            ossClient.setObjectAcl(bucketName, objectName, CannedAccessControlList.PublicRead);
        } catch (OSSException oe) {
            System.out.println("Caught an OSSException, which means your request made it to OSS, "
                + "but was rejected with an error response for some reason.");
            System.out.println("Error Message:" + oe.getErrorMessage());
            System.out.println("Error Code:" + oe.getErrorCode());
            System.out.println("Request ID:" + oe.getRequestId());
            System.out.println("Host ID:" + oe.getHostId());
        } catch (ClientException ce) {
            System.out.println("Caught an ClientException, which means the client encountered "
                + "a serious internal problem while trying to communicate with OSS, "
                + "such as not being able to access the network.");
            System.out.println("Error Message:" + ce.getMessage());
        } finally {
            if (ossClient != null) {
                ossClient.shutdown();
            }
        }
    }
}

```

Query the ACL of an object

The following code provides an example on how to query the ACL of a specified object:

```
import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.ObjectAcl;
public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the bucket name. Example: examplebucket.
        String bucketName = "examplebucket";
        // Specify the full path of the object. The full path of the object cannot contain the bucket name. Example: testfolder/exampleobject.txt.
        String objectName = "testfolder/exampleobject.txt";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);
    };
    try {
        // Query the ACL of the object.
        ObjectAcl objectAcl = ossClient.getObjectAcl(bucketName, objectName);
        System.out.println(objectAcl.getPermission().toString());
    } catch (OSSException oe) {
        System.out.println("Caught an OSSException, which means your request made it to OSS, "
            + "but was rejected with an error response for some reason.");
        System.out.println("Error Message:" + oe.getErrorMessage());
        System.out.println("Error Code:" + oe.getErrorCode());
        System.out.println("Request ID:" + oe.getRequestId());
        System.out.println("Host ID:" + oe.getHostId());
    } catch (ClientException ce) {
        System.out.println("Caught an ClientException, which means the client encountered "
            + "a serious internal problem while trying to communicate with OSS, "
            + "such as not being able to access the network.");
        System.out.println("Error Message:" + ce.getMessage());
    } finally {
        if (ossClient != null) {
            ossClient.shutdown();
        }
    }
}
```

References

- For the complete sample code that is used to manage the ACL of an object, visit [GitHub](#).

- For more information about the API operation that you can call to configure the ACL of an object, see [Put Object ACL](#).
- For more information about the API operation that you can call to query the ACL of an object, see [Get Object ACL](#).

2.8.4. Manage object metadata

Object Storage Service (OSS) uses object metadata to describe object attributes. Object metadata includes standard HTTP headers and user metadata. You can configure HTTP headers to customize HTTP request policies, such as cache policies and policies for forced object download. You can also configure user metadata to identify the purposes or attributes of objects.

Configure object metadata

The following examples show how to configure the HTTP headers and user metadata of an object.

- Configure HTTP headers

The following code provides an example on how to configure the HTTP headers of an object:

```
import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.common.utils.BinaryUtil;
import com.aliyun.oss.common.utils.DateUtil;
import com.aliyun.oss.model.ObjectMetadata;
import java.io.ByteArrayInputStream;

public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify
        // your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API oper-
        // ations. Using these credentials to perform operations in Object Storage Service (OSS) is
        // a high-risk operation. We recommend that you use a RAM user to call API operations or per-
        // form routine O&M. To create a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the bucket name. Example: examplebucket.
        String bucketName = "examplebucket";
        // Specify the full path of the object. The full path of the object cannot contain
        // the bucket name. Example: testfolder/exampleobject.txt.
        String objectName = "testfolder/exampleobject.txt";
        String content = "Hello OSS";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);

        try {
            // Create the metadata of the object. You can configure the HTTP headers of the
            // object in the object metadata.
            ObjectMetadata meta = new ObjectMetadata();
            String md5 = BinaryUtil.toBase64String(BinaryUtil.calculateMd5(content.getBytes()));
            // Enable MD5 verification. After MD5 verification is enabled, OSS calculates
```

the MD5 hash of the object and compares this MD5 hash with the MD5 hash provided in the request. If the two values are different, an error occurs.

```

        meta.setContentMD5(md5);
        // Specify the type of the content that you want to upload. The browser determines the format and encoding type that are used to read the object based on the content type of the object. If the content type is not specified, a content type is generated based on the object name extension. If no extension is available, the default value application/octet-stream is used as the content type.
        meta.setContentType("text/plain");
        // Specify a name for the object when the content is downloaded.
        meta.setContentDisposition("attachment; filename=\"DownloadFilename\"");
        // Specify the length of the object content to upload. If the actual object length is greater than the configured length, the object is truncated. Only content of the configured length is uploaded. If the configured length is greater than the actual object length, all parts of the object are uploaded.
        meta.setContentLength(content.length());
        // Specify the caching behavior of the website when the content is downloaded
        .
        meta.setCacheControl("Download Action");
        // Specify the expiration time of the cache in GMT.
        meta.setExpirationTime(DateUtil.parseIso8601Date("2022-10-12T00:00:00.000Z"));
    ;

    // Specify the content encoding format when the content is downloaded.
    meta.setContentEncoding("utf-8");
    // Configure the HTTP header.
    meta.setHeader("yourHeader", "yourHeaderValue");
    // Upload the object.
    ossClient.putObject(bucketName, objectName, new ByteArrayInputStream(content.getBytes()), meta);
    } catch (OSSEException oe) {
        System.out.println("Caught an OSSEException, which means your request made it to OSS, "
            + "but was rejected with an error response for some reason.");
        System.out.println("Error Message:" + oe.getErrorMessage());
        System.out.println("Error Code:" + oe.getErrorCode());
        System.out.println("Request ID:" + oe.getRequestId());
        System.out.println("Host ID:" + oe.getHostId());
    } catch (ClientException ce) {
        System.out.println("Caught an ClientException, which means the client encountered "
            + "a serious internal problem while trying to communicate with OSS, "
            + "such as not being able to access the network.");
        System.out.println("Error Message:" + ce.getMessage());
    } finally {
        if (ossClient != null) {
            ossClient.shutdown();
        }
    }
}
}

```

For more information about HTTP headers, see [RFC 2616](#).

- Configure user metadata

You can configure the user metadata of an object to describe the object.

The following code provides an example on how to configure the user metadata of an object:

```
package com.aliyun.oss.demo;
import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.ObjectMetadata;
import java.io.ByteArrayInputStream;

public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify
        // your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API oper-
        // ations. Using these credentials to perform operations in OSS is a high-risk operation. We
        // recommend that you use a RAM user to call API operations or perform routine O&M. To creat
        // e a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the bucket name. Example: examplebucket.
        String bucketName = "examplebucket";
        // Specify the full path of the object. The full path of the object cannot contain
        // the bucket name. Example: testfolder/exampleobject.txt.
        String objectName = "testfolder/exampleobject.txt";
        String content = "Hello OSS";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);

        try {
            // Create the metadata of the object.
            ObjectMetadata meta = new ObjectMetadata();
            // Set the property parameter of user metadata to property-value. We recommend
            // that you use the Base64 encoding method.
            meta.addUserMetadata("property", "property-value");
            // Upload the object.
            ossClient.putObject(bucketName, objectName, new ByteArrayInputStream(content.getBytes()), meta);
            // Query the metadata of the object.
            ossClient.getObjectMetadata(bucketName, objectName);
        } catch (OSSException oe) {
            System.out.println("Caught an OSSException, which means your request made it
            to OSS, "
                + "but was rejected with an error response for some reason.");
            System.out.println("Error Message:" + oe.getErrorMessage());
            System.out.println("Error Code:" + oe.getErrorCode());
            System.out.println("Request ID:" + oe.getRequestId());
            System.out.println("Host ID:" + oe.getHostId());
        } catch (ClientException ce) {
            System.out.println("Caught an ClientException, which means the client encountered "
                + "a serious internal problem while trying to communicate with OSS, "
                + "such as not being able to access the network.");
        }
    }
}
```

```

        System.out.println("Error Message:" + ce.getMessage());
    } finally {
        if (ossClient != null) {
            ossClient.shutdown();
        }
    }
}
}
}

```

The metadata of an object is downloaded along with the object. An object can have multiple pieces of object metadata. The total size of object metadata cannot exceed 8 KB.

Modify object metadata

The following code provides an example on how to modify the metadata of an object:

```

import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.common.utils.DateUtil;
import com.aliyun.oss.model.CopyObjectRequest;
import com.aliyun.oss.model.ObjectMetadata;

public class Demo {

    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the name of the source bucket.
        String sourceBucketName = "yourSourceBucketName";
        // Specify the full path of the source object.
        String sourceObjectName = "yourSourceObjectName";
        // Specify the name of the destination bucket. The destination bucket must be in the same region as the source bucket.
        String destinationBucketName = "yourDestinationBucketName";
        // Specify the full path of the destination object.
        String destinationObjectName = "yourDestinationObjectName";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);

        try {
            // Replace the source object with the destination object. Call ossClient.copyObject to modify object metadata.
            CopyObjectRequest request = new CopyObjectRequest(sourceBucketName, sourceObjectName, destinationBucketName, destinationObjectName);
            ObjectMetadata meta = new ObjectMetadata();
            // Specify the type of the content that you want to upload. The browser determines the format and encoding type that are used to read the object based on the content type of the object. If the content type is not specified, a content type is generated based on the

```

of the object. If the content type is not specified, a content type is generated based on the object name extension. If no extension is available, the default value application/octet-stream is used as the content type.

```
meta.setContentType("text/plain");
// Specify a name for the object when the content is downloaded.
meta.setContentDisposition("Download File Name");
// Specify the caching behavior of the website when the content is downloaded.
meta.setCacheControl("Download Action");
// Specify the expiration time of the cache in GMT.
meta.setExpirationTime(DateUtil.parseIso8601Date("2022-10-12T00:00:00.000Z"));
// Specify the content encoding format when the content is downloaded.
meta.setContentEncoding("utf-8");
// Configure the header.
meta.setHeader("<yourHeader>", "<yourHeaderValue>");
// Set the property parameter of user metadata to property-value.
meta.addUserMetadata("property", "property-value");
request.setNewObjectMetadata(meta);
// Modify the metadata of the object.
ossClient.copyObject(request);
} catch (OSSException oe) {
    System.out.println("Caught an OSSException, which means your request made it to
OSS, "
        + "but was rejected with an error response for some reason.");
    System.out.println("Error Message:" + oe.getErrorMessage());
    System.out.println("Error Code:" + oe.getErrorCode());
    System.out.println("Request ID:" + oe.getRequestId());
    System.out.println("Host ID:" + oe.getHostId());
} catch (ClientException ce) {
    System.out.println("Caught an ClientException, which means the client encounter
ed "
        + "a serious internal problem while trying to communicate with OSS, "
        + "such as not being able to access the network.");
    System.out.println("Error Message:" + ce.getMessage());
} finally {
    if (ossClient != null) {
        ossClient.shutdown();
    }
}
}
```

Query object metadata

The following table provides the methods you can use to query the metadata of an object.

Method	Description	Advantage
<code>ossClient.getSimplifiedObjectMetadata</code>	Queries part of object metadata such as the ETag, Size, and LastModified (the last modified time) values of the object.	More lightweight and faster
<code>ossClient.getObjectMetadata</code>	Queries all object metadata.	N/A

The following code provides an example on how to query the metadata of an object:

```
import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.ObjectMetadata;
import com.aliyun.oss.model.SimplifiedObjectMeta;
public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the bucket name. Example: examplebucket.
        String bucketName = "examplebucket";
        // Specify the full path of the object. The full path of the object cannot contain the bucket name. Example: testfolder/exampleobject.txt.
        String objectName = "testfolder/exampleobject.txt";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);

        try {
            // Specify the bucket name and the full path of the object in the bucket.
            // Query part of object metadata.
            SimplifiedObjectMeta objectMeta = ossClient.getSimplifiedObjectMeta(bucketName, objectName);
            System.out.println(objectMeta.getSize());
            System.out.println(objectMeta.getETag());
            System.out.println(objectMeta.getLastModified());
            // Query all object metadata.
            ObjectMetadata metadata = ossClient.getObjectMetadata(bucketName, objectName);
            System.out.println(metadata.getContentType());
            System.out.println(metadata.getLastModified());
            System.out.println(metadata.getExpirationTime());
        } catch (OSSException oe) {
            System.out.println("Caught an OSSException, which means your request made it to OSS, "
                + "but was rejected with an error response for some reason.");
            System.out.println("Error Message:" + oe.getErrorMessage());
            System.out.println("Error Code:" + oe.getErrorCode());
            System.out.println("Request ID:" + oe.getRequestId());
            System.out.println("Host ID:" + oe.getHostId());
        } catch (ClientException ce) {
            System.out.println("Caught an ClientException, which means the client encountered "
                + "a serious internal problem while trying to communicate with OSS, "
                + "such as not being able to access the network.");
            System.out.println("Error Message:" + ce.getMessage());
        }
    }
}
```



```

        } finally {
            if (ossClient != null) {
                ossClient.shutdown();
            }
        }
    }
}

```

References

- For the complete sample code that is used to configure and query object metadata, visit [GitHub](#).
- For more information about the API operation that you can call to configure object metadata in simple upload, see [Put Object](#).
- For more information about the API operation that you can call to query object metadata, see [Get Object Meta](#).

2.8.5. Convert the storage classes of objects

OSS provides the following storage classes to cover various data storage scenarios from hot data to cold data: Standard, Infrequent Access (IA), Archive and Cold Archive. This topic describes how to convert the storage class of an object.

Sample code

The following section provides examples on how to convert the storage class of an object.

- The following code provides an example on how to convert the storage class of an object from Standard or IA to Archive:

```

import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.CopyObjectRequest;
import com.aliyun.oss.model.CopyObjectResult;
import com.aliyun.oss.model.ObjectMetadata;
import com.aliyun.oss.model.StorageClass;

public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify
        // the actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API oper-
        // ations. Using these credentials to perform operations in OSS is a high-risk operation. We
        // recommend that you use a RAM user to call API operations or perform routine O&M. To creat
        // e a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // In this example, a bucket and a Standard or an IA object have been created.
        // Specify the name of the bucket. Example: examplebucket.
        String bucketName = "examplebucket";
        // Specify the full path of the object. The full path of the object cannot contain
        // the bucket name. Example: exampleobject.txt.
        String objectName = "exampleobject.txt";
    }
}

```

```

        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);

        try {
            // Create a CopyObjectRequest object.
            CopyObjectRequest request = new CopyObjectRequest(bucketName, objectName, bucketName, objectName);
            // Create an ObjectMetadata object.
            ObjectMetadata objectMetadata = new ObjectMetadata();
            // Encapsulate the header. In this example, the storage class is converted to Archive.
            objectMetadata.setHeader("x-oss-storage-class", StorageClass.Archive);
            request.setNewObjectMetadata(objectMetadata);
            // Convert the storage class of the object.
            CopyObjectResult result = ossClient.copyObject(request);
        } catch (OSSException oe) {
            System.out.println("Caught an OSSException, which means your request made it to OSS, "
                + "but was rejected with an error response for some reason.");
            System.out.println("Error Message:" + oe.getErrorMessage());
            System.out.println("Error Code:" + oe.getErrorCode());
            System.out.println("Request ID:" + oe.getRequestId());
            System.out.println("Host ID:" + oe.getHostId());
        } catch (ClientException ce) {
            System.out.println("Caught an ClientException, which means the client encountered "
                + "a serious internal problem while trying to communicate with OSS, "
                + "such as not being able to access the network.");
            System.out.println("Error Message:" + ce.getMessage());
        } finally {
            if (ossClient != null) {
                ossClient.shutdown();
            }
        }
    }
}

```

- The following code provides an example on how to convert the storage class of an object from Archive to IA:

```

package com.aliyun.oss.demo;
import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.CopyObjectRequest;
import com.aliyun.oss.model.CopyObjectResult;
import com.aliyun.oss.model.ObjectMetadata;
import com.aliyun.oss.model.StorageClass;
public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify the actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKeyPair of an Alibaba Cloud account has permissions on all API operations.
    }
}

```

```

// The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
String accessKeyId = "yourAccessKeyId";
String accessKeySecret = "yourAccessKeySecret";
// In this example, a bucket and an Archive object have been created.
// Specify the name of the bucket. Example: examplebucket.
String bucketName = "examplebucket";
// Specify the full path of the object. The full path of the object cannot contain the bucket name. Example: exampleobject.txt.
String objectName = "exampleobject.txt";
// Create an OSSClient instance.
OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);

try {
    // Query the object metadata.
    ObjectMetadata objectMetadata = ossClient.getObjectMetadata(bucketName, objectName);

    // Check whether the storage class of the object for which you want to convert the storage class is Archive. If the storage class of the object is Archive, you must restore the object before you can convert the storage class.
    StorageClass storageClass = objectMetadata.getObjectStorageClass();
    System.out.println("storage type:" + storageClass);
    if (storageClass == StorageClass.Archive) {
        // Restore the object.
        ossClient.restoreObject(bucketName, objectName);
        // Wait until the object is restored.
        do {
            Thread.sleep(1000);
            objectMetadata = ossClient.getObjectMetadata(bucketName, objectName);
        } while (!objectMetadata.isRestoreCompleted());
    }
    // Create a CopyObjectRequest object.
    CopyObjectRequest request = new CopyObjectRequest(bucketName, objectName, bucketName, objectName);
    // Create an ObjectMetadata object.
    objectMetadata = new ObjectMetadata();
    // Encapsulate the header. In this example, the storage class is converted to IA.
    objectMetadata.setHeader("x-oss-storage-class", StorageClass.IA);
    request.setNewObjectMetadata(objectMetadata);
    // Convert the storage class of the object.
    CopyObjectResult result = ossClient.copyObject(request);
} catch (OSSException oe) {
    System.out.println("Caught an OSSException, which means your request made it to OSS, "
        + "but was rejected with an error response for some reason.");
    System.out.println("Error Message:" + oe.getErrorMessage());
    System.out.println("Error Code:" + oe.getErrorCode());
    System.out.println("Request ID:" + oe.getRequestId());
    System.out.println("Host ID:" + oe.getHostId());
} catch (ClientException ce) {
    System.out.println("Caught an ClientException, which means the client encountered "

```

```
        + "a serious internal problem while trying to communicate with OSS, "  
        + "such as not being able to access the network.");  
        System.out.println("Error Message:" + ce.getMessage());  
    } finally {  
        if (ossClient != null) {  
            ossClient.shutdown();  
        }  
    }  
}
```

References

- For more information about the complete sample code that is used to convert the storage class of an object, visit [GitHub](#).
- For more information about the API operation that you can call to convert the storage class of an object, see [CopyObject](#).

2.8.6. List objects

This topic describes how to list all objects in a bucket, a specified number of objects in a bucket, and objects whose names contain a specified prefix in a bucket.

Background information

You can call the GetBucket (ListObjects) or GetBucketV2 (ListObjectsV2) operation to list up to 1,000 objects in a bucket at a time. This operation provides several parameters to customize the output. For example, you can configure parameters to list objects from a specified start position, list objects and subdirectories in a specified directory, and list objects by page. The GetBucket (ListObjects) and GetBucketV2 (ListObjectsV2) operations are different in the following aspects:

- By default, when you call the GetBucket (ListObjects) operation to list objects, the information about the bucket owner is included in the response.
- When you call the GetBucketV2 (ListObjectsV2) operation to list objects, you can set the fetchOwner parameter to specify whether to include the information about the bucket owner in the response.

 **Note** If you want to list objects in a versioned bucket, we recommend that you call the GetBucketV2 (ListObjectsV2) operation.

The following tables describe the parameters that you can configure when you call the GetBucket (ListObjects) or GetBucketV2 (ListObjectsV2) operation to list objects.

- Call GetBucket (ListObjects) to list objects

You can call GetBucket (ListObjects) in one of the following formats:

- ObjectListing listObjects(String bucketName): lists all objects in a bucket. By default, up to 100 objects can be listed by a single request.
- ObjectListing listObjects(String bucketName, String prefix): lists objects whose names contain a specified prefix in a bucket. By default, up to 100 objects can be listed by a single request.
- ObjectListing listObjects(ListObjectsRequest listObjectsRequest): lists objects with multiple conditions specified. You can use this format to list objects in a flexible manner.

The following table describes the parameters that you can configure for the GetBucket (ListObjects) operation.

Parameter	Description	Method
objectSummaries	The returned object metadata.	List<OSSObjectSummary> getObjectSummaries()
prefix	The prefix that must be included in the names of returned objects.	String getPrefix()
delimiter	The character that is used to group the returned objects by name.	String getDelimiter()
marker	The position from which the list operation starts.	String getMarker()
maxKeys	The maximum number of objects that can be listed each time.	int getMaxKeys()
nextMarker	The position from which the next list operation starts.	String getNextMarker()
isTruncated	Indicates whether the listed objects are truncated. Valid values: ◦ false: All objects are listed without truncation. ◦ true: Only part of the objects are listed.	boolean isTruncated()
commonPrefixes	A set of substrings of object names. Objects whose names end with the delimiter and contain the same prefix are grouped as a single result element in commonPrefixes.	List<String> getCommonPrefixes()
encodingType	The encoding type of the returned object names.	String getEncodingType()

- Call GetBucketV2 (ListObjectsV2) to list objects

You can call GetBucketV2 (ListObjectsV2) in one of the following formats:

- ListObjectsV2Result listObjectsV2(String bucketName): lists all objects in a bucket. By default, up to 100 objects can be listed by a single request.
- ListObjectsV2Result listObjectsV2(String bucketName, String prefix): lists objects whose names contain a specified prefix in a bucket. By default, up to 100 objects can be listed by a single request.

- `ListObjectsV2Result listObjectsV2(ListObjectsRequest listObjectsRequest)`: lists objects with multiple conditions specified. You can use this format to list objects in a flexible manner.

The following table describes the parameters that you can configure for the `GetBucketV2` (`ListObjectsV2`) operation.

Parameter	Description	Method
<code>objectSummaries</code>	The returned object metadata.	<code>List<OSSObjectSummary> getObjectSummaries()</code>
<code>prefix</code>	The prefix that must be included in the names of returned objects.	<code>String getPrefix()</code>
<code>delimiter</code>	The character that is used to group the returned objects by name.	<code>String getDelimiter()</code>
<code>startAfter</code>	The position from which the list operation starts.	<code>String getStartAfter()</code>
<code>maxKeys</code>	The maximum number of objects that can be listed each time.	<code>int getMaxKeys()</code>
<code>continuationToken</code>	The continuationToken that is used in this list operation.	<code>String getContinuationToken()</code>
<code>nextContinuationToken</code>	The continuationToken that is used in the next list operation.	<code>String getNextContinuationToken()</code>
<code>isTruncated</code>	Indicates whether the listed objects are truncated. Valid values: ◦ <code>false</code>: All objects are listed without truncation. ◦ <code>true</code>: Only part of the objects are listed.	<code>boolean isTruncated()</code>
<code>commonPrefixes</code>	A set of substrings of object names. Objects whose names end with the delimiter and contain the same prefix are grouped as a single result element in <code>commonPrefixes</code> .	<code>List<String> getCommonPrefixes()</code>
<code>encodingType</code>	The encoding type of the returned object names.	<code>String getEncodingType()</code>

Parameter	Description	Method
fetchOwner	Specifies whether to include the object owner information in the response. Valid values: ◦ true: The response includes the owner information. ◦ false: The response does not include the owner information.	String getFetchOwner()

Simple list

You can call GetBucket (ListObjects) or GetBucketV2 (ListObjectsV2) to list objects in a specified bucket.

- Call GetBucket (ListObjects) to list objects

The following code provides an example on how to call GetBucket (ListObjects) to list objects in a specified bucket. By default, 100 objects are listed by a request.

```
import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.*;
import java.util.List;
public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify
        // your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API oper-
        // ations. Using these credentials to perform operations in Object Storage Service (OSS) is
        // a high-risk operation. We recommend that you use a RAM user to call API operations or per-
        // form routine O&M. To create a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the bucket name. Example: examplebucket.
        String bucketName = "examplebucket";
        // Specify the prefix that must be included in the names of objects to list. Exam-
        // ple: exampledir/object.
        String keyPrefix = "exampledir/object";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecr
        et);
        try {
            // List objects. If you do not specify keyPrefix, all objects in the bucket a
            // re listed. If you specify keyPrefix, the objects whose names contain the specified prefix
            // are listed.
            ObjectListing objectListing = ossClient.listObjects(bucketName, keyPrefix);
            List<OSSObjectSummary> sums = objectListing.getObjectSummaries();
            for (OSSObjectSummary s : sums) {
                System.out.println("\t" + s.getKey());
            }
        }
    }
}
```

```

    } catch (OSSEException oe) {
        System.out.println("Caught an OSSEException, which means your request made it
to OSS, "
            + "but was rejected with an error response for some reason.");
        System.out.println("Error Message:" + oe.getErrorMessage());
        System.out.println("Error Code:" + oe.getErrorCode());
        System.out.println("Request ID:" + oe.getRequestId());
        System.out.println("Host ID:" + oe.getHostId());
    } catch (ClientException ce) {
        System.out.println("Caught an ClientException, which means the client encount
ered "
            + "a serious internal problem while trying to communicate with OSS, "
            + "such as not being able to access the network.");
        System.out.println("Error Message:" + ce.getMessage());
    } finally {
        if (ossClient != null) {
            ossClient.shutdown();
        }
    }
}
}

```

- Call GetBucketV2 (ListObjectsV2) to list objects

The following code provides an example on how to call GetBucketV2 (ListObjectsV2) to list objects in a specified bucket. By default, 100 objects are listed by a request.

```

import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSEException;
import com.aliyun.oss.model.*;
import java.util.List;

public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify
your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API oper
ations. Using these credentials to perform operations in OSS is a high-risk operation. We
recommend that you use a RAM user to call API operations or perform routine O&M. To creat
e a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the bucket name. Example: examplebucket.
        String bucketName = "examplebucket";
        // Specify the prefix that must be included in the names of objects to list. Exam
ple: exampledir/object.
        String keyPrefix = "exampledir/object";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecr
et);
        try {
            // List objects. If you do not specify keyPrefix, all objects in the bucket a
re listed. If you specify keyPrefix, the objects whose names contain the specified prefix

```



```
are listed.
        ListObjectsV2Result result = ossClient.listObjectsV2(bucketName, keyPrefix);
        List<OSSObjectSummary> ossObjectSummaries = result.getObjectSummaries();
        for (OSSObjectSummary s : ossObjectSummaries) {
            System.out.println("\t" + s.getKey());
        }
    } catch (OSSException oe) {
        System.out.println("Caught an OSSException, which means your request made it
to OSS, "
            + "but was rejected with an error response for some reason.");
        System.out.println("Error Message:" + oe.getErrorMessage());
        System.out.println("Error Code:" + oe.getErrorCode());
        System.out.println("Request ID:" + oe.getRequestId());
        System.out.println("Host ID:" + oe.getHostId());
    } catch (ClientException ce) {
        System.out.println("Caught an ClientException, which means the client encount
ered "
            + "a serious internal problem while trying to communicate with OSS, "
            + "such as not being able to access the network.");
        System.out.println("Error Message:" + ce.getMessage());
    } finally {
        if (ossClient != null) {
            ossClient.shutdown();
        }
    }
}
```

Call ListObjectsRequest to list objects

You can configure parameters for the ListObjectsRequest operation to list objects in a flexible manner. The following table describes the parameters that you can configure for the ListObjectsRequest operation.

Parameter	Description	Method
prefix	The prefix that must be included in the names of returned objects. For information about how to use this parameter, see List objects whose names contain a specified prefix .	setPrefix(String prefix)

Parameter	Description	Method
delimiter	The character that is used to group the returned objects by name. Objects whose names contain the same string that ranges from the specified prefix to the next occurrence of the delimiter are grouped as a single result element in <code>commonPrefixes</code> . For information about how to use this parameter, see List objects and subdirectories in the specified directory .	<code>setDelimiter(String delimiter)</code>
marker	The position from which the list operation starts. Objects whose names are alphabetically after the value of marker are listed. For information about how to use this parameter, see List objects whose names are alphabetically after the object specified by marker .	<code>setMarker(String marker)</code>
maxKeys	The maximum number of objects that can be listed at a time. After you specify this parameter, objects are listed in alphabetical order. Maximum value: 1000. Default value: 100. For information about how to use this parameter, see List all objects in a bucket by page .	<code>setMaxKeys(Integer maxKeys)</code>
encodingType	The encoding type of the object names in the response. Only URL encoding is supported. For information about how to use this parameter, see List objects and specify the encoding type of the object names .	<code>setEncodingType(String encodingType)</code>

- List a specified number of objects

The following code provides an example on how to list a specified number of objects:

```

import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.*;
import java.util.List;

public class Demo {
    public static void main(String[] args){
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify
        // your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API oper
        // ations. Using these credentials to perform operations in OSS is a high-risk operation. We
        // recommend that you use a RAM user to call API operations or perform routine O&M. To creat
        // e a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the bucket name. Example: examplebucket.
        String bucketName = "examplebucket";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecr
        et);

        try {
            // Specify the maximum number of objects that can be listed each time.
            final int maxKeys = 200;
            // List objects.
            ObjectListing objectListing = ossClient.listObjects(new ListObjectsRequest(bu
            cketName).withMaxKeys(maxKeys));
            List<OSSObjectSummary> sums = objectListing.getObjectSummaries();
            for (OSSObjectSummary s : sums) {
                System.out.println("\t" + s.getKey());
            }
        } catch (OSSException oe) {
            System.out.println("Caught an OSSException, which means your request made it
            to OSS, "
                + "but was rejected with an error response for some reason.");
            System.out.println("Error Message:" + oe.getErrorMessage());
            System.out.println("Error Code:" + oe.getErrorCode());
            System.out.println("Request ID:" + oe.getRequestId());
            System.out.println("Host ID:" + oe.getHostId());
        } catch (ClientException ce) {
            System.out.println("Caught an ClientException, which means the client encount
            ered "
                + "a serious internal problem while trying to communicate with OSS, "
                + "such as not being able to access the network.");
            System.out.println("Error Message:" + ce.getMessage());
        } finally {
            if (ossClient != null) {
                ossClient.shutdown();
            }
        }
    }
}

```

- List objects whose names contain a specified prefix

The following code provides an example on how to list objects whose names contain a specified prefix:

```
import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.*;
import java.util.List;

public class Demo {
    public static void main(String[] args){
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify
        // your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API oper-
        // ations. Using these credentials to perform operations in OSS is a high-risk operation. We
        // recommend that you use a RAM user to call API operations or perform routine O&M. To creat-
        // e a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the bucket name. Example: examplebucket.
        String bucketName = "examplebucket";
        // Specify the prefix that must be included in the names of objects to list. Exam-
        // ple: exampledir/object.
        String keyPrefix = "exampledir/object";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);

        try {
            // List objects whose names contain the specified prefix. By default, 100 obj-
            // ects are listed.
            ObjectListing objectListing = ossClient.listObjects(new ListObjectsRequest(bucketName).withPrefix(keyPrefix));
            List<OSSObjectSummary> sums = objectListing.getObjectSummaries();
            for (OSSObjectSummary s : sums) {
                System.out.println("\t" + s.getKey());
            }
        } catch (OSSException oe) {
            System.out.println("Caught an OSSException, which means your request made it
            to OSS, "
                + "but was rejected with an error response for some reason.");
            System.out.println("Error Message:" + oe.getErrorMessage());
            System.out.println("Error Code:" + oe.getErrorCode());
            System.out.println("Request ID:" + oe.getRequestId());
            System.out.println("Host ID:" + oe.getHostId());
        } catch (ClientException ce) {
            System.out.println("Caught an ClientException, which means the client encount-
            ered "
                + "a serious internal problem while trying to communicate with OSS, "
                + "such as not being able to access the network.");
            System.out.println("Error Message:" + ce.getMessage());
        } finally {
            if (ossClient != null) {

```

```

        ossClient.shutdown();
    }
}
}
}

```

- List objects whose names are alphabetically after the object specified by marker

You can configure the marker parameter to specify the name of the object after which the list operation starts. The following code provides an example on how to list objects whose names are alphabetically after the object specified by marker:

```

import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.*;
import java.util.List;

public class Demo {
    public static void main(String[] args){
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify
        // your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API oper-
        // ations. Using these credentials to perform operations in OSS is a high-risk operation. We
        // recommend that you use a RAM user to call API operations or perform routine O&M. To creat-
        // e a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the bucket name. Example: examplebucket.
        String bucketName = "examplebucket";
        // Specify the marker parameter. Example: exampleobject.txt.
        String marker = "exampleobject.txt";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecr-
        et);

        try {
            // List objects whose names are alphabetically after the object specified by
            // marker. By default, 100 objects are listed.
            ObjectListing objectListing = ossClient.listObjects(new ListObjectsRequest(bu-
            cketName).withMarker(marker));
            List<OSSObjectSummary> sums = objectListing.getObjectSummaries();
            for (OSSObjectSummary s : sums) {
                System.out.println("\t" + s.getKey());
            }
        } catch (OSSException oe) {
            System.out.println("Caught an OSSException, which means your request made it
            to OSS, "
                + "but was rejected with an error response for some reason.");
            System.out.println("Error Message:" + oe.getErrorMessage());
            System.out.println("Error Code:" + oe.getErrorCode());
            System.out.println("Request ID:" + oe.getRequestId());
            System.out.println("Host ID:" + oe.getHostId());
        } catch (ClientException ce) {
            System.out.println("Caught an ClientException, which means the client encount-

```

```

        } catch (ClientException ce) {
            System.out.println("Caught an ClientException, which means the client encountered "
                + "a serious internal problem while trying to communicate with OSS, "
                + "such as not being able to access the network.");
            System.out.println("Error Message:" + ce.getMessage());
        } finally {
            if (ossClient != null) {
                ossClient.shutdown();
            }
        }
    }
}

```

- List all objects in a bucket by page

The following code provides an example on how to list all objects in a specified bucket by page. You can set `maxKeys` to specify the maximum number of objects that can be listed on each page.

```

import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.*;
import java.util.List;

public class Demo {
    public static void main(String[] args){
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify
        // your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API oper-
        // ations. Using these credentials to perform operations in OSS is a high-risk operation. We
        // recommend that you use a RAM user to call API operations or perform routine O&M. To creat-
        // e a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the bucket name. Example: examplebucket.
        String bucketName = "examplebucket";
        // Set maxKeys to 200.
        int maxKeys = 200;
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySec-
        et);

        try {
            String nextMarker = null;
            ObjectListing objectListing;
            do {
                objectListing = ossClient.listObjects(new ListObjectsRequest(bucketName).
                withMarker(nextMarker).withMaxKeys(maxKeys));
                List<OSSObjectSummary> sums = objectListing.getObjectSummaries();
                for (OSSObjectSummary s : sums) {
                    System.out.println("\t" + s.getKey());
                }
                nextMarker = objectListing.getNextMarker();
            } while (objectListing.isTruncated());
        } catch (OSSException oe) {
            System.out.println("Caught an OSSException, which means your request made it

```

```

        System.out.println("Caught an OSSException, which means your request made it
to OSS, "
            + "but was rejected with an error response for some reason.");
        System.out.println("Error Message:" + oe.getErrorMessage());
        System.out.println("Error Code:" + oe.getErrorCode());
        System.out.println("Request ID:" + oe.getRequestId());
        System.out.println("Host ID:" + oe.getHostId());
    } catch (ClientException ce) {
        System.out.println("Caught an ClientException, which means the client encount
ered "
            + "a serious internal problem while trying to communicate with OSS, "
            + "such as not being able to access the network.");
        System.out.println("Error Message:" + ce.getMessage());
    } finally {
        if (ossClient != null) {
            ossClient.shutdown();
        }
    }
}
}
}

```

- List objects whose names contain a specified prefix by page

The following code provides an example on how to list objects whose names contain a specified prefix by page:

```

import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.*;
import java.util.List;
public class Demo {
    public static void main(String[] args){
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify
your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API oper
ations. Using these credentials to perform operations in OSS is a high-risk operation. We
recommend that you use a RAM user to call API operations or perform routine O&M. To creat
e a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the bucket name. Example: examplebucket.
        String bucketName = "examplebucket";
        // Specify that up to 200 objects can be listed on each page.
        int maxKeys = 200;
        // Specify the prefix that must be included in the names of objects to list. Exam
ple: exampledir/object.
        String keyPrefix = "exampledir/object";
        // Specify the marker parameter. Example: objecttest.txt.
        String nextMarker = "objecttest.txt";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecr
et);
        try {

```

```

try {
    ObjectListing objectListing;
    do {
        objectListing = ossClient.listObjects(new ListObjectsRequest(bucketName).
            withPrefix(keyPrefix).withMarker(nextMarker).withMaxKeys(maxKeys)
        );

        List<OSSObjectSummary> sums = objectListing.getObjectSummaries();
        for (OSSObjectSummary s : sums) {
            System.out.println("\t" + s.getKey());
        }
        nextMarker = objectListing.getNextMarker();
    } while (objectListing.isTruncated());
} catch (OSSException oe) {
    System.out.println("Caught an OSSException, which means your request made it
to OSS, "
        + "but was rejected with an error response for some reason.");
    System.out.println("Error Message:" + oe.getErrorMessage());
    System.out.println("Error Code:" + oe.getErrorCode());
    System.out.println("Request ID:" + oe.getRequestId());
    System.out.println("Host ID:" + oe.getHostId());
} catch (ClientException ce) {
    System.out.println("Caught an ClientException, which means the client encount
ered "
        + "a serious internal problem while trying to communicate with OSS, "
        + "such as not being able to access the network.");
    System.out.println("Error Message:" + ce.getMessage());
} finally {
    if (ossClient != null) {
        ossClient.shutdown();
    }
}
}
}

```

- List objects and specify the encoding type of the object names

An object name that contains one of the following special characters must be encoded before the object is transmitted. OSS supports only URL encoding.

- Single quotation marks (')
- Double quotations marks (")
- Ampersands (&)
- Angle brackets (<>)
- Pause markers (,)
- Chinese characters

The following code provides an example on how to list objects and specify the encoding type of the object names:

```

import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.*;
import java.net.URLEncoder;

```



```

import java.net.URLDecoder;

public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify
        // your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API oper-
        // ations. Using these credentials to perform operations in OSS is a high-risk operation. We
        // recommend that you use a RAM user to call API operations or perform routine O&M. To creat
        // e a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the bucket name. Example: examplebucket.
        String bucketName = "examplebucket";
        // Specify that up to 200 objects can be listed on each page.
        int maxKeys = 200;
        // Specify the prefix that must be included in the names of objects to list. Exam-
        // ple: exampledir/object.
        String keyPrefix = "exampledir/object";
        // Specify the marker parameter. Example: objecttest.txt.
        String nextMarker = "objecttest.txt";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecr
        et);

        try {
            ObjectListing objectListing;
            do {
                ListObjectsRequest listObjectsRequest = new ListObjectsRequest(bucketName
                );

                listObjectsRequest.setPrefix(keyPrefix);
                listObjectsRequest.setMaxKeys(maxKeys);
                listObjectsRequest.setMarker(nextMarker);
                // Specify the encoding type of the object names.
                listObjectsRequest.setEncodingType("url");
                objectListing = ossClient.listObjects(listObjectsRequest);
                // Decode the value of Key in the response.
                for (OSSObjectSummary objectSummary: objectListing.getObjectSummaries())
                {
                    System.out.println("Key:" + URLDecoder.decode(objectSummary.getKey(),
                    "UTF-8"));
                }
                // Decode the value of commonPrefixes in the response.
                for (String commonPrefixes: objectListing.getCommonPrefixes()) {
                    System.out.println("CommonPrefixes:" + URLDecoder.decode(commonPrefix
                    es, "UTF-8"));
                }
                // Decode the value of nextMarker in the response.
                if (objectListing.getNextMarker() != null) {
                    nextMarker = URLDecoder.decode(objectListing.getNextMarker(), "UTF-8"
                    );
                }
            } while (objectListing.isTruncated());
        } catch (OSSException oe) {
            System.out.println("Caught an OSSException, which means your request made it
            to OSS, "

```

```

        + "but was rejected with an error response for some reason.");
        System.out.println("Error Message:" + oe.getErrorMessage());
        System.out.println("Error Code:" + oe.getErrorCode());
        System.out.println("Request ID:" + oe.getRequestId());
        System.out.println("Host ID:" + oe.getHostId());
    } catch (ClientException ce) {
        System.out.println("Caught an ClientException, which means the client encount
ered "
            + "a serious internal problem while trying to communicate with OSS, "
            + "such as not being able to access the network.");
        System.out.println("Error Message:" + ce.getMessage());
    } finally {
        if (ossClient != null) {
            ossClient.shutdown();
        }
    }
}
}
}

```

Call ListObjectsV2Request to list objects

You can configure the parameters for the ListObjectsV2Request operation to list objects in a flexible manner. For example, you can list a specified number of objects, objects whose names contain a specified prefix, or all objects in a bucket by page.

The following table describes the parameters that you can configure for the ListObjectsV2Request operation.

Parameter	Description	Method
prefix	The prefix that must be included in the names of returned objects. For information about how to use this parameter, see List objects whose names contain a specified prefix .	setPrefix(String prefix)
delimiter	The character that is used to group the returned objects by name. Objects whose names contain the same string that ranges from the specified prefix to the next occurrence of the delimiter are grouped as a single result element in commonPrefixes. For information about how to use this parameter, see List objects and subdirectories in the specified directory .	setDelimiter(String delimiter)
maxKeys	The maximum number of objects that can be listed at a time. After you specify this parameter, objects are listed in alphabetical order. Maximum value: 1000. Default value: 100. For information about how to use this parameter, see List all objects in a bucket by page .	setMaxKeys(Integer maxKeys)

Parameter	Description	Method
startAfter	The position from which the list operation starts. Objects whose names are alphabetically after the value of startAfter are listed. For information about how to use this parameter, see List objects whose names are alphabetically after the object specified by startAfter .	setStartAfter(String startAfter)
continuationToken	The continuationToken that is used in this list operation. For information about how to use this parameter, see List all objects in a bucket by page .	setContinuationToken(String continuationToken)
encodingType	The encoding type of the object names in the response. Only URL encoding is supported. For information about how to use this parameter, see List objects and specify the encoding type of the object names .	setEncodingType(String encodingType)
fetchOwner	Specifies whether to include the object owner information in the response. For information about how to use this parameter, see List objects and their owner information .	setFetchOwner(boolean fetchOwner)

- List a specified number of objects

The following code provides an example on how to list a specified number of objects:

```
import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.*;
import java.util.List;

public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify
        // your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API oper-
        // ations. Using these credentials to perform operations in OSS is a high-risk operation. We
        // recommend that you use a RAM user to call API operations or perform routine O&M. To creat
        // e a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the bucket name. Example: examplebucket.
        String bucketName = "examplebucket";
        // Specify the maximum number of objects that can be listed each time.
        int maxKeys = 200;
        // Create an OSSClient instance.
```

```

    OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);

    try {
        // List objects. By default, 100 objects are returned by a single request. In
        // this example, up to 200 objects can be returned each time.
        ListObjectsV2Request listObjectsV2Request = new ListObjectsV2Request(bucketName);

        listObjectsV2Request.setMaxKeys(maxKeys);
        ListObjectsV2Result result = ossClient.listObjectsV2(listObjectsV2Request);
        List<OSSObjectSummary> ossObjectSummaries = result.getObjectSummaries();
        for (OSSObjectSummary s : ossObjectSummaries) {
            System.out.println("\t" + s.getKey());
        }
    } catch (OSSEException oe) {
        System.out.println("Caught an OSSEException, which means your request made it
        to OSS, "
            + "but was rejected with an error response for some reason.");
        System.out.println("Error Message:" + oe.getErrorMessage());
        System.out.println("Error Code:" + oe.getErrorCode());
        System.out.println("Request ID:" + oe.getRequestId());
        System.out.println("Host ID:" + oe.getHostId());
    } catch (ClientException ce) {
        System.out.println("Caught an ClientException, which means the client encountered "
            + "a serious internal problem while trying to communicate with OSS, "
            + "such as not being able to access the network.");
        System.out.println("Error Message:" + ce.getMessage());
    } finally {
        if (ossClient != null) {
            ossClient.shutdown();
        }
    }
}

```

- List objects whose names contain a specified prefix

The following code provides an example on how to list objects whose names contain a specified prefix:

```

import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSEException;
import com.aliyun.oss.model.*;
import java.util.List;

public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify
        // your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";

        // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations.
        // Using these credentials to perform operations in OSS is a high-risk operation. We
        // recommend that you use a RAM user to call API operations or perform routine O&M. To create
        // a RAM user, log on to the RAM console.
    }
}

```

```

        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the bucket name. Example: examplebucket.
        String bucketName = "examplebucket";
        // Specify the prefix that must be included in the names of objects to list. Exam
ple: exampledir/object.
        String prefix = "exampledir/object";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecr
et);
        try {
            // List objects whose names contain the specified prefix.
            ListObjectsV2Request listObjectsV2Request = new ListObjectsV2Request(bucketNa
me);

            listObjectsV2Request.setPrefix(prefix);
            ListObjectsV2Result result = ossClient.listObjectsV2(listObjectsV2Request);
            List<OSSObjectSummary> ossObjectSummaries = result.getObjectSummaries();
            for (OSSObjectSummary s : ossObjectSummaries) {
                System.out.println("\t" + s.getKey());
            }
        } catch (OSSEException oe) {
            System.out.println("Caught an OSSEException, which means your request made it
to OSS, "
                + "but was rejected with an error response for some reason.");
            System.out.println("Error Message:" + oe.getErrorMessage());
            System.out.println("Error Code:" + oe.getErrorCode());
            System.out.println("Request ID:" + oe.getRequestId());
            System.out.println("Host ID:" + oe.getHostId());
        } catch (ClientException ce) {
            System.out.println("Caught an ClientException, which means the client encount
ered "
                + "a serious internal problem while trying to communicate with OSS, "
                + "such as not being able to access the network.");
            System.out.println("Error Message:" + ce.getMessage());
        } finally {
            if (ossClient != null) {
                ossClient.shutdown();
            }
        }
    }
}

```

- List objects whose names are alphabetically after the object specified by startAfter

The following code provides an example on how to list objects whose names are alphabetically after the object specified by startAfter:

```

import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSEException;
import com.aliyun.oss.model.*;
import java.util.List;
public class Demo {
    public static void main(String[] args) throws Exception {

```

```

// In this example, the endpoint of the China (Hangzhou) region is used. Specify
your actual endpoint.
String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
// The AccessKey pair of an Alibaba Cloud account has permissions on all API oper-
ations. Using these credentials to perform operations in OSS is a high-risk operation. We
recommend that you use a RAM user to call API operations or perform routine O&M. To creat
e a RAM user, log on to the RAM console.
String accessKeyId = "yourAccessKeyId";
String accessKeySecret = "yourAccessKeySecret";
// Specify the bucket name. Example: examplebucket.
String bucketName = "examplebucket";
// Create an OSSClient instance.
OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecr
et);
try {
    // List objects whose names are alphabetically after the object specified by
startAfter. The value of startAfter can be the name of an existing object or other string
s.
    // For example, if you set startAfter to test-obj, objects whose names are al
phabetically after the test-obj string are returned. The object named test-obj is not lis
ted even if it exists in the bucket.
    ListObjectsV2Request listObjectsV2Request = new ListObjectsV2Request(bucketNa
me);
    listObjectsV2Request.setStartAfter("test-obj");
    ListObjectsV2Result result = ossClient.listObjectsV2(listObjectsV2Request);
    List<OSSObjectSummary> ossObjectSummaries = result.getObjectSummaries();
    for (OSSObjectSummary s : ossObjectSummaries) {
        System.out.println("\t" + s.getKey());
    }
} catch (OSSEException oe) {
    System.out.println("Caught an OSSEException, which means your request made it
to OSS, "
        + "but was rejected with an error response for some reason.");
    System.out.println("Error Message:" + oe.getErrorMessage());
    System.out.println("Error Code:" + oe.getErrorCode());
    System.out.println("Request ID:" + oe.getRequestId());
    System.out.println("Host ID:" + oe.getHostId());
} catch (ClientException ce) {
    System.out.println("Caught an ClientException, which means the client encount
ered "
        + "a serious internal problem while trying to communicate with OSS, "
        + "such as not being able to access the network.");
    System.out.println("Error Message:" + ce.getMessage());
} finally {
    if (ossClient != null) {
        ossClient.shutdown();
    }
}
}
}

```

- List objects and their owner information

The following code provides an example on how to list objects and their owner information:

```

import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.*;
import java.util.List;
public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify
        // your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API oper-
        // ations. Using these credentials to perform operations in OSS is a high-risk operation. We
        // recommend that you use a RAM user to call API operations or perform routine O&M. To creat-
        // e a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the bucket name. Example: examplebucket.
        String bucketName = "examplebucket";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySec-
        et);

        try {
            // By default, the owner information of the objects is not listed. To include
            // the object owner information in the response, set the fetchOwner parameter to true.
            ListObjectsV2Request listObjectsV2Request = new ListObjectsV2Request(bucketNa-
            me);

            listObjectsV2Request.setFetchOwner(true);
            ListObjectsV2Result result = ossClient.listObjectsV2(listObjectsV2Request);
            List<OSSObjectSummary> ossObjectSummaries = result.getObjectSummaries();
            for (OSSObjectSummary s : ossObjectSummaries) {
                System.out.println("\t" + s.getKey());
                if (s.getOwner() != null) {
                    System.out.println("owner id:" + s.getOwner().getId());
                    System.out.println("name:" + s.getOwner().getDisplayName());
                }
            }
        } catch (OSSException oe) {
            System.out.println("Caught an OSSException, which means your request made it
            to OSS, "
                + "but was rejected with an error response for some reason.");
            System.out.println("Error Message:" + oe.getErrorMessage());
            System.out.println("Error Code:" + oe.getErrorCode());
            System.out.println("Request ID:" + oe.getRequestId());
            System.out.println("Host ID:" + oe.getHostId());
        } catch (ClientException ce) {
            System.out.println("Caught an ClientException, which means the client encount-
            ered "
                + "a serious internal problem while trying to communicate with OSS, "
                + "such as not being able to access the network.");
            System.out.println("Error Message:" + ce.getMessage());
        } finally {
            if (ossClient != null) {
                ossClient.shutdown();
            }
        }
    }
}

```

```

    }
}
}
}

```

- List all objects in a bucket by page

The following code provides an example on how to list all objects in a specified bucket by page. You can set `maxKeys` to specify the maximum number of objects that can be listed on each page.

```

import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.*;
import java.util.List;

public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify
        // your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API oper-
        // ations. Using these credentials to perform operations in OSS is a high-risk operation. We
        // recommend that you use a RAM user to call API operations or perform routine O&M. To creat-
        // e a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the bucket name. Example: examplebucket.
        String bucketName = "examplebucket";
        // Specify the maximum number of objects that can be listed each time.
        int maxKeys = 200;
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecr-
        et);

        try {
            String nextContinuationToken = null;
            ListObjectsV2Result result = null;
            // List objects by page by using the nextContinuationToken parameter included
            // in the response of the previous list operation.
            do {
                ListObjectsV2Request listObjectsV2Request = new ListObjectsV2Request(bucket-
                etName).withMaxKeys(maxKeys);
                listObjectsV2Request.setContinuationToken(nextContinuationToken);
                result = ossClient.listObjectsV2(listObjectsV2Request);
                List<OSSObjectSummary> sums = result.getObjectSummaries();
                for (OSSObjectSummary s : sums) {
                    System.out.println("\t" + s.getKey());
                }
                nextContinuationToken = result.getNextContinuationToken();
            } while (result.isTruncated());
        } catch (OSSException oe) {
            System.out.println("Caught an OSSException, which means your request made it
            to OSS, "
                + "but was rejected with an error response for some reason.");
            System.out.println("Error Message:" + oe.getErrorMessage());

```



```

        System.out.println("Error Code:" + oe.getErrorCode());
        System.out.println("Request ID:" + oe.getRequestId());
        System.out.println("Host ID:" + oe.getHostId());
    } catch (ClientException ce) {
        System.out.println("Caught an ClientException, which means the client encount
ered "
            + "a serious internal problem while trying to communicate with OSS, "
            + "such as not being able to access the network.");
        System.out.println("Error Message:" + ce.getMessage());
    } finally {
        if (ossClient != null) {
            ossClient.shutdown();
        }
    }
}
}

```

- List objects whose names contain a specified prefix by page

The following code provides an example on how to list objects whose names contain a specified prefix by page:

```

import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.*;
import java.util.List;

public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify
        your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API oper
        ations. Using these credentials to perform operations in OSS is a high-risk operation. We
        recommend that you use a RAM user to call API operations or perform routine O&M. To creat
        e a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the bucket name. Example: examplebucket.
        String bucketName = "examplebucket";
        // Specify the prefix that must be included in the names of objects to list. Exam
        ple: exampledir/object.
        String keyPrefix = "exampledir/object";
        // Specify the maximum number of objects that can be listed each time.
        int maxKeys = 200;
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecr
        et);

        try {
            String nextContinuationToken = null;
            ListObjectsV2Result result = null;
            // List the objects whose names contain the specified prefix by page.
            do {
                ListObjectsV2Request listObjectsV2Request = new ListObjectsV2Request(buck

```

```

etName).withMaxKeys(maxKeys);
        listObjectsV2Request.setPrefix(keyPrefix);
        listObjectsV2Request.setContinuationToken(nextContinuationToken);
        result = ossClient.listObjectsV2(listObjectsV2Request);
        List<OSSObjectSummary> sums = result.getObjectSummaries();
        for (OSSObjectSummary s : sums) {
            System.out.println("\t" + s.getKey());
        }
        nextContinuationToken = result.getNextContinuationToken();
    } while (result.isTruncated());
} catch (OSSEException oe) {
    System.out.println("Caught an OSSEException, which means your request made it
to OSS, "
        + "but was rejected with an error response for some reason.");
    System.out.println("Error Message:" + oe.getErrorMessage());
    System.out.println("Error Code:" + oe.getErrorCode());
    System.out.println("Request ID:" + oe.getRequestId());
    System.out.println("Host ID:" + oe.getHostId());
} catch (ClientException ce) {
    System.out.println("Caught an ClientException, which means the client encount
ered "
        + "a serious internal problem while trying to communicate with OSS, "
        + "such as not being able to access the network.");
    System.out.println("Error Message:" + ce.getMessage());
} finally {
    if (ossClient != null) {
        ossClient.shutdown();
    }
}
}
}

```

- List objects and specify the encoding type of the object names

 **Note** For more information, see [List objects and specify the encoding type of the object names](#).

The following code provides an example on how to list objects and specify the encoding type of the object names:

```

import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSEException;
import com.aliyun.oss.model.*;
import java.net.URLDecoder;

public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify
        your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API oper
        ations. Using these credentials to perform operations in OSS is a high-risk operation. We
        recommend that you use a RAM user to call API operations or perform routine O&M. To creat
    }
}

```

```

e a RAM user, log on to the RAM console.
    String accessKeyId = "yourAccessKeyId";
    String accessKeySecret = "yourAccessKeySecret";
    // Specify the bucket name. Example: examplebucket.
    String bucketName = "examplebucket";
    // Specify the prefix that must be included in the names of objects to list. Exam
    ple: exampledir/object.
    String keyPrefix = "exampledir/object";
    // Specify the maximum number of objects that can be listed each time.
    int maxKeys = 200;
    // Create an OSSClient instance.
    OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecr
    et);

    try {
        String nextContinuationToken = null;
        ListObjectsV2Result result = null;
        // Specify that the returned results are URL-encoded. In this case, you must
        decode the values of the prefix, delimiter, startAfter, key, and commonPrefix parameters
        in the response.
        do {
            ListObjectsV2Request listObjectsV2Request = new ListObjectsV2Request(bucket
            etName).withMaxKeys(maxKeys);
            listObjectsV2Request.setPrefix(keyPrefix);
            listObjectsV2Request.setEncodingType("url");
            listObjectsV2Request.setContinuationToken(nextContinuationToken);
            result = ossClient.listObjectsV2(listObjectsV2Request);
            // Decode the value of prefix in the response.
            if (result.getPrefix() != null) {
                String prefix = URLDecoder.decode(result.getPrefix(), "UTF-8");
                System.out.println("prefix: " + prefix);
            }
            // Decode the value of delimiter in the response.
            if (result.getDelimiter() != null) {
                String delimiter = URLDecoder.decode(result.getDelimiter(), "UTF-8");
                System.out.println("delimiter: " + delimiter);
            }
            // Decode the value of startAfter in the response.
            if (result.getStartAfter() != null) {
                String startAfter = URLDecoder.decode(result.getStartAfter(), "UTF-8"
            );

                System.out.println("startAfter: " + startAfter);
            }
            // Decode the value of key in the response.
            for (OSSObjectSummary s : result.getObjectSummaries()) {
                String decodedKey = URLDecoder.decode(s.getKey(), "UTF-8");
                System.out.println("key: " + decodedKey);
            }
            // Decode the value of commonPrefixes in the response.
            for (String commonPrefix: result.getCommonPrefixes()) {
                String decodeCommonPrefix = URLDecoder.decode(commonPrefix, "UTF-8");
                System.out.println("CommonPrefix:" + decodeCommonPrefix);
            }
            nextContinuationToken = result.getNextContinuationToken();
        } while (result.isTruncated());
    }

```

```

    } catch (OSSException oe) {
        System.out.println("Caught an OSSException, which means your request made it
to OSS, "
            + "but was rejected with an error response for some reason.");
        System.out.println("Error Message:" + oe.getErrorMessage());
        System.out.println("Error Code:" + oe.getErrorCode());
        System.out.println("Request ID:" + oe.getRequestId());
        System.out.println("Host ID:" + oe.getHostId());
    } catch (ClientException ce) {
        System.out.println("Caught an ClientException, which means the client encount
ered "
            + "a serious internal problem while trying to communicate with OSS, "
            + "such as not being able to access the network.");
        System.out.println("Error Message:" + ce.getMessage());
    } finally {
        if (ossClient != null) {
            ossClient.shutdown();
        }
    }
}
}

```

List objects by directory

OSS uses a flat structure for objects instead of a hierarchical structure to store objects. A directory is an object whose size is 0 and whose name ends with a forward slash (/). You can upload and download this object. By default, objects whose names end with a forward slash (/) are displayed as directories in the OSS console. For the complete sample code that is used to create directories, visit [GitHub](#).

You can specify the delimiter and prefix parameters to list objects by directory.

- If you set prefix to a directory name in the request, objects and subdirectories whose names contain the prefix are listed.
- If you also set delimiter to a forward slash (/) in the request, the objects and subdirectories whose names start with the specified prefix in the directory are listed. Each subdirectory is listed as a single result element in commonPrefixes. The objects and directories in these subdirectories are not listed.

Example: The following four objects are stored in a bucket: `oss.jpg`, `fun/test.jpg`, `fun/movie/001.avi`, and `fun/movie/007.avi`. The forward slash (/) is specified as the directory delimiter. The following code provides examples on how to list objects by directory.

- List all objects in a bucket
 - The following code provides an example on how to call GetBucket (ListObjects) to list all objects in a bucket:

```

import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.*;
public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specif
y your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";

```

```

// The AccessKey pair of an Alibaba Cloud account has permissions on all API op
erations. Using these credentials to perform operations in OSS is a high-risk operation
. We recommend that you use a RAM user to call API operations or perform routine O&M. T
o create a RAM user, log on to the RAM console.
String accessKeyId = "yourAccessKeyId";
String accessKeySecret = "yourAccessKeySecret";
// Specify the bucket name. Example: examplebucket.
String bucketName = "examplebucket";
// Create an OSSClient instance.
OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySe
cret);
try {
    // Create a request to list objects.
    ListObjectsRequest listObjectsRequest = new ListObjectsRequest(bucketName);
    // List objects.
    ObjectListing listing = ossClient.listObjects(listObjectsRequest);
    // Traverse all objects.
    System.out.println("Objects:");
    for (OSSObjectSummary objectSummary : listing.getObjectSummaries()) {
        System.out.println(objectSummary.getKey());
    }
    // Traverse all commonPrefix elements.
    System.out.println("CommonPrefixes:");
    for (String commonPrefix : listing.getCommonPrefixes()) {
        System.out.println(commonPrefix);
    }
} catch (OSSException oe) {
    System.out.println("Caught an OSSException, which means your request made i
t to OSS, "
        + "but was rejected with an error response for some reason.");
    System.out.println("Error Message:" + oe.getErrorMessage());
    System.out.println("Error Code:" + oe.getErrorCode());
    System.out.println("Request ID:" + oe.getRequestId());
    System.out.println("Host ID:" + oe.getHostId());
} catch (ClientException ce) {
    System.out.println("Caught an ClientException, which means the client encou
ntered "
        + "a serious internal problem while trying to communicate with OSS,
"
        + "such as not being able to access the network.");
    System.out.println("Error Message:" + ce.getMessage());
} finally {
    if (ossClient != null) {
        ossClient.shutdown();
    }
}
}

```

- o The following code provides an example on how to call GetBucketV2 (ListObjectsV2) to list all objects in a specified bucket:

```

import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;

```

```

import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.*;
public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the bucket name. Example: examplebucket.
        String bucketName = "examplebucket";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);

        try {
            // List objects.
            ListObjectsV2Request listObjectsV2Request = new ListObjectsV2Request(bucketName);

            ListObjectsV2Result result = ossClient.listObjectsV2(listObjectsV2Request);
            // Traverse all objects.
            System.out.println("Objects:");
            for (OSSObjectSummary objectSummary : result.getObjectSummaries()) {
                System.out.println(objectSummary.getKey());
            }
            // Traverse all commonPrefix elements.
            System.out.println("CommonPrefixes:");
            for (String commonPrefix : result.getCommonPrefixes()) {
                System.out.println(commonPrefix);
            }
        } catch (OSSException oe) {
            System.out.println("Caught an OSSException, which means your request made it to OSS, "
                + "but was rejected with an error response for some reason.");
            System.out.println("Error Message:" + oe.getErrorMessage());
            System.out.println("Error Code:" + oe.getErrorCode());
            System.out.println("Request ID:" + oe.getRequestId());
            System.out.println("Host ID:" + oe.getHostId());
        } catch (ClientException ce) {
            System.out.println("Caught an ClientException, which means the client encountered "
                + "a serious internal problem while trying to communicate with OSS, "
                + "such as not being able to access the network.");
            System.out.println("Error Message:" + ce.getMessage());
        } finally {
            if (ossClient != null) {
                ossClient.shutdown();
            }
        }
    }
}

```

```
}
```

- Returned results

The following response is returned when you call the preceding two operations to list all objects in the bucket:

```
Objects:
fun/movie/001.avi
fun/movie/007.avi
fun/test.jpg
oss.jpg
CommonPrefixes:
```

- List all objects in a specified directory

- The following code provides an example on how to call GetBucket (List Objects) to list all objects in a specified directory:

```
import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.*;
public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the bucket name. Example: examplebucket.
        String bucketName = "examplebucket";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);
        try {
            // Create a request to list objects.
            ListObjectsRequest listObjectsRequest = new ListObjectsRequest(bucketName);
            // Set prefix to fun/ to list all objects in the fun/ directory.
            listObjectsRequest.setPrefix("fun/");
            // Lists all objects in the fun/ directory.
            ObjectListing listing = ossClient.listObjects(listObjectsRequest);
            // Traverse all objects.
            System.out.println("Objects:");
            for (OSSObjectSummary objectSummary : listing.getObjectSummaries()) {
                System.out.println(objectSummary.getKey());
            }
            // Traverse all commonPrefix elements.
            System.out.println("\nCommonPrefixes:");
            for (String commonPrefix : listing.getCommonPrefixes()) {
                System.out.println(commonPrefix);
            }
        } catch (OSSException oe) {
            System.out.println("An exception occurred: " + oe.getMessage());
        } catch (ClientException ce) {
            System.out.println("An exception occurred: " + ce.getMessage());
        }
    }
}
```

```

    }
    } catch (OSSEException oe) {
        System.out.println("Caught an OSSEException, which means your request made i
t to OSS, "
            + "but was rejected with an error response for some reason.");
        System.out.println("Error Message:" + oe.getErrorMessage());
        System.out.println("Error Code:" + oe.getErrorCode());
        System.out.println("Request ID:" + oe.getRequestId());
        System.out.println("Host ID:" + oe.getHostId());
    } catch (ClientException ce) {
        System.out.println("Caught an ClientException, which means the client encou
ntered "
            + "a serious internal problem while trying to communicate with OSS,
"
            + "such as not being able to access the network.");
        System.out.println("Error Message:" + ce.getMessage());
    } finally {
        if (ossClient != null) {
            ossClient.shutdown();
        }
    }
}
}
}

```

- o The following code provides an example on how to call GetBucketV2 (ListObjectsV2) to list all objects in a directory:

```

import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSEException;
import com.aliyun.oss.model.*;
public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specif
y your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API op
erations. Using these credentials to perform operations in OSS is a high-risk operation
. We recommend that you use a RAM user to call API operations or perform routine O&M. T
o create a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the bucket name. Example: examplebucket.
        String bucketName = "examplebucket";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySe
cret);
        try {
            // Create a request to list objects.
            ListObjectsV2Request listObjectsV2Request = new ListObjectsV2Request(bucket
Name);

            // Set prefix to fun/ to list all objects in the fun/ directory.
            listObjectsV2Request.setPrefix("fun/");

```



```

        // Initiate a list request.
        ListObjectsV2Result result = ossClient.listObjectsV2(listObjectsV2Request);
        // Traverse all objects.
        System.out.println("Objects:");
        for (OSSObjectSummary objectSummary : result.getObjectSummaries()) {
            System.out.println(objectSummary.getKey());
        }
        // Traverse all commonPrefix elements.
        System.out.println("\nCommonPrefixes:");
        for (String commonPrefix : result.getCommonPrefixes()) {
            System.out.println(commonPrefix);
        }
    } catch (OSSException oe) {
        System.out.println("Caught an OSSException, which means your request made it to OSS, "
            + "but was rejected with an error response for some reason.");
        System.out.println("Error Message:" + oe.getErrorMessage());
        System.out.println("Error Code:" + oe.getErrorCode());
        System.out.println("Request ID:" + oe.getRequestId());
        System.out.println("Host ID:" + oe.getHostId());
    } catch (ClientException ce) {
        System.out.println("Caught an ClientException, which means the client encountered "
            + "a serious internal problem while trying to communicate with OSS, "
            + "such as not being able to access the network.");
        System.out.println("Error Message:" + ce.getMessage());
    } finally {
        if (ossClient != null) {
            ossClient.shutdown();
        }
    }
}
}

```

- The following response is returned when you call the preceding two operations to list all objects in the specified directory:

```

Objects:
fun/movie/001.avi
fun/movie/007.avi
fun/test.jpg
CommonPrefixes:

```

- List objects and subdirectories in the specified directory
 - The following code provides an example on how to call GetBucket (List Objects) to list objects and subdirectories in a directory:

```

import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.*;
public class Demo {
    public static void main(String[] args) throws Exception {

```

```

public static void main(String[] args) throws Exception {
    // In this example, the endpoint of the China (Hangzhou) region is used. Specify
    // your actual endpoint.
    String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
    // The AccessKey pair of an Alibaba Cloud account has permissions on all API op-
    // erations. Using these credentials to perform operations in OSS is a high-risk operation
    // . We recommend that you use a RAM user to call API operations or perform routine O&M. To
    // create a RAM user, log on to the RAM console.
    String accessKeyId = "yourAccessKeyId";
    String accessKeySecret = "yourAccessKeySecret";
    // Specify the bucket name. Example: examplebucket.
    String bucketName = "examplebucket";
    // Create an OSSClient instance.
    OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySe-
    cret);

    try {
        // Create a request to list objects.
        ListObjectsRequest listObjectsRequest = new ListObjectsRequest(bucketName);
        // Set delimiter to a forward slash (/).
        listObjectsRequest.setDelimiter("/");
        // List all objects and subdirectories in the fun/ directory.
        listObjectsRequest.setPrefix("fun/");
        ObjectListing listing = ossClient.listObjects(listObjectsRequest);
        // Traverse all objects.
        System.out.println("Objects:");
        // In the response, objectSummaries lists the objects in the fun/ directory
        .
        for (OSSObjectSummary objectSummary : listing.getObjectSummaries()) {
            System.out.println(objectSummary.getKey());
        }
        // Traverse all commonPrefix elements.
        System.out.println("\nCommonPrefixes:");
        // CommonPrefixes lists all subdirectories in the fun/ directory. The fun/mo-
        // vie/001.avi and fun/movie/007.avi objects are not listed because they are in the movie
        // subdirectory of the fun/ directory.
        for (String commonPrefix : listing.getCommonPrefixes()) {
            System.out.println(commonPrefix);
        }
    } catch (OSSException oe) {
        System.out.println("Caught an OSSException, which means your request made it
        to OSS, "
            + "but was rejected with an error response for some reason.");
        System.out.println("Error Message:" + oe.getErrorMessage());
        System.out.println("Error Code:" + oe.getErrorCode());
        System.out.println("Request ID:" + oe.getRequestId());
        System.out.println("Host ID:" + oe.getHostId());
    } catch (ClientException ce) {
        System.out.println("Caught an ClientException, which means the client encou-
        ntered "
            + "a serious internal problem while trying to communicate with OSS,
        "
            + "such as not being able to access the network.");
        System.out.println("Error Message:" + ce.getMessage());
    } finally {
        if (ossClient != null) {

```

```

        ossClient.shutdown();
    }
}
}
}

```

- o The following code provides an example on how to call GetBucketV2 (ListObjectsV2) to list objects and subdirectories in a directory:

```

import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.*;
public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the bucket name. Example: examplebucket.
        String bucketName = "examplebucket";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);
        try {
            // Create a request to list objects.
            ListObjectsV2Request listObjectsV2Request = new ListObjectsV2Request(bucketName);
            // Set prefix to fun/ to list all objects and subdirectories in the fun/ directory.
            listObjectsV2Request.setPrefix("fun/");
            // Set delimiter to a forward slash (/).
            listObjectsV2Request.setDelimiter("/");
            // Initiate a list request.
            ListObjectsV2Result result = ossClient.listObjectsV2(listObjectsV2Request);
            // Traverse all objects.
            System.out.println("Objects:");
            // In the response, objectSummaries lists the objects in the fun/ directory.
            for (OSSObjectSummary objectSummary : result.getObjectSummaries()) {
                System.out.println(objectSummary.getKey());
            }
            // Traverse all commonPrefix elements.
            System.out.println("\nCommonPrefixes:");
            // CommonPrefixes lists all subdirectories in the fun/ directory. The fun/movie/001.avi and fun/movie/007.avi objects are not listed because they are in the movie subdirectory of the fun/ directory.
            for (String commonPrefix : result.getCommonPrefixes()) {

```

```

        System.out.println(commonPrefix);
    }
} catch (OSSException oe) {
    System.out.println("Caught an OSSException, which means your request made i
t to OSS, "
        + "but was rejected with an error response for some reason.");
    System.out.println("Error Message:" + oe.getErrorMessage());
    System.out.println("Error Code:" + oe.getErrorCode());
    System.out.println("Request ID:" + oe.getRequestId());
    System.out.println("Host ID:" + oe.getHostId());
} catch (ClientException ce) {
    System.out.println("Caught an ClientException, which means the client encou
ntered "
        + "a serious internal problem while trying to communicate with OSS,
"
        + "such as not being able to access the network.");
    System.out.println("Error Message:" + ce.getMessage());
} finally {
    if (ossClient != null) {
        ossClient.shutdown();
    }
}
}
}

```

- The following response is returned when you call the preceding two operations to list objects and subdirectories in the specified directory:

```

Objects:
fun/test.jpg
CommonPrefixes:
fun/movie/

```

- Obtain the sizes of objects in the specified directory
 - The following code provides an example on how to call GetBucket (List Objects) to obtain the sizes of objects in a specified directory:

```

import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.*;
import java.util.List;

public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specif
y your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API op
erations. Using these credentials to perform operations in OSS is a high-risk operation
. We recommend that you use a RAM user to call API operations or perform routine O&M. T
o create a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the bucket name. Example: examplebucket.

```

```

// Specify the bucket name. Example: examplebucket.
String bucketName = "examplebucket";
// Specify the prefix that must be included in the names of objects to list. Ex
ample: exampledir/object. If you want to traverse the sizes of objects and directories
within the root directory, leave this parameter empty.
String keyPrefix = "exampledir/object";
// Create an OSSClient instance.
OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySe
cret);

try {
    ObjectListing objectListing = null;
    do {
        // By default, 100 objects or directories are listed each time.
        ListObjectsRequest request = new ListObjectsRequest(bucketName).withDel
imiter("/").withPrefix(keyPrefix);
        if (objectListing != null) {
            request.setMarker(objectListing.getNextMarker());
        }
        objectListing = ossClient.listObjects(request);
        List<String> folders = objectListing.getCommonPrefixes();
        for (String folder : folders) {
            System.out.println(folder + " : " + (calculateFolderLength(ossClien
t, bucketName, folder) / 1024) + "KB");
        }
        List<OSSObjectSummary> sums = objectListing.getObjectSummaries();
        for (OSSObjectSummary s : sums) {
            System.out.println(s.getKey() + " : " + (s.getSize() / 1024) + "KB"
);
        }
    } while (objectListing.isTruncated());
} catch (OSSEException oe) {
    System.out.println("Caught an OSSEException, which means your request made i
t to OSS, "

        + "but was rejected with an error response for some reason.");
    System.out.println("Error Message:" + oe.getErrorMessage());
    System.out.println("Error Code:" + oe.getErrorCode());
    System.out.println("Request ID:" + oe.getRequestId());
    System.out.println("Host ID:" + oe.getHostId());
} catch (ClientException ce) {
    System.out.println("Caught an ClientException, which means the client encou
ntered "

        + "a serious internal problem while trying to communicate with OSS,

        + "such as not being able to access the network.");
    System.out.println("Error Message:" + ce.getMessage());
} finally {
    if (ossClient != null) {
        ossClient.shutdown();
    }
}

// Query the sizes of objects within a specified directory in a bucket.
private static long calculateFolderLength(OSS ossClient, String bucketName, String
folder) {
    long size = 0L;

```

```

        ObjectListing objectListing = null;
        do {
            // The default value of maxKeys is 100. The maximum value of maxKeys is 1000.
            ListObjectsRequest request = new ListObjectsRequest(bucketName).withPrefix(folder).withMaxKeys(1000);
            if (objectListing != null) {
                request.setMarker(objectListing.getNextMarker());
            }
            objectListing = ossClient.listObjects(request);
            List<OSSObjectSummary> sums = objectListing.getObjectSummaries();
            for (OSSObjectSummary s : sums) {
                size += s.getSize();
            }
        } while (objectListing.isTruncated());
        return size;
    }
}

```

- o The following code provides an example on how to call GetBucketV2 (ListObjectsV2) to query the sizes of objects in a specified directory:

```

import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.*;
import java.util.List;

public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the bucket name. Example: examplebucket.
        String bucketName = "examplebucket";
        // Specify the prefix that must be included in the names of objects to list. Example: exampledir/object. If you want to traverse the sizes of objects and directories within the root directory, leave this parameter empty.
        String keyPrefix = "exampledir/object";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);

        try {
            ListObjectsV2Result result = null;
            do {
                // By default, 100 objects or directories are listed each time.
                ListObjectsV2Request request = new ListObjectsV2Request(bucketName).withDelimiter("/") .withPrefix(keyPrefix);
                result = ossClient.listObjectsV2(request);
            } while (result.isTruncated());
        } catch (OSSException oe) {
            System.out.println("Caught an OSSException, which means that some SDK errors occurred but the OSS servers did not respond with an error code.");
            System.out.println(oe.getMessage());
        } catch (ClientException ce) {
            System.out.println("Caught a ClientException, which means the client's request was valid but a server-side error occurred. This error code should not be caught and should be reported to a customer representative as a bug.");
            System.out.println(ce.getMessage());
        }
    }
}

```

```

        if (result != null) {
            request.setContinuationToken(result.getNextContinuationToken());
        }
        result = ossClient.listObjectsV2(request);
        List<String> folders = result.getCommonPrefixes();
        for (String folder : folders) {
            System.out.println(folder + " : " + (calculateFolderLength(ossClient, bucketName, folder) / 1024) + "KB");
        }
        List<OSSObjectSummary> sums = result.getObjectSummaries();
        for (OSSObjectSummary s : sums) {
            System.out.println(s.getKey() + " : " + (s.getSize() / 1024) + "KB");
        }
    } while (result.isTruncated());
} catch (OSSEException oe) {
    System.out.println("Caught an OSSEException, which means your request made it to OSS, "
        + "but was rejected with an error response for some reason.");
    System.out.println("Error Message:" + oe.getErrorMessage());
    System.out.println("Error Code:" + oe.getErrorCode());
    System.out.println("Request ID:" + oe.getRequestId());
    System.out.println("Host ID:" + oe.getHostId());
} catch (ClientException ce) {
    System.out.println("Caught an ClientException, which means the client encountered "
        + "a serious internal problem while trying to communicate with OSS, "
        + "such as not being able to access the network.");
    System.out.println("Error Message:" + ce.getMessage());
} finally {
    if (ossClient != null) {
        ossClient.shutdown();
    }
}

// Query the sizes of objects within a specified directory in a bucket.
private static long calculateFolderLength(OSS ossClient, String bucketName, String folder) {
    long size = 0L;
    ListObjectsV2Result result = null;
    do {
        // The default value of maxKeys is 100. The maximum value of maxKeys is 1000.
        ListObjectsV2Request request = new ListObjectsV2Request(bucketName).withPrefix(folder).withMaxKeys(1000);
        if (result != null) {
            request.setContinuationToken(result.getNextContinuationToken());
        }
        result = ossClient.listObjectsV2(request);
        List<OSSObjectSummary> sums = result.getObjectSummaries();
        for (OSSObjectSummary s : sums) {
            size += s.getSize();
        }
    } while (result.isTruncated());
}

```

```

        } while (result.isTruncated());
        return size;
    }
}

```

References

- For the complete sample code that is used to list objects, visit [GitHub](#).
- For more information about the API operations that you can call to list objects, see [GetBucket \(List Objects\)](#) and [GetBucketV2 \(List ObjectsV2\)](#).

2.8.7. Query objects

This topic describes how to call `SelectObject` to query CSV and JSON objects in a bucket by using Object Storage Service (OSS) SDK for Java.

 **Note** For more information about the `SelectObject` operation, see [Call the SelectObject operation to query objects](#) in OSS Developer Guide and [SelectObject](#) in OSS API Reference.

The following code provides an example on how to query CSV and JSON objects:

```

import com.aliyun.oss.model.*;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import java.io.BufferedReader;
import java.io.ByteArrayInputStream;
import java.io.InputStreamReader;
/**
 * Examples of create select object metadata and select object.
 *
 */
public class SelectObjectSample {
    // Set yourEndpoint to the endpoint of the region in which the bucket is located. For e
    // xample, if your bucket is located in the China (Hangzhou) region, set yourEndpoint to https
    // ://oss-cn-hangzhou.aliyuncs.com.
    private static String endpoint = "yourEndpoint";
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account t
    // o access OSS because the account has permissions on all API operations. We recommend that y
    // ou use a Resource Access Management (RAM) user to call API operations or perform routine O&
    // M. To create a RAM user, log on to the RAM console.
    private static String accessKeyId = "yourAccessKeyId";
    private static String accessKeySecret = "yourAccessKeySecret";
    // Specify the name of the bucket. Example: examplebucket.
    private static String bucketName = "examplebucket";
    public static void main(String[] args) throws Exception {
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret
    );

        // Specify the full path of the object that you want to query, and then query the d
        // ata of the object by using SELECT statements. The full path of the object cannot contain bu
        // cket names.
        // Specify the full path of the CSV object.
        selectCsvSample("test.csv", ossClient);
        // Specify the full path of the JSON object.
        selectJsonSample("test.json", ossClient);
    }
}

```



```

        selectCsvSample( test.json , ossClient,
        ossClient.shutdown();
    }

    private static void selectCsvSample(String key, OSS ossClient) throws Exception {
        // Specify the content of the sample object you want to upload.
        String content = "name,school,company,age\r\n" +
            "Lora Francis,School A,Staples Inc,27\r\n" +
            "Eleanor Little,School B,\"Conectiv, Inc\",43\r\n" +
            "Rosie Hughes,School C,Western Gas Resources Inc,44\r\n" +
            "Lawrence Ross,School D,MetLife Inc.,24";
        ossClient.putObject(bucketName, key, new ByteArrayInputStream(content.getBytes()));
        SelectObjectMetadata selectObjectMetadata = ossClient.createSelectObjectMetadata(
            new CreateSelectObjectMetadataRequest(bucketName, key)
                .withInputSerialization(
                    new InputSerialization().withCsvInputFormat(
                        // Specify the delimiter between different records
                        in the content. Example: \r\n.
                        new CSVFormat().withHeaderInfo(CSVFormat.Header.Use
                            ).withRecordDelimiter("\r\n"))));
        System.out.println(selectObjectMetadata.getCsvObjectMetadata().getTotalLines());
        System.out.println(selectObjectMetadata.getCsvObjectMetadata().getSplits());
        SelectObjectRequest selectObjectRequest =
            new SelectObjectRequest(bucketName, key)
                .withInputSerialization(
                    new InputSerialization().withCsvInputFormat(
                        new CSVFormat().withHeaderInfo(CSVFormat.Header.Use
                            ).withRecordDelimiter("\r\n")))
                .withOutputSerialization(new OutputSerialization().withCsvOutputFormat(
                    new CSVFormat()));
        selectObjectRequest.setExpression("select * from ossobject where _4 > 40"); // Use
        // SELECT statements to query data in the object.
        OSSObject ossObject = ossClient.selectObject(selectObjectRequest);
        // Read the content of the object.
        BufferedReader reader = new BufferedReader(new InputStreamReader(ossObject.getObjectContent()));
        while (true) {
            String line = reader.readLine();
            if (line == null) {
                break;
            }
            System.out.println(line);
        }
        reader.close();
        ossClient.deleteObject(bucketName, key);
    }

    private static void selectJsonSample(String key, OSS ossClient) throws Exception {
        // Specify the content to upload.
        final String content = "{\n" +
            "\t\"name\": \"Lora Francis\",\n" +
            "\t\"age\": 27,\n" +
            "\t\"company\": \"Staples Inc\"\n" +
            "}\n" +
            "{\n" +
            "\t\"name\": \"Eleanor Little\",\n" +
            "\t\"age\": 43,\n" +

```

```

        "\t\t\"company\": \"Conectiv, Inc\\\"\\n\" +
        \"}\\n\" +
        \"{\\n\" +
        \"\t\t\"name\": \"Rosie Hughes\\\",\\n\" +
        \"\t\t\"age\": 44,\\n\" +
        \"\t\t\"company\": \"Western Gas Resources Inc\\\"\\n\" +
        \"}\\n\" +
        \"{\\n\" +
        \"\t\t\"name\": \"Lawrence Ross\\\",\\n\" +
        \"\t\t\"age\": 24,\\n\" +
        \"\t\t\"company\": \"MetLife Inc.\\\"\\n\" +
        \"}\";

ossClient.putObject(bucketName, key, new ByteArrayInputStream(content.getBytes()));
SelectObjectRequest selectObjectRequest =
    new SelectObjectRequest(bucketName, key)
        .withInputSerialization(new InputSerialization()
            .withCompressionType(CompressionType.NONE)
            .withJsonInputFormat(new JsonFormat().withJsonType(JsonType
.LINES)))
        .withOutputSerialization(new OutputSerialization()
            .withCrcEnabled(true)
            .withJsonOutputFormat(new JsonFormat()))
        .withExpression("select * from ossobject as s where s.age > 40"); //
/ Use SELECT statements to query data in the object.
OSSObject ossObject = ossClient.selectObject(selectObjectRequest);
// Read the content of the object.
BufferedReader reader = new BufferedReader(new InputStreamReader(ossObject.getObjec
tContent()));
while (true) {
    String line = reader.readLine();
    if (line == null) {
        break;
    }
    System.out.println(line);
}
reader.close();
ossClient.deleteObject(bucketName, key);
}
}

```

2.8.8. Rename objects

After the hierarchical namespace feature is enabled for a bucket, you can rename objects in the bucket. This topic describes how to rename an object.

 **Note** For more information about specific operations related to the hierarchical namespace feature of buckets, see [Create a bucket](#).

The following code provides an example on how to change the name of exampleobject.txt object in examplebucket to newexampleobject.txt.

```
// Set yourEndpoint to the endpoint of the region where the bucket is located. For example,
// if your bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-c
n-hangzhou.aliyuncs.com.
String endpoint = "yourEndpoint";
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to lo
g on to OSS because the account has permissions on all API operations. We recommend that yo
u use your RAM user's credentials to call API operations or perform routine operations and
maintenance. To create a RAM user, log on to the RAM console.
String accessKeyId = "yourAccessKeyId";
String accessKeySecret = "yourAccessKeySecret";
// Specify the bucket name.
String bucketName = "examplebucket";
// Set the absolute path of the source object. The absolute path of the directory cannot co
ntain the bucket name.
String sourceObject = "exampleobject.txt";
// Set the absolute path of the destination object in the same bucket as that of the source
object. The absolute path of the object cannot contain the bucket name.
String destinationObject = "newexampleobject.txt";
// Create an OSSClient instance.
OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);
// Change the absolute path of the source object in the bucket to the absolute path of the
destination object.
RenameObjectRequest renameObjectRequest = new RenameObjectRequest(bucketName, sourceObject,
destinationObject);
ossClient.renameObject(renameObjectRequest);
// Shut down the OSSClient instance.
ossClient.shutdown();
```

For more information about how to rename objects, see [Rename](#).

2.8.9. Delete objects

You can delete a single object, multiple specified objects, objects whose names contain a specified prefix, or a specified directory and all objects in the directory.



Warning Deleted objects cannot be recovered. Exercise caution when you delete objects.

Delete a single object

The following code provides an example on how to delete an object named *exampleobject.txt* from a bucket named *examplebucket*:

```

import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in Object Storage Service (OSS) is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the bucket name. Example: examplebucket.
        String bucketName = "examplebucket";
        // Specify the full path of the object. The full path of the object cannot contain the bucket name.
        String objectName = "exampleobject.txt";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);

        try {
            // Delete the object or the directory. Before you delete a directory, make sure that the directory contains no objects.
            ossClient.deleteObject(bucketName, objectName);
        } catch (OSSException oe) {
            System.out.println("Caught an OSSException, which means your request made it to OSS, "
                + "but was rejected with an error response for some reason.");
            System.out.println("Error Message:" + oe.getErrorMessage());
            System.out.println("Error Code:" + oe.getErrorCode());
            System.out.println("Request ID:" + oe.getRequestId());
            System.out.println("Host ID:" + oe.getHostId());
        } catch (ClientException ce) {
            System.out.println("Caught an ClientException, which means the client encountered "
                + "a serious internal problem while trying to communicate with OSS, "
                + "such as not being able to access the network.");
            System.out.println("Error Message:" + ce.getMessage());
        } finally {
            if (ossClient != null) {
                ossClient.shutdown();
            }
        }
    }
}

```

Delete multiple objects

You can delete up to 1,000 objects at a time. You can delete multiple specified objects, objects whose names contain a specified prefix, or a specified directory and all objects in the directory.

You can also configure lifecycle rules to automatically delete objects. For more information, see [Lifecycle rules based on the last modified time](#).

Parameters that you can configure to delete multiple objects

- Request parameters

Parameter	Description	Method
Keys	The objects that you want to delete.	setKeys(List<String>)
quiet	The mode in which the result is returned. Default value: false. The result can be returned in one of the following modes: - verbose: A list of all deleted objects is returned. If quiet is not set or set to false, the result is returned in this mode. - quiet: No message body is returned. If quiet is set to true, the result is returned in this mode.	setQuiet(boolean)
encodingType	The method that is used to encode the object names in the result. Only URL encoding is supported.	setEncodingType(String)

- Response parameters

Parameter	Description	Method
deletedObjects	The result of the delete operation. In the verbose mode, a list of deleted objects is returned. In the quiet mode, a list of objects that fail to be deleted is returned.	List<String> getDeletedObjects()
encodingType	The method that is used to encode the object names in the result. If this parameter is empty, the object names are not encoded.	getEncodingType()

Delete multiple specified objects

The following code provides an example on how to delete multiple specified objects:

```
import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.DeleteObjectsRequest;
import com.aliyun.oss.model.DeleteObjectsResult;
import java.io.UnsupportedEncodingException;
import java.net.URLDecoder;
import java.util.ArrayList;
import java.util.List;
public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify yo
```

```

ur actual endpoint.
    String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
    // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
    String accessKeyId = "yourAccessKeyId";
    String accessKeySecret = "yourAccessKeySecret";
    // Specify the bucket name. Example: examplebucket.
    String bucketName = "examplebucket";
    // Create an OSSClient instance.
    OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);

    try {
        // Delete the objects.
        // Specify the full paths of the objects that you want to delete. The full paths of the objects cannot contain the bucket name.
        List<String> keys = new ArrayList<String>();
        keys.add("exampleobjecta.txt");
        keys.add("testfolder/sampleobject.txt");
        keys.add("exampleobjectb.txt");
        DeleteObjectsResult deleteObjectsResult = ossClient.deleteObjects(new DeleteObjectsRequest(bucketName).withKeys(keys).withEncodingType("url"));
        List<String> deletedObjects = deleteObjectsResult.getDeletedObjects();
        try {
            for(String obj : deletedObjects) {
                String deleteObj = URLDecoder.decode(obj, "UTF-8");
                System.out.println(deleteObj);
            }
        } catch (UnsupportedEncodingException e) {
            e.printStackTrace();
        }
    } catch (OSSException oe) {
        System.out.println("Caught an OSSException, which means your request made it to OSS, "
            + "but was rejected with an error response for some reason.");
        System.out.println("Error Message:" + oe.getErrorMessage());
        System.out.println("Error Code:" + oe.getErrorCode());
        System.out.println("Request ID:" + oe.getRequestId());
        System.out.println("Host ID:" + oe.getHostId());
    } catch (ClientException ce) {
        System.out.println("Caught an ClientException, which means the client encountered "
            + "a serious internal problem while trying to communicate with OSS, "
            + "such as not being able to access the network.");
        System.out.println("Error Message:" + ce.getMessage());
    } finally {
        if (ossClient != null) {
            ossClient.shutdown();
        }
    }
}

```

Delete multiple objects with a specified prefix or in a specified directory

The following code provides an example on how to delete multiple objects with a specified prefix and how to delete a specified directory and all objects in the directory.

 **Warning** If the prefix is not specified or is set to NULL in the following code, all objects in the bucket are deleted. Exercise caution when you configure the prefix.

```
import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.*;
import java.io.UnsupportedEncodingException;
import java.net.URLDecoder;
import java.util.ArrayList;
import java.util.List;
public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the bucket name. Example: examplebucket.
        String bucketName = "examplebucket";
        // Specify the prefix of the object that you want to delete. If you want to delete all objects whose names contain the src prefix, set the prefix to src. After you set the prefix to src, all non-directory objects whose names contain the src prefix, the src directory, and all objects in the src directory are deleted.
        String prefix = "src";
        // If you want to delete only the src directory and all objects in the directory, set the prefix to src/.
        // String prefix = "src/";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);
        try {
            // List all objects whose names contain the specified prefix and then delete the objects.
            String nextMarker = null;
            ObjectListing objectListing = null;
            do {
                ListObjectsRequest listObjectsRequest = new ListObjectsRequest(bucketName)
                    .withPrefix(prefix)
                    .withMarker(nextMarker);
                objectListing = ossClient.listObjects(listObjectsRequest);
                if (objectListing.getObjectSummaries().size() > 0) {
```

```

        List<String> keys = new ArrayList<String>();
        for (OSSObjectSummary s : objectListing.getObjectSummaries()) {
            System.out.println("key name: " + s.getKey());
            keys.add(s.getKey());
        }
        DeleteObjectsRequest deleteObjectsRequest = new DeleteObjectsRequest(bucketName).withKeys(keys).withEncodingType("url");
        DeleteObjectsResult deleteObjectsResult = ossClient.deleteObjects(deleteObjectsRequest);
        List<String> deletedObjects = deleteObjectsResult.getDeletedObjects();
        try {
            for(String obj : deletedObjects) {
                String deleteObj = URLDecoder.decode(obj, "UTF-8");
                System.out.println(deleteObj);
            }
        } catch (UnsupportedEncodingException e) {
            e.printStackTrace();
        }
        nextMarker = objectListing.getNextMarker();
    } while (objectListing.isTruncated());
} catch (OSSException oe) {
    System.out.println("Caught an OSSException, which means your request made it to
OSS, "
        + "but was rejected with an error response for some reason.");
    System.out.println("Error Message:" + oe.getErrorMessage());
    System.out.println("Error Code:" + oe.getErrorCode());
    System.out.println("Request ID:" + oe.getRequestId());
    System.out.println("Host ID:" + oe.getHostId());
} catch (ClientException ce) {
    System.out.println("Caught an ClientException, which means the client encounter
ed "
        + "a serious internal problem while trying to communicate with OSS, "
        + "such as not being able to access the network.");
    System.out.println("Error Message:" + ce.getMessage());
} finally {
    if (ossClient != null) {
        ossClient.shutdown();
    }
}
}
}

```

References

- Delete a single object

For more information about the API operation that you can call to delete a single object, see [DeleteObject](#).

- Delete multiple objects

- For the complete sample code that is used to delete multiple objects, visit [GitHub](#).

- For more information about the API operation that you can call to delete multiple objects, see [DeleteMultipleObjects](#).

2.8.10. Copy objects

This topic describes how to copy an object within a bucket or across buckets in the same region.

Precautions

- To copy an object, you must have read permissions on the source object and read and write permissions on the destination bucket.
- The source bucket and the destination bucket must have no retention policies configured. Otherwise, the error message `The object you specified is immutable.` is returned.
- The source bucket and the destination bucket must be in the same region. For example, objects cannot be copied from a bucket located in the China (Hangzhou) region to another bucket located in the China (Qingdao) region.

Copy a small object

You can use the `ossClient.copyObject` method to copy an object that is smaller than 1 GB in size. The following table describes the configuration methods for `ossClient.copyObject`.

Configuration method	Description
<code>CopyObjectResult copyObject(String sourceBucketName, String sourceKey, String destinationBucketName, String destinationKey)</code>	Allows you to specify the source bucket, source object, destination bucket, and destination object. After an object is copied, the content and metadata of the destination object are the same as the content and metadata of the source object. This method is called simple copy.
<code>CopyObjectResult copyObject(CopyObjectRequest copyObjectRequest)</code>	Allows you to specify the metadata of the destination object and copy conditions. If the URLs of the source object and destination object are the same, the metadata that is specified in the request replaces the metadata of the source object.

The following table describes the parameters that you can configure for `CopyObjectRequest`.

Parameter	Description	Method
<code>sourceBucketName</code>	The name of the source bucket.	<code>setSourceBucketName(String sourceBucketName)</code>
<code>sourceKey</code>	The name of the source object.	<code>setSourceKey(String sourceKey)</code>
<code>destinationBucketName</code>	The name of the destination bucket.	<code>setDestinationBucketName(String destinationBucketName)</code>
<code>destinationKey</code>	The name of the destination object.	<code>setDestinationKey(String destinationKey)</code>

Parameter	Description	Method
newObjectMetadata	The metadata of the destination object.	setNewObjectMetadata(ObjectMetadata newObjectMetadata)
matchingETagConstraints	The condition for the copy operation. If the ETag value of the source object is the same as the ETag value that is specified in the request, Object Storage Service (OSS) copies the object. Otherwise, an error is returned.	setMatchingETagConstraints(List<String> matchingETagConstraints)
nonmatchingETagConstraints	The condition for the copy operation. If the ETag value of the source object is different from the ETag value that is specified in the request, OSS copies the object. Otherwise, an error is returned.	setNonmatchingETagConstraints(List<String> nonmatchingETagConstraints)
unmodifiedSinceConstraint	The condition for the copy operation. If the time that is specified in the request is later than or equal to the actual modified time of the source object, OSS copies the object. Otherwise, an error is returned.	setUnmodifiedSinceConstraint(Date unmodifiedSinceConstraint)
modifiedSinceConstraint	The condition for the copy operation. If the source object is modified after the time that is specified in the request, OSS copies the object. Otherwise, an error is returned.	setModifiedSinceConstraint(Date modifiedSinceConstraint)

The following table describes the parameters that you can configure for CopyObjectResult.

Parameter	Description	Method
etag	The unique tag of the object.	String getETag()
lastModified	The last modified time of the object.	Date getLastModified()

You can use one of the following methods to copy small objects:

- Simple copy

The following code provides an example on how to use simple copy to copy an object named srcexampleobject.txt from the srcexamplebucket bucket to an object named desexampleobject.txt in the desexamplebucket bucket:

```
// Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com.
String endpoint = "yourEndpoint";
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
String accessKeyId = "yourAccessKeyId";
String accessKeySecret = "yourAccessKeySecret";
// Specify the name of the source bucket.
String sourceBucketName = "srcexamplebucket";
// Specify the full path of the source object. The full path of the object cannot contain the bucket name.
String sourceKey = "srcexampleobject.txt";
// Specify the name of the destination bucket. The destination bucket must be in the same region as the source bucket.
String destinationBucketName = "desexamplebucket";
// Specify the full path of the destination object. The full path of the object cannot contain the bucket name.
String destinationKey = "desexampleobject.txt";
// Create an OSSClient instance.
OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);
// Copy the object.
CopyObjectResult result = ossClient.copyObject(sourceBucketName, sourceKey, destinationBucketName, destinationKey);
System.out.println("ETag: " + result.getETag() + " LastModified: " + result.getLastModified());
// Shut down the OSSClient instance.
ossClient.shutdown();
```

- Use CopyObjectRequest to copy objects

The following code provides an example on how to use CopyObjectRequest to copy an object named srcexampleobject.txt from the srcexamplebucket bucket to an object named desexampleobject.txt in the desexamplebucket bucket:

```
// Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com.
String endpoint = "yourEndpoint";
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
String accessKeyId = "yourAccessKeyId";
String accessKeySecret = "yourAccessKeySecret";
// Specify the name of the source bucket.
String sourceBucketName = "srcexamplebucket";
// Specify the full path of the source object. The full path of the object cannot contain the bucket name.
String sourceKey = "srcexampleobject.txt";
// Specify the name of the destination bucket. The destination bucket must be in the same region as the source bucket.
String destinationBucketName = "desexamplebucket";
```

```

String destinationBucketName = "deexamplebucket";
// Specify the full path of the destination object. The full path of the object cannot contain the bucket name.
String destinationKey = "deexampleobject.txt";
// Create an OSSClient instance.
OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);
// Create a CopyObjectRequest object.
CopyObjectRequest copyObjectRequest = new CopyObjectRequest(sourceBucketName, sourceKey, destinationBucketName, destinationKey);
// Configure the metadata of the destination object.
ObjectMetadata meta = new ObjectMetadata();
meta.setContentType("text/txt");
// Specify whether the CopyObject operation overwrites the object with the same name. In this example, this parameter is set to true, which indicates that the object with the same name cannot be overwritten.
// meta.setHeader("x-oss-forbid-overwrite", "true");
// Specify the path of the source object.
// meta.setHeader(OSSHeaders.COPY_OBJECT_SOURCE, "/examplebucket/recode-test.txt");
// If the ETag value of the source object is the same as the ETag value that is specified in the request, OSS copies the object and returns 200 OK.
// meta.setHeader(OSSHeaders.COPY_OBJECT_SOURCE_IF_MATCH, "5B3C1A2E053D763E1B002CC607C5**");
// If the ETag value of the source object is different from the ETag value that is specified in the request, OSS copies the object and returns 200 OK.
// meta.setHeader(OSSHeaders.COPY_OBJECT_SOURCE_IF_NONE_MATCH, "5B3C1A2E053D763E1B002CC607C5****");
// If the time that is specified in the request is later than or equal to the actual modified time of the source object, OSS copies the object and returns 200 OK.
// meta.setHeader(OSSHeaders.COPY_OBJECT_SOURCE_IF_UNMODIFIED_SINCE, "2021-12-09T07:01:56.000Z");
// If the source object is modified after the time that is specified in the request, OSS copies the object.
// meta.setHeader(OSSHeaders.COPY_OBJECT_SOURCE_IF_MODIFIED_SINCE, "2021-12-09T07:01:56.000Z");
// Specify the method that is used to configure the metadata of the destination object. In this example, the method is set to COPY, which indicates that the metadata of the source object is copied to the destination object.
// meta.setHeader(OSSHeaders.COPY_OBJECT_METADATA_DIRECTIVE, "COPY");
// Specify the server-side encryption algorithm that is used to encrypt the destination object when OSS creates the object.
// meta.setHeader(OSSHeaders.OSS_SERVER_SIDE_ENCRYPTION, ObjectMetadata.KMS_SERVER_SIDE_ENCRYPTION);
// Specify the customer master key (CMK) that is managed by Key Management Service (KMS). This parameter takes effect only when you set x-oss-server-side-encryption to KMS.
// meta.setHeader(OSSHeaders.OSS_SERVER_SIDE_ENCRYPTION_KEY_ID, "9468da86-3509-4f8d-a61e-6eableac****");
// Specify the access control list (ACL) of the destination object. In this example, the ACL is set to private, which indicates that only the owner of the object and authorized users have read and write permissions on the object.
// meta.setHeader(OSSHeaders.OSS_OBJECT_ACL, CannedAccessControlList.Private);
// Specify the storage class of the destination object. In this example, the storage class is set to Standard.
// meta.setHeader(OSSHeaders.OSS_STORAGE_CLASS, StorageClass.Standard);
// Specify tags for the destination object. You can specify multiple tags for the destination object at the same time.

```

```
// meta.setHeader(OSSHeaders.OSS_TAGGING, "a:1");
// Specify the method that is used to configure tags for the destination object. In this
// example, the method is set to COPY, which indicates that the tags of the source object ar
// e copied to the destination object.
// meta.setHeader(OSSHeaders.COPY_OBJECT_TAGGING_DIRECTIVE, "COPY");
copyObjectRequest.setNewObjectMetadata(meta);
// Copy the object.
CopyObjectResult result = ossClient.copyObject(copyObjectRequest);
System.out.println("ETag: " + result.getETag() + " LastModified: " + result.getLastModifi
ed());
// Shut down the OSSClient instance.
ossClient.shutdown();
```

Copy a large object

To copy an object that is larger than 1 GB in size, you must split the object into parts and sequentially copy the parts by using UploadPartCopy. To implement multipart copy, perform the following steps:

1. Use `ossClient.initiateMultipartUpload` to initiate a multipart copy task.
2. Use `ossClient.uploadPartCopy` to copy the parts. All parts except for the last part must be larger than 100 KB.
3. Use `ossClient.completeMultipartUpload` to complete the multipart copy task.

The following code provides an example on how to use multipart copy to copy an object named `srcexampleobject.txt` from the `srcexamplebucket` bucket to an object named `desexampleobject.txt` object in the `desexamplebucket` bucket:

```
// Set yourEndpoint to the endpoint of the region in which the bucket is located. For exampl
// e, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://os
// s-cn-hangzhou.aliyuncs.com.
String endpoint = "yourEndpoint";
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to ac
// cess OSS because the account has permissions on all API operations. We recommend that you u
// se a RAM user to call API operations or perform routine O&M. To create a RAM user, log on t
// o the RAM console.
String accessKeyId = "yourAccessKeyId";
String accessKeySecret = "yourAccessKeySecret";
// Specify the name of the source bucket.
String sourceBucketName = "srcexamplebucket";
// Specify the full path of the source object. The full path of the object cannot contain t
// he bucket name.
String sourceKey = "srcexampleobject.txt";
// Specify the name of the destination bucket. The destination bucket must be in the same r
// egion as the source bucket.
String destinationBucketName = "desexamplebucket";
// Specify the full path of the destination object. The full path of the object cannot cont
// ain the bucket name.
String destinationKey = "desexampleobject.txt";
// Create an OSSClient instance.
OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);
ObjectMetadata objectMetadata = ossClient.getObjectMetadata(sourceBucketName, sourceKey);
// Query the size of the object that you want to copy.
long contentLength = objectMetadata.getContentLength();
// Set the size of each part to 10 MB. Unit: bytes.
```

```

long partSize = 1024 * 1024 * 10;
// Calculate the total number of parts.
int partCount = (int) (contentLength / partSize);
if (contentLength % partSize != 0) {
    partCount++;
}
System.out.println("total part count:" + partCount);
// Initiate a multipart copy task. You can use InitiateMultipartUploadRequest to specify the
// metadata of the destination object.
InitiateMultipartUploadRequest initiateMultipartUploadRequest = new InitiateMultipartUploadRequest(
    destinationBucketName, destinationKey);
InitiateMultipartUploadResult initiateMultipartUploadResult = ossClient.initiateMultipartUpload(
    initiateMultipartUploadRequest);
String uploadId = initiateMultipartUploadResult.getUploadId();
// Start the multipart copy task.
List<PartETag> partETags = new ArrayList<PartETag>();
for (int i = 0; i < partCount; i++) {
    // Calculate the size of each part.
    long skipBytes = partSize * i;
    long size = partSize < contentLength - skipBytes ? partSize : contentLength - skipBytes;

    // Create an UploadPartCopyRequest object. You can use UploadPartCopyRequest to specify
    // conditions.
    UploadPartCopyRequest uploadPartCopyRequest =
        new UploadPartCopyRequest(sourceBucketName, sourceKey, destinationBucketName, destinationKey);
    uploadPartCopyRequest.setUploadId(uploadId);
    uploadPartCopyRequest.setPartSize(size);
    uploadPartCopyRequest.setBeginIndex(skipBytes);
    uploadPartCopyRequest.setPartNumber(i + 1);
    //Map headers = new HashMap();
    // Specify the path of the source object.
    // headers.put(OSSHeaders.COPY_OBJECT_SOURCE, "/examplebucket/desexampleobject.txt");
    // Specify the range of data that you want to copy. For example, if you set bytes to 0~
    // 1023, the first 1,024 bytes of the source object are copied.
    // headers.put(OSSHeaders.COPY_SOURCE_RANGE, "bytes=0~1023");
    // If the ETag value of the source object is the same as the ETag value that is specified
    // in the request, OSS copies the object and returns 200 OK.
    // headers.put(OSSHeaders.COPY_OBJECT_SOURCE_IF_MATCH, "5B3C1A2E053D763E1B002CC607C5***");
    // If the ETag value of the source object is different from the ETag value that is specified
    // in the request, OSS copies the object and returns 200 OK.
    // headers.put(OSSHeaders.COPY_OBJECT_SOURCE_IF_NONE_MATCH, "5B3C1A2E053D763E1B002CC607C5*****");
    // If the time that is specified in the request is later than or equal to the actual modified
    // time of the source object, OSS copies the object and returns 200 OK.
    // headers.put(OSSHeaders.COPY_OBJECT_SOURCE_IF_UNMODIFIED_SINCE, "2021-12-09T07:01:56.000Z");
    // If the source object is modified after the time that is specified in the request, OSS
    // copies the object.
    // headers.put(OSSHeaders.COPY_OBJECT_SOURCE_IF_MODIFIED_SINCE, "2021-12-09T07:01:56.000Z");
    // uploadPartCopyRequest.setHeaders(headers);
    UploadPartCopyResult uploadPartCopyResult = ossClient.uploadPartCopy(uploadPartCopyRequest);
}

```

```
est);
    // Store the returned part ETags in partETags.
    partETags.add(uploadPartCopyResult.getPartETag());
}
// Complete the multipart copy task.
CompleteMultipartUploadRequest completeMultipartUploadRequest = new CompleteMultipartUploadRequest(
    destinationBucketName, destinationKey, uploadId, partETags);
ossClient.completeMultipartUpload(completeMultipartUploadRequest);
// Shut down the OSSClient instance.
ossClient.shutdown();
```

References

- Copy a small object

For more information about the API operations that you can call to copy a small object, see [CopyObject](#).

- Copy a large object

- For more information about the complete sample code that is used to copy a large object, visit [GitHub](#).
- For more information about the API operations that you can call to copy a large object, see [UploadPartCopy](#).

2.8.11. Prevent objects from being overwritten by objects that have the same names

By default, if you upload an object that has the same name as an existing object, the existing object is overwritten by the uploaded object. This topic describes how to configure the `x-oss-forbid-overwrite` request header to prevent objects from being overwritten by objects with the same names when you copy objects or perform simple upload or multipart upload.

Simple upload

The following code provides an example on how to prevent objects from being overwritten by objects that have the same names in simple upload:

```
import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.*;
import java.io.ByteArrayInputStream;

public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to access Object Storage Service (OSS) is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To
```

```

create a RAM user, log on to the RAM console.
    String accessKeyId = "yourAccessKeyId";
    String accessKeySecret = "yourAccessKeySecret";
    // Specify the bucket name. Example: examplebucket.
    String bucketName = "examplebucket";
    // Specify the full path of the object. The full path of the object cannot contain
the bucket name.
    String objectName = "exampledir/object";
    String content = "Hello OSS!";
    // Create an OSSClient instance.
    OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret
);
    try {
        // Create a PutObjectRequest object.
        PutObjectRequest putObjectRequest = new PutObjectRequest(bucketName, objectName
, new ByteArrayInputStream(content.getBytes()));
        // Specify whether to overwrite the existing object with the same name.
        // By default, if x-oss-forbid-overwrite is not specified, existing objects are
overwritten by the objects with the same names.
        // If x-oss-forbid-overwrite is set to false, existing objects are overwritten
by the objects with the same names.
        // If x-oss-forbid-overwrite is set to true, existing objects are not overwritt
en by the objects with the same names. If an object with the same name already exists, an e
rror is reported.
        ObjectMetadata metadata = new ObjectMetadata();
        metadata.setHeader("x-oss-forbid-overwrite", "true");
        putObjectRequest.setMetadata(metadata);
        // Upload the object.
        ossClient.putObject(putObjectRequest);
    } catch (OSSException oe) {
        System.out.println("Caught an OSSException, which means your request made it to
OSS, "
            + "but was rejected with an error response for some reason.");
        System.out.println("Error Message:" + oe.getErrorMessage());
        System.out.println("Error Code:" + oe.getErrorCode());
        System.out.println("Request ID:" + oe.getRequestId());
        System.out.println("Host ID:" + oe.getHostId());
    } catch (ClientException ce) {
        System.out.println("Caught an ClientException, which means the client encounter
ed "
            + "a serious internal problem while trying to communicate with OSS, "
            + "such as not being able to access the network.");
        System.out.println("Error Message:" + ce.getMessage());
    } finally {
        if (ossClient != null) {
            ossClient.shutdown();
        }
    }
}
}

```

Copy objects

- Use CopyObjectRequest to copy objects

The following code provides an example on how to prevent an existing object from being overwritten by a small object with the same name when you copy the small object by using CopyObjectRequest:

```
import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.*;
public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify
        // your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API oper-
        // ations. Using these credentials to access OSS is a high-risk operation. We recommend that
        // you use a RAM user to call API operations or perform routine O&M. To create a RAM user, l-
        // og on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the name of the source bucket.
        String sourceBucketName = "srcexamplebucket";
        // Specify the full path of the source object. The full path of the object cannot
        // contain the bucket name.
        String sourceObjectName = "srcexampleobject.txt";
        // Specify the name of the destination bucket. The destination bucket must be in
        // the same region as the source bucket.
        String destinationBucketName = "desexamplebucket";
        // Specify the full path of the destination object. The full path of the object c-
        // annot contain the bucket name.
        String destinationObjectName = "desexampleobject.txt";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecr-
        et);
        try {
            // Create a CopyObjectRequest object.
            CopyObjectRequest copyObjectRequest = new CopyObjectRequest(sourceBucketName,
            sourceObjectName, destinationBucketName, destinationObjectName);
            // Configure the metadata of the destination object.
            ObjectMetadata metadata = new ObjectMetadata();
            metadata.setContentType("text/html");
            // Specify whether to overwrite the existing object with the same name.
            // By default, if x-oss-forbid-overwrite is not specified, existing objects a-
            // re overwritten by the objects with the same names.
            // If x-oss-forbid-overwrite is set to false, existing objects are overwritte-
            // n by the objects with the same names.
            // If x-oss-forbid-overwrite is set to true, existing objects are not overwri-
            // tten by the objects with the same names. If an object with the same name already exists,
            // an error is reported.
            metadata.setHeader("x-oss-forbid-overwrite", "true");
            copyObjectRequest.setNewObjectMetadata(metadata);
            // Copy the object.
            CopyObjectResult result = ossClient.copyObject(copyObjectRequest);
            System.out.println("ETag: " + result.getETag() + " LastModified: " + result.g
```

```

etLastModified());
    } catch (OSSEException oe) {
        System.out.println("Caught an OSSEException, which means your request made it
to OSS, "
            + "but was rejected with an error response for some reason.");
        System.out.println("Error Message:" + oe.getErrorMessage());
        System.out.println("Error Code:" + oe.getErrorCode());
        System.out.println("Request ID:" + oe.getRequestId());
        System.out.println("Host ID:" + oe.getHostId());
    } catch (ClientException ce) {
        System.out.println("Caught an ClientException, which means the client encount
ered "
            + "a serious internal problem while trying to communicate with OSS, "
            + "such as not being able to access the network.");
        System.out.println("Error Message:" + ce.getMessage());
    } finally {
        if (ossClient != null) {
            ossClient.shutdown();
        }
    }
}
}

```

- Copy a large object

The following code provides an example on how to prevent an existing object from being overwritten by a large object with the same name when you copy the large object by using multipart copy:

```

import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSEException;
import com.aliyun.oss.model.*;
import java.util.ArrayList;
import java.util.List;
public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify
your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API oper
ations. Using these credentials to access OSS is a high-risk operation. We recommend that
you use a RAM user to call API operations or perform routine O&M. To create a RAM user, l
og on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the name of the source bucket.
        String sourceBucketName = "srcexamplebucket";
        // Specify the full path of the source object. The full path of the object cannot
contain the bucket name.
        String sourceObjectName = "srcexampleobject.txt";
        // Specify the name of the destination bucket. The destination bucket must be in
the same region as the source bucket.
        String destinationBucketName = "desexamplebucket";
    }
}

```

```

        // Specify the full path of the destination object. The full path of the object cannot contain the bucket name.
        String destinationObjectName = "desexampleobject.txt";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);

        try {
            ObjectMetadata objectMetadata = ossClient.getObjectMetadata(sourceBucketName, sourceObjectName);
            // Obtain the size of the object to copy.
            long contentLength = objectMetadata.getContentLength();
            // Set the size of each part to 10 MB.
            long partSize = 1024 * 1024 * 10;
            // Calculate the total number of parts.
            int partCount = (int) (contentLength / partSize);
            if (contentLength % partSize != 0) {
                partCount++;
            }
            System.out.println("total part count:" + partCount);
            // Initiate a multipart copy task. You can use InitiateMultipartUploadRequest to specify the metadata of the destination object.
            InitiateMultipartUploadRequest initiateMultipartUploadRequest = new InitiateMultipartUploadRequest(destinationBucketName, destinationObjectName);
            // Specify whether to overwrite the existing object with the same name.
            // By default, if x-oss-forbid-overwrite is not specified, existing objects are overwritten by the objects with the same names.
            // If x-oss-forbid-overwrite is set to false, existing objects are overwritten by the objects with the same names.
            // If x-oss-forbid-overwrite is set to true, existing objects are not overwritten by the objects with the same names. If an object with the same name already exists, an error is reported.
            ObjectMetadata metadata = new ObjectMetadata();
            metadata.setHeader("x-oss-forbid-overwrite", "true");
            initiateMultipartUploadRequest.setObjectMetadata(metadata);
            InitiateMultipartUploadResult initiateMultipartUploadResult = ossClient.initiateMultipartUpload(initiateMultipartUploadRequest);
            String uploadId = initiateMultipartUploadResult.getUploadId();
            // Start the multipart copy task.
            List<PartETag> partETags = new ArrayList<PartETag>();
            for (int i = 0; i < partCount; i++) {
                // Calculate the size of each part.
                long skipBytes = partSize * i;
                long size = partSize < contentLength - skipBytes ? partSize : contentLength - skipBytes;
                // Create an UploadPartCopyRequest object. You can use UploadPartCopyRequest to specify conditions.
                UploadPartCopyRequest uploadPartCopyRequest =
                    new UploadPartCopyRequest(sourceBucketName, sourceObjectName, destinationBucketName, destinationObjectName);
                uploadPartCopyRequest.setUploadId(uploadId);
                uploadPartCopyRequest.setPartSize(size);
                uploadPartCopyRequest.setBeginIndex(skipBytes);
                uploadPartCopyRequest.setPartNumber(i + 1);
                UploadPartCopyResult uploadPartCopyResult = ossClient.uploadPartCopy(uploadPartCopyRequest);
            }
        } catch (Exception e) {
            e.printStackTrace();
        }
    }
}

```

```

adPartCopyRequest);
    // Store the returned part ETags in partETags.
    partETags.add(uploadPartCopyResult.getPartETag());
}
// Complete the multipart copy task.
CompleteMultipartUploadRequest completeMultipartUploadRequest = new CompleteM
ultipartUploadRequest(
    destinationBucketName, destinationObjectName, uploadId, partETags);
// Specify whether to overwrite the existing object with the same name.
// By default, if x-oss-forbid-overwrite is not specified, existing objects a
re overwritten by the objects with the same names.
// If x-oss-forbid-overwrite is set to false, existing objects are overwritte
n by the objects with the same names.
// If x-oss-forbid-overwrite is set to true, existing objects are not overwri
tten by the objects with the same names. If an object with the same name already exists,
an error is reported.
completeMultipartUploadRequest.addHeader("x-oss-forbid-overwrite", "true");
ossClient.completeMultipartUpload(completeMultipartUploadRequest);
} catch (OSSException oe) {
    System.out.println("Caught an OSSException, which means your request made it
to OSS, "
        + "but was rejected with an error response for some reason.");
    System.out.println("Error Message:" + oe.getErrorMessage());
    System.out.println("Error Code:" + oe.getErrorCode());
    System.out.println("Request ID:" + oe.getRequestId());
    System.out.println("Host ID:" + oe.getHostId());
} catch (ClientException ce) {
    System.out.println("Caught an ClientException, which means the client encount
ered "
        + "a serious internal problem while trying to communicate with OSS, "
        + "such as not being able to access the network.");
    System.out.println("Error Message:" + ce.getMessage());
} finally {
    if (ossClient != null) {
        ossClient.shutdown();
    }
}
}
}

```

Multipart upload

The following code provides an example on how to prevent objects from being overwritten by newly uploaded objects that have the same names in multipart upload:

```

import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.*;
import java.io.File;
import java.io.FileInputStream;
import java.io.InputStream;
import java.util.ArrayList;
import java.util.List;

```

```

import java.util.List;

public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to access OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the bucket name. Example: examplebucket.
        String bucketName = "examplebucket";
        // Specify the full path of the object. The full path of the object cannot contain the bucket name.
        String objectName = "exemplerdir/object";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);

        try {
            // Create an InitiateMultipartUploadRequest object.
            InitiateMultipartUploadRequest request = new InitiateMultipartUploadRequest(bucketName, objectName);
            // Specify whether to overwrite the existing object with the same name.
            // By default, if x-oss-forbid-overwrite is not specified, existing objects are overwritten by the objects with the same names.
            // If x-oss-forbid-overwrite is set to false, existing objects are overwritten by the objects with the same names.
            // If x-oss-forbid-overwrite is set to true, existing objects are not overwritten by the objects with the same names. If an object with the same name already exists, an error is reported.
            ObjectMetadata metadata = new ObjectMetadata();
            metadata.setHeader("x-oss-forbid-overwrite", "true");
            request.setObjectMetadata(metadata);
            // Initiate a multipart upload task.
            InitiateMultipartUploadResult upresult = ossClient.initiateMultipartUpload(request);

            // Obtain the upload ID. The upload ID is the unique identification of a multipart upload task. You can initiate related requests based on the upload ID, such as canceling a multipart upload task and querying a multipart upload task.
            String uploadId = upresult.getUploadId();
            // partETags is the set of PartETags. A PartETag consists of the part number and ETag of an uploaded part.
            List<PartETag> partETags = new ArrayList<PartETag>();
            // Calculate the total number of parts to upload.
            final long partSize = 1 * 1024 * 1024L; // 1MB
            final File sampleFile = new File("<localFile>");
            long fileLength = sampleFile.length();
            int partCount = (int) (fileLength / partSize);
            if (fileLength % partSize != 0) {
                partCount++;
            }
            // Upload each part until all parts are uploaded.
            for (int i = 0; i < partCount; i++) {

```

```

        long startPos = i * partSize;
        long curPartSize = (i + 1 == partCount) ? (fileLength - startPos) : partSize;

        InputStream instream = new FileInputStream(sampleFile);
        // Skip the parts that have been uploaded.
        instream.skip(startPos);
        UploadPartRequest uploadPartRequest = new UploadPartRequest();
        uploadPartRequest.setBucketName(bucketName);
        uploadPartRequest.setKey(objectName);
        uploadPartRequest.setUploadId(uploadId);
        uploadPartRequest.setInputStream(instream);
        // Configure the size of each part. All parts except for the last part must
        be at least 5 MB in size.
        uploadPartRequest.setPartSize(curPartSize);
        // Configure part numbers. Each part is configured with a part number. The
        value can be from 1 to 10,000. If you configure a number beyond the range, OSS returns an InvalidArgument
        error code.
        uploadPartRequest.setPartNumber(i + 1);
        // Parts are not necessarily uploaded in sequence. They can be uploaded from
        different OSS clients. OSS sorts the parts based on their part numbers and combines them
        into a complete object.
        UploadPartResult uploadPartResult = ossClient.uploadPart(uploadPartRequest);

        // After you upload a part, OSS returns a result that contains a PartETag.
        The PartETag is stored in partETags.
        partETags.add(uploadPartResult.getPartETag());
    }
    // Create a CompleteMultipartUploadRequest object.
    // When the multipart upload task is completed, you must provide all valid Part
    ETags. After OSS receives the PartETags, OSS verifies all parts one by one. After all parts
    are verified, OSS combines these parts into a complete object.
    CompleteMultipartUploadRequest completeMultipartUploadRequest =
        new CompleteMultipartUploadRequest(bucketName, objectName, uploadId, partETags);

    // Specify whether to overwrite the existing object with the same name.
    // By default, if x-oss-forbid-overwrite is not specified, existing objects are
    overwritten by the objects with the same names.
    // If x-oss-forbid-overwrite is set to false, existing objects are overwritten
    by the objects with the same names.
    // If x-oss-forbid-overwrite is set to true, existing objects are not overwritten
    by the objects with the same names. If an object with the same name already exists, an error
    is reported.
    completeMultipartUploadRequest.addHeader("x-oss-forbid-overwrite", "true");
    // Complete the multipart upload task.
    CompleteMultipartUploadResult completeMultipartUploadResult = ossClient.completeMultipartUpload(completeMultipartUploadRequest);
} catch (OSSException oe) {
    System.out.println("Caught an OSSException, which means your request made it to
    OSS, "

        + "but was rejected with an error response for some reason.");
    System.out.println("Error Message:" + oe.getErrorMessage());
    System.out.println("Error Code:" + oe.getErrorCode());
    System.out.println("Request ID:" + oe.getRequestId());
    System.out.println("Host ID:" + oe.getHostId());
}

```

```
    } catch (ClientException ce) {
        System.out.println("Caught an ClientException, which means the client encounter  
ed "
            + "a serious internal problem while trying to communicate with OSS, "  
            + "such as not being able to access the network.");
        System.out.println("Error Message:" + ce.getMessage());
    } finally {
        if (ossClient != null) {
            ossClient.shutdown();
        }
    }
}
```

References

- For the complete sample code that is used to prevent existing objects from being overwritten by objects that have the same names in object upload scenarios, visit [GitHub](#).
- For more information about the API operation that you can call to perform simple upload, see [Put Object](#).
- For more information about the API operation that you can call to copy an object, see [Copy Object](#).
- For more information about the API operations that you can call to perform multipart upload, see [InitiateMultipartUpload](#) and [CompleteMultipartUpload](#).

2.8.12. Restore objects

You must restore an Archive or a Cold Archive object before you can read it. This topic describes how to restore an Archive or a Cold Archive object.

 **Notice** Do not call `RestoreObject` to restore an object whose storage class is not Archive or Cold Archive.

Restoration

If you want to read an Archive object or a Cold Archive object, you must restore the object in advance. It takes several minutes to restore an Archive object and takes several hours to restore a Cold Archive object.

The following section describes the status of an Archive object or a Cold Archive object throughout the restoration process:

1. By default, an Archive or a Cold Archive object is in the frozen state before restoration.
2. After you submit a restore request, the object is in the restoring state.
3. After the server completes the restore task, the object enters the restored state and you can read the object.
 - Archive objects

For Archive objects, the restored state lasts 24 hours by default. During the 24 hours, if you call `RestoreObject` again, the restored state is extended by 24 hours. You can extend the restored state to seven days by calling `RestoreObject` six times during the first 24 hours of restored state. You can also configure the duration of the restored state in days by calling `RestoreObject` once. You can specify a duration of at most seven days.

- Cold Archive objects

For Cold Archive objects, you can specify the duration of restored state and restoration priority. The duration of the restored state must be at least one day and at most 365 days. The time required to restore a Cold Archive object to the readable state is determined based on the restoration priority of the object:

- Expedited: The object is restored within one hour.
- Standard: The object is restored within two to five hours. If the `JobParameters` element is not passed in, the default restoration priority is Standard.
- Bulk: The object is restored within five to twelve hours.

4. After the restored state expires, the object returns to the frozen state.

Restore Archive objects

The following code provides an example on how to restore an Archive object:


```
// Specify the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set endpoint to https://oss-cn-hangzhou.aliyuncs.com.
String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
String accessKeyId = "yourAccessKeyId";
String accessKeySecret = "yourAccessKeySecret";
// Specify the name of the bucket.
String bucketName = "yourBucketName";
// Specify the full path of the Archive object. The path cannot contain the bucket name.
String objectName = "yourArchiveObjectName";
// Create an OSSClient instance.
OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);
ObjectMetadata objectMetadata = ossClient.getObjectMetadata(bucketName, objectName);
// Check whether the object is an Archive object.
StorageClass storageClass = objectMetadata.getObjectStorageClass();
if (storageClass == StorageClass.Archive) {
    // Restores the object.
    ossClient.restoreObject(bucketName, objectName);
    // Wait until the object is restored.
    do {
        Thread.sleep(1000);
        objectMetadata = ossClient.getObjectMetadata(bucketName, objectName);
    } while (!objectMetadata.isRestoreCompleted());
}
// Obtain the restored object.
OSSObject ossObject = ossClient.getObject(bucketName, objectName);
ossObject.getObjectContent().close();
// Shut down the OSSClient instance.
ossClient.shutdown();
```

Restore a Cold Archive object

The following code provides an example on how to restore a Cold Archive object:

```
// Specify the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set endpoint to https://oss-cn-hangzhou.aliyuncs.com.
String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
String accessKeyId = "yourAccessKeyId";
String accessKeySecret = "yourAccessKeySecret";
// Specify the name of the bucket.
String bucketName = "yourBucketName";
// Specify the full path of the object. The path cannot contain the bucket name.
String objectName = "yourColdArchiveObjectName";
// Create an OSSClient instance.
OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);
// Refer to the following code if you want to set the storage class of the object to Cold Archive when you upload the object.
// Create a PutObjectRequest object.
// PutObjectRequest putObjectRequest = new PutObjectRequest(bucketName, objectName, new ByteArrayInputStream("<yourContent>".getBytes()));
// Set the storage class of the object to Cold Archive in the object metadata.
// ObjectMetadata metadata = new ObjectMetadata();
// metadata.setHeader(OSSHeaders.OSS_STORAGE_CLASS, StorageClass.ColdArchive.toString());
// putObjectRequest.setMetadata(metadata);
// Set the storage class of the object when you upload the object.
// ossClient.putObject(putObjectRequest);
// Set the restoration priority of the Cold Archive object.
// RestoreTier.RESTORE_TIER_EXPEDITED: The object is restored within an hour.
// RestoreTier.RESTORE_TIER_STANDARD: The object is restored within two to five hours.
// RestoreTier.RESTORE_TIER_STANDARD: The object is restored within five to twelve hours.
RestoreJobParameters jobParameters = new RestoreJobParameters(restoreTier);
// Configure parameters. For example, set the restoration priority of the object to Standard, and set the duration for which the object can remain in the restored state to two days.
// The first parameter indicates the validity period of the restored state. The default value is one day. This parameter applies to Archive and Cold Archive objects.
// jobParameters indicates the restoration priority of the object and applies only to Cold Archive objects.
RestoreConfiguration configuration = new RestoreConfiguration(2, jobParameters);
// Initiate a restore request.
ossClient.restoreObject(bucketName, objectName, configuration);
// Shut down the OSSClient instance.
ossClient.shutdown();
```

References

- For more information about the complete sample code that is used to restore Archive and Cold Archive objects, visit [Git Hub](#).
- For more information about the API operation that you can call to restore Archive and Cold Archive objects, see [RestoreObject](#).

2.8.13. Manage symbolic links

Symbolic links can be used to access objects that are frequently used in buckets. After you create a symbolic link for an object, you can use the symbolic link to access the object.

Create a symbolic link

The following code provides an example on how to create a symbolic link:

```
import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.*;

public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access Object Storage Service (OSS) because the account has permissions on all API operations. We recommend that you use a Resource Access Management (RAM) user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the name of the bucket. Example: examplebucket.
        String bucketName = "examplebucket";
        // Specify the name of the symbolic link that you want to create.
        String symLink = "yourSymLink";
        // Specify the name of the object to which the symbolic link points.
        String destinationObjectName = "yourDestinationObjectName";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);

        try {
            // Create metadata for the object to upload.
            ObjectMetadata metadata = new ObjectMetadata();
            metadata.setContentType("text/plain");
            // Set the value of the property parameter for the user metadata to property-value.
            metadata.addUserMetadata("property", "property-value");
            // Specify whether to overwrite the object with the same name when you create the symbolic link.
            // metadata.setHeader("x-oss-forbid-overwrite", "true");
            // Specify the access control list (ACL) of the object.
            // metadata.setHeader(OSSHeaders.OSS_OBJECT_ACL, CannedAccessControlList.Default);

            // Specify the storage class of the object.
            // metadata.setHeader(OSSHeaders.OSS_STORAGE_CLASS, StorageClass.Standard);
            // Create a request to create the symbolic link.
            CreateSymlinkRequest createSymlinkRequest = new CreateSymlinkRequest(bucketName, symLink, destinationObjectName);
            // Configure the metadata.
            createSymlinkRequest.setMetadata(metadata);
            // Create the symbolic link.
            ossClient.createSymlink(createSymlinkRequest);
        } catch (OSSException oe) {
```

```
        } catch (OSSException oe) {  
            System.out.println("Caught an OSSException, which means your request made it to  
OSS, "  
                + "but was rejected with an error response for some reason.");  
            System.out.println("Error Message:" + oe.getErrorMessage());  
            System.out.println("Error Code:" + oe.getErrorCode());  
            System.out.println("Request ID:" + oe.getRequestId());  
            System.out.println("Host ID:" + oe.getHostId());  
        } catch (ClientException ce) {  
            System.out.println("Caught an ClientException, which means the client encounter  
ed "  
                + "a serious internal problem while trying to communicate with OSS, "  
                + "such as not being able to access the network.");  
            System.out.println("Error Message:" + ce.getMessage());  
        } finally {  
            if (ossClient != null) {  
                ossClient.shutdown();  
            }  
        }  
    }  
}
```

Query a symbolic link

To query a symbolic link, you must have read permissions on the symbolic link. The following code provides an example on how to query a symbolic link and the name of the object to which the symbolic link points:

```

import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.*;
public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the name of the bucket. Example: examplebucket.
        String bucketName = "examplebucket";
        // Specify the name of the symbolic link to query.
        String symLink = "yourSymLink";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);

        try {
            // Query the symbolic link and the name of the object to which the symbolic link points.
            OSSSymlink symbolicLink = ossClient.getSymlink(bucketName, symLink);
            System.out.println(symbolicLink.getSymlink());
            System.out.println(symbolicLink.getTarget());
            System.out.println(symbolicLink.getRequestId());
        } catch (OSSException oe) {
            System.out.println("Caught an OSSException, which means your request made it to OSS, "
                + "but was rejected with an error response for some reason.");
            System.out.println("Error Message:" + oe.getErrorMessage());
            System.out.println("Error Code:" + oe.getErrorCode());
            System.out.println("Request ID:" + oe.getRequestId());
            System.out.println("Host ID:" + oe.getHostId());
        } catch (ClientException ce) {
            System.out.println("Caught an ClientException, which means the client encountered "
                + "a serious internal problem while trying to communicate with OSS, "
                + "such as not being able to access the network.");
            System.out.println("Error Message:" + ce.getMessage());
        } finally {
            if (ossClient != null) {
                ossClient.shutdown();
            }
        }
    }
}

```

References

- For more information about the API operation that you can call to create a symbolic link, see [PutSymlink](#).
- For more information about the API operation that you can call to query a symbolic link, see [GetSymlink](#).

2.9. Manage directories

After the hierarchical namespace feature is enabled for a bucket, you can create, rename, and delete a directory for the bucket. This topic describes how to create, rename, and delete a directory for a bucket.

 **Note** For more information about how to enable the hierarchical namespace feature for a bucket, see [Create a bucket](#).

Create a directory

The following code provides an example on how to create a directory named `exemplerdir` for `examplebucket`:

```
// Set yourEndpoint to the endpoint of the region where the bucket is located. For example,
// if your bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-c
n-hangzhou.aliyuncs.com.
String endpoint = "yourEndpoint";
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to lo
g on to OSS because the account has permissions on all API operations. We recommend that yo
u use your RAM user's credentials to call API operations or perform routine operations and
maintenance. To create a RAM user, log on to the RAM console.
String accessKeyId = "yourAccessKeyId";
String accessKeySecret = "yourAccessKeySecret";
// Specify the bucket name.
String bucketName = "examplebucket";
// Specify the absolute path of the directory. The absolute path of the directory cannot co
ntain bucket names.
String directoryName = "exemplerdir";
// Create an OSSClient instance.
OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);
// Create a directory.
CreateDirectoryRequest createDirectoryRequest = new CreateDirectoryRequest(bucketName, dire
ctoryName);
ossClient.createDirectory(createDirectoryRequest);
// Shut down the OSSClient instance.
ossClient.shutdown();
```

For more information about how to create a directory, see [CreateDirectory](#).

Rename a directory

After the hierarchical feature is enabled for a bucket, you can rename directories in the bucket.

The following code provides an example on how to rename the `exemplerdir` directory `newexemplerdir` for `examplebucket`:

```
// Set yourEndpoint to the endpoint of the region where the bucket is located. For example,
// if your bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-c
n-hangzhou.aliyuncs.com.
String endpoint = "yourEndpoint";
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to lo
g on to OSS because the account has permissions on all API operations. We recommend that yo
u use your RAM user's credentials to call API operations or perform routine operations and
maintenance. To create a RAM user, log on to the RAM console.
String accessKeyId = "yourAccessKeyId";
String accessKeySecret = "yourAccessKeySecret";
// Specify the bucket name.
String bucketName = "examplebucket";
// Specify the absolute path of the source directory. The absolute path of the directory ca
nnot contain bucket names.
String sourceDir = "exampledir";
// Specify the absolute path of the destination directory that is in the same bucket as the
source directory. The absolute path of the directory cannot contain bucket names.
String destinationDir = "newexampledir";
// Create an OSSClient instance.
OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);
// Change the absolute path of the source directory to that of the destination directory.
RenameObjectRequest renameObjectRequest = new RenameObjectRequest(bucketName, sourceDir, de
stnationDir);
ossClient.renameObject(renameObjectRequest);
// Shut down the OSSClient instance.
ossClient.shutdown();
```

For more information about how to rename a directory, see [Rename](#).

Delete a directory

After the hierarchical namespace feature is enabled for a bucket, you can use the non-recursive delete or recursive delete method to delete directories from the bucket.

- Non-recursive delete

If you use the non-recursive delete method to delete a directory, you can delete the directory only when the directory is empty.

The following code provides an example on how to use the non-recursive delete method to delete the exampledir directory from examplebucket:

```
// Set yourEndpoint to the endpoint of the region where the bucket is located. For example, if your bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com.
String endpoint = "yourEndpoint";
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log on to OSS because the account has permissions on all API operations. We recommend that you use your RAM user's credentials to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
String accessKeyId = "yourAccessKeyId";
String accessKeySecret = "yourAccessKeySecret";
// Specify the bucket name.
String bucketName = "examplebucket";
// Specify the absolute path of the directory. The absolute path of the directory cannot contain bucket names.
String directoryName = "exemplerdir";
// Create an OSSClient instance.
OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);
// Delete the directory. By default, the non-recursive delete method is used. Make sure that all objects and subdirectories are deleted from this directory.
DeleteDirectoryRequest deleteDirectoryRequest = new DeleteDirectoryRequest(bucketName, directoryName);
DeleteDirectoryResult deleteDirectoryResult = ossClient.deleteDirectory(deleteDirectoryRequest);
// Display the absolute path of the deleted directory.
System.out.println("delete dir name :" + deleteDirectoryResult.getDirectoryName());
// Display the total number of deleted objects and directories.
System.out.println("delete number:" + deleteDirectoryResult.getDeleteNumber());
// Shut down the OSSClient instance.
ossClient.shutdown();
```

- Recursive delete

If you use the recursive delete method to delete a directory, you can delete the directory along with the objects and subdirectories from this directory. Exercise caution when you use recursive delete.

The following code provides an example on how to use the recursive delete method to delete the exemplerdir directory as well as the objects and subdirectories from examplebucket:


```
// Set yourEndpoint to the endpoint of the region where the bucket is located. For example, if your bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com.
String endpoint = "yourEndpoint";
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log on to OSS because the account has permissions on all API operations. We recommend that you use your RAM user's credentials to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
String accessKeyId = "yourAccessKeyId";
String accessKeySecret = "yourAccessKeySecret";
// Specify the bucket name.
String bucketName = "examplebucket";
// Specify the absolute path of the directory. The absolute path of the directory cannot contain bucket names.
String directoryName = "examplerdir";
// Create an OSSClient instance.
OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);
// Use the recursive delete method to delete the directory.
DeleteDirectoryRequest deleteDirectoryRequest = new DeleteDirectoryRequest(bucketName, directoryName);
deleteDirectoryRequest.setDeleteRecursive(true);
DeleteDirectoryResult deleteDirectoryResult = ossClient.deleteDirectory(deleteDirectoryRequest);
// Display the deletion result.
// The maximum total number of directories and objects that can be deleted at a time is 1000. If only part of directories and objects are returned, the server returns nextDeleteToken. You can use nextDeleteToken to continue deleting remaining data.
// nextDeleteToken helps the server find the object or directory after which the delete operation begins.
String nextDeleteToken = deleteDirectoryResult.getNextDeleteToken();
System.out.println("delete next token:" + nextDeleteToken);
// Display the absolute path of the deleted directory.
System.out.println("delete dir name : " + deleteDirectoryResult.getDirectoryName());
// Display the total number of deleted objects and directories.
System.out.println("delete number:" + deleteDirectoryResult.getDeleteNumber());
// Shut down the OSSClient instance.
ossClient.shutdown();
```

For more information about how to delete a directory, see [DeleteDirectory](#).

2.10. Versioning

2.10.1. Versioning

The versioning state of a bucket applies to all objects in the bucket. If you enable versioning for a bucket, you can recover any previous version of an object in the bucket when you accidentally overwrite or delete the object.

Configure versioning for a bucket

The following code provides an example on how to enable or suspend versioning for a bucket:

```

import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.*;
public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in Object Storage Service (OSS) is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the bucket name. Example: examplebucket.
        String bucketName = "examplebucket";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);

        try {
            // Enable versioning for the bucket by setting the BucketVersioningConfiguration parameter to ENABLED. To suspend versioning for the bucket, set the BucketVersioningConfiguration parameter to SUSPENDED.
            BucketVersioningConfiguration configuration = new BucketVersioningConfiguration();
            configuration.setStatus(BucketVersioningConfiguration.ENABLED);
            SetBucketVersioningRequest request = new SetBucketVersioningRequest(bucketName, configuration);
            ossClient.setBucketVersioning(request);
        } catch (OSSException oe) {
            System.out.println("Caught an OSSException, which means your request made it to OSS, "
                + "but was rejected with an error response for some reason.");
            System.out.println("Error Message:" + oe.getErrorMessage());
            System.out.println("Error Code:" + oe.getErrorCode());
            System.out.println("Request ID:" + oe.getRequestId());
            System.out.println("Host ID:" + oe.getHostId());
        } catch (ClientException ce) {
            System.out.println("Caught an ClientException, which means the client encountered "
                + "a serious internal problem while trying to communicate with OSS, "
                + "such as not being able to access the network.");
            System.out.println("Error Message:" + ce.getMessage());
        } finally {
            if (ossClient != null) {
                ossClient.shutdown();
            }
        }
    }
}

```

Query the versioning state of a bucket

The following code provides an example on how to query the versioning state of a bucket:

```
import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.*;
public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the bucket name. Example: examplebucket.
        String bucketName = "examplebucket";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);

        try {
            // Query the versioning state of the bucket.
            BucketVersioningConfiguration versionConfiguration = ossClient.getBucketVersioning(bucketName);
            System.out.println("bucket versioning status: " + versionConfiguration.getStatus());
        } catch (OSSException oe) {
            System.out.println("Caught an OSSException, which means your request made it to OSS, "
                + "but was rejected with an error response for some reason.");
            System.out.println("Error Message:" + oe.getErrorMessage());
            System.out.println("Error Code:" + oe.getErrorCode());
            System.out.println("Request ID:" + oe.getRequestId());
            System.out.println("Host ID:" + oe.getHostId());
        } catch (ClientException ce) {
            System.out.println("Caught an ClientException, which means the client encountered "
                + "a serious internal problem while trying to communicate with OSS, "
                + "such as not being able to access the network.");
            System.out.println("Error Message:" + ce.getMessage());
        } finally {
            if (ossClient != null) {
                ossClient.shutdown();
            }
        }
    }
}
```

References

- For the complete sample code that is used to configure versioning for a bucket, visit [GitHub](#).
- For more information about the API operation that you can call to configure versioning for a bucket, see [PutBucketVersioning](#).
- For more information about the API operation that you can call to query the versioning state of a bucket, see [GetBucketVersioning](#).

2.10.2. Upload objects

This topic describes how to upload objects to a versioned bucket.

Simple upload

If you initiate a PutObject request to a versioning-enabled bucket, Object Storage Service (OSS) generates a unique version ID for the object to upload and includes the version ID in the x-oss-version-id response header. In a versioning-suspended bucket, the version ID generated for the object is null. If you upload an object with the same name as an existing object to a versioning-suspended bucket, the existing object is overwritten. This way, each object has only a single version whose version ID is null.

The following code provides an example on how to upload an object to a versioned bucket in simple upload:

```

import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.*;
import java.io.ByteArrayInputStream;

public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a Resource Access Management (RAM) user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the name of the bucket. Example: examplebucket.
        String bucketName = "examplebucket";
        // Specify the full path of the object. The full path of the object cannot contain the bucket name.
        String objectName = "exemplerdir/object";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);

        try {
            // Upload a string to the object in this example.
            String content = "Hello OSS";
            PutObjectResult result = ossClient.putObject(bucketName, objectName, new ByteArrayInputStream(content.getBytes()));
            // Query the version ID of the object.
            System.out.println("result.versionid: " + result.getVersionId());
        } catch (OSSException oe) {
            System.out.println("Caught an OSSException, which means your request made it to OSS, "
                + "but was rejected with an error response for some reason.");
            System.out.println("Error Message:" + oe.getErrorMessage());
            System.out.println("Error Code:" + oe.getErrorCode());
            System.out.println("Request ID:" + oe.getRequestId());
            System.out.println("Host ID:" + oe.getHostId());
        } catch (ClientException ce) {
            System.out.println("Caught an ClientException, which means the client encountered "
                + "a serious internal problem while trying to communicate with OSS, "
                + "such as not being able to access the network.");
            System.out.println("Error Message:" + ce.getMessage());
        } finally {
            if (ossClient != null) {
                ossClient.shutdown();
            }
        }
    }
}

```

Append upload

In a versioning-enabled bucket, the AppendObject operation can be performed only on the current version of an append object.

Note

- When you perform the AppendObject operation on the current version of an append object, OSS does not generate a previous version for the object.
- When you perform the PutObject or DeleteObject operation on the current version of an append object, OSS saves the append object as a previous version that cannot be appended any more.
- The append operation on the current version of a non-append object such as a normal object or a delete marker is not supported.

The following code provides an example on how to upload an object in append upload:

```
import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.*;
import java.io.ByteArrayInputStream;

public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the name of the bucket. Example: examplebucket.
        String bucketName = "examplebucket";
        // Specify the full path of the object. The full path of the object cannot contain the bucket name.
        String objectName = "exampledir/object";
        String content1 = "Hello OSS A \n";
        String content2 = "Hello OSS B \n";
        String content3 = "Hello OSS C \n";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);

        try {
            ObjectMetadata meta = new ObjectMetadata();
            // Specify the type of content to upload.
            meta.setContentType("text/plain");
            // Configure multiple parameters by using AppendObjectRequest.
            AppendObjectRequest appendObjectRequest = new AppendObjectRequest(bucketName, objectName, new ByteArrayInputStream(content1.getBytes()), meta);
            // Configure a single parameter by using AppendObjectRequest.
```

```

        // Specify the bucket name.
        //appendObjectRequest.setBucketName("<yourBucketName>");
        // Specify the object name.
        //appendObjectRequest.setKey("<yourObjectName>");
        // Configure the type of the content to append. Two content types are available
: InputStream and File. The following code provides an example on how to specify the conten
t of the InputStream type.
        //appendObjectRequest.setInputStream(new ByteArrayInputStream(content1.getBytes
()));
        // Configure the type of content to append. Two content types are available: In
putStream and File. The following code provides an example on how to specify the content of
the File type.
        //appendObjectRequest.setFile(new File("<yourLocalFile>"));
        // Specify the metadata of the object. You can specify the metadata only when y
ou perform the first append operation on the object.
        //appendObjectRequest.setMetadata(meta);
        // Perform the first append operation.
        // Specify the position from which the append operation starts.
        appendObjectRequest.setPosition(0L);
        AppendObjectResult appendObjectResult = ossClient.appendObject(appendObjectRequ
est);
        // Calculate the 64-bit CRC value of the object. The value is calculated based
on the ECMA-182 standard.
        System.out.println(appendObjectResult.getObjectCRC());
        // Perform the second append operation.
        // nextPosition indicates the position from which the next append operation sta
rts. This is the current length of the object.
        appendObjectRequest.setPosition(appendObjectResult.getNextPosition());
        appendObjectRequest.setInputStream(new ByteArrayInputStream(content2.getBytes()
));
        appendObjectResult = ossClient.appendObject(appendObjectRequest);
        // Perform the third append operation.
        appendObjectRequest.setPosition(appendObjectResult.getNextPosition());
        appendObjectRequest.setInputStream(new ByteArrayInputStream(content3.getBytes()
));
        appendObjectResult = ossClient.appendObject(appendObjectRequest);
    } catch (OSSException oe) {
        System.out.println("Caught an OSSException, which means your request made it to
OSS, "
            + "but was rejected with an error response for some reason.");
        System.out.println("Error Message:" + oe.getErrorMessage());
        System.out.println("Error Code:" + oe.getErrorCode());
        System.out.println("Request ID:" + oe.getRequestId());
        System.out.println("Host ID:" + oe.getHostId());
    } catch (ClientException ce) {
        System.out.println("Caught an ClientException, which means the client encounter
ed "
            + "a serious internal problem while trying to communicate with OSS, "
            + "such as not being able to access the network.");
        System.out.println("Error Message:" + ce.getMessage());
    } finally {
        if (ossClient != null) {
            ossClient.shutdown();
        }
    }
}

```

```

    }
}

```

Multipart upload

If you call `CompleteMultipartUpload` to complete the MultipartUpload task for an object in a versioned bucket, OSS generates a unique version ID for the object and returns it as the `x-oss-version-id` response header.

The following code provides an example on how to upload an object to a versioned bucket in multipart upload:

```

import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.*;
import java.io.File;
import java.io.FileInputStream;
import java.io.InputStream;
import java.util.ArrayList;
import java.util.List;
public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the name of the bucket. Example: examplebucket.
        String bucketName = "examplebucket";
        // Specify the full path of the object. The full path of the object cannot contain the bucket name.
        String objectName = "exampledir/object";
        String localFile = "D://example.txt";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);
        try {
            /* Step 1: Initiate a multipart upload event.
             */
            InitiateMultipartUploadRequest request = new InitiateMultipartUploadRequest(bucketName, objectName);
            InitiateMultipartUploadResult upresult = ossClient.initiateMultipartUpload(request);
            // Obtain an upload ID. The upload ID uniquely identifies the multipart upload task. You can use the upload ID to initiate related requests such as canceling and querying the multipart upload task.
            String uploadId = upresult.getUploadId();
            /* Step 2: Upload parts.

```



```

/* Step 2: Upload parts.
 */
// partETags is the set of PartETags. A PartETag consists of the part number and ETag of an uploaded part
List<PartETag> partETags = new ArrayList<PartETag>();
// Calculate the total number of parts to upload.
final long partSize = 1 * 1024 * 1024L; // 1MB
final File sampleFile = new File(localFile);
long fileLength = sampleFile.length();
int partCount = (int) (fileLength / partSize);
if (fileLength % partSize != 0) {
    partCount++;
}
// Upload each part until all parts are uploaded.
for (int i = 0; i < partCount; i++) {
    long startPos = i * partSize;
    long curPartSize = (i + 1 == partCount) ? (fileLength - startPos) : partSize;

    InputStream instream = new FileInputStream(sampleFile);
    // Skip the parts that have been uploaded.
    instream.skip(startPos);
    UploadPartRequest uploadPartRequest = new UploadPartRequest();
    uploadPartRequest.setBucketName(bucketName);
    uploadPartRequest.setKey(objectName);
    uploadPartRequest.setUploadId(uploadId);
    uploadPartRequest.setInputStream(instream);
    // Configure the size of each part. Each part except for the last part must be larger than 100 KB in size.
    uploadPartRequest.setPartSize(curPartSize);
    // Configure part numbers. Each part is configured with a part number. The value ranges from 1 to 10000. If you configure a number beyond the range, OSS returns an InvalidArgument error code.
    uploadPartRequest.setPartNumber(i + 1);
    // Parts are not necessarily uploaded in sequence. They can be uploaded from different OSS clients. OSS sorts the parts based on their part numbers and combines them into a complete object.
    UploadPartResult uploadPartResult = ossClient.uploadPart(uploadPartRequest);

    // After you upload a part, OSS returns a result that contains a PartETag. The PartETag is stored in partETags.
    partETags.add(uploadPartResult.getPartETag());
}
/* Step 3: Complete multipart upload.
 */
// You must provide all valid PartETags when you perform this operation. OSS verifies the validity of all parts one by one after it receives the PartETags. After all parts are verified, OSS combines these parts into a complete object.
CompleteMultipartUploadRequest completeMultipartUploadRequest =
    new CompleteMultipartUploadRequest(bucketName, objectName, uploadId, partETags);
CompleteMultipartUploadResult completeMultipartUploadResult = ossClient.completeMultipartUpload(completeMultipartUploadRequest);
// Query the version ID of the uploaded object.
System.out.println("restore object versionid: " + completeMultipartUploadResult.getVersionId());

```

```

        } catch (OSSException oe) {
            System.out.println("Caught an OSSException, which means your request made it to
OSS, "

                + "but was rejected with an error response for some reason.");
            System.out.println("Error Message:" + oe.getErrorMessage());
            System.out.println("Error Code:" + oe.getErrorCode());
            System.out.println("Request ID:" + oe.getRequestId());
            System.out.println("Host ID:" + oe.getHostId());
        } catch (ClientException ce) {
            System.out.println("Caught an ClientException, which means the client encounter
ed "

                + "a serious internal problem while trying to communicate with OSS, "
                + "such as not being able to access the network.");
            System.out.println("Error Message:" + ce.getMessage());
        } finally {
            if (ossClient != null) {
                ossClient.shutdown();
            }
        }
    }
}

```

References

- For more information about the API operation that you can call to perform simple upload, see [Put Object](#).
- For more information about the API operation that you can call to perform append upload, see [Append Object](#).
- For more information about the API operation that you can call to perform multipart upload, see [Complete Multipart Upload](#).

2.10.3. Download objects

By default, when you perform the `GetObject` operation on an object in a versioned bucket, only the current version of the object is returned.

Usage notes

When you perform the `GetObject` operation on an object in a bucket:

- If the current version of the object is a delete marker, the 404 Not Found error is returned.
- If you specify the `versionId` of the target object in the request, the specified version of the object is returned. For example, if the `versionId` is specified as null in the request, the version of which the `versionId` is null is returned.
- If the version specified in the request is a delete marker, the 405 Method Not Allowed error is returned.

Examples

Use the following code to download an object:

```

import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;

```

```

import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.*;
import java.io.BufferedReader;
import java.io.InputStreamReader;
public class Demo {
    public static void main(String[] args) {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify the
        // actual endpoint. For more information about the endpoints of other regions, see Regions and
        // endpoints.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account
        // to access Object Storage Service (OSS) because the account has permissions on all API op-
        // erations. We recommend that you use a RAM user to call API operations or perform routine O&
        // M. To create a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the name of the bucket. Example: examplebucket.
        String bucketName = "examplebucket";
        // Specify the full path of the object. Example: exampledir/exampleobject.txt. The
        // full path of the object cannot contain the bucket name.
        String objectName = "exampledir/exampleobject.txt";
        // Specify the version ID of the object. Example: CAEQARiBgID8rumR2hYiIGUyOTAYZGY2MzU5MjQ5Zj
        // lhYzQzZjNlYTAyZDE3****. For more information about how to obtain the version ID of
        // an object, see GetBucketVersions(ListObjectVersions).
        String versionid = "CAEQARiBgID8rumR2hYiIGUyOTAYZGY2MzU5MjQ5ZjlhYzQzZjNlYTAyZDE3****";

        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);

        try {
            // Encapsulate the GetObject request.
            GetObjectRequest getObjectRequest = new GetObjectRequest(bucketName, objectName);

            getObjectRequest.setVersionId(versionid);
            // ossObject specifies the bucket name, object name, object metadata, and an input
            // stream.
            OSSObject ossObject = ossClient.getObject(getObjectRequest);
            // View the version ID of the downloaded object.
            System.out.println("Get Object versionid:" + ossObject.getObjectMetadata().getVersionId());
            // View the content of the object that has the specified version ID.
            System.out.println("Object content:");
            BufferedReader reader = new BufferedReader(new InputStreamReader(ossObject.getObjectContent()));
            while (true) {
                String line = reader.readLine();
                if (line == null) break;
                System.out.println("\n" + line);
            }
        } catch (OSSException oe) {
            System.out.println("Caught an OSSException, which means your request made it to
            OSS, "
                + "but was rejected with an error response for some reason.");
            System.out.println("Error Message:" + oe.getMessage());
        }
    }
}

```

```

        System.out.println("Error Code:" + oe.getErrorCode());
        System.out.println("Request ID:" + oe.getRequestId());
        System.out.println("Host ID:" + oe.getHostId());
    } catch (Throwable ce) {
        System.out.println("Caught an ClientException, which means the client encounter
ed "
            + "a serious internal problem while trying to communicate with OSS, "
            + "such as not being able to access the network.");
        System.out.println("Error Message:" + ce.getMessage());
    } finally {
        // Shut down the OSSClient instance.
        if (ossClient != null) {
            ossClient.shutdown();
        }
    }
}
}

```

For more information about the API operation that you can call to download an object, see [GetObject](#).

2.10.4. Copy objects

This topic describes how to copy objects in a versioned bucket. You can use `CopyObject` to copy an object that is up to 1 GB in size and use `UploadPartCopy` to copy an object larger than 1 GB in size.

Copy a small object

You can use `CopyObject` to copy an object that is up to 1 GB in size from a source bucket to a destination bucket within the same region.

- By default, the `x-oss-copy-source` header in a `CopyObject` request specifies the current version of the object to copy. If the current version of the object is a delete marker, Object Storage Service (OSS) returns 404 Not Found, which indicates that the object does not exist. You can add a version ID to `x-oss-copy-source` to copy the specified object version. Delete markers cannot be copied.
- You can copy a previous version of an object to the same bucket. The copied previous version becomes the new current version of the object. This way, the previous version of the object is recovered.
- If versioning is enabled for the destination bucket, OSS generates a unique version ID for the destination object. The version ID is returned in the response as the `x-oss-version-id` header value. If versioning is disabled or suspended for the destination bucket, OSS generates a version whose version ID is null for the destination object and overwrites the original version whose version ID is null.
- Appendable objects in a bucket for which versioning is enabled or suspended cannot be copied.

The following code provides an example on how to copy a small object:

```

import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.*;
public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify yo

```

```

ur actual endpoint.
    String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
    // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
    String accessKeyId = "yourAccessKeyId";
    String accessKeySecret = "yourAccessKeySecret";
    // Specify the name of the source bucket.
    String sourceBucketName = "srcexamplebucket";
    // Specify the full path of the source object. The full path of the object cannot contain the bucket name.
    String sourceObjectName = "srcexampleobject.txt";
    // Specify the name of the destination bucket. The destination bucket must be in the same region as the source bucket.
    String destinationBucketName = "desexamplebucket";
    // Specify the full path of the destination object. The full path of the object cannot contain the bucket name.
    String destinationObjectName = "desexampleobject.txt";
    String versionId = "CAEQMxiBgICAof2D0BYiIDJhMGE3N2M1YTI1NDQzOGY5NTkyNTI3MGYyMzJm**";

    // Create an OSSClient instance.
    OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);

    try {
        CopyObjectRequest copyObjectRequest = new CopyObjectRequest(sourceBucketName, sourceObjectName, destinationBucketName, destinationObjectName);
        copyObjectRequest.setSourceVersionId(versionId);
        CopyObjectResult copyObjectResult = ossClient.copyObject(copyObjectRequest);
        System.out.println("ETag: " + copyObjectResult.getETag() + " LastModified: " + copyObjectResult.getLastModified());
        System.out.println("dest object versionid: " + copyObjectResult.getVersionId());
    } catch (OSSException oe) {
        System.out.println("Caught an OSSException, which means your request made it to OSS, "
            + "but was rejected with an error response for some reason.");
        System.out.println("Error Message:" + oe.getErrorMessage());
        System.out.println("Error Code:" + oe.getErrorCode());
        System.out.println("Request ID:" + oe.getRequestId());
        System.out.println("Host ID:" + oe.getHostId());
    } catch (ClientException ce) {
        System.out.println("Caught an ClientException, which means the client encountered "
            + "a serious internal problem while trying to communicate with OSS, "
            + "such as not being able to access the network.");
        System.out.println("Error Message:" + ce.getMessage());
    } finally {
        if (ossClient != null) {
            ossClient.shutdown();
        }
    }
}

```

Copy a large object

To copy an object that is larger than 1 GB in size, you must split the object into parts and sequentially copy the parts by using UploadPartCopy.

By default, the UploadPartCopy operation uploads a part by copying data from the current version of an existing object. You can add versionId to the x-oss-copy-source request header as the condition to upload a part by copying data from the specified version of an existing object. Example: x-oss-copy-source : /SourceBucketName/SourceObjectName?versionId=111111.

 **Note** The value of SourceObjectName must be URL-encoded. The version ID of the copied object version is returned as the x-oss-copy-source-version-id header value in the response.

If versionId is not specified in the request and the current version of the source object is a delete marker, OSS returns 404 Not Found. If versionId is specified in the request and the specified version of the source object is a delete marker, OSS returns 400 Bad Request.

The following code provides an example on how to use multipart copy to copy a large object:

```
import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.*;
import java.util.ArrayList;
import java.util.List;

public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the name of the source bucket.
        String sourceBucketName = "srcexamplebucket";
        // Specify the full path of the source object. The full path of the object cannot contain the bucket name.
        String sourceObjectName = "srcexampleobject.txt";
        // Specify the name of the destination bucket. The destination bucket must be in the same region as the source bucket.
        String destinationBucketName = "desexamplebucket";
        // Specify the full path of the destination object. The full path of the object cannot contain the bucket name.
        String destinationObjectName = "desexampleobject.txt";
        String sourceObjectVersionId = "CAEQMxiBgICAof2D0BYiIDJhMGE3N2M1YTI1NDQzOGY5NTkyNTI3MGYyMzM****";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);

        try {
            ObjectMetadata objectMetadata = ossClient.getObjectMetadata(sourceBucketName, s
```

```

        ObjectMetadata objectMetadata = ossClient.getObjectMetadata(sourceBucketName, sourceObjectName);
        // Query the size of the object to copy.
        long contentLength = objectMetadata.getContentLength();
        // Set the size of each part to 10 MB.
        long partSize = 1024 * 1024 * 10;
        // Calculate the total number of parts.
        int partCount = (int) (contentLength / partSize);
        if (contentLength % partSize != 0) {
            partCount++;
        }
        System.out.println("total part count:" + partCount);
        // Initiate a multipart copy task. You can use InitiateMultipartUploadRequest to specify the metadata of the destination object.
        InitiateMultipartUploadRequest initiateMultipartUploadRequest = new InitiateMultipartUploadRequest(destinationBucketName, destinationObjectName);
        InitiateMultipartUploadResult initiateMultipartUploadResult = ossClient.initiateMultipartUpload(initiateMultipartUploadRequest);
        String uploadId = initiateMultipartUploadResult.getUploadId();
        // Start the multipart copy task.
        List<PartETag> partETags = new ArrayList<PartETag>();
        for (int i = 0; i < partCount; i++) {
            // Calculate the size of each part.
            long skipBytes = partSize * i;
            long size = partSize < contentLength - skipBytes ? partSize : contentLength - skipBytes;
            // Create an UploadPartCopyRequest object. You can use UploadPartCopyRequest to specify conditions.
            UploadPartCopyRequest uploadPartCopyRequest =
                new UploadPartCopyRequest(sourceBucketName, sourceObjectName, destinationBucketName, destinationObjectName);
            uploadPartCopyRequest.setUploadId(uploadId);
            uploadPartCopyRequest.setPartSize(size);
            uploadPartCopyRequest.setBeginIndex(skipBytes);
            uploadPartCopyRequest.setPartNumber(i + 1);
            // Specify the version ID of the source object to that you want to copy.
            uploadPartCopyRequest.setSourceVersionId(sourceObjectVersionId);
            UploadPartCopyResult uploadPartCopyResult = ossClient.uploadPartCopy(uploadPartCopyRequest);
            // Store the returned part ETags in partETags.
            partETags.add(uploadPartCopyResult.getPartETag());
        }
        // Complete the multipart copy task.
        CompleteMultipartUploadRequest completeMultipartUploadRequest = new CompleteMultipartUploadRequest(
            destinationBucketName, destinationObjectName, uploadId, partETags);
        CompleteMultipartUploadResult completeMultipartUploadResult = ossClient.completeMultipartUpload(completeMultipartUploadRequest);
        // Display the version ID of the destination object.
        System.out.println("object versionid: " + completeMultipartUploadResult.getVersionId());
    } catch (OSSException oe) {
        System.out.println("Caught an OSSException, which means your request made it to OSS, "
            + "but was rejected with an error response for some reason.");
    }
}

```

```

        System.out.println("Error Message:" + oe.getErrorMessage());
        System.out.println("Error Code:" + oe.getErrorCode());
        System.out.println("Request ID:" + oe.getRequestId());
        System.out.println("Host ID:" + oe.getHostId());
    } catch (ClientException ce) {
        System.out.println("Caught an ClientException, which means the client encounter
ed "
            + "a serious internal problem while trying to communicate with OSS, "
            + "such as not being able to access the network.");
        System.out.println("Error Message:" + ce.getMessage());
    } finally {
        if (ossClient != null) {
            ossClient.shutdown();
        }
    }
}
}

```

References

- For more information about the API operation that you can call to copy a small object, see [CopyObject](#).
- For more information about the API operation that you can call to copy a large object, see [UploadPartCopy](#).

2.10.5. List objects

This topic describes how to list objects in a versioned bucket, including all objects, a specified number of objects, and objects whose names contain a specified prefix.

List the versions of all objects in a bucket

The following code provides an example on how to list the versions of all objects and delete markers within a specified bucket:

```

import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.*;
public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in Object Storage Service (OSS) is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the bucket name. Example: examplebucket.
        String bucketName = "examplebucket";
    }
}

```



```

// Create an OSSClient instance.
OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret
);

try {
    // List the versions of all objects and delete markers within the bucket.
    String nextKeyMarker = null;
    String nextVersionMarker = null;
    VersionListing versionListing = null;
    do {
        ListVersionsRequest listVersionsRequest = new ListVersionsRequest()
            .withBucketName(bucketName)
            .withKeyMarker(nextKeyMarker)
            .withVersionIdMarker(nextVersionMarker);
        versionListing = ossClient.listVersions(listVersionsRequest);
        for (OSSVersionSummary ossVersion : versionListing.getVersionSummaries()) {
            System.out.println("key name: " + ossVersion.getKey());
            System.out.println("versionid: " + ossVersion.getVersionId());
            System.out.println("Is latest: " + ossVersion.isLatest());
            System.out.println("Is delete marker: " + ossVersion.isDeleteMarker());
        }
        nextKeyMarker = versionListing.getNextKeyMarker();
        nextVersionMarker = versionListing.getNextVersionIdMarker();
    } while (versionListing.isTruncated());
} catch (OSSException oe) {
    System.out.println("Caught an OSSException, which means your request made it to
OSS, "

        + "but was rejected with an error response for some reason.");
    System.out.println("Error Message:" + oe.getErrorMessage());
    System.out.println("Error Code:" + oe.getErrorCode());
    System.out.println("Request ID:" + oe.getRequestId());
    System.out.println("Host ID:" + oe.getHostId());
} catch (ClientException ce) {
    System.out.println("Caught an ClientException, which means the client encounter
ed "

        + "a serious internal problem while trying to communicate with OSS, "
        + "such as not being able to access the network.");
    System.out.println("Error Message:" + ce.getMessage());
} finally {
    if (ossClient != null) {
        ossClient.shutdown();
    }
}
}
}

```

List the versions of objects whose names contain a specified prefix

The following code provides an example on how to list the versions of objects whose names contain a specified prefix:

```

import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;

```

```

import com.aliyun.oss.internal.OSSConstants;
import com.aliyun.oss.model.*;
public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the bucket name. Example: examplebucket.
        String bucketName = "examplebucket";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);

        try {
            // List the versions of objects whose names contain the "test-" prefix.
            String prefix = "test-";
            String nextKeyMarker = null;
            String nextVersionMarker = null;
            VersionListing versionListing = null;
            do {
                ListVersionsRequest listVersionsRequest = new ListVersionsRequest()
                    .withBucketName(bucketName)
                    .withKeyMarker(nextKeyMarker)
                    .withVersionIdMarker(nextVersionMarker)
                    .withEncodingType(OSSConstants.URL_ENCODING)
                    .withPrefix(prefix);
                versionListing = ossClient.listVersions(listVersionsRequest);
                // Display the versions of the listed objects.
                for (OSSVersionSummary ossVersion : versionListing.getVersionSummaries()) {
                    System.out.println("key name: " + ossVersion.getKey());
                    System.out.println("versionid: " + ossVersion.getVersionId());
                    System.out.println("Is latest: " + ossVersion.isLatest());
                    System.out.println("Is delete marker: " + ossVersion.isDeleteMarker());
                }
                nextKeyMarker = versionListing.getNextKeyMarker();
                nextVersionMarker = versionListing.getNextVersionIdMarker();
            } while (versionListing.isTruncated());
        } catch (OSSException oe) {
            System.out.println("Caught an OSSException, which means your request made it to OSS, "
                + "but was rejected with an error response for some reason.");
            System.out.println("Error Message:" + oe.getErrorMessage());
            System.out.println("Error Code:" + oe.getErrorCode());
            System.out.println("Request ID:" + oe.getRequestId());
            System.out.println("Host ID:" + oe.getHostId());
        } catch (ClientException ce) {
            System.out.println("Caught an ClientException, which means the client encountered "
                + "a serious internal problem while trying to communicate with OSS, "

```

```

        + "such as not being able to access the network.");
        System.out.println("Error Message:" + ce.getMessage());
    } finally {
        if (ossClient != null) {
            ossClient.shutdown();
        }
    }
}
}
}

```

List the versions of a specified number of objects

The following code provides an example on how to list the versions of a specified number of objects:

```

import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.*;
public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the bucket name. Example: examplebucket.
        String bucketName = "examplebucket";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);

        try {
            // Specify to return up to 200 versions.
            ListVersionsRequest listVersionsRequest = new ListVersionsRequest()
                .withBucketName(bucketName)
                .withMaxResults(200);
            VersionListing versionListing = ossClient.listVersions(listVersionsRequest);
            for (OSSVersionSummary ossVersion : versionListing.getVersionSummaries()) {
                System.out.println("key name: " + ossVersion.getKey());
                // If versioning is not enabled for the bucket, the version IDs of the listed objects are none.
                System.out.println("versionid: " + ossVersion.getVersionId());
                System.out.println("Is latest: " + ossVersion.isLatest());
                System.out.println("Is delete marker: " + ossVersion.isDeleteMarker());
            }
        } catch (OSSException oe) {
            System.out.println("Caught an OSSException, which means your request made it to OSS, "
                + "but was rejected with an error response for some reason.");
            System.out.println("Error Message:" + oe.getErrorMessage());
        }
    }
}

```

```

        System.out.println("Error Code:" + oe.getErrorCode());
        System.out.println("Request ID:" + oe.getRequestId());
        System.out.println("Host ID:" + oe.getHostId());
    } catch (ClientException ce) {
        System.out.println("Caught an ClientException, which means the client encounter
ed "
            + "a serious internal problem while trying to communicate with OSS, "
            + "such as not being able to access the network.");
        System.out.println("Error Message:" + ce.getMessage());
    } finally {
        if (ossClient != null) {
            ossClient.shutdown();
        }
    }
}
}

```

List the versions of all objects by page

The following code provides an example on how to list the versions of all objects by page:

```

import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.*;

public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the bucket name. Example: examplebucket.
        String bucketName = "examplebucket";
        // List up to 200 versions on each page.
        int maxKeys = 200;
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);

        try {
            // List the versions of all objects in the bucket by page.
            String nextKeyMarker = null;
            String nextVersionMarker = null;
            VersionListing versionListing = null;
            do {
                ListVersionsRequest listVersionsRequest = new ListVersionsRequest()
                    .withBucketName(bucketName)
                    .withKeyMarker(nextKeyMarker)
                    .withMaxResults(maxKeys)
            }
        }
    }
}

```

```

        .withVersionIdMarker(nextVersionMarker);
        versionListing = ossClient.listVersions(listVersionsRequest);
        for (OSSVersionSummary ossVersion : versionListing.getVersionSummaries()) {
            System.out.println("key name: " + ossVersion.getKey());
            System.out.println("versionid: " + ossVersion.getVersionId());
            System.out.println("Is latest: " + ossVersion.isLatest());
            System.out.println("Is delete marker: " + ossVersion.isDeleteMarker());
        }
        nextKeyMarker = versionListing.getNextKeyMarker();
        nextVersionMarker = versionListing.getNextVersionIdMarker();
    } while (versionListing.isTruncated());
} catch (OSSException oe) {
    System.out.println("Caught an OSSException, which means your request made it to
OSS, "
        + "but was rejected with an error response for some reason.");
    System.out.println("Error Message:" + oe.getErrorMessage());
    System.out.println("Error Code:" + oe.getErrorCode());
    System.out.println("Request ID:" + oe.getRequestId());
    System.out.println("Host ID:" + oe.getHostId());
} catch (ClientException ce) {
    System.out.println("Caught an ClientException, which means the client encounter
ed "
        + "a serious internal problem while trying to communicate with OSS, "
        + "such as not being able to access the network.");
    System.out.println("Error Message:" + ce.getMessage());
} finally {
    if (ossClient != null) {
        ossClient.shutdown();
    }
}
}
}

```

List the versions of objects whose names contain special characters

If the name of an object contains any of the following special characters, you must encode the object name before you transmit the object. OSS supports only URL encoding.

- Single quotation marks (')
- Double quotations marks (")
- Ampersands (&)
- Angle brackets (<>)
- Pause markers
- Chinese characters

The following code provides an example on how to list the versions of objects whose names contain special characters:

```

import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.internal.OSSConstants;

```

```

import com.aliyun.oss.model.*;
public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the bucket name. Example: examplebucket.
        String bucketName = "examplebucket";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);

        try {
            // Specify that the object names in the response are URL-encoded.
            String nextKeyMarker = null;
            String nextVersionMarker = null;
            VersionListing versionListing = null;
            do {
                ListVersionsRequest listVersionsRequest = new ListVersionsRequest()
                    .withBucketName(bucketName)
                    .withKeyMarker(nextKeyMarker)
                    .withVersionIdMarker(nextVersionMarker)
                    .withEncodingType(OSSConstants.URL_ENCODING);
                versionListing = ossClient.listVersions(listVersionsRequest);
                // Display the versions of the listed objects.
                for (OSSVersionSummary ossVersion : versionListing.getVersionSummaries()) {
                    System.out.println("key name: " + ossVersion.getKey());
                    System.out.println("versionid: " + ossVersion.getVersionId());
                    System.out.println("Is latest: " + ossVersion.isLatest());
                    System.out.println("Is delete marker: " + ossVersion.isDeleteMarker());
                }
                nextKeyMarker = versionListing.getNextKeyMarker();
                nextVersionMarker = versionListing.getNextVersionIdMarker();
            } while (versionListing.isTruncated());
        } catch (OSSException oe) {
            System.out.println("Caught an OSSException, which means your request made it to OSS, "
                + "but was rejected with an error response for some reason.");
            System.out.println("Error Message:" + oe.getErrorMessage());
            System.out.println("Error Code:" + oe.getErrorCode());
            System.out.println("Request ID:" + oe.getRequestId());
            System.out.println("Host ID:" + oe.getHostId());
        } catch (ClientException ce) {
            System.out.println("Caught an ClientException, which means the client encountered "
                + "a serious internal problem while trying to communicate with OSS, "
                + "such as not being able to access the network.");
            System.out.println("Error Message:" + ce.getMessage());
        } finally {

```

```

        if (ossClient != null) {
            ossClient.shutdown();
        }
    }
}

```

List objects by directory

OSS uses a flat structure for objects instead of a hierarchical structure to store objects. A directory is an object whose size is 0 and whose name ends with a forward slash (/). You can upload and download this object. By default, objects whose names end with a forward slash (/) are displayed as directories in the OSS console.

You can specify the delimiter and prefix parameters to list objects by directory.

- If you set prefix to a directory name in the request, objects and subdirectories whose names contain the prefix are listed.
- If you also set delimiter to a forward slash (/) in the request, objects and subdirectories whose names start with the specified prefix in the directory are listed. Each subdirectory is listed as a single result element in CommonPrefixes. The objects and directories in these subdirectories are not listed.

Example: The following four objects are stored in a bucket: *oss.jpg*, *fun/test.jpg*, *fun/movie/001.avi*, and *fun/movie/007.avi*. The forward slash (/) is specified as the directory delimiter. The following examples describe how to list objects in simulated directories.

- List the versions of objects within the root directory of a bucket

The following code provides an example on how to list the versions of objects within the root directory of a bucket:

```

import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.internal.OSSConstants;
import com.aliyun.oss.model.*;

public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify
        // your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API oper-
        // ations. Using these credentials to perform operations in OSS is a high-risk operation. We
        // recommend that you use a RAM user to call API operations or perform routine O&M. To creat
        // e a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the bucket name. Example: examplebucket.
        String bucketName = "examplebucket";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecr
        et);

        try {
            // Set the delimiter parameter to a forward slash (/) to list the versions of
            // objects and the names of subdirectories within the root directory.

```

```

Objects and the names of subdirectories within the root directory.
    String delimiter = "/";
    String nextKeyMarker = null;
    String nextVersionMarker = null;
    VersionListing versionListing = null;
    do {
        ListVersionsRequest listVersionsRequest = new ListVersionsRequest()
            .withBucketName(bucketName)
            .withKeyMarker(nextKeyMarker)
            .withVersionIdMarker(nextVersionMarker)
            .withEncodingType(OSSConstants.URL_ENCODING)
            .withDelimiter(delimiter);
        versionListing = ossClient.listVersions(listVersionsRequest);
        // Display the versions of the listed objects.
        for (OSSVersionSummary ossVersion : versionListing.getVersionSummaries())
        {
            System.out.println("key name: " + ossVersion.getKey());
            System.out.println("versionid: " + ossVersion.getVersionId());
            System.out.println("Is latest: " + ossVersion.isLatest());
            System.out.println("Is delete marker: " + ossVersion.isDeleteMarker());
        }
        // Display the names of directories whose names end with a forward slash
        (/) in the root directory.
        for (String commonPrefix : versionListing.getCommonPrefixes()) {
            System.out.println("commonPrefix: " + commonPrefix);
        }
        nextKeyMarker = versionListing.getNextKeyMarker();
        nextVersionMarker = versionListing.getNextVersionIdMarker();
    } while (versionListing.isTruncated());
} catch (OSSException oe) {
    System.out.println("Caught an OSSException, which means your request made it
to OSS, "
        + "but was rejected with an error response for some reason.");
    System.out.println("Error Message:" + oe.getErrorMessage());
    System.out.println("Error Code:" + oe.getErrorCode());
    System.out.println("Request ID:" + oe.getRequestId());
    System.out.println("Host ID:" + oe.getHostId());
} catch (ClientException ce) {
    System.out.println("Caught an ClientException, which means the client encount
ered "
        + "a serious internal problem while trying to communicate with OSS, "
        + "such as not being able to access the network.");
    System.out.println("Error Message:" + ce.getMessage());
} finally {
    if (ossClient != null) {
        ossClient.shutdown();
    }
}
}
}

```

Returned results


```
key name: oss.jpg
versionid: CAEQEHibgMCw8Y7FqBciIGIzMDE3MTEzOWRiMDRmZmFhMmRlMj1jZWlOMWU4****
Is latest: true
Is delete marker: false
commonPrefix: fun/
```

- List objects and subdirectories in a specified directory

The following code provides an example on how to list objects and subdirectories in a specified directory:

```
import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.internal.OSSConstants;
import com.aliyun.oss.model.*;
public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify
        // your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API oper-
        // ations. Using these credentials to perform operations in OSS is a high-risk operation. We
        // recommend that you use a RAM user to call API operations or perform routine O&M. To creat
        // e a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the bucket name. Example: examplebucket.
        String bucketName = "examplebucket";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecr
        et);
        try {
            // Configure the prefix parameter to list all objects and subdirectories with
            // in the fun directory. Set the delimiter parameter to a forward slash (/).
            String prefix = "fun/";
            String delimiter = "/";
            String nextKeyMarker = null;
            String nextVersionMarker = null;
            VersionListing versionListing = null;
            do {
                ListVersionsRequest listVersionsRequest = new ListVersionsRequest()
                    .withBucketName(bucketName)
                    .withKeyMarker(nextKeyMarker)
                    .withVersionIdMarker(nextVersionMarker)
                    .withEncodingType(OSSConstants.URL_ENCODING)
                    .withDelimiter(delimiter)
                    .withPrefix(prefix);
                versionListing = ossClient.listVersions(listVersionsRequest);
                // Display the versions of the listed objects.
                for (OSSVersionSummary ossVersion : versionListing.getVersionSummaries())
                {
                    System.out.println("key name: " + ossVersion.getKey());
                    System.out.println("versionid: " + ossVersion.getVersionId());
```

```

        System.out.println("versionid: " + ossVersion.getVersionId());
        System.out.println("Is latest: " + ossVersion.isLatest());
        System.out.println("Is delete marker: " + ossVersion.isDeleteMarker()
    );
    }
    // Display the names of the subdirectories within the fun directory.
    for (String commonPrefix : versionListing.getCommonPrefixes()) {
        System.out.println("commonPrefix: " + commonPrefix);
    }
    nextKeyMarker = versionListing.getNextKeyMarker();
    nextVersionMarker = versionListing.getNextVersionIdMarker();
    } while (versionListing.isTruncated());
} catch (OSSException oe) {
    System.out.println("Caught an OSSException, which means your request made it
to OSS, "
        + "but was rejected with an error response for some reason.");
    System.out.println("Error Message:" + oe.getErrorMessage());
    System.out.println("Error Code:" + oe.getErrorCode());
    System.out.println("Request ID:" + oe.getRequestId());
    System.out.println("Host ID:" + oe.getHostId());
} catch (ClientException ce) {
    System.out.println("Caught an ClientException, which means the client encount
ered "
        + "a serious internal problem while trying to communicate with OSS, "
        + "such as not being able to access the network.");
    System.out.println("Error Message:" + ce.getMessage());
} finally {
    if (ossClient != null) {
        ossClient.shutdown();
    }
}
}
}
}

```

Returned results

```

key name: fun/test.jpg
versionid: CAEQEhiBgIC48Y7FqBciIDU4NjAzMjczMtY5NDRjYmVhNGY4NTM2YTdjY2Ji****
Is latest: true
Is delete marker: false
commonPrefix: fun/movie/

```

2.10.6. Delete objects

This topic describes how to delete a single object, multiple objects, or objects whose names contain a specified prefix from a versioned bucket.

Delete operations in a versioned bucket

When you delete an object from a versioned bucket, you must determine whether to specify a version ID in the request.

- Delete an object without specifying a version ID (temporary deletion)

By default, if you do not specify the version ID of the object that you want to delete in the request, Object Storage Service (OSS) does not delete the current version of the object but adds a delete marker to the object as the new current version. If you perform the `GetObject` operation on the object, OSS identifies the current version of the object as a delete marker and returns `404 Not Found`. Additionally, the `x-oss-delete-marker` header and the `x-oss-version-id` header that indicates the version ID of the created delete marker are included in the response to the `DeleteObject` request.

The value of the `x-oss-delete-marker` header in the response is true, which indicates that the `x-oss-version-id` value is the version ID of a delete marker.

- Delete an object by specifying a version ID (permanent deletion)

If you specify the version ID of the object that you want to delete in the request, OSS permanently deletes the version specified by the `versionId` field in the `params` parameter. To delete a version whose ID is null, add `params['versionId'] = "null"` to `params` in the request. OSS identifies the string "null" as the ID of the version to delete and deletes the version whose ID is null.

Delete a single object

The following examples demonstrate how to permanently or temporarily delete a single object from a versioned bucket.

- Permanently delete an object from a versioned bucket

The following code provides an example on how to permanently delete an object from a versioned bucket by specifying the version ID of the object in the request:

```

import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSEException;
public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify
        // your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API oper-
        // ations. Using these credentials to perform operations in OSS is a high-risk operation. We
        // recommend that you use a RAM user to call API operations or perform routine O&M. To creat-
        // e a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the bucket name. Example: examplebucket.
        String bucketName = "examplebucket";
        // Specify the full path of the object. The full path of the object cannot contain
        // the bucket name.
        String objectName = "exampledir/object";
        String versionId = "CAEQMxiBgICAof2D0BYiIDJhMGE3N2M1YTI1NDQzOGY5NTkyNTI3MGYyMzJm
        *****";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);
        try {
            // Delete the object that has the specified version ID.
            ossClient.deleteVersion(bucketName, objectName, versionId);
        } catch (OSSEException oe) {
            System.out.println("Caught an OSSEException, which means your request made it
            to OSS, "
                + "but was rejected with an error response for some reason.");
            System.out.println("Error Message:" + oe.getErrorMessage());
            System.out.println("Error Code:" + oe.getErrorCode());
            System.out.println("Request ID:" + oe.getRequestId());
            System.out.println("Host ID:" + oe.getHostId());
        } catch (ClientException ce) {
            System.out.println("Caught an ClientException, which means the client encountered "
                + "a serious internal problem while trying to communicate with OSS, "
                + "such as not being able to access the network.");
            System.out.println("Error Message:" + ce.getMessage());
        } finally {
            if (ossClient != null) {
                ossClient.shutdown();
            }
        }
    }
}

```

- Temporarily delete an object from a versioned bucket

The following code provides an example on how to temporarily delete an object from a versioned bucket by sending a request in which no version ID is specified:

```

import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify
        // your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API oper-
        // ations. Using these credentials to perform operations in OSS is a high-risk operation. We
        // recommend that you use a RAM user to call API operations or perform routine O&M. To creat-
        // e a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the bucket name. Example: examplebucket.
        String bucketName = "examplebucket";
        // Specify the full path of the object. The full path of the object cannot contain
        // the bucket name.
        String objectName = "exampledir/object";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);
        try {
            // Temporarily delete the object. A delete marker is added to the object.
            ossClient.deleteObject(bucketName, objectName);
        } catch (OSSException oe) {
            System.out.println("Caught an OSSException, which means your request made it
            to OSS, "
                + "but was rejected with an error response for some reason.");
            System.out.println("Error Message:" + oe.getErrorMessage());
            System.out.println("Error Code:" + oe.getErrorCode());
            System.out.println("Request ID:" + oe.getRequestId());
            System.out.println("Host ID:" + oe.getHostId());
        } catch (ClientException ce) {
            System.out.println("Caught an ClientException, which means the client encountered "
                + "a serious internal problem while trying to communicate with OSS, "
                + "such as not being able to access the network.");
            System.out.println("Error Message:" + ce.getMessage());
        } finally {
            if (ossClient != null) {
                ossClient.shutdown();
            }
        }
    }
}

```

Delete multiple objects

The following examples demonstrate how to permanently or temporarily delete multiple objects from a versioned bucket.

- Permanently delete multiple objects from a versioned bucket

The following code provides an example on how to permanently delete multiple objects or delete markers from a versioned bucket by specifying the version IDs of the objects or delete markers in the request:

```
import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.DeleteVersionsRequest;
import com.aliyun.oss.model.DeleteVersionsResult;
import java.net.URLDecoder;
import java.util.ArrayList;
import java.util.List;
public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify
        // your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API oper
        // ations. Using these credentials to perform operations in OSS is a high-risk operation. We
        // recommend that you use a RAM user to call API operations or perform routine O&M. To creat
        // e a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the bucket name. Example: examplebucket.
        String bucketName = "examplebucket";
        // Specify the full path of the object. The full path of the object cannot contain
        // the bucket name.
        String objectName = "exampledir/object";
        String object2Name = "exampledir/object2";
        String yourObjectNameVersionId = "CAEQMxiBgICAof2D0BYiIDJhMGE3N2M1YTI1NDQzOGY5NT
        kyNTI3MGYyMzJm****";
        String yourObject2DelMarkerNameVersionId = "MGE3N2M1YgICAof2D0BYiID3N2M1YTITI1ND
        QzOGY5NTN2M1YTI1NDQz****";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);
        try {
            // Permanently delete the objects or delete markers that have specified version
            // IDs.
            List<DeleteVersionsRequest.KeyVersion> keyVersionsList = new ArrayList<Delete
            VersionsRequest.KeyVersion>();
            keyVersionsList.add(new DeleteVersionsRequest.KeyVersion(objectName,yourObject
            NameVersionId));
            keyVersionsList.add(new DeleteVersionsRequest.KeyVersion(object2Name,yourObject
            2DelMarkerNameVersionId));
            DeleteVersionsRequest delVersionsRequest = new DeleteVersionsRequest(bucketName);
            delVersionsRequest.setKeys(keyVersionsList);
            // Initiate a deleteVersions request.
            DeleteVersionsResult delVersionsResult = ossClient.deleteVersions(delVersions
            Request);
            // Display the deletion result.
            for (DeleteVersionsResult.DeletedVersion delVer : delVersionsResult.getDelete
```

```

dVersions()) {
    String keyName = URLDecoder.decode(delVer.getKey(), "UTF-8");
    String keyVersionId = delVer.getVersionId();
    String keyDelMarkerId = delVer.getDeleteMarkerVersionId();
    System.out.println("delete key: " + keyName);
    System.out.println("delete key versionId: " + keyVersionId);
    if(keyDelMarkerId != null && keyDelMarkerId.length() != 0){
        System.out.println("delete key del_marker versionId: " + keyDelMarker
Id);
    }
}
} catch (OSSEException oe) {
    System.out.println("Caught an OSSEException, which means your request made it
to OSS, "
        + "but was rejected with an error response for some reason.");
    System.out.println("Error Message:" + oe.getErrorMessage());
    System.out.println("Error Code:" + oe.getErrorCode());
    System.out.println("Request ID:" + oe.getRequestId());
    System.out.println("Host ID:" + oe.getHostId());
} catch (ClientException ce) {
    System.out.println("Caught an ClientException, which means the client encount
ered "
        + "a serious internal problem while trying to communicate with OSS, "
        + "such as not being able to access the network.");
    System.out.println("Error Message:" + ce.getMessage());
} finally {
    if (ossClient != null) {
        ossClient.shutdown();
    }
}
}
}
}

```

- Temporarily delete multiple objects from a versioned bucket

The following code provides an example on how to delete multiple objects from a versioned bucket by sending a request in which no version IDs are specified:

```

import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSEException;
import com.aliyun.oss.model.DeleteObjectsRequest;
import com.aliyun.oss.model.DeleteObjectsResult;
import java.net.URLDecoder;
import java.util.ArrayList;
import java.util.List;
public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify
your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API oper
ations. Using these credentials to perform operations in OSS is a high-risk operation. We
recommend that you use a RAM user to call API operations or perform routine O&M. To creat

```

```

e a RAM user, log on to the RAM console.
    String accessKeyId = "yourAccessKeyId";
    String accessKeySecret = "yourAccessKeySecret";
    // Specify the bucket name. Example: examplebucket.
    String bucketName = "examplebucket";
    // Specify the full path of the object. The full path of the object cannot contain the bucket name.
    String objectName = "exemplerdir/object";
    String object2Name = "exemplerdir/object2";
    // Create an OSSClient instance.
    OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);

    try {
        // Delete multiple objects at a time.
        List<String> KeysList = new ArrayList<String>();
        KeysList.add(objectName);
        KeysList.add(object2Name);
        DeleteObjectsRequest request = new DeleteObjectsRequest(bucketName);
        request.setKeys(KeysList);
        // Initiate a deleteObjects request.
        DeleteObjectsResult delObjResult = ossClient.deleteObjects(request);
        // Display the deletion result.
        for (String o : delObjResult.getDeletedObjects()) {
            String keyName = URLDecoder.decode(o, "UTF-8");
            System.out.println("delete key name: " + keyName);
        }
    } catch (OSSException oe) {
        System.out.println("Caught an OSSException, which means your request made it to OSS, "
            + "but was rejected with an error response for some reason.");
        System.out.println("Error Message:" + oe.getErrorMessage());
        System.out.println("Error Code:" + oe.getErrorCode());
        System.out.println("Request ID:" + oe.getRequestId());
        System.out.println("Host ID:" + oe.getHostId());
    } catch (ClientException ce) {
        System.out.println("Caught an ClientException, which means the client encountered "
            + "a serious internal problem while trying to communicate with OSS, "
            + "such as not being able to access the network.");
        System.out.println("Error Message:" + ce.getMessage());
    } finally {
        if (ossClient != null) {
            ossClient.shutdown();
        }
    }
}

```

Delete objects whose names contain a specified prefix

The following code provides an example on how to delete objects whose names contain a specified prefix:

```

import com.aliyun.oss.ClientException;

```



```

import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.*;
import java.util.ArrayList;
import java.util.List;

public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the bucket name. Example: examplebucket.
        String bucketName = "examplebucket";
        // Specify the prefix that is contained in the names of the objects that you want to delete.
        String prefix = "yourkeyPrefix";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);

        try {
            // List the versions of all objects whose names contain the specified prefix and delete the versions.
            String nextKeyMarker = null;
            String nextVersionMarker = null;
            VersionListing versionListing = null;
            do {
                ListVersionsRequest listVersionsRequest = new ListVersionsRequest()
                    .withBucketName(bucketName)
                    .withKeyMarker(nextKeyMarker)
                    .withVersionIdMarker(nextVersionMarker)
                    .withPrefix(prefix);
                versionListing = ossClient.listVersions(listVersionsRequest);
                if (versionListing.getVersionSummaries().size() > 0) {
                    List<DeleteVersionsRequest.KeyVersion> keyVersionsList = new ArrayList<DeleteVersionsRequest.KeyVersion>();
                    for (OSSVersionSummary ossVersion : versionListing.getVersionSummaries()) {
                        System.out.println("key name: " + ossVersion.getKey());
                        System.out.println("versionid: " + ossVersion.getVersionId());
                        System.out.println("Is delete marker: " + ossVersion.isDeleteMarker());

                        keyVersionsList.add(new DeleteVersionsRequest.KeyVersion(ossVersion.getKey(), ossVersion.getVersionId()));
                    }
                    DeleteVersionsRequest delVersionsRequest = new DeleteVersionsRequest(bucketName).withKeys(keyVersionsList);
                    ossClient.deleteVersions(delVersionsRequest);
                }
                nextKeyMarker = versionListing.getNextKeyMarker();
            } while (nextKeyMarker != null);
        } catch (OSSException oe) {
            System.out.println("An error occurred: " + oe.getMessage());
            oe.printStackTrace();
        }
    }
}

```

```

        nextKeyMarker = versionListing.getNextKeyMarker();
        nextVersionMarker = versionListing.getNextVersionIdMarker();
    } while (versionListing.isTruncated());
} catch (OSSException oe) {
    System.out.println("Caught an OSSException, which means your request made it to
OSS, "

        + "but was rejected with an error response for some reason.");
    System.out.println("Error Message:" + oe.getErrorMessage());
    System.out.println("Error Code:" + oe.getErrorCode());
    System.out.println("Request ID:" + oe.getRequestId());
    System.out.println("Host ID:" + oe.getHostId());
} catch (ClientException ce) {
    System.out.println("Caught an ClientException, which means the client encounter
ed "

        + "a serious internal problem while trying to communicate with OSS, "
        + "such as not being able to access the network.");
    System.out.println("Error Message:" + ce.getMessage());
} finally {
    if (ossClient != null) {
        ossClient.shutdown();
    }
}
}
}
}

```

References

- For more information about the API operation that you can call to delete a single object, see [DeleteObject](#).
- For more information about the API operation that you can call to delete multiple objects, see [DeleteMultipleObjects](#).

2.10.7. Restore objects

In a versioned bucket, the storage classes of different versions of an object can be different. By default, when you perform the `RestoreObject` operation on an object, the current version of the object is restored. You can specify a version ID in the request to restore the specified version of an object.

Sample code

The following code provides an example on how to restore an object:

```

import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.*;
public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify yo
ur actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API operat
ions. Using these credentials to perform operations in Object Storage Service (OSS) is a hi

```

```

gh-risk operation. We recommend that you use a RAM user to call API operations or perform r
outine O&M. To create a RAM user, log on to the RAM console.
    String accessKeyId = "yourAccessKeyId";
    String accessKeySecret = "yourAccessKeySecret";
    // Specify the bucket name. Example: examplebucket.
    String bucketName = "examplebucket";
    // Specify the full path of the object. The full path of the object cannot contain
the bucket name.
    String objectName = "exampledir/object";
    // Specify the version ID of the Archive object that you want to restore.
    String versionId = "CAEQMxiBgICAof2D0BYiIDJhMGE3N2MlYTI1NDQzOGY5NTkyNTI3MGYyMzJm***
*";
    // Create an OSSClient instance.
    OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret
);
    try {
        // Query part of metadata of the specified version of the object.
        GenericRequest getObjectMetadataRequest = new GenericRequest(bucketName, objectN
ame, versionId);
        ObjectMetadata objectMetadata = ossClient.getObjectMetadata(getObjectMetadataRe
quest);
        // Check whether the specified version of the object is an Archive object.
        StorageClass storageClass = objectMetadata.getObjectStorageClass();
        System.out.println("object storage class:" + objectMetadata.getObjectStorageCla
ss());
        if (storageClass == StorageClass.Archive) {
            // Restore the object.
            GenericRequest genericRequest = new GenericRequest(bucketName, objectName,
versionId);
            RestoreObjectResult restoreObjectResult = ossClient.restoreObject(genericRe
quest);
            System.out.println("restor versionid: " + restoreObjectResult.getVersionId(
));
            // Wait until the object is restored.
            do {
                Thread.sleep(1000);
                objectMetadata = ossClient.getObjectMetadata(getObjectMetadataRequest);
                System.out.println("is competed:" + objectMetadata.isRestoreCompleted()
);
            } while (!objectMetadata.isRestoreCompleted());
        }
        // Specify the version ID of the object that you want to restore.
        GetObjectRequest getObjectRequest = new GetObjectRequest(bucketName, objectName
);
        getObjectRequest.setVersionId(versionId);
        OSSObject ossObject = ossClient.getObject(getObjectRequest);
        ossObject.getObjectContent().close();
    } catch (OSSException oe) {
        System.out.println("Caught an OSSException, which means your request made it to
OSS, "
            + "but was rejected with an error response for some reason.");
        System.out.println("Error Message:" + oe.getErrorMessage());
        System.out.println("Error Code:" + oe.getErrorCode());
        System.out.println("Request ID:" + oe.getRequestId());
    }
}

```

```

        System.out.println("Host ID:" + oe.getHostId());
    } catch (ClientException ce) {
        System.out.println("Caught an ClientException, which means the client encounter
ed "
            + "a serious internal problem while trying to communicate with OSS, "
            + "such as not being able to access the network.");
        System.out.println("Error Message:" + ce.getMessage());
    } finally {
        if (ossClient != null) {
            ossClient.shutdown();
        }
    }
}
}

```

References

For more information about the API operation that you can call to restore an object, see [RestoreObject](#).

2.10.8. Query object metadata

By default, when you call the `HeadObject` operation on an object in a versioned bucket, the metadata of only the current version of the object is returned.

 **Note** If the current version of the object is a delete marker, OSS returns 404 Not Found. If you specify the version ID in the request, OSS returns the metadata of the specified version of the object.

Sample code

The following code provides an example on how to query object metadata:

```

import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.*;
public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify the
        // actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations.
        // Using these credentials to access OSS is a high-risk operation. We recommend that you
        // use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on
        // to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the name of the bucket. Example: examplebucket.
        String bucketName = "examplebucket";
        // Specify the full path of the object. The full path of the object cannot contain
        // the bucket name.
        String objectName = "exampledir/object";
    }
}

```

```

// Specify the version ID of the Archive object whose metadata you want to query.
String versionId = "CAEQMxiBgICAof2D0BYiIDJhMGE3N2MlYTI1NDQzOGY5NTkyNTI3MGYyMzJm***
*";

// Create an OSSClient instance.
OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret
);

try {
    // Query part of the metadata of the specified version of the object.
    GenericRequest getSimplifiedObjectMetaRequest = new GenericRequest(bucketName,
objectName, versionId);
    SimplifiedObjectMeta objectMeta = ossClient.getSimplifiedObjectMeta(getSimplifi
edObjectMetaRequest);
    System.out.println(objectMeta.getSize());
    System.out.println(objectMeta.getETag());
    System.out.println(objectMeta.getLastModified());
    // Query all metadata of the specified version of the object.
    GenericRequest getObjectMetadataRequest = new GenericRequest(bucketName, object
Name, versionId);
    ObjectMetadata metadata = ossClient.getObjectMetadata(getObjectMetadataRequest)
;

    System.out.println(metadata.getContentType());
    System.out.println(metadata.getLastModified());
    System.out.println(metadata.getExpirationTime());
} catch (OSSException oe) {
    System.out.println("Caught an OSSException, which means your request made it to
OSS, "

        + "but was rejected with an error response for some reason.");
    System.out.println("Error Message:" + oe.getErrorMessage());
    System.out.println("Error Code:" + oe.getErrorCode());
    System.out.println("Request ID:" + oe.getRequestId());
    System.out.println("Host ID:" + oe.getHostId());
} catch (ClientException ce) {
    System.out.println("Caught an ClientException, which means the client encounter
ed "

        + "a serious internal problem while trying to communicate with OSS, "
        + "such as not being able to access the network.");
    System.out.println("Error Message:" + ce.getMessage());
} finally {
    if (ossClient != null) {
        ossClient.shutdown();
    }
}
}
}

```

References

For more information about the API operations that you can call to query the metadata of an object, see [HeadObject](#) and [GetObjectMeta](#).

2.10.9. Manage object ACLs

This topic describes how to manage the access control lists (ACLs) of objects in a versioned bucket.

Configure the ACL of an object

By default, the PutObjectACL operation is called to configure the ACL of the current version of an object. If the current version of the object is a delete marker, OSS returns 404 Not Found. You can specify a version ID in the request to configure the ACL of the specified version of an object.

The following code provides an example on how to configure the ACL of an object:

```
import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.*;

public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify the actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to access OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the name of the bucket. Example: examplebucket.
        String bucketName = "examplebucket";
        // Specify the full path of the object. The full path of the object cannot contain the bucket name.
        String objectName = "exampledir/object";
        // Specify the version ID of the object for which you want to configure the ACL.
        String versionId = "CAEQMxiBgICAof2D0BYiIDJhMGE3N2MlYTI1NDQzOGY5NTkyNTI3MGYyMzJm***";

        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);

        try {
            // Create a SetObjectAclRequest object. In this example, the ACL of the object is set to public-read.
            SetObjectAclRequest setObjectAclRequest = new SetObjectAclRequest(bucketName, objectName,
                versionId, CannedAccessControlList.PublicRead);
            // Configure the ACL of the object with the specified version ID.
            ossClient.setObjectAcl(setObjectAclRequest);
        } catch (OSSException oe) {
            System.out.println("Caught an OSSException, which means your request made it to OSS, "
                + "but was rejected with an error response for some reason.");
            System.out.println("Error Message:" + oe.getErrorMessage());
            System.out.println("Error Code:" + oe.getErrorCode());
            System.out.println("Request ID:" + oe.getRequestId());
            System.out.println("Host ID:" + oe.getHostId());
        } catch (ClientException ce) {
            System.out.println("Caught an ClientException, which means the client encountered "
                + "a serious internal problem while trying to communicate with OSS, "
```

```

        + "such as not being able to access the network.");
        System.out.println("Error Message:" + ce.getMessage());
    } finally {
        if (ossClient != null) {
            ossClient.shutdown();
        }
    }
}
}
}

```

Query the ACL of an object.

By default, the GetObjectACL operation is called to query the ACL of the current version of an object. If the current version of the object is a delete marker, OSS returns 404 Not Found. You can specify a version ID in the request to query the ACL of the specified version of an object.

The following code provides an example on how to query the ACL of an object:

```

import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.*;
public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify the
        // actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations.
        // Using these credentials to access OSS is a high-risk operation. We recommend that you
        // use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on
        // to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the name of the bucket. Example: examplebucket.
        String bucketName = "examplebucket";
        // Specify the full path of the object. The full path of the object cannot contain
        // the bucket name.
        String objectName = "exampledir/object";
        // Specify the version ID of the object whose ACL you want to query.
        String versionId = "CAEQMxiBgICAof2D0BYiIDJhMGE3N2M1YTI1NDQzOGY5NTkyNTI3MGYyMzJm***";

        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);

        try {
            // Query the ACL of the object.
            GenericRequest genericRequest = new GenericRequest(bucketName, objectName, versionId);

            ObjectAcl objectAcl = ossClient.getObjectAcl(genericRequest);
            System.out.println("get object acl: " + objectAcl.getPermission().toString());
            System.out.println("object versionid: " + objectAcl.getVersionId());
        } catch (OSSException oe) {
            System.out.println("Caught an OSSException, which means your request made it to

```

```

OSS, "
        + "but was rejected with an error response for some reason.");
    System.out.println("Error Message:" + oe.getErrorMessage());
    System.out.println("Error Code:" + oe.getErrorCode());
    System.out.println("Request ID:" + oe.getRequestId());
    System.out.println("Host ID:" + oe.getHostId());
} catch (ClientException ce) {
    System.out.println("Caught an ClientException, which means the client encounter
ed "
        + "a serious internal problem while trying to communicate with OSS, "
        + "such as not being able to access the network.");
    System.out.println("Error Message:" + ce.getMessage());
} finally {
    if (ossClient != null) {
        ossClient.shutdown();
    }
}
}
}

```

References

- For more information about the API operation that you can call to configure the ACL of an object, see [Put Object ACL](#).
- For more information about the API operation that you can call to query the ACL of an object, see [Get Object ACL](#).

2.10.10. Manage symbolic links

This topic describes how to manage symbolic links in a versioned bucket.

Create a symbolic link

You can create a symbolic link to which the current version of an object points.

 **Note** In a versioned bucket, a symbolic link cannot be created for a delete marker.

A symbolic link can have multiple versions. You can call the PutSymLink operation to point each version of a symbolic link to different objects. In this case, the IDs of the versions are generated by OSS and are returned in the response as the x-oss-version-id header.

The following code provides an example on how to create a symbolic link:

```

import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.*;
public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify the
        // actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // ...
    }
}

```



```

// The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to access OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
String accessKeyId = "yourAccessKeyId";
String accessKeySecret = "yourAccessKeySecret";
// Specify the name of the bucket. Example: examplebucket.
String bucketName = "examplebucket";
// Specify the name of the symbolic link that you want to create.
String symLink = "yourSymLink";
// Specify the full path of the object. The full path of the object cannot contain the bucket name.
String destinationObjectName = "exampledir/object";
// Create an OSSClient instance.
OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);

try {
    // Create metadata for the object that you want to upload.
    ObjectMetadata metadata = new ObjectMetadata();
    metadata.setContentType("text/plain");
    // Set the value of the property parameter for the user metadata to property-value.
    metadata.addUserMetadata("property", "property-value");
    // Create a request (CreateSymlinkRequest) to create the symbolic link.
    CreateSymlinkRequest createSymlinkRequest = new CreateSymlinkRequest(bucketName, symLink, destinationObjectName);
    // Configure the metadata.
    createSymlinkRequest.setMetadata(metadata);
    // Create the symbolic link.
    ossClient.createSymlink(createSymlinkRequest);
} catch (OSSException oe) {
    System.out.println("Caught an OSSException, which means your request made it to OSS, "
        + "but was rejected with an error response for some reason.");
    System.out.println("Error Message:" + oe.getErrorMessage());
    System.out.println("Error Code:" + oe.getErrorCode());
    System.out.println("Request ID:" + oe.getRequestId());
    System.out.println("Host ID:" + oe.getHostId());
} catch (ClientException ce) {
    System.out.println("Caught an ClientException, which means the client encountered "
        + "a serious internal problem while trying to communicate with OSS, "
        + "such as not being able to access the network.");
    System.out.println("Error Message:" + ce.getMessage());
} finally {
    if (ossClient != null) {
        ossClient.shutdown();
    }
}
}

```

Query a symbolic link

By default, the GetSymlink operation is called to query the current version of a symbolic link. You can specify the version ID in the request to query the specified version of a symbolic link. If the current version of the symbolic link is a delete marker, OSS returns 404 Not Found and includes x-oss-delete-marker = true and x-oss-version-id in the response header.

 **Note** To query a symbolic link, you must have read permissions on the symbolic link.

The following code provides an example on how to query a symbolic link:

```
import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.*;

public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify the actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to access OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the name of the bucket. Example: examplebucket.
        String bucketName = "examplebucket";
        // Specify the name of the symbolic link that you want to query.
        String symLink = "yourSymLink";
        // Specify the version ID of the symbolic link that you want to query.
        String versionId = "CAEQMxiBgICAof2D0BYiIDJhMGE3N2M1YTI1NDQzOGY5NTkyNTI3MGYyMzMjMm**";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);

        try {
            // Query the content of the symbolic link with the specified version ID.
            GenericRequest genericRequest = new GenericRequest(bucketName, symLink, versionId);

            OSSSymlink symbolicLink = ossClient.getSymlink(genericRequest);
            // Display the content of the object to which the symbolic link points.
            System.out.println(symbolicLink.getSymlink());
            System.out.println(symbolicLink.getTarget());
            System.out.println(symbolicLink.getRequestId());
            System.out.println(symbolicLink.getMetadata().getVersionId());
        } catch (OSSException oe) {
            System.out.println("Caught an OSSException, which means your request made it to OSS, "
                + "but was rejected with an error response for some reason.");
            System.out.println("Error Message:" + oe.getErrorMessage());
            System.out.println("Error Code:" + oe.getErrorCode());
            System.out.println("Request ID:" + oe.getRequestId());
            System.out.println("Host ID:" + oe.getHostId());
        } catch (ClientException ce) {
            // ...
        }
    }
}
```

```

        } catch (ClientException ce) {
            System.out.println("Caught an ClientException, which means the client encounter
ed "
                                + "a serious internal problem while trying to communicate with OSS, "
                                + "such as not being able to access the network.");
            System.out.println("Error Message:" + ce.getMessage());
        } finally {
            if (ossClient != null) {
                ossClient.shutdown();
            }
        }
    }
}

```

References

- For more information about the API operation that you can call to create a symbolic link, see [PutSymlink](#).
- For more information about the API operation that you can call to query a symbolic link, see [GetSymlink](#).

2.11. Object tagging

2.11.1. Configure object tagging

Object Storage Service (OSS) allows you to configure object tags to classify objects. Tags allow you to configure lifecycle rules and access control lists (ACLs) for objects that have the same tag.

Background information

When you configure object tagging, take note of the following items:

- You can add tags to an object when and after you upload the object. If the object already has tags, the existing tags are overwritten. For more information about object tags, see [PutObjectTagging](#).
- To add tags to an object, you must have permissions to call the PutObjectTagging operation.

You can create a custom policy by using scripts to grant the permissions to RAM users. For more information, see [Attach a custom policy to a RAM user](#).

- The Last-Modified value of an object is not updated when the tags are changed.
- You can add up to 10 tags to an object. The tags added to an object must have unique keys.
- A tag key can be up to 128 characters in length. A tag value can be up to 256 characters in length.
- Tag keys and tag values are case-sensitive.
- The key and the value of a tag can contain letters, digits, spaces, and the following special characters:

+ - = . _ : /

 **Note** If you configure the tags in the HTTP header and the tags contain special characters, you can encode the keys and the values of the tags by using URL encoding.

Object tagging uses a key-value pair to identify objects. For more information about object tags, see [Object tagging](#).

Add tags to an object when you upload the object

The following examples describe how to add tags to objects in simple upload, multipart upload, append upload, and resumable upload:

- Add tags to an object when you upload it by using simple upload

The following code provides an example on how to add tags to an object when you upload it by using simple upload:

```
import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.*;
import java.io.ByteArrayInputStream;
import java.util.HashMap;
import java.util.Map;
public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify
        the actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API oper
        ations. Using these credentials to perform operations in OSS is a high-risk operation. We
        recommend that you use a RAM user to call API operations or perform routine O&M. To creat
        e a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the name of the bucket. Example: examplebucket.
        String bucketName = "examplebucket";
        // Specify the full path of the object, not including the bucket name. Example: e
        xamplendir/exampleobject.txt.
        String objectName = "exampledir/exampleobject.txt";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecr
        et);
        try {
            Map<String, String> tags = new HashMap<String, String>();
            // Specify the key and the value of the object tag. For example, set the key
            to owner and the value to John.
            tags.put("owner", "John");
            tags.put("type", "document");
            // Configure the tags in the HTTP header.
            ObjectMetadata metadata = new ObjectMetadata();
            metadata.setObjectTagging(tags);
            // Upload the object and add tags to it.
            String content = "<yourContent>";
            ossClient.putObject(bucketName, objectName, new ByteArrayInputStream(content.
            getBytes()), metadata);
        } catch (OSSException oe) {
            System.out.println("Caught an OSSException, which means your request made it
            to OSS "
```

```

    to OSS,
        + "but was rejected with an error response for some reason.");
    System.out.println("Error Message:" + oe.getErrorMessage());
    System.out.println("Error Code:" + oe.getErrorCode());
    System.out.println("Request ID:" + oe.getRequestId());
    System.out.println("Host ID:" + oe.getHostId());
    } catch (ClientException ce) {
        System.out.println("Caught an ClientException, which means the client encount
ered "
            + "a serious internal problem while trying to communicate with OSS, "
            + "such as not being able to access the network.");
        System.out.println("Error Message:" + ce.getMessage());
    } finally {
        if (ossClient != null) {
            ossClient.shutdown();
        }
    }
}
}
}

```

- Add tags to an object when you upload it by using multipart upload

The following code provides an example on how to add tags to an object when you upload it by using multipart upload:

```

import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.*;
import java.io.*;
import java.util.*;
public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify
the actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API oper
ations. Using these credentials to perform operations in OSS is a high-risk operation. We
recommend that you use a RAM user to call API operations or perform routine O&M. To creat
e a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the name of the bucket. Example: examplebucket.
        String bucketName = "examplebucket";
        // Specify the full path of the object, not including the bucket name. Example: e
xamplendir/exampleobject.txt.
        String objectName = "exampledir/exampleobject.txt";
        // Specify the full path of the local file that you want to upload. If you do not
specify the full path of the local file, the local file is uploaded from the path of the
project to which the sample program belongs.
        String localFile = "D:\\localpath\\examplefile.txt";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecr
et);
    }
}

```

```

try {
    /*
     * Step 1: Initialize a multipart upload task.
     */
    // Configure the tags in the HTTP header.
    Map<String, String> tags = new HashMap<String, String>();
    // Specify the key and the value of the object tag. For example, set the key
    to owner and the value to John.
    tags.put("owner", "John");
    tags.put("type", "document");
    ObjectMetadata metadata = new ObjectMetadata();
    metadata.setObjectTagging(tags);
    // Send an InitiateMultipartUploadRequest request and add tags to the object.

    InitiateMultipartUploadRequest request = new InitiateMultipartUploadRequest(b
ucketName, objectName, metadata);
    InitiateMultipartUploadResult result = ossClient.initiateMultipartUpload(requ
est);

    // Obtain the upload ID, which uniquely identifies the multipart upload task.
    The upload ID can be used to query or cancel a multipart upload task.
    String uploadId = result.getUploadId();
    /*
     * Step 2: Upload parts.
     */
    // partETags is the set of PartETags. A PartETag consists of an ETag and a pa
rt number.

    List<PartETag> partETags = new ArrayList<PartETag>();
    // Calculate the total number of parts to upload.
    final long partSize = 1 * 1024 * 1024L; // Set the part size to 1 MB.
    final File sampleFile = new File(localFile);
    long fileLength = sampleFile.length();
    int partCount = (int) (fileLength / partSize);
    if (fileLength % partSize != 0) {
        partCount++;
    }
    // Upload each part until all parts are uploaded.
    for (int i = 0; i < partCount; i++) {
        long startPos = i * partSize;
        long curPartSize = (i + 1 == partCount) ? (fileLength - startPos) : partS
ize;

        InputStream instream = null;
        try {
            instream = new FileInputStream(sampleFile);
            // Skip the parts that have been uploaded.
            instream.skip(startPos);
        } catch (FileNotFoundException e) {
            e.printStackTrace();
        } catch (IOException e) {
            e.printStackTrace();
        }
        UploadPartRequest uploadPartRequest = new UploadPartRequest();
        uploadPartRequest.setBucketName(bucketName);
        uploadPartRequest.setKey(objectName);
        uploadPartRequest.setUploadId(uploadId);
        uploadPartRequest.setInputStream(instream);
    }
}

```

```

        // Set the size available for each part. Each part except the last part must be equal to or greater than 100 KB.
        uploadPartRequest.setPartSize(curPartSize);
        // Set part numbers. Each part is configured with a part number. The number ranges from 1 to 10000. If you configure a part number that exceeds this range, OSS returns an InvalidArgument error code.
        uploadPartRequest.setPartNumber(i + 1);
        // Parts are not necessarily uploaded in order. They can be uploaded from different OSS clients. OSS sorts the parts based on their part numbers and combines them into a complete object.
        UploadPartResult uploadPartResult = ossClient.uploadPart(uploadPartRequest);

        // After you upload a part, OSS returns a result that contains a PartETag. The PartETag is stored in partETags.
        partETags.add(uploadPartResult.getPartETag());
    }
    /* Step 3: Complete multipart upload.
    */
    // PartETags stored in partETags are sorted in ascending order by part numbers.
    Collections.sort(partETags, new Comparator<PartETag>() {
        public int compare(PartETag p1, PartETag p2) {
            return p1.getPartNumber() - p2.getPartNumber();
        }
    });
    // You must provide all valid PartETags when you perform this operation. OSS verifies the validity of all parts one by one after it receives PartETags. After all parts are verified, OSS combines these parts into a complete object.
    CompleteMultipartUploadRequest completeMultipartUploadRequest =
        new CompleteMultipartUploadRequest(bucketName, objectName, uploadId, partETags);
    ossClient.completeMultipartUpload(completeMultipartUploadRequest);
    // View the tags added to the object.
    TagSet tagSet = ossClient.getObjectTagging(bucketName, objectName);
    Map<String, String> getTags = tagSet.getAllTags();
    System.out.println("object tagging: " + getTags.toString());
} catch (OSSException oe) {
    System.out.println("Caught an OSSException, which means your request made it to OSS, "
        + "but was rejected with an error response for some reason.");
    System.out.println("Error Message:" + oe.getErrorMessage());
    System.out.println("Error Code:" + oe.getErrorCode());
    System.out.println("Request ID:" + oe.getRequestId());
    System.out.println("Host ID:" + oe.getHostId());
} catch (ClientException ce) {
    System.out.println("Caught an ClientException, which means the client encountered "
        + "a serious internal problem while trying to communicate with OSS, "
        + "such as not being able to access the network.");
    System.out.println("Error Message:" + ce.getMessage());
} finally {
    if (ossClient != null) {
        ossClient.shutdown();
    }
}

```

```

    }
}
}

```

- Add tags to an object when you upload it by using append upload

The following code provides an example on how to add tags to an object when you upload it by using append upload:

```

import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.*;
import java.io.*;
import java.util.*;

public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify
        // the actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API oper-
        // ations. Using these credentials to perform operations in OSS is a high-risk operation. We
        // recommend that you use a RAM user to call API operations or perform routine O&M. To creat
        // e a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the name of the bucket. Example: examplebucket.
        String bucketName = "examplebucket";
        // Specify the full path of the object, not including the bucket name. Example: e
        // xamplendir/exampleobject.txt.
        String objectName = "exampledir/exampleobject.txt";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecr
        et);

        try {
            String content1 = "Hello OSS A \n";
            String content2 = "Hello OSS B \n";
            String content3 = "Hello OSS C \n";
            Map<String, String> tags = new HashMap<String, String>();
            // Specify the key and the value of the object tag. For example, set the key
            // to owner and the value to John.
            tags.put("owner", "John");
            tags.put("type", "document");
            ObjectMetadata meta = new ObjectMetadata();
            // Configure tags for the object that you want to upload.
            meta.setObjectTagging(tags);
            // Specify the type of content that you want to upload.
            meta.setContentType("text/plain");
            // Configure parameters by using AppendObjectRequest.
            AppendObjectRequest appendObjectRequest = new AppendObjectRequest(bucketName,
            objectName, new ByteArrayInputStream(content1.getBytes()), meta);
            // Configure a single parameter by using AppendObjectRequest.
            //appendObjectRequest.setBucketName(bucketName);
            //appendObjectRequest.setKey(objectName);

```



```

        // Specify the content that you want to append. Two types of content are available: InputStream and File. The following code provides an example on how to specify the content of the InputStream type:
        //appendObjectRequest.setInputStream(new ByteArrayInputStream(content1.getBytes()));

        // Specify the content that you want to append. Two types of content are available: InputStream and File. The following code provides an example on how to specify the content of the File type:
        // Specify the full path of the local file that you want to upload. If you do not specify the full path of the local file, the local file is uploaded from the path of the project to which the sample program belongs.
        //appendObjectRequest.setFile(new File("D:\\localpath\\examplefile.txt"));
        // Specify the object metadata. You can specify the metadata of an object only when you perform the append operation on the object for the first time.
        //appendObjectRequest.setMetadata(meta);
        // Start the first append. Only the tags configured when the object is appended for the first time are added to the object.
        // Configure the position from which the append operation starts.
        appendObjectRequest.setPosition(0L);
        AppendObjectResult appendObjectResult = ossClient.appendObject(appendObjectRequest);

        // Calculate the CRC-64 of the object. The value is calculated based on the ECMA-182 specification.
        System.out.println(appendObjectResult.getObjectCRC());
        // Start the second append.
        // NextPosition specifies the position from which the next append operation starts, which is the current length of the object.
        appendObjectRequest.setPosition(appendObjectResult.getNextPosition());
        appendObjectRequest.setInputStream(new ByteArrayInputStream(content2.getBytes()));

        appendObjectResult = ossClient.appendObject(appendObjectRequest);
        // Start the third append.
        appendObjectRequest.setPosition(appendObjectResult.getNextPosition());
        appendObjectRequest.setInputStream(new ByteArrayInputStream(content3.getBytes()));

        appendObjectResult = ossClient.appendObject(appendObjectRequest);
        // View the tags added to the uploaded object.
        TagSet tagSet = ossClient.getObjectTagging(bucketName, objectName);
        Map<String, String> getTags = tagSet.getAllTags();
        System.out.println("object tagging: " + getTags.toString());
    } catch (OSSException oe) {
        System.out.println("Caught an OSSException, which means your request made it to OSS, "
            + "but was rejected with an error response for some reason.");
        System.out.println("Error Message:" + oe.getErrorMessage());
        System.out.println("Error Code:" + oe.getErrorCode());
        System.out.println("Request ID:" + oe.getRequestId());
        System.out.println("Host ID:" + oe.getHostId());
    } catch (ClientException ce) {
        System.out.println("Caught an ClientException, which means the client encountered "
            + "a serious internal problem while trying to communicate with OSS, "
            + "such as not being able to access the network.");
        System.out.println("Error Message:" + ce.getMessage());
    }
}

```

```
        } finally {  
            if (ossClient != null) {  
                ossClient.shutdown();  
            }  
        }  
    }  
}
```

- Add tags to an object when you upload it by using resumable upload

The following code provides an example on how to add tags to an object when you upload it by using resumable upload:

```
import com.aliyun.oss.ClientException;  
import com.aliyun.oss.OSS;  
import com.aliyun.oss.OSSClientBuilder;  
import com.aliyun.oss.OSSException;  
import com.aliyun.oss.model.*;  
import java.util.*;  
public class Demo {  
    public static void main(String[] args) throws Throwable {  
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify  
        // the actual endpoint.  
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";  
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API oper-  
        // ations. Using these credentials to perform operations in OSS is a high-risk operation. We  
        // recommend that you use a RAM user to call API operations or perform routine O&M. To creat-  
        // e a RAM user, log on to the RAM console.  
        String accessKeyId = "yourAccessKeyId";  
        String accessKeySecret = "yourAccessKeySecret";  
        // Specify the name of the bucket. Example: examplebucket.  
        String bucketName = "examplebucket";  
        // Specify the full path of the object, not including the bucket name. Example: e-  
        // xamplendir/exampleobject.txt.  
        String objectName = "exampledir/exampleobject.txt";  
        // Specify the full path of the local file that you want to upload. If you do not  
        // specify the full path of the local file, the local file is uploaded from the path of the  
        // project to which the sample program belongs.  
        String localFile = "D:\\\\localpath\\\\examplefile.txt";  
        // Specify the file that records the upload result of each part. Example: D:\\\\loca-  
        // lpath\\\\uploadFile.ucp.  
        String yourCheckpointFile = "D:\\\\localpath\\\\uploadFile.ucp";  
        // Create an OSSClient instance.  
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySec-  
        et);  
        try {  
            // Configure the tags to be added to the object.  
            Map<String, String> tags = new HashMap<String, String>();  
            // Specify the key and the value of the object tag. For example, set the key  
            // to owner and the value to John.  
            tags.put("owner", "John");  
            tags.put("type", "document");  
            ObjectMetadata meta = new ObjectMetadata();  
            // Specify the type of content that you want to upload.  
            meta.setContentType("text/plain");
```

```

        // Add the tags to the object.
        meta.setObjectTagging(tags);
        // Configure parameters by using UploadFileRequest.
        UploadFileRequest uploadFileRequest = new UploadFileRequest(bucketName,object
Name);

        // Configure a single parameter by using UploadFileRequest.
        // Specify the bucket name.
        //uploadFileRequest.setBucketName(bucketName);
        // Specify the full path of the object, not including the bucket name.
        //uploadFileRequest.setKey(objectName);
        // Specify the local file that you want to upload.
        uploadFileRequest.setUploadFile(localFile);
        // Specify the number of threads for the resumable upload task. Default value
: 1.

        uploadFileRequest.setTaskNum(5);
        // Specify the size of each part to upload. Valid values: 100 KB to 5 GB. The
default size of each part is calculated based on the following formula: Default size of e
ach part = Object size/10,000.
        uploadFileRequest.setPartSize(1 * 1024 * 1024);
        // Enable resumable upload. By default, resumable upload is disabled.
        uploadFileRequest.setEnableCheckpoint(true);
        // Specify the file that records the upload result of each part. You need to
configure this parameter when you enable resumable upload. This file stores progress info
rmation. If a part upload fails, it can be continued based on the progress information re
corded in this file. After the local file is uploaded, the file that stores progress info
rmation is deleted. By default, this file shares the same directory (uploadFile.ucp) as t
he file you want to upload.
        uploadFileRequest.setCheckpointFile(yourCheckpointFile);
        // Configure object metadata.
        uploadFileRequest.setObjectMetadata(meta);
        // Configure callback after successful uploads. The parameter type is Callbac
k.

        //uploadFileRequest.setCallback("yourCallbackEvent");
        // Start the resumable upload task and configure the tags to be added to the
object.

        ossClient.uploadFile(uploadFileRequest);
        // View the tags added to the object.
        TagSet tagSet = ossClient.getObjectTagging(bucketName, objectName);
        Map<String, String> getTags = tagSet.getAllTags();
        System.out.println("object tagging: " + getTags.toString());
    } catch (OSSEException oe) {
        System.out.println("Caught an OSSEException, which means your request made it
to OSS, "

            + "but was rejected with an error response for some reason.");
        System.out.println("Error Message:" + oe.getErrorMessage());
        System.out.println("Error Code:" + oe.getErrorCode());
        System.out.println("Request ID:" + oe.getRequestId());
        System.out.println("Host ID:" + oe.getHostId());
    } catch (ClientException ce) {
        System.out.println("Caught an ClientException, which means the client encount
ered "

            + "a serious internal problem while trying to communicate with OSS, "
            + "such as not being able to access the network.");
        System.out.println("Error Message:" + ce.getMessage());
    }
}

```

```
        } finally {  
            if (ossClient != null) {  
                ossClient.shutdown();  
            }  
        }  
    }  
}
```

Add tags to or modify the tags of an existing object

If an existing object has no tags or the added tags of the object do not meet your requirements, you can add tags to or modify the tags of the existing object.

The following code provides an example on how to add tags to or modify the tags of an existing object:

```

import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import java.util.*;
public class Demo {
    public static void main(String[] args) throws Throwable {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify the actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the name of the bucket. Example: examplebucket.
        String bucketName = "examplebucket";
        // Specify the full path of the object, not including the bucket name. Example: exampledir/exampleobject.txt.
        String objectName = "exampledir/exampleobject.txt";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);

        try {
            Map<String, String> tags = new HashMap<String, String>();
            // Specify the key and the value of the object tag. For example, set the key to owner and the value to John.
            tags.put("owner", "John");
            tags.put("type", "document");
            // Add tags to the object.
            ossClient.setObjectTagging(bucketName, objectName, tags);
        } catch (OSSException oe) {
            System.out.println("Caught an OSSException, which means your request made it to OSS, "
                + "but was rejected with an error response for some reason.");
            System.out.println("Error Message:" + oe.getErrorMessage());
            System.out.println("Error Code:" + oe.getErrorCode());
            System.out.println("Request ID:" + oe.getRequestId());
            System.out.println("Host ID:" + oe.getHostId());
        } catch (ClientException ce) {
            System.out.println("Caught an ClientException, which means the client encountered "
                + "a serious internal problem while trying to communicate with OSS, "
                + "such as not being able to access the network.");
            System.out.println("Error Message:" + ce.getMessage());
        } finally {
            if (ossClient != null) {
                ossClient.shutdown();
            }
        }
    }
}

```

Add tags to or modify the tags of a specified version of an object

If versioning is enabled for the bucket, you can add tags to or modify the tags of a specified version of an object by specifying the version ID of the object.

The following code provides an example on how to add tags to or modify the tags of a specified version of an object:

 **Note** For more information about how to obtain the version IDs of an object, see [List objects](#).

```
import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.SetObjectTaggingRequest;
import java.util.*;

public class Demo {
    public static void main(String[] args) throws Throwable {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify the actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the name of the bucket. Example: examplebucket.
        String bucketName = "examplebucket";
        // Specify the full path of the object, not including the bucket name. Example: exampledir/exampleobject.txt.
        String objectName = "exampledir/exampleobject.txt";
        // Specify the version ID of the object. Example: CAEQMxiBgICAof2D0BYiIDJhMGE3N2M1YTI1NDQzOGY5NTkyNTI3MGYyMzJm****.
        String versionId = "CAEQMxiBgICAof2D0BYiIDJhMGE3N2M1YTI1NDQzOGY5NTkyNTI3MGYyMzJm****";

        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);

        try {
            Map<String, String> tags = new HashMap<String, String>(1);
            // Specify the key and the value of the object tag. For example, set the key to owner and the value to John.
            tags.put("owner", "John");
            tags.put("type", "document");
            SetObjectTaggingRequest setObjectTaggingRequest = new SetObjectTaggingRequest(bucketName, objectName, tags);
            setObjectTaggingRequest.setVersionId(versionId);
            ossClient.setObjectTagging(setObjectTaggingRequest);
        } catch (OSSException oe) {
            System.out.println("Caught an OSSException, which means your request made it to OSS, "
                + "but was rejected with an error response for some reason.");
            System.out.println("Error Message:" + oe.getErrorMessage());
        }
    }
}
```

```

        System.out.println("Error Code:" + oe.getErrorCode());
        System.out.println("Request ID:" + oe.getRequestId());
        System.out.println("Host ID:" + oe.getHostId());
    } catch (ClientException ce) {
        System.out.println("Caught an ClientException, which means the client encounter
ed "
            + "a serious internal problem while trying to communicate with OSS, "
            + "such as not being able to access the network.");
        System.out.println("Error Message:" + ce.getMessage());
    } finally {
        if (ossClient != null) {
            ossClient.shutdown();
        }
    }
}
}

```

Add tags to an object when you copy the object

The following examples describe how to add tags to an object smaller than 1 GB in simple copy mode and larger than 1 GB in multipart copy mode:

- Add tags to an object when you copy the object in simply copy mode

The following code provides an example on how to add tags to an object smaller than 1 GB when you copy it in simply copy mode:

```

import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.internal.OSSHeaders;
import com.aliyun.oss.model.CopyObjectRequest;
import com.aliyun.oss.model.CopyObjectResult;
import com.aliyun.oss.model.TagSet;
import java.util.*;
public class Demo {
    public static void main(String[] args) throws Throwable {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify
the actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API oper
ations. Using these credentials to perform operations in OSS is a high-risk operation. We
recommend that you use a RAM user to call API operations or perform routine O&M. To creat
e a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the name of the source bucket. Example: srcexamplebucket.
        String sourceBucketName = "srcexamplebucket";
        // Specify the full path of the object in the source bucket, not including the bu
cket name. Example: srcexampledir/exampleobject.txt.
        String sourceObjectName = "srcexampledir/exampleobject.txt";
        // Specify the name of the destination bucket. Example: destexamplebucket.
        String destinationBucketName = "destexamplebucket";
        // Specify the full path of the object in the destination bucket, not including t

```

```

// Specify the full path of the object in the destination bucket, not including t
he bucket name. Example: destexampledir/exampleobject.txt.
String destinationObjectName = "destexampledir/exampleobject.txt";
// Create an OSSClient instance.
OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecr
et);

try {
    // Create a CopyObjectRequest object.
    CopyObjectRequest copyObjectRequest = new CopyObjectRequest(sourceBucketName,
sourceObjectName, destinationBucketName, destinationObjectName);
    // Configure the tags to be added to the destination object. If the headers a
re not configured, the tags of the source object are added to the destination object by d
efault.

    Map<String, String> headers = new HashMap<String, String>();
    headers.put(OSSHeaders.COPY_OBJECT_TAGGING_DIRECTIVE, "REPLACE");
    headers.put(OSSHeaders.OSS_TAGGING, "key1=value1&key2=value2");
    copyObjectRequest.setHeaders(headers);
    // Copy the object.
    CopyObjectResult result = ossClient.copyObject(copyObjectRequest);
    System.out.println("ETag: " + result.getETag() + " LastModified: " + result.g
etLastModified());
    // View the tags added to the destination object.
    TagSet tagSet = ossClient.getObjectTagging(destinationBucketName, destination
ObjectName);
    Map<String, String> getTags = tagSet.getAllTags();
    System.out.println("dest object tagging: "+ getTags.toString());
} catch (OSSEException oe) {
    System.out.println("Caught an OSSEException, which means your request made it
to OSS, "
        + "but was rejected with an error response for some reason.");
    System.out.println("Error Message:" + oe.getErrorMessage());
    System.out.println("Error Code:" + oe.getErrorCode());
    System.out.println("Request ID:" + oe.getRequestId());
    System.out.println("Host ID:" + oe.getHostId());
} catch (ClientException ce) {
    System.out.println("Caught an ClientException, which means the client encount
ered "
        + "a serious internal problem while trying to communicate with OSS, "
        + "such as not being able to access the network.");
    System.out.println("Error Message:" + ce.getMessage());
} finally {
    if (ossClient != null) {
        ossClient.shutdown();
    }
}
}

```

- Add tags to an object when you copy the object in multipart copy mode

The following code provides an example on how to add tags to an object larger than 1 GB when you copy it in multipart copy mode:

```

import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;

```



```

import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.*;
import java.util.*;

public class Demo {
    public static void main(String[] args) throws Throwable {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify
        the actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API oper
        ations. Using these credentials to perform operations in OSS is a high-risk operation. We
        recommend that you use a RAM user to call API operations or perform routine O&M. To creat
        e a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the name of the source bucket. Example: srcexamplebucket.
        String sourceBucketName = "srcexamplebucket";
        // Specify the full path of the object in the source bucket, not including the bu
        cket name. Example: srcexampledir/exampleobject.txt.
        String sourceObjectName = "srcexampledir/exampleobject.txt";
        // Specify the name of the destination bucket. Example: destexamplebucket.
        String destinationBucketName = "destexamplebucket";
        // Specify the full path of the object in the destination bucket, not including t
        he bucket name. Example: destexampledir/exampleobject.txt.
        String destinationObjectName = "destexampledir/exampleobject.txt";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecr
        et);

        try {
            ObjectMetadata objectMetadata = ossClient.getObjectMetadata(sourceBucketName,
            sourceObjectName);
            // Query the size of the object to copy.
            long contentLength = objectMetadata.getContentLength();
            // Set the size of each part to 10 MB.
            long partSize = 1024 * 1024 * 10;
            // Calculate the total number of parts.
            int partCount = (int) (contentLength / partSize);
            if (contentLength % partSize != 0) {
                partCount++;
            }
            System.out.println("total part count:" + partCount);
            // Configure the tags in the HTTP header.
            Map<String, String> tags2 = new HashMap<String, String>();
            // Specify the key and value of the object tag. For example, set the key to o
            wner and the value to Lily.
            tags2.put("owner", "Lily");
            tags2.put("type", "document");
            ObjectMetadata metadata = new ObjectMetadata();
            metadata.setObjectTagging(tags2);
            // Initiate a multipart copy task and configure the tags to be added to the d
            estination object.
            InitiateMultipartUploadRequest initiateMultipartUploadRequest = new InitiateM
            ultipartUploadRequest(destinationBucketName, destinationObjectName, metadata);
            InitiateMultipartUploadResult initiateMultipartUploadResult = ossClient.initi

```

```

    initiateMultipartUpload(initiateMultipartUploadRequest);
    String uploadId = initiateMultipartUploadResult.getUploadId();
    // Start the multipart copy task.
    List<PartETag> partETags = new ArrayList<PartETag>();
    for (int i = 0; i < partCount; i++) {
        // Calculate the size of each part.
        long skipBytes = partSize * i;
        long size = partSize < contentLength - skipBytes ? partSize : contentLength - skipBytes;
        // Create an UploadPartCopyRequest object. You can call the UploadPartCopyRequest operation to specify conditions.
        UploadPartCopyRequest uploadPartCopyRequest =
            new UploadPartCopyRequest(sourceBucketName, sourceObjectName, destinationBucketName, destinationObjectName);
        uploadPartCopyRequest.setUploadId(uploadId);
        uploadPartCopyRequest.setPartSize(size);
        uploadPartCopyRequest.setBeginIndex(skipBytes);
        uploadPartCopyRequest.setPartNumber(i + 1);
        UploadPartCopyResult uploadPartCopyResult = ossClient.uploadPartCopy(uploadPartCopyRequest);
        // Store the returned PartETags in partETags.
        partETags.add(uploadPartCopyResult.getPartETag());
    }
    // Complete the multipart copy task.
    CompleteMultipartUploadRequest completeMultipartUploadRequest = new CompleteMultipartUploadRequest(
        destinationBucketName, destinationObjectName, uploadId, partETags);
    CompleteMultipartUploadResult completeMultipartUploadResult = ossClient.completeMultipartUpload(completeMultipartUploadRequest);
    System.out.println("versionId: " + completeMultipartUploadResult.getVersionId());
};

// View the tags of the source object.
TagSet tagSet = ossClient.getObjectTagging(sourceBucketName, sourceObjectName);

Map<String, String> getTags = tagSet.getAllTags();
System.out.println("src object tagging: " + getTags.toString());
// View the tags added to the destination object.
tagSet = ossClient.getObjectTagging(destinationBucketName, destinationObjectName);

getTags = tagSet.getAllTags();
System.out.println("dest object tagging: " + getTags.toString());
} catch (OSSEException oe) {
    System.out.println("Caught an OSSEException, which means your request made it to OSS, "
        + "but was rejected with an error response for some reason.");
    System.out.println("Error Message:" + oe.getErrorMessage());
    System.out.println("Error Code:" + oe.getErrorCode());
    System.out.println("Request ID:" + oe.getRequestId());
    System.out.println("Host ID:" + oe.getHostId());
} catch (ClientException ce) {
    System.out.println("Caught an ClientException, which means the client encountered "
        + "a serious internal problem while trying to communicate with OSS, "
        + "such as not being able to access the network.");
}

```

```

        System.out.println("Error Message:" + ce.getMessage());
    } finally {
        if (ossClient != null) {
            ossClient.shutdown();
        }
    }
}
}
}

```

Add tags to a symbolic link

The following code provides an example on how to add tags to a symbolic link:

```

import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.*;
import java.util.*;

public class Demo {
    public static void main(String[] args) throws Throwable {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify the actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the name of the bucket. Example: examplebucket.
        String bucketName = "examplebucket";
        // Specify the full path of the symbolic link. Example: shortcut/myobject.txt.
        String symLink = "shortcut/myobject.txt";
        // Specify the full path of the object, not including the bucket name. Example: exampledir/exampleobject.txt.
        String destinationObjectName = "exampledir/exampleobject.txt";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);

        try {
            // Configure the tags to be added to the symbolic link.
            Map<String, String> tags = new HashMap<String, String>();
            // Specify the key and the value of the object tag. For example, set the key to owner and the value to John.
            tags.put("owner", "John");
            tags.put("type", "document");
            // Create metadata for the object to upload.
            ObjectMetadata metadata = new ObjectMetadata();
            metadata.setObjectTagging(tags);
            // Create a request to create the symbolic link.
            CreateSymlinkRequest createSymlinkRequest = new CreateSymlinkRequest(bucketName, symLink, destinationObjectName);
            // Configure the metadata.

```

```

        createSymlinkRequest.setMetadata(metadata);
        // Create the symbolic link.
        ossClient.createSymlink(createSymlinkRequest);
        // View the tags added to the symbolic link.
        TagSet tagSet = ossClient.getObjectTagging(bucketName, symlink);
        Map<String, String> getTags = tagSet.getAllTags();
        System.out.println("symlink tagging: " + getTags.toString());
    } catch (OSSException oe) {
        System.out.println("Caught an OSSException, which means your request made it to
OSS, "
            + "but was rejected with an error response for some reason.");
        System.out.println("Error Message:" + oe.getErrorMessage());
        System.out.println("Error Code:" + oe.getErrorCode());
        System.out.println("Request ID:" + oe.getRequestId());
        System.out.println("Host ID:" + oe.getHostId());
    } catch (ClientException ce) {
        System.out.println("Caught an ClientException, which means the client encounter
ed "
            + "a serious internal problem while trying to communicate with OSS, "
            + "such as not being able to access the network.");
        System.out.println("Error Message:" + ce.getMessage());
    } finally {
        if (ossClient != null) {
            ossClient.shutdown();
        }
    }
}
}
}

```

References

- For more information about the complete sample code that is used to configure object tagging, visit [GitHub](#).
- For more information about the API operation that you can call to configure object tagging, see [Put Object Tagging](#).

2.11.2. Query the tags of an object

After you configure tags for an object, you can query the tags of the object. If versioning is enabled for a bucket that stores the object whose tags you want to query, Object Storage Service (OSS) returns the tags of the current version of the object by default. To query the tags of a specified version of the object, you must specify the version ID of the object.

Query the tags of an object

If versioning is not enabled for the bucket that stores the object whose tags you want to query, you can query the tags of the object based on your requirements. If versioning is enabled for the bucket that stores the object whose tags you want to query, OSS returns the tags of the current version of the object by default.

The following code provides an example on how to query the tags of the `exampleobject.txt` object in the `exampledir` directory of the `examplebucket` bucket:

```

import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.*;
import java.util.*;

public class Demo {
    public static void main(String[] args) throws Throwable {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify the actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to access OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the name of the bucket. Example: examplebucket.
        String bucketName = "examplebucket";
        // Specify the full path of the object. Example: exampledir/exampleobject.txt. The full path of the object cannot contain the bucket name.
        String objectName = "exampledir/exampleobject.txt";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);

        try {
            // Query the tags of the object.
            TagSet tagSet = ossClient.getObjectTagging(bucketName, objectName);
            Map<String, String> tags = tagSet.getAllTags();
            // View the tags of the object.
            System.out.println("object tagging: " + tags.toString());
        } catch (OSSException oe) {
            System.out.println("Caught an OSSException, which means your request made it to OSS, "
                + "but was rejected with an error response for some reason.");
            System.out.println("Error Message:" + oe.getErrorMessage());
            System.out.println("Error Code:" + oe.getErrorCode());
            System.out.println("Request ID:" + oe.getRequestId());
            System.out.println("Host ID:" + oe.getHostId());
        } catch (ClientException ce) {
            System.out.println("Caught an ClientException, which means the client encountered "
                + "a serious internal problem while trying to communicate with OSS, "
                + "such as not being able to access the network.");
            System.out.println("Error Message:" + ce.getMessage());
        } finally {
            if (ossClient != null) {
                ossClient.shutdown();
            }
        }
    }
}

```

Query the tags of a specified version of an object

If versioning is enabled for the bucket that stores the object whose tags you want to query, you can query the tags of a specified version of the object by specifying the version ID of the object.

The following code provides an example on how to query the tags of a specified version of the `exampleobject.txt` object in the `exampledir` directory of the `examplebucket` bucket:

 **Note** For more information about how to query the version IDs of an object, see [List objects](#).

```
import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.*;

public class Demo {
    public static void main(String[] args) throws Throwable {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify the actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to access OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the name of the bucket. Example: examplebucket.
        String bucketName = "examplebucket";
        // Specify the full path of the object. Example: exampledir/exampleobject.txt. The full path of the object cannot contain the bucket name.
        String objectName = "exampledir/exampleobject.txt";
        // Specify the version ID of the object. Example: CAEQMxiBgICAof2D0BYiIDJhMGE3N2M1YTI1NDQzOGY5NTkyNTI3MGYyMzJm****.
        String versionId = "CAEQMxiBgICAof2D0BYiIDJhMGE3N2M1YTI1NDQzOGY5NTkyNTI3MGYyMzJm****";

        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);

        try {
            GenericRequest genericRequest = new GenericRequest(bucketName, objectName, versionId);
            TagSet tagSet = ossClient.getObjectTagging(genericRequest);
            // View the tags of the object.
            System.out.println("object tagging: " + tagSet.toString());
        } catch (OSSException oe) {
            System.out.println("Caught an OSSException, which means your request made it to OSS, "
                + "but was rejected with an error response for some reason.");
            System.out.println("Error Message:" + oe.getErrorMessage());
            System.out.println("Error Code:" + oe.getErrorCode());
            System.out.println("Request ID:" + oe.getRequestId());
            System.out.println("Host ID:" + oe.getHostId());
        } catch (ClientException ce) {
            System.out.println("Caught an ClientException, which means the client encounter
```

```
ed "
        + "a serious internal problem while trying to communicate with OSS, "
        + "such as not being able to access the network.");
    System.out.println("Error Message:" + ce.getMessage());
} finally {
    if (ossClient != null) {
        ossClient.shutdown();
    }
}
}
```

References

- For more information about the complete sample code that is used to query the tags of an object, visit [Git Hub](#).
- For more information about the API operation that you can call to query the tags of an object, see [Get Object Tagging](#).

2.11.3. Delete the tags of an object

You can delete unnecessary tags of an object based on your requirements. If versioning is enabled for a bucket that stores the object whose tags you want to delete, Object Storage Service (OSS) deletes the tags of the current version of the object by default. To delete the tags of a specified version of the object, you must specify the version ID of the object.

Delete the tags of an object

If versioning is not enabled for the bucket that stores the object whose tags you want to delete, you can delete the tags of the object based on your requirements. If versioning is enabled for the bucket that stores the object whose tags you want to delete, OSS deletes the tags of the current version of the object by default.

The following code provides an example on how to delete the tags of the `exampleobject.txt` object in the `exampledir` directory of the `examplebucket` bucket:

```

import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
public class Demo {
    public static void main(String[] args) throws Throwable {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify the
        // actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to access OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the name of the bucket. Example: examplebucket.
        String bucketName = "examplebucket";
        // Specify the full path of the object. Example: exampledir/exampleobject.txt. The full path of the object cannot contain the bucket name.
        String objectName = "exampledir/exampleobject.txt";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);
        try {
            // Delete the tags of the object.
            ossClient.deleteObjectTagging(bucketName, objectName);
        } catch (OSSException oe) {
            System.out.println("Caught an OSSException, which means your request made it to OSS, "
                + "but was rejected with an error response for some reason.");
            System.out.println("Error Message:" + oe.getErrorMessage());
            System.out.println("Error Code:" + oe.getErrorCode());
            System.out.println("Request ID:" + oe.getRequestId());
            System.out.println("Host ID:" + oe.getHostId());
        } catch (ClientException ce) {
            System.out.println("Caught an ClientException, which means the client encountered "
                + "a serious internal problem while trying to communicate with OSS, "
                + "such as not being able to access the network.");
            System.out.println("Error Message:" + ce.getMessage());
        } finally {
            if (ossClient != null) {
                ossClient.shutdown();
            }
        }
    }
}

```

Delete the tags of a specified version of an object

If versioning is enabled for the bucket that stores the object whose tags you want to delete, you can delete the tags of a specified version of the object by specifying the version ID of the object.

The following code provides an example on how to delete the tags of a specified version of the `exampleobject.txt` object in the `exampledir` directory of the `examplebucket` bucket:

```

import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.GenericRequest;
public class Demo {
    public static void main(String[] args) throws Throwable {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify the actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to access OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the name of the bucket. Example: examplebucket.
        String bucketName = "examplebucket";
        // Specify the full path of the object. Example: exampledir/exampleobject.txt. The full path of the object cannot contain the bucket name.
        String objectName = "exampledir/exampleobject.txt";
        // Specify the version ID of the object. Example: CAEQMxiBgICAof2D0BYiIDJhMGE3N2M1YTI1NDQzOGY5NTkyNTI3MGYyMzJm****.
        String versionId = "CAEQMxiBgICAof2D0BYiIDJhMGE3N2M1YTI1NDQzOGY5NTkyNTI3MGYyMzJm****";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);
        try {
            GenericRequest genericRequest = new GenericRequest(bucketName, objectName, versionId);
            ossClient.deleteObjectTagging(genericRequest);
        } catch (OSSException oe) {
            System.out.println("Caught an OSSException, which means your request made it to OSS, "
                + "but was rejected with an error response for some reason.");
            System.out.println("Error Message:" + oe.getErrorMessage());
            System.out.println("Error Code:" + oe.getErrorCode());
            System.out.println("Request ID:" + oe.getRequestId());
            System.out.println("Host ID:" + oe.getHostId());
        } catch (ClientException ce) {
            System.out.println("Caught an ClientException, which means the client encountered "
                + "a serious internal problem while trying to communicate with OSS, "
                + "such as not being able to access the network.");
            System.out.println("Error Message:" + ce.getMessage());
        } finally {
            if (ossClient != null) {
                ossClient.shutdown();
            }
        }
    }
}

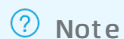
```

References

- For more information about the complete sample code that is used to delete the tags of an object, visit [Git Hub](#).
- For more information about the API operation that you can call to delete the tags of an object, see [DeleteObjectTagging](#).

2.11.4. Object tagging and lifecycle management

Lifecycle rules can take effect on objects configured with specified prefixes or tags. You can also specify a combination of prefixes and tags as conditions for lifecycle rules to take effect.



Add tagging configurations to a lifecycle rule

The following code provides an example on how to add tagging configurations to a lifecycle rule:

```
import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.common.utils.DateUtil;
import com.aliyun.oss.model.LifecycleRule;
import com.aliyun.oss.model.SetBucketLifecycleRequest;
import java.util.HashMap;
import java.util.Map;
public class Demo {
    public static void main(String[] args) throws Throwable {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify the actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to access OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the name of the bucket. Example: examplebucket.
        String bucketName = "examplebucket";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);

        try {
            // Create a request by using SetBucketLifecycleRequest.
            SetBucketLifecycleRequest request = new SetBucketLifecycleRequest(bucketName);
            // Specify the rule IDs and the object prefixes and tags to match the rules.
            String ruleId0 = "rule0";
            String matchPrefix0 = "A0/";
            Map<String, String> matchTags0 = new HashMap<String, String>();
            matchTags0.put("key0", "value0");
            String ruleId1 = "rule1";
            String matchPrefix1 = "A1/";
```

```

        Map<String, String> matchTags1 = new HashMap<String, String>();
        matchTags1.put("key1", "value1");
        String ruleId2 = "rule2";
        String matchPrefix2 = "A2/";
        String ruleId3 = "rule3";
        String matchPrefix3 = "A3/";
        // Set the expiration date to three days after the last modified time.
        LifecycleRule rule = new LifecycleRule(ruleId0, matchPrefix0, LifecycleRule.RuleStatus.Enabled, 3);
        rule.setTags(matchTags0);
        request.AddLifecycleRule(rule);
        // Specify that objects that are created before the specified date expire.
        rule = new LifecycleRule(ruleId1, matchPrefix1, LifecycleRule.RuleStatus.Enabled);
        rule.setCreatedBeforeDate(DateUtil.parseIso8601Date("2022-10-12T00:00:00.000Z"));
        rule.setTags(matchTags1);
        request.AddLifecycleRule(rule);
        // Set the expiration date of parts to three days later.
        rule = new LifecycleRule(ruleId2, matchPrefix2, LifecycleRule.RuleStatus.Enabled);
        LifecycleRule.AbortMultipartUpload abortMultipartUpload = new LifecycleRule.AbortMultipartUpload();
        abortMultipartUpload.setExpirationDays(3);
        rule.setAbortMultipartUpload(abortMultipartUpload);
        request.AddLifecycleRule(rule);
        // Specify that parts that are generated before the specified date expire.
        rule = new LifecycleRule(ruleId3, matchPrefix3, LifecycleRule.RuleStatus.Enabled);
        abortMultipartUpload = new LifecycleRule.AbortMultipartUpload();
        abortMultipartUpload.setCreatedBeforeDate(DateUtil.parseIso8601Date("2022-10-12T00:00:00.000Z"));
        rule.setAbortMultipartUpload(abortMultipartUpload);
        request.AddLifecycleRule(rule);
        ossClient.setBucketLifecycle(request);
    } catch (OSSException oe) {
        System.out.println("Caught an OSSException, which means your request made it to OSS, "
            + "but was rejected with an error response for some reason.");
        System.out.println("Error Message:" + oe.getErrorMessage());
        System.out.println("Error Code:" + oe.getErrorCode());
        System.out.println("Request ID:" + oe.getRequestId());
        System.out.println("Host ID:" + oe.getHostId());
    } catch (ClientException ce) {
        System.out.println("Caught an ClientException, which means the client encountered "
            + "a serious internal problem while trying to communicate with OSS, "
            + "such as not being able to access the network.");
        System.out.println("Error Message:" + ce.getMessage());
    } finally {
        if (ossClient != null) {
            ossClient.shutdown();
        }
    }
}

```

```

    }
}

```

Query the tagging configurations in a lifecycle rule

The following code provides an example on how to query tagging configurations in a lifecycle rule:

```

import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.LifecycleRule;
import java.util.List;
public class Demo {
    public static void main(String[] args) throws Throwable {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify the actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to access OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the name of the bucket. Example: examplebucket.
        String bucketName = "examplebucket";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);

        try {
            // Query the lifecycle rules that are configured for the bucket.
            List<LifecycleRule> rules = ossClient.getBucketLifecycle(bucketName);
            // Query the specified lifecycle rule.
            for (LifecycleRule rule1 : rules) {
                // Query the rule ID.
                System.out.println("rule id: " + rule1.getId());
                // Query the prefixes in the names of the objects that match the rule.
                System.out.println("rule prefix: " + rule1.getPrefix());
                // Query the tags of the objects that match the rule.
                if (rule1.hasTags()) {
                    System.out.println("rule tagging: " + rule1.getTags().toString());
                }
                // Query the number of days after which the objects that match the rule expire.
                if (rule1.hasExpirationDays()) {
                    System.out.println("rule expiration days: " + rule1.getExpirationDays());
                }
                // Query the date on which the objects that match the rule expire.
                if (rule1.hasCreatedBeforeDate()) {
                    System.out.println("rule expiration create before days: " + rule1.getCreatedBeforeDate());
                }
                // Query the rules based on which the parts that match the rules expire.
            }
        }
    }
}

```

```

        if(rule1.hasAbortMultipartUpload()) {
            if(rule1.getAbortMultipartUpload().hasExpirationDays()) {
                System.out.println("rule abort uppart days: " + rule1.getAbortMulti
partUpload().getExpirationDays());
            }
            if (rule1.getAbortMultipartUpload().hasCreatedBeforeDate()) {
                System.out.println("rule abort uppart create before date: " + rule1
.getAbortMultipartUpload().getCreatedBeforeDate());
            }
        }
    } catch (OSSException oe) {
        System.out.println("Caught an OSSException, which means your request made it to
OSS, "
            + "but was rejected with an error response for some reason.");
        System.out.println("Error Message:" + oe.getErrorMessage());
        System.out.println("Error Code:" + oe.getErrorCode());
        System.out.println("Request ID:" + oe.getRequestId());
        System.out.println("Host ID:" + oe.getHostId());
    } catch (ClientException ce) {
        System.out.println("Caught an ClientException, which means the client encounter
ed "
            + "a serious internal problem while trying to communicate with OSS, "
            + "such as not being able to access the network.");
        System.out.println("Error Message:" + ce.getMessage());
    } finally {
        if (ossClient != null) {
            ossClient.shutdown();
        }
    }
}
}

```

2.12. Single-connection bandwidth throttling

This topic describes how to add parameters in an object upload or download request to set the limit of upload or download bandwidth. This ensures sufficient bandwidth for other applications.

Configure bandwidth throttling for simple upload and download

The following sample code provides an example on how to configure single-connection bandwidth throttling for simple upload and download:

```

import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSException;
import com.aliyun.oss.model.GetObjectRequest;
import com.aliyun.oss.model.PutObjectRequest;
import java.io.File;
import java.io.FileInputStream;

```

```

import java.io.InputStream;
public class Demo {
    public static void main(String[] args) throws Throwable {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify the actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the name of the bucket. Example: examplebucket.
        String bucketName = "examplebucket";
        // Specify the full path of the object. The path cannot contain the bucket name. Example: exampledir/exampleobject.txt.
        String objectName = "exampledir/exampleobject.txt";
        // Specify the full path of the local file that you want to upload. Example: D:\\localpath\\examplefile.txt.
        // By default, if the path of the local file is not specified, the local file is uploaded from the path of the project to which the sample program belongs.
        String localFileName = "D:\\localpath\\examplefile.txt";
        // Specify the full path of the local file to which you want the object to be downloaded. If the specified local file exists, the object to download overwrites the file. Otherwise, a file is created.
        // If the path for the object is not specified, the downloaded object is saved to the path of the project to which the sample program belongs.
        String downloadFileName = "D:\\localpath\\exampleobject.txt";
        // Set the bandwidth limit to 100 KB/s.
        int limitSpeed = 100 * 1024 * 8;
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);

        try {
            // Configure bandwidth throttling for object upload.
            InputStream inputStream = new FileInputStream(localFileName);
            PutObjectRequest putObjectRequest = new PutObjectRequest(bucketName, objectName, inputStream);
            putObjectRequest.setTrafficLimit(limitSpeed);
            ossClient.putObject(putObjectRequest);
            // Configure bandwidth throttling for object download.
            GetObjectRequest getObjectRequest = new GetObjectRequest(bucketName, objectName);
            getObjectRequest.setTrafficLimit(limitSpeed);
            File localFile = new File(downloadFileName);
            ossClient.getObject(getObjectRequest, localFile);
        } catch (OSSException oe) {
            System.out.println("Caught an OSSException, which means your request made it to OSS, "
                + "but was rejected with an error response for some reason.");
            System.out.println("Error Message:" + oe.getErrorMessage());
            System.out.println("Error Code:" + oe.getErrorCode());
            System.out.println("Request ID:" + oe.getRequestId());
            System.out.println("Host ID:" + oe.getHostId());
        }
    }
}

```

```

    } catch (ClientException ce) {
        System.out.println("Caught an ClientException, which means the client encounter
ed "
            + "a serious internal problem while trying to communicate with OSS, "
            + "such as not being able to access the network.");
        System.out.println("Error Message:" + ce.getMessage());
    } finally {
        if (ossClient != null) {
            ossClient.shutdown();
        }
    }
}
}

```

Configure bandwidth throttling for upload and download that use signed URLs

The following sample code provides an example on how to configure single-connection bandwidth throttling when you use signed URLs to upload or download objects:

```

import com.aliyun.oss.*;
import com.aliyun.oss.model.GeneratePresignedUrlRequest;
import com.aliyun.oss.model.GetObjectRequest;
import java.io.File;
import java.io.FileInputStream;
import java.io.InputStream;
import java.net.URL;
import java.util.Date;
public class Demo {
    public static void main(String[] args) throws Throwable {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify the
        // actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations.
        // Using these credentials to perform operations in OSS is a high-risk operation. We recommend
        // that you use a RAM user to call API operations or perform routine O&M. To create a RAM user,
        // log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the name of the bucket. Example: examplebucket.
        String bucketName = "examplebucket";
        // Specify the full path of the object. The path cannot contain the bucket name. Example:
        // example: exampledir/exampleobject.txt.
        String objectName = "exampledir/exampleobject.txt";
        // Specify the full path of the local file that you want to upload. Example: D:\\localpath\\
        // examplefile.txt.
        // By default, if the path of the local file is not specified, the local file is uploaded from
        // the path of the project to which the sample program belongs.
        String localFileName = "D:\\localpath\\examplefile.txt";
        // Specify the full path of the local file to which you want the object to be downloaded. If
        // the specified local file exists, the object to download overwrites the file. Otherwise, a
        // file is created.
        // If the path for the object is not specified, the downloaded object is saved to the path of
        // the project to which the sample program belongs.
    }
}

```



```

ne path or the project to which the sample program belongs.
String downloadFileName = "D:\\localpath\\exampleobject.txt";
// Set the bandwidth limit to 100 KB/s.
int limitSpeed = 100 * 1024 * 8;
// Create an OSSClient instance.
OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret
);

try {
    // Generate a signed URL used to upload an object with bandwidth throttling con
    figured and set the validity period of the URL to 60 seconds.
    Date date = new Date();
    date.setTime(date.getTime() + 60 * 1000);
    GeneratePresignedUrlRequest request = new GeneratePresignedUrlRequest(bucketNam
e, objectName, HttpMethod.PUT);
    request.setExpiration(date);
    request.setTrafficLimit(limitSpeed);
    URL signedUrl = ossClient.generatePresignedUrl(request);
    System.out.println("put object url" + signedUrl);
    // Configure bandwidth throttling for object upload.
    InputStream inputStream = new FileInputStream(localFileName);
    ossClient.putObject(signedUrl, inputStream, -1, null, true);
    // Generate a signed URL used to download the object with bandwidth throttling
    configured and set the validity period of the URL to 60 seconds.
    date = new Date();
    date.setTime(date.getTime() + 60 * 1000);
    request = new GeneratePresignedUrlRequest(bucketName, objectName, HttpMethod.GE
T);
    request.setExpiration(date);
    request.setTrafficLimit(limitSpeed);
    signedUrl = ossClient.generatePresignedUrl(request);
    System.out.println("get object url" + signedUrl);
    // Configure bandwidth throttling for object download.
    GetObjectRequest getObjectRequest = new GetObjectRequest(signedUrl, null);
    ossClient.getObject(getObjectRequest, new File(downloadFileName));
} catch (OSSEException oe) {
    System.out.println("Caught an OSSEException, which means your request made it to
OSS, "
        + "but was rejected with an error response for some reason.");
    System.out.println("Error Message:" + oe.getErrorMessage());
    System.out.println("Error Code:" + oe.getErrorCode());
    System.out.println("Request ID:" + oe.getRequestId());
    System.out.println("Host ID:" + oe.getHostId());
} catch (ClientException ce) {
    System.out.println("Caught an ClientException, which means the client encounter
ed "
        + "a serious internal problem while trying to communicate with OSS, "
        + "such as not being able to access the network.");
    System.out.println("Error Message:" + ce.getMessage());
} finally {
    if (ossClient != null) {
        ossClient.shutdown();
    }
}
}

```

References

For more information about the complete sample code of single-connection bandwidth throttling, visit [GitHub](#).

2.13. Data encryption

2.13.1. Server-side encryption

OSS supports server-side encryption for uploaded data. When you upload data, OSS encrypts the data and stores the encrypted data. When you download data, OSS decrypts the data and returns the original data. The returned HTTP request header declares that the data is encrypted on the server.

Background information

OSS provides the following server-side encryption methods:

- Server-side encryption by using CMKs managed by KMS (SSE-KMS)

When you upload an object, you can use a specified CMK ID or the default CMK managed by KMS to encrypt and decrypt a large amount of data. This method is cost-effective because you do not need to send user data to the KMS server by using networks for encryption and decryption.

 **Notice** You are charged for calling API operations when you use CMKs to encrypt or decrypt data.

- Server-side encryption by using OSS-managed keys (SSE-OSS)

When you upload an object, OSS encrypts the object on the server side by using AES-256 managed by OSS. OSS server-side encryption uses AES-256 to encrypt objects by using different data keys. AES-256 uses master keys that are regularly rotated to encrypt data keys.

-  **Notice**
- Only one server-side encryption method can be used for an object at a time.
 - If you configure server-side encryption for a bucket, you can still configure the encryption method for a single object when you upload or copy it. In this case, the encryption method configured for the object takes precedence. For more information, see [PutObject](#).
 - For more information about server-side encryption, see [Server-side encryption](#).

Configure server-side encryption for a bucket

The following code provides an example on how to configure the default encryption method for a bucket. After the method is configured, all objects that are uploaded to the bucket without encryption methods configured are encrypted by using the default encryption method.

```
// The endpoint of the China (Hangzhou) region is used in this example. Specify the actual endpoint.
String endpoint = "http://oss-cn-hangzhou.aliyuncs.com";
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log on to OSS because the account has permissions on all API operations. We recommend that you use your RAM user's credentials to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
String accessKeyId = "<yourAccessKeyId>";
String accessKeySecret = "<yourAccessKeySecret>";
// Create an OSSClient instance.
OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);
// Configure server-side encryption for the bucket.
ServerSideEncryptionByDefault applyServerSideEncryptionByDefault = new ServerSideEncryptionByDefault(SSEAlgorithm.KMS);
applyServerSideEncryptionByDefault.setKMSEncryptionMasterKeyID("<yourTestKmsId>");
ServerSideEncryptionConfiguration sseConfig = new ServerSideEncryptionConfiguration();
sseConfig.setApplyServerSideEncryptionByDefault(applyServerSideEncryptionByDefault);
SetBucketEncryptionRequest request = new SetBucketEncryptionRequest("<yourBucketName>", sseConfig);
ossClient.setBucketEncryption(request);
// Shut down the OSSClient instance.
ossClient.shutdown();
```

Query the server-side encryption configurations of a bucket

The following code provides an example on how to query the server-side encryption configurations of a bucket:

```
// The endpoint of the China (Hangzhou) region is used in this example. Specify the actual endpoint.
String endpoint = "http://oss-cn-hangzhou.aliyuncs.com";
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log on to OSS because the account has permissions on all API operations. We recommend that you use your RAM user's credentials to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
String accessKeyId = "<yourAccessKeyId>";
String accessKeySecret = "<yourAccessKeySecret>";
// Create an OSSClient instance.
OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);
// Query the server-side encryption configurations of the bucket.
ServerSideEncryptionConfiguration sseConfig = ossClient.getBucketEncryption("<yourBucketName>");
System.out.println("get Algorithm: " + sseConfig.getApplyServerSideEncryptionByDefault().getSSEAlgorithm());
System.out.println("get kmsid: " + sseConfig.getApplyServerSideEncryptionByDefault().getKMSEncryptionMasterKeyID());
// Shut down the OSSClient instance.
ossClient.shutdown();
```

Delete the server-side encryption configurations of a bucket

The following code provides an example on how to delete the server-side encryption configurations of a bucket:

```
// The endpoint of the China (Hangzhou) region is used in this example. Specify the actual endpoint.
String endpoint = "http://oss-cn-hangzhou.aliyuncs.com";
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log on to OSS because the account has permissions on all API operations. We recommend that you use your RAM user's credentials to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
String accessKeyId = "<yourAccessKeyId>";
String accessKeySecret = "<yourAccessKeySecret>";
// Create an OSSClient instance.
OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);
// Delete the server-side encryption configurations of the bucket.
ossClient.deleteBucketEncryption("<yourBucketName>");
// Shut down the OSSClient instance.
ossClient.shutdown();
```

2.13.2. Client-side encryption

Client-side encryption is used to encrypt objects on the local client before they are uploaded to Object Storage Service (OSS).

Background information

Notice

- Client-side encryption supports multipart upload for objects larger than 5 GB in size. When you use multipart upload to upload an object, you must specify the total size of the object and the size of each part. The size of each part except for the last part must be the same and be a multiple of 16 bytes.
- After you upload objects encrypted on the client, object metadata related to client-side encryption is protected and cannot be modified by calling [CopyObject](#).

Encryption methods

Object metadata related to client-side encryption

Parameter	Description	Required
x-oss-meta-client-side-encryption-key	The encrypted data key. The encrypted data key is a string encrypted by using a customer master key (CMK) and encoded in Base64.	Yes

Parameter	Description	Required
x-oss-meta-client-side-encryption-start	The initial value that is randomly generated for data encryption. The initial value is a string encrypted by using a CMK and encoded in Base64.	Yes
x-oss-meta-client-side-encryption-cek-alg	The algorithm used to encrypt data.	Yes
x-oss-meta-client-side-encryption-wrap-alg	The algorithm used to encrypt the data key.	Yes
x-oss-meta-client-side-encryption-matdesc	The description of the CMK in the JSON format.  Warning We recommend that you configure a description for each CMK and store the mapping relationship between the CMK and its description. A CMK without a specified description cannot be replaced.	No
x-oss-meta-client-side-encryption-unencrypted-content-length	The length of data before encryption. This parameter is not generated if Content-Length is not specified.	No
x-oss-meta-client-side-encryption-unencrypted-content-md5	The MD5 hash of the data before encryption. This parameter is not generated if Content-MD5 is not specified.	No
x-oss-meta-client-side-encryption-data-size	The total size of the data to encrypt for multipart upload when init_multipart is called.	Yes (for multipart upload)
x-oss-meta-client-side-encryption-part-size	The size of each part to encrypt for multipart upload when init_multipart is called.  Notice The size of each part must be a multiple of 16 bytes.	Yes (for multipart upload)

Create a client for client-side encryption

 **Note** When you use client-side encryption, make sure that you are using OSS SDK for Java V3.9.1 or later and have added the Bouncy Castle Crypto package to your project. For example, you can add the following dependency to the pom.xml file:

```
<dependency>
  <groupId>org.bouncycastle</groupId>
  <artifactId>bcprov-jdk15on</artifactId>
  <version>1.62</version>
</dependency>
```

If the `java.security.InvalidKeyException: Illegal key size or default parameters` exception occurs during the operation, you must add the Java Cryptography Extension (JCE) jurisdiction policy file of Oracle and deploy this file in the JRE environment.

Download one of the following files based on the JDK version that you use. Then, decompress the file and save them to the `jre/lib/security` directory.

- [Java Cryptography Extension \(JCE\) Unlimited Strength Jurisdiction Policy Files 6](#)
- [Java Cryptography Extension \(JCE\) Unlimited Strength Jurisdiction Policy Files 7](#)
- [Java Cryptography Extension \(JCE\) Unlimited Strength Jurisdiction Policy Files 8](#)

The following sections provide complete sample code on how to create a client for RSA-based encryption and for KMS-based encryption.

- Create a client for RSA-based encryption

Before you create a client for RSA-based encryption, you must create an asymmetric key pair. OSS SDK for Java provides methods for converting a PKCS #1-encoded or PKCS #8-encoded private key string in the PEM format to an RSA private key, and methods for converting an X.509-encoded public key string in the PEM format to an RSA public key.

You can use the following methods to convert keys that meet the preceding requirements:

```
RSAPrivateKey SimpleRSAEncryptionMaterials.getPrivateKeyFromPemPKCS1(String privateKeyStr)
;
RSAPrivateKey SimpleRSAEncryptionMaterials.getPrivateKeyFromPemPKCS8(String privateKeyStr)
;
RSAPublicKey SimpleRSAEncryptionMaterials.getPublicKeyFromPemX509(String publicKeyStr);
```

The following code provides an example on how to create a client for RSA-based encryption:

```
import com.aliyun.oss.*;
import com.aliyun.oss.crypto.SimpleRSAEncryptionMaterials;
import java.security.KeyPair;
import java.security.interfaces.RSAPrivateKey;
import java.security.interfaces.RSAPublicKey;
import java.util.HashMap;
import java.util.Map;
public class Demo {
    public static void main(String[] args) throws Throwable {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify
        // your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API oper
```

ations. Using these credentials to access OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.

```
String accessKeyId = "yourAccessKeyId";
String accessKeySecret = "yourAccessKeySecret";

// You can run the following commands to separately generate the PEM files for the
// private key and public key. Then, set the values of PRIVATE_PKCS1_PEM and PUBLIC_X509_PEM
// to the strings included in the PEM files for the private key and public key.
// openssl genrsa -out private_key.pem 2048
// openssl rsa -in private_key.pem -out rsa_public_key.pem -pubout
// Enter your RSA private key string. You can generate the string by using OpenSSL.
// The following code shows a sample RSA private key string.
```

```
final String PRIVATE_PKCS1_PEM =
    "-----BEGIN RSA PRIVATE KEY-----\n" +
    "MIICWwIBAAKBgQCokfiAVXXf5ImFzKDw+XO/UByW6mse2QsIgz3ZwBtMNu59fR5z\n" +
    "ttSx+8fB7vR4CN3bTztrP9A6bjoN0FFnhlQ3vNJC5MF01PByrE/MNd5AAfSVba93\n" +
    "I6sx8NSk5MzUCA4NJzAUqYOEwGtGBcom6kEF6MmR1EKib1Id8hpooY5xaQIDAQAB\n" +
    "AoGAOPUZgkNeEMinrw3lU3b2JS5sepG6oDG2CKpPu8OtdZMaAkzEfVTJiVoJpP2Y\n" +
    "nPziADhFW3e0ZAnak9BPsSsySRaSNmR465cG9tbqpXFKh9Rp/sCPo4Jq2n65yood\n" +
    "JBBrnGr6/xhYvNa14sQ6xjJfSgRNBsXD1XXNF4kALwgZyCAECQQDV7t4bTx9FbEs5\n" +
    "36nAxPsPM6aACXaOkv6d9LXI7A0J8Zf42FeBV6RK0q7QG5iNNd1WJHSXIITUizVF\n" +
    "6aX5NnvFAkEAYbeXNowUvYtkgx4s28s6gn1lc5HZw4/a8vZm2tXXK/QfTQrJVXp\n" +
    "VwxmSr0FAajWAlcYN/fGkXlpWA04lCKFVQJAG08ozzekeEpAuByTIOaEXgZr5MBQ\n" +
    "gBbHpgZNB18Lsw9CJSQI15wGfv6yDiLXsH8FyC9TKs+d5Tv4Cvquk0efOQJAd9OC\n" +
    "lCKFs48hdaiz9yEDsc57PdrvRFepVdj/gpGzD14mVerJbOiOF6aSV19ot27u4on\n" +
    "Td/3aifYs0CveHzFPQJAWb4LCDwqLctfzZiG7/S7Z74gyq5qZF4FUElOAZkz123E\n" +
    "yZvADwuz/4aK0od0lX9c4Jp7Mo5vQ4TvdoBnPuGo****\n" +
    "-----END RSA PRIVATE KEY-----";

// Enter your RSA public key string. You can generate the string by using OpenSSL.
// The following code shows a sample RSA public key string.
```

```
final String PUBLIC_X509_PEM =
    "-----BEGIN PUBLIC KEY-----\n" +
    "MIGfMA0GCSqGSIb3DQEBAQUAA4GNADCBiQKBgQCokfiAVXXf5ImFzKDw+XO/UByW\n" +
    "6mse2QsIgz3ZwBtMNu59fR5zttSx+8fB7vR4CN3bTztrP9A6bjoN0FFnhlQ3vNJC\n" +
    "5MF01PByrE/MNd5AAfSVba93I6sx8NSk5MzUCA4NJzAUqYOEwGtGBcom6kEF6MnR\n" +
    "1EKib1Id8hpooY5xaQID****\n" +
    "-----END PUBLIC KEY-----";

// Create an RSA key pair.
RSAPrivateKey privateKey = SimpleRSAEncryptionMaterials.getPrivateKeyFromPemPKCS1
(PRIVATE_PKCS1_PEM);
RSAPublicKey publicKey = SimpleRSAEncryptionMaterials.getPublicKeyFromPemX509(PUB
LIC_X509_PEM);
KeyPair keyPair = new KeyPair(publicKey, privateKey);

// Specify the description of the CMK. The description cannot be modified after i
t is specified. You can specify only one description for a CMK.

// If all objects share a CMK, you can leave the description of the CMK empty. In
this case, the CMK cannot be replaced.

// If the description of the CMK is empty, the client cannot determine which CMK
to use for decryption.

// We recommend that you configure a description for each CMK and store the mappi
ng relationship between the CMK and the description in the client. The server does not st
ore the relationship.
Map<String, String> matDesc = new HashMap<String, String>();
matDesc.put("desc-key", "desc-value");
```

```

        // Create RSA encryption materials.
        SimpleRSAEncryptionMaterials encryptionMaterials = new SimpleRSAEncryptionMaterials(keyPair, matDesc);
        // To download and decrypt objects encrypted by using other CMKs, add these CMKs and their descriptions to the encryption materials.
        // encryptionMaterials.addKeyPairDescMaterial(<otherKeyPair>, <otherKeyPairMatDesc>);

        // Create a client for RSA-based encryption.
        OSSEncryptionClient ossEncryptionClient = new OSSEncryptionClientBuilder().
            build(endpoint, accessKeyId, accessKeySecret, encryptionMaterials);
        try {
            // Develop your business code.
        } catch (OSSEException oe) {
            System.out.println("Caught an OSSEException, which means your request made it to OSS, "
                + "but was rejected with an error response for some reason.");
            System.out.println("Error Message:" + oe.getErrorMessage());
            System.out.println("Error Code:" + oe.getErrorCode());
            System.out.println("Request ID:" + oe.getRequestId());
            System.out.println("Host ID:" + oe.getHostId());
        } catch (ClientException ce) {
            System.out.println("Caught an ClientException, which means the client encountered "
                + "a serious internal problem while trying to communicate with OSS, "
                + "such as not being able to access the network.");
            System.out.println("Error Message:" + ce.getMessage());
        } finally {
            if (ossEncryptionClient != null) {
                ossEncryptionClient.shutdown();
            }
        }
    }
}

```

- Create a client for KMS-based encryption

The following code provides an example on how to create a client for KMS-based encryption:

```

import com.aliyun.oss.*;
import com.aliyun.oss.crypto.KmsEncryptionMaterials;
import java.util.HashMap;
import java.util.Map;

public class Demo {
    public static void main(String[] args) throws Throwable {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to access OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify your CMK ID. Example: e1935511-cf88-1123-a0f8-1be8d251****.
        String cmk = "e1935511-cf88-1123-a0f8-1be8d251****";
    }
}

```



```

// Specify the region to which the CMK belongs.
String region = "cn-hangzhou";
// Specify the description of the CMK. The description cannot be modified after i
t is specified. You can specify only one description and one region for a CMK.
// If all objects share a CMK, you can leave the description of the CMK empty. In
this case, the CMK cannot be replaced.
// If the description of the CMK is empty, the client cannot determine which CMK
to use for decryption.
// We recommend that you configure a description for each CMK and store the mappi
ng relationship between the CMK and the description in the client. The server does not st
ore the relationship.
Map<String, String> matDesc = new HashMap<String, String>();
matDesc.put("desc-key", "desc-value");
// Create KMS encryption materials.
KmsEncryptionMaterials encryptionMaterials = new KmsEncryptionMaterials(region, c
mk, matDesc);
// To download and decrypt objects encrypted by using other CMKs, add the region
names and descriptions of the CMKs to the KMS encryption materials.
// encryptionMaterials.addKmsDescMaterial(<otherKmsRegion>, <otherKmsMatDesc>);
// In scenarios where the account used to access KMS is different from the accoun
t used to access OSS, if you want to download and decrypt objects encrypted by using othe
r CMKs, add the region names, credentials, and descriptions of the CMKs to the KMS encryp
tion materials.
// encryptionMaterials.addKmsDescMaterial(<otherKmsRegion>, <otherKmsCredentialsP
rovider>, <otherKmsMatDesc>);
// Create a client for KMS-based encryption.
OSSEncryptionClient ossEncryptionClient = new OSSEncryptionClientBuilder().
    build(endpoint, accessKeyId, accessKeySecret, encryptionMaterials);
try {
    // Develop your business code.
} catch (OSSEException oe) {
    System.out.println("Caught an OSSEException, which means your request made it
to OSS, "
        + "but was rejected with an error response for some reason.");
    System.out.println("Error Message:" + oe.getErrorMessage());
    System.out.println("Error Code:" + oe.getErrorCode());
    System.out.println("Request ID:" + oe.getRequestId());
    System.out.println("Host ID:" + oe.getHostId());
} catch (ClientException ce) {
    System.out.println("Caught an ClientException, which means the client encount
ered "
        + "a serious internal problem while trying to communicate with OSS, "
        + "such as not being able to access the network.");
    System.out.println("Error Message:" + ce.getMessage());
} finally {
    if (ossEncryptionClient != null) {
        ossEncryptionClient.shutdown();
    }
}
}
}

```

The following sections provide complete sample code for using customer-managed CMKs (RSA-based) to perform client-side encryption in simple upload and download, multipart upload, and range download.

 **Note** KMS-based client-side encryption differs from RSA-based client-side encryption only in the creation of `OSSEncryptionClient`.

Perform RSA-based client-side encryption in simple upload and download

The following code provides an example on how to use customer-managed CMKs to encrypt objects in simple upload or decrypt objects in simple download:

```
import com.aliyun.oss.*;
import com.aliyun.oss.crypto.SimpleRSAEncryptionMaterials;
import com.aliyun.oss.model.OSSObject;
import java.io.BufferedReader;
import java.io.ByteArrayInputStream;
import java.io.InputStreamReader;
import java.security.KeyPair;
import java.security.interfaces.RSAPrivateKey;
import java.security.interfaces.RSAPublicKey;
import java.util.HashMap;
import java.util.Map;
public class Demo {
    public static void main(String[] args) throws Throwable {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to access OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the bucket name. Example: examplebucket.
        String bucketName = "examplebucket";
        // Specify the full path of the object. Example: exampleobject.txt. The full path of the object cannot contain the bucket name.
        String objectName = "exampleobject.txt";
        String content = "Hello OSS!";
        // Enter your RSA private key string. You can generate the string by using OpenSSL. The following code shows a sample RSA private key string.
        final String PRIVATE_PKCS1_PEM =
            "-----BEGIN RSA PRIVATE KEY-----\n" +
            "MIICWwIBAAKBgQCokfiAVXXf5ImFzKDw+XO/UByW6mse2QsIgz3ZwBtMNu59fR5z\n" +
            "ttSx+8fB7vR4CN3bTztrP9A6bjoN0FFnhlQ3vNJC5MF01PByrE/MNd5AAfSVba93\n" +
            "I6sx8NSk5MzUCA4NJzAUqYOEWTGBcom6kEF6MmR1EKib1Id8hpooY5xaQIDAQAB\n" +
            "AoGAOPUZgkNeEMinrw31U3b2JS5sepG6oDG2CKpPu8OtdZMaAkzEfVTJiVoJpP2Y\n" +
            "nPZiAdhFW3e0ZAnak9BPSSySRaSNmR465cG9tbqpXFKh9Rp/sCPo4Jq2n65yood\n" +
            "JBrnGr6/xhYvNa14sQ6xjjfSgRNBSXD1XXNF4kALwgZyCAECQQDV7t4bTx9FbEs5\n" +
            "36nAxPsPM6aACXaOkv6d9LXI7A0J8Zf42FeBV6RK0q7QG5iNNd1WJHSXIITUizVF\n" +
            "6aX5NnvFAkEaybeXNOwUvYtkgxF4s28s6gn11c5HZw4/a8vZm2tXXK/QfTQrJVXp\n" +
```

```

        "VwxmSr0FAajWAlcYN/fGkXlpWA041CKFVQJAG08ozzekeEpAuByTIOaEXgZr5MBQ\n" +
        "gBbHpgZNB18Lsw9CJSQI15wGfv6yDiLXsH8FyC9TKs+d5Tv4Cvquk0efOQJAd9OC\n" +
        "lCKFs48hdyaiZ9yEDsc57PdrvRFepVdj/gpGzD14mVerJb0iOF6aSV19ot27u4on\n" +
        "Td/3aifYs0CveHzFPQJAWb4LCDwqLctfzziG7/S7Z74gyq5qZF4FUElOAZkz123E\n" +
        "yZvADwuz/4aK0od0lX9c4Jp7Mo5vQ4TvdoBnPuGo****\n" +
        "-----END RSA PRIVATE KEY-----";

    // Enter your RSA public key string. You can generate the string by using OpenSSL.
    The following code shows a sample RSA public key string.
    final String PUBLIC_X509_PEM =
        "-----BEGIN PUBLIC KEY-----\n" +
        "MIGfMA0GCSqGSIb3DQEBAQUAA4GNADCBiQKBgQCokfiAVXXf5ImFzKDw+XO/UByW\n" +
        "6mse2QsIgz3ZwBtMNu59fR5zttSx+8fB7vR4CN3bTztrP9A6bjoN0FFnhlQ3vNJC\n" +
        "5MFO1PByrE/MNd5AAfSVba93I6sx8NSk5MzUCA4NJzAUqYOEWGtGBcom6kEF6MnR\n" +
        "1EKib1Id8hpooY5xaQID****\n" +
        "-----END PUBLIC KEY-----";

    // Create an RSA key pair.
    RSAPrivateKey privateKey = SimpleRSAEncryptionMaterials.getPrivateKeyFromPemPKCS1(P
RIVATE_PKCS1_PEM);
    RSAPublicKey publicKey = SimpleRSAEncryptionMaterials.getPublicKeyFromPemX509(PUBLI
C_X509_PEM);
    KeyPair keyPair = new KeyPair(publicKey, privateKey);
    // Specify the description of the CMK. The description cannot be modified after it
    is specified. You can specify only one description for a CMK.
    // If all objects share a CMK, you can leave the description of the CMK empty. In t
    his case, the CMK cannot be replaced.
    // If the description of the CMK is empty, the client cannot determine which CMK to
    use for decryption.
    // We recommend that you configure a description for each CMK and store the mapping
    relationship between the CMK and the description in the client. The server does not store t
    he relationship.
    Map<String, String> matDesc = new HashMap<String, String>();
    matDesc.put("desc-key", "desc-value");
    // Create RSA encryption materials.
    SimpleRSAEncryptionMaterials encryptionMaterials = new SimpleRSAEncryptionMaterials
(keyPair, matDesc);
    // To download and decrypt objects encrypted by using other CMKs, add these CMKs an
    d their descriptions to the encryption materials.
    // encryptionMaterials.addKeyPairDescMaterial(<otherKeyPair>, <otherKeyPairMatDesc>
);

    // Create a client for client-side encryption.
    OSSEncryptionClient ossEncryptionClient = new OSSEncryptionClientBuilder().
        build(endpoint, accessKeyId, accessKeySecret, encryptionMaterials);

    try {
        // Encrypt the object that you want to upload.
        ossEncryptionClient.putObject(bucketName, objectName, new ByteArrayInputStream(
content.getBytes()));
        // Download the object. The object is automatically decrypted.
        OSSObject ossObject = ossEncryptionClient.getObject(bucketName, objectName);
        BufferedReader reader = new BufferedReader(new InputStreamReader(ossObject.getO
bjectContent()));
        StringBuffer buffer = new StringBuffer();
        String line;
        while ((line = reader.readLine()) != null) {
            buffer.append(line);

```

```

    }
    reader.close();
    // Check whether the decrypted content is the same as the uploaded object in pl
aintext.
    System.out.println("Put plain text: " + content);
    System.out.println("Get and decrypted text: " + buffer.toString());
} catch (OSSException oe) {
    System.out.println("Caught an OSSException, which means your request made it to
OSS, "
        + "but was rejected with an error response for some reason.");
    System.out.println("Error Message:" + oe.getErrorMessage());
    System.out.println("Error Code:" + oe.getErrorCode());
    System.out.println("Request ID:" + oe.getRequestId());
    System.out.println("Host ID:" + oe.getHostId());
} catch (ClientException ce) {
    System.out.println("Caught an ClientException, which means the client encounter
ed "
        + "a serious internal problem while trying to communicate with OSS, "
        + "such as not being able to access the network.");
    System.out.println("Error Message:" + ce.getMessage());
} finally {
    if (ossEncryptionClient != null) {
        ossEncryptionClient.shutdown();
    }
}
}
}

```

Perform RSA-based client-side encryption in multipart upload

The following code provides an example on how to use a customer-managed CMK (RSA-based) to encrypt objects in multipart upload:

```

import com.aliyun.oss.*;
import com.aliyun.oss.crypto.MultipartUploadCryptoContext;
import com.aliyun.oss.crypto.SimpleRSAEncryptionMaterials;
import com.aliyun.oss.model.*;
import java.io.*;
import java.security.KeyPair;
import java.security.interfaces.RSAPrivateKey;
import java.security.interfaces.RSAPublicKey;
import java.util.ArrayList;
import java.util.HashMap;
import java.util.List;
import java.util.Map;
public class Demo {
    public static void main(String[] args) throws Throwable {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify yo
ur actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API operat
ions. Using these credentials to access OSS is a high-risk operation. We recommend that you
use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on
to the RAM console.

```

```

String accessKeyId = "yourAccessKeyId";
String accessKeySecret = "yourAccessKeySecret";
// Specify the bucket name. Example: examplebucket.
String bucketName = "examplebucket";
// Specify the full path of the object. Example: exampleobject.txt. The full path o
f the object cannot contain the bucket name.
String objectName = "exampleobject.txt";
// Specify the full path of the local file. Example: D:\\localpath\\examplefile.txt
.

String localFile = "D:\\localpath\\examplefile.txt";
// Enter your RSA private key string. You can generate the string by using OpenSSL.
The following code shows a sample RSA private key string.
final String PRIVATE_PKCS1_PEM =
    "-----BEGIN RSA PRIVATE KEY-----\n" +
    "MIICWwIBAAKBgQCokfiAVXXf5ImFzKDw+XO/UByW6mse2QsIgz3ZwBtMNU59fR5z\n" +
    "ttSx+8fB7vR4CN3bTztrP9A6bjoN0FFnhlQ3vNJC5MF01PByrE/MNd5AAfSVba93\n" +
    "I6sx8NSk5MzUCA4NJzAUqYOEWGtGBcom6kEF6MmR1EKib1Id8hpooY5xaQIDAQAB\n" +
    "AoGAOPUZgkNeEMinrw31U3b2JS5sepG6oDG2CKpPu8OtdZMaAkzEfVTJiVoJpP2Y\n" +
    "nPZiAdhFW3e0ZAnak9BPsSsySRaSNmr465cG9tbqpXFKh9Rp/sCPo4Jq2n65yood\n" +
    "JBrnGr6/xhYvNa14sQ6xjJfSgRNBSXD1XXNF4kALwgZyCAECQQDV7t4bTx9FbEs5\n" +
    "36nAxPsPM6aACXaOkv6d9LXI7A0J8Zf42FeBV6RK0q7QG5iNNd1WJHSXIITUizVF\n" +
    "6aX5NnvFAkEAYbeXNOwUvYtkgxF4s28s6gn11c5HZw4/a8vZm2tXXK/QfTQrJVXp\n" +
    "VwxmSr0FAajWalcYN/fGkXlpWA041CKFVQJAG08ozzekeEpAuByTIOaEXgZr5MBQ\n" +
    "gBbHpgZNB18Lsw9CJSQI15wGfv6yDiLXsH8FyC9TKs+d5Tv4Cvquk0efOQJAd9OC\n" +
    "lCKFs48hdyaiZ9yEDsc57PdrvRFepVdj/gpGzD14mVerJbOi0F6aSV19ot27u4on\n" +
    "Td/3aifYs0CveHzFPQJAWb4LCDwqLctfzziG7/S7Z74gyq5qZF4FUElOAZkz123E\n" +
    "yZvAdwuz/4aK0od01X9c4Jp7Mo5vQ4TvdoBnPuGo****\n" +
    "-----END RSA PRIVATE KEY-----";

// Enter your RSA public key string. You can generate the string by using OpenSSL.
The following code shows a sample RSA public key string.
final String PUBLIC_X509_PEM =
    "-----BEGIN PUBLIC KEY-----\n" +
    "MIGfMA0GCSqGSIb3DQEBAQUAA4GNADCBiQKBgQCokfiAVXXf5ImFzKDw+XO/UByW\n" +
    "6mse2QsIgz3ZwBtMNU59fR5zttSx+8fB7vR4CN3bTztrP9A6bjoN0FFnhlQ3vNJC\n" +
    "5MF01PByrE/MNd5AAfSVba93I6sx8NSk5MzUCA4NJzAUqYOEWGtGBcom6kEF6MnR\n" +
    "1EKib1Id8hpooY5xaQID****\n" +
    "-----END PUBLIC KEY-----";

// Create an RSA key pair.
RSAPrivateKey privateKey = SimpleRSAEncryptionMaterials.getPrivateKeyFromPemPKCS1(P
RIVATE_PKCS1_PEM);
RSAPublicKey publicKey = SimpleRSAEncryptionMaterials.getPublicKeyFromPemX509(PUBLI
C_X509_PEM);
KeyPair keyPair = new KeyPair(publicKey, privateKey);
// Specify the description of the CMK. The description cannot be modified after it
is specified. You can specify only one description for a CMK.
// If all objects share a CMK, you can leave the description of the CMK empty. In t
his case, the CMK cannot be replaced.
// If the description of the CMK is empty, the client cannot determine which CMK to
use for decryption.
// We recommend that you configure a description for each CMK and store the mapping
relationship between the CMK and the description in the client. The server does not store t
he relationship.
Map<String, String> matDesc = new HashMap<String, String>();
matDesc.put("desc-key", "desc-value");

```

```

        // Create RSA encryption materials.
        SimpleRSAEncryptionMaterials encryptionMaterials = new SimpleRSAEncryptionMaterials
(keyPair, matDesc);
        // To download and decrypt objects encrypted by using other CMKs, add these CMKs an
d their descriptions to the encryption materials.
        // encryptionMaterials.addKeyPairDescMaterial(<otherKeyPair>, <otherKeyPairMatDesc>
);
        // Create a client for client-side encryption.
        OSSEncryptionClient ossEncryptionClient = new OSSEncryptionClientBuilder().
            build(endpoint, accessKeyId, accessKeySecret, encryptionMaterials);
        try {
            // Create a MultipartUploadCryptoContext object and specify the sizes of the ob
ject and each part. The part size must be a multiple of 16 bytes.
            File file = new File(localFile);
            long fileLength = file.length();
            final long partSize = 100 * 1024L;    // 100K
            MultipartUploadCryptoContext context = new MultipartUploadCryptoContext();
            context.setPartSize(partSize);
            context.setDataSize(fileLength);
            // Initiate a multipart upload task.
            InitiateMultipartUploadRequest initiateMultipartUploadRequest = new InitiateMul
tipartUploadRequest(bucketName, objectName);
            // Upload the created MultipartUploadCryptoContext object.
            InitiateMultipartUploadResult upresult = ossEncryptionClient.initiateMultipartU
pload(initiateMultipartUploadRequest, context);
            String uploadId = upresult.getUploadId();
            // Create a set of PartETags. A PartETag consists of the part number and ETag o
f an uploaded part.
            List<PartETag> partETags = new ArrayList<PartETag>();
            int partCount = (int) (fileLength / partSize);
            if (fileLength % partSize != 0) {
                partCount++;
            }
            // Upload each part until all parts are uploaded.
            for (int i = 0; i < partCount; i++) {
                long startPos = i * partSize;
                long curPartSize = (i + 1 == partCount) ? (fileLength - startPos) : partSiz
e;

                InputStream instream = new FileInputStream(file);
                instream.skip(startPos);
                UploadPartRequest uploadPartRequest = new UploadPartRequest();
                uploadPartRequest.setBucketName(bucketName);
                uploadPartRequest.setKey(objectName);
                uploadPartRequest.setUploadId(uploadId);
                uploadPartRequest.setInputStream(instream);
                uploadPartRequest.setPartSize(curPartSize);
                uploadPartRequest.setPartNumber(i + 1);
                // Upload the created MultipartUploadCryptoContext object.
                UploadPartResult uploadPartResult = ossEncryptionClient.uploadPart(uploadPa
rtRequest, context);
                partETags.add(uploadPartResult.getPartETag());
            }
            // Complete the multipart upload task.
            CompleteMultipartUploadRequest completeMultipartUploadRequest =
                new CompleteMultipartUploadRequest(bucketName, objectName, uploadId, pa

```

```

        new CompleteMultipartUploadRequest(bucketName, objectName, uploadId, pa
rtETags);
        ossEncryptionClient.completeMultipartUpload(completeMultipartUploadRequest);
    } catch (OSSException oe) {
        System.out.println("Caught an OSSException, which means your request made it to
OSS, "
            + "but was rejected with an error response for some reason.");
        System.out.println("Error Message:" + oe.getErrorMessage());
        System.out.println("Error Code:" + oe.getErrorCode());
        System.out.println("Request ID:" + oe.getRequestId());
        System.out.println("Host ID:" + oe.getHostId());
    } catch (ClientException ce) {
        System.out.println("Caught an ClientException, which means the client encounter
ed "
            + "a serious internal problem while trying to communicate with OSS, "
            + "such as not being able to access the network.");
        System.out.println("Error Message:" + ce.getMessage());
    } finally {
        if (ossEncryptionClient != null) {
            ossEncryptionClient.shutdown();
        }
    }
}
}

```

Perform RSA-based client-side encryption in resumable upload

The following code provides an example on how to use a customer-managed CMK (RSA-based) to encrypt objects in resumable upload:

```

import com.aliyun.oss.*;
import com.aliyun.oss.crypto.SimpleRSAEncryptionMaterials;
import com.aliyun.oss.model.*;
import java.security.KeyPair;
import java.security.interfaces.RSAPrivateKey;
import java.security.interfaces.RSAPublicKey;
import java.util.HashMap;
import java.util.Map;

public class Demo {
    public static void main(String[] args) throws Throwable {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify yo
ur actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API operat
ions. Using these credentials to access OSS is a high-risk operation. We recommend that you
use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on
to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the bucket name. Example: examplebucket.
        String bucketName = "examplebucket";
        // Specify the full path of the object. Example: exampleobject.txt. The full path o
f the object cannot contain the bucket name.
        String objectName = "exampleobject.txt";

```

```

        // Specify the full path of the local file. Example: D:\\localpath\\examplefile.txt
        .

        String localFile = "D:\\localpath\\examplefile.txt";
        // Enter your RSA private key string. You can generate the string by using OpenSSL.
        The following code shows a sample RSA private key string.
        final String PRIVATE_PKCS1_PEM =
            "-----BEGIN RSA PRIVATE KEY-----\n" +
            "MIICWwIBAAKBgQCokfiAVXXf5ImFzKDw+XO/UByW6mse2QsIgz3ZwBtMNU59fR5z\n" +
            "ttSx+8fB7vR4CN3bTztrP9A6bjoN0FFnhlQ3vNJC5MF01PByrE/MNd5AAfSVba93\n" +
            "I6sx8NSk5MzUCA4NJzAUqYOEwGtGBcom6kEF6MmR1EKib1Id8hpooY5xaQIDAQAB\n" +
            "AoGAOPUZgkNeEMinrw31U3b2JS5sepG6oDG2CKpPu8OtdZMaAkzEfVTJiVoJpP2Y\n" +
            "nPziAdhFW3e0ZAnak9BPSSySRaSNmR465cG9tbqpXFKh9Rp/sCPo4Jq2n65yood\n" +
            "JBrnGr6/xhYvNa14sQ6xjJfSgRNBSXD1XXNF4kALwgZyCAECQQDV7t4bTx9FbEs5\n" +
            "36nAxPsPM6aACXaOkv6d9LXI7A0J8Zf42FeBV6RK0q7QG5iNNd1WJHSXIITUizVF\n" +
            "6aX5NnvFAkEaybeXNOwUvYtkgxF4s28s6gn11c5HZw4/a8vZm2tXXK/QftQrJVXp\n" +
            "VwxmSr0FAajWalcYN/fGkXlpWA041CKFVQJAG08ozzekeEpAuByTIOaEXgZr5MBQ\n" +
            "gBbHpgZNB18Lsw9CJSQI15wGfv6yDiLXsH8FyC9TKs+d5Tv4Cvquk0efOQJAd9OC\n" +
            "lCKFs48hdyaiZ9yEDsc57PdrvRFepVdj/gpGzD14mVerJbOiOF6aSV19ot27u4on\n" +
            "Td/3aifys0CveHzFPQJAWb4LCDwqLctfzziG7/S7Z74gyq5qZF4FUElOAZkz123E\n" +
            "yZvAdwuz/4aK0od01X9c4Jp7Mo5vQ4TvdoBnPuGo****\n" +
            "-----END RSA PRIVATE KEY-----";

        // Enter your RSA public key string. You can generate the string by using OpenSSL.
        The following code shows a sample RSA public key string.
        final String PUBLIC_X509_PEM =
            "-----BEGIN PUBLIC KEY-----\n" +
            "MIGfMA0GCSqGSIb3DQEBAQUAA4GNADCBiQKBgQCokfiAVXXf5ImFzKDw+XO/UByW\n" +
            "6mse2QsIgz3ZwBtMNU59fR5zttSx+8fB7vR4CN3bTztrP9A6bjoN0FFnhlQ3vNJC\n" +
            "5MF01PByrE/MNd5AAfSVba93I6sx8NSk5MzUCA4NJzAUqYOEwGtGBcom6kEF6MmR\n" +
            "1EKib1Id8hpooY5xaQID****\n" +
            "-----END PUBLIC KEY-----";

        // Create an RSA key pair.
        RSAPrivateKey privateKey = SimpleRSAEncryptionMaterials.getPrivateKeyFromPemPKCS1(P
        RIVATE_PKCS1_PEM);
        RSAPublicKey publicKey = SimpleRSAEncryptionMaterials.getPublicKeyFromPemX509(PUBLI
        C_X509_PEM);
        KeyPair keyPair = new KeyPair(publicKey, privateKey);
        // Specify the description of the CMK. The description cannot be modified after it
        is specified. You can specify only one description for a CMK.
        // If all objects share a CMK, you can leave the description of the CMK empty. In t
        his case, the CMK cannot be replaced.
        // If the description of the CMK is empty, the client cannot determine which CMK to
        use for decryption.
        // We recommend that you configure a description for each CMK and store the mapping
        relationship between the CMK and the description in the client. The server does not store t
        he relationship.
        Map<String, String> matDesc = new HashMap<String, String>();
        matDesc.put("desc-key", "desc-value");
        // Create RSA encryption materials.
        SimpleRSAEncryptionMaterials encryptionMaterials = new SimpleRSAEncryptionMaterials
        (keyPair, matDesc);
        // To download and decrypt objects encrypted by using other CMKs, add these CMKs an
        d their descriptions to the encryption materials.
        // encryptionMaterials.addKeyPairDescMaterial(<otherKeyPair>, <otherKeyPairMatDesc>
        );
        // Create a client for client-side encryption

```



```

// Create a client for client-side encryption.
OSSEncryptionClient ossEncryptionClient = new OSSEncryptionClientBuilder().
    build(endpoint, accessKeyId, accessKeySecret, encryptionMaterials);
try {
    // Create an UploadFileRequest object.
    UploadFileRequest uploadFileRequest = new UploadFileRequest(bucketName, objectName);

    // Specify the path of the object that you want to upload.
    uploadFileRequest.setUploadFile(localFile);
    // Specify the size of each part to upload. Valid values: 100 KB to 5 GB. The default size of each part is calculated based on the following formula: Default size of each part = Object size/10000.
    uploadFileRequest.setPartSize(100 * 1024);
    // Specify whether to enable resumable upload. By default, resumable upload is disabled.
    uploadFileRequest.setEnableCheckpoint(true);
    // Configure the checkpoint file. If you do not specify the checkpoint file, the default name localfile + ".ucp" is used. localfile + ".ucp" is in the same directory as localfile.

    This file stores the information about upload progress. If a part fails to be uploaded, the part upload continues based on the recorded progress. After the object is uploaded, the checkpoint file is deleted.
    uploadFileRequest.setCheckpointFile("test-upload.ucp");
    // Start resumable upload.
    ossEncryptionClient.uploadFile(uploadFileRequest);
} catch (OSSEException oe) {
    System.out.println("Caught an OSSEException, which means your request made it to OSS, "
        + "but was rejected with an error response for some reason.");
    System.out.println("Error Message:" + oe.getErrorMessage());
    System.out.println("Error Code:" + oe.getErrorCode());
    System.out.println("Request ID:" + oe.getRequestId());
    System.out.println("Host ID:" + oe.getHostId());
} catch (ClientException ce) {
    System.out.println("Caught an ClientException, which means the client encountered "
        + "a serious internal problem while trying to communicate with OSS, "
        + "such as not being able to access the network.");
    System.out.println("Error Message:" + ce.getMessage());
} finally {
    if (ossEncryptionClient != null) {
        ossEncryptionClient.shutdown();
    }
}
}

```

Perform RSA-based client-side encryption in resumable download

The following code provides an example on how to use a customer-managed CMK (RSA-based) to decrypt objects in resumable download:

```

import com.aliyun.oss.*;
import com.aliyun.oss.crypto.SimpleRSAEncryptionMaterials;

```

```

import com.aliyun.oss.model.*;
import java.security.KeyPair;
import java.security.interfaces.RSAPrivateKey;
import java.security.interfaces.RSAPublicKey;
import java.util.HashMap;
import java.util.Map;
public class Demo {
    public static void main(String[] args) throws Throwable {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to access OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the bucket name. Example: examplebucket.
        String bucketName = "examplebucket";
        // Specify the full path of the object. Example: exampleobject.txt. The full path of the object cannot contain the bucket name.
        String objectName = "exampleobject.txt";
        // Enter your RSA private key string. You can generate the string by using OpenSSL. The following code shows a sample RSA private key string.
        final String PRIVATE_PKCS1_PEM =
            "-----BEGIN RSA PRIVATE KEY-----\n" +
            "MIICWwIBAAKBgQCokfiAVXXf5ImFzKDw+XO/UByW6mse2QsIgz3ZwBtMNU59fR5z\n" +
            "ttSx+8fB7vR4CN3bTztrP9A6bjoN0FFnhlQ3vNJC5MF01PByrE/MNd5AAfSVba93\n" +
            "I6sx8NSk5MzUCA4NJzAUqYOEwGtGBcom6kEF6MmR1EKib1Id8hpooY5xaQIDAQAB\n" +
            "AoGAOPUZgkNeEMinrw31U3b2JS5sepG6oDG2CKpPu8OtdZMaAkzEfVTJiVoJpP2Y\n" +
            "nPziAdhFW3e0ZAnak9BPSSySRaSNmR465cG9tbqpXFKh9Rp/sCPo4Jq2n65yood\n" +
            "JBrnGr6/xhYvNa14sQ6xjJfSgrNBSXD1XXNF4kALwgZyCAECQQDV7t4bTx9FbEs5\n" +
            "36nAxPsPM6aACXaOkv6d9LXI7A0J8Zf42FeBV6RK0q7QG5iNnd1WJHSXIITUizVF\n" +
            "6aX5NnvFAkEaybeXN0wUvYtkgxF4s28s6gn11c5HZw4/a8vZm2tXXK/QftQrJVXp\n" +
            "VwxmSr0FAajWAlcYN/fGkXlpWA041CKFVQJAG08ozzekeEpAuByTIOaEXgZr5MBQ\n" +
            "gBbHpgZNB18Lsw9CJSQI15wGfv6yDiLXsH8FyC9TKs+d5Tv4CvquK0efOQJad9OC\n" +
            "lCKFs48hdyaiZ9yEDsc57PdrvRFepVdj/gpGzD14mVerJbOioF6aSV19ot27u4on\n" +
            "Td/3aifYs0CveHzFPQJAWb4LCDwqLctfzZig7/S7Z74gyq5qZF4FUElOAZkz123E\n" +
            "yZvAdwuz/4aK0od01X9c4Jp7Mo5vQ4TvdoBnPuGo****\n" +
            "-----END RSA PRIVATE KEY-----";

        // Enter your RSA public key string. You can generate the string by using OpenSSL. The following code shows a sample RSA public key string.
        final String PUBLIC_X509_PEM =
            "-----BEGIN PUBLIC KEY-----\n" +
            "MIGfMA0GCSqGSIb3DQEBAQUAA4GNADCBiQKBgQCokfiAVXXf5ImFzKDw+XO/UByW\n" +
            "6mse2QsIgz3ZwBtMNU59fR5zttSx+8fB7vR4CN3bTztrP9A6bjoN0FFnhlQ3vNJC\n" +
            "5MF01PByrE/MNd5AAfSVba93I6sx8NSk5MzUCA4NJzAUqYOEwGtGBcom6kEF6MnR\n" +
            "1EKib1Id8hpooY5xaQID****\n" +
            "-----END PUBLIC KEY-----";

        // Create an RSA key pair.
        RSAPrivateKey privateKey = SimpleRSAEncryptionMaterials.getPrivateKeyFromPemPKCS1(PRIVATE_PKCS1_PEM);
        RSAPublicKey publicKey = SimpleRSAEncryptionMaterials.getPublicKeyFromPemX509(PUBLIC_X509_PEM);
        KeyPair keyPair = new KeyPair(publicKey, privateKey);

```

```

    KeyPair keyPair = new KeyPair(publicKey, privateKey);
    // Specify the description of the CMK. The description cannot be modified after it
    // is specified. You can specify only one description for a CMK.
    // If all objects share a CMK, you can leave the description of the CMK empty. In t
    // his case, the CMK cannot be replaced.
    // If the description of the CMK is empty, the client cannot determine which CMK to
    // use for decryption.
    // We recommend that you configure a description for each CMK and store the mapping
    // relationship between the CMK and the description in the client. The server does not store t
    // he relationship.
    Map<String, String> matDesc = new HashMap<String, String>();
    matDesc.put("desc-key", "desc-value");
    // Create RSA encryption materials.
    SimpleRSAEncryptionMaterials encryptionMaterials = new SimpleRSAEncryptionMaterials
    (keyPair, matDesc);
    // To download and decrypt objects encrypted by using other CMKs, add these CMKs an
    // d their descriptions to the encryption materials.
    // encryptionMaterials.addKeyPairDescMaterial(<otherKeyPair>, <otherKeyPairMatDesc>
    );
    // Create a client for client-side encryption.
    OSSEncryptionClient ossEncryptionClient = new OSSEncryptionClientBuilder().
        build(endpoint, accessKeyId, accessKeySecret, encryptionMaterials);
    try {
        // Send a request to start resumable download in which 10 download tasks run co
        ncurrently.
        DownloadFileRequest downloadFileRequest = new DownloadFileRequest(bucketName, o
        bjectName);
        downloadFileRequest.setDownloadFile("<yourDownloadFile>");
        downloadFileRequest.setPartSize(1 * 1024 * 1024);
        downloadFileRequest.setTaskNum(10);
        downloadFileRequest.setEnableCheckpoint(true);
        downloadFileRequest.setCheckpointFile("<yourCheckpointFile>");
        // Start resumable download.
        DownloadFileResult downloadRes = ossEncryptionClient.downloadFile(downloadFileR
        equest);
    } catch (OSSEException oe) {
        System.out.println("Caught an OSSEException, which means your request made it to
        OSS, "
            + "but was rejected with an error response for some reason.");
        System.out.println("Error Message:" + oe.getErrorMessage());
        System.out.println("Error Code:" + oe.getErrorCode());
        System.out.println("Request ID:" + oe.getRequestId());
        System.out.println("Host ID:" + oe.getHostId());
    } catch (ClientException ce) {
        System.out.println("Caught an ClientException, which means the client encounter
        ed "
            + "a serious internal problem while trying to communicate with OSS, "
            + "such as not being able to access the network.");
        System.out.println("Error Message:" + ce.getMessage());
    } finally {
        if (ossEncryptionClient != null) {
            ossEncryptionClient.shutdown();
        }
    }
}

```

```
}

```

Perform RSA-based client-side encryption in range download

The following code provides an example on how to use a customer-managed CMK (RSA-based) to decrypt objects in range download:

```
import com.aliyun.oss.*;
import com.aliyun.oss.crypto.SimpleRSAEncryptionMaterials;
import com.aliyun.oss.model.*;
import java.io.BufferedReader;
import java.io.ByteArrayInputStream;
import java.io.InputStreamReader;
import java.security.KeyPair;
import java.security.interfaces.RSAPrivateKey;
import java.security.interfaces.RSAPublicKey;
import java.util.HashMap;
import java.util.Map;
public class Demo {
    public static void main(String[] args) throws Throwable {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to access OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the bucket name. Example: examplebucket.
        String bucketName = "examplebucket";
        // Specify the full path of the object. Example: exampleobject.txt. The full path of the object cannot contain the bucket name.
        String objectName = "exampleobject.txt";
        String content = "test-range-get-content-82042795hlnf12s8yhfs976y2nfoshhnsdfs235bvs\nmnhstskbcfd!";
        // Enter your RSA private key string. You can generate the string by using OpenSSL. The following code shows a sample RSA private key string.
        final String PRIVATE_PKCS1_PEM =
            "-----BEGIN RSA PRIVATE KEY-----\n" +
            "MIICWwIBAAKqCokfiAVXXf5ImFzKDw+XO/UByW6mse2QsIgz3ZwBtMNu59fR5z\n" +
            "ttSx+8fB7vR4CN3bTztrP9A6bjoN0FFnhlQ3vNJC5MF01PByrE/MNd5AAfSVba93\n" +
            "I6sx8NSk5MzUCA4NJzAUqYOEWGtGBcom6kEF6MmR1EKib1Id8hpooY5xaQIDAQAB\n" +
            "AoGAOPUZgkNeEMinrw31U3b2JS5sepG6oDG2CKpPu8OtdZMaAkzEfVTJiVoJpP2Y\n" +
            "nPziAdhFW3e0ZAnak9BPSSsySRaSNmR465cG9tbqpXFKh9Rp/sCPo4Jq2n65yood\n" +
            "JBrnGr6/xhYvNa14sQ6xjJfSgRNBSXD1XXNF4kALwgZyCAECQQDV7t4bTx9FbEs5\n" +
            "36nAxPsPM6aACXaOkv6d9LXI7A0J8Zf42FeBV6RK0q7QG5iNNd1WJHSXIITUizVF\n" +
            "6aX5NnvFAkEaybeXN0wUvYtkgxF4s28s6gn11c5HZw4/a8vZm2tXXK/QfTQrJVXp\n" +
            "VwxmSr0FAajWAlcYN/fGkX1pWA041CKFVQJAG08ozzekeEpAuByTIOaEXgZr5MBQ\n" +
            "gBbHpgZNB18Lsw9CJSQI15wGfv6yDiLXsH8FyC9TKs+d5Tv4Cvquk0efOQJAd9OC\n" +
            "lCKFs48hdyaiZ9yEDsc57PdrvRFepVdj/gpGzD14mVerJbOiOF6aSV19ot27u4on\n" +
            "Td/3aifYs0CveHzFPQJAWb4LCDwqLctfzZiG7/S7Z74gyq5qZF4FUElOAZkz123E\n" +
            "yZvAdwuz/4aK0od0lX9c4Jp7Mo5vQ4TvdoBnPuGo****\n" +
            "-----END RSA PRIVATE KEY-----";
    }
}
```

```

// Enter your RSA public key string. You can generate the string by using OpenSSL.
The following code shows a sample RSA public key string.
final String PUBLIC_X509_PEM =
    "-----BEGIN PUBLIC KEY-----\n" +
    "MIGfMA0GCSqGSIb3DQEBAQUAA4GNADCBiQKBgQCokfiAVXXf5ImFzKDw+XO/UByW\n" +
    "6mse2QsIgZ3ZwBtMnu59fR5zttSx+8fB7vR4CN3bTztrP9A6bjoN0FFnhlQ3vNJC\n" +
    "5MF01PByrE/MNd5AAfSVba93I6sx8NSk5MzUCA4NJzAUqYOEWGtGBcom6kEF6MnR\n" +
    "1EKib1Id8hpooY5xaQID****\n" +
    "-----END PUBLIC KEY-----";

// Create an RSA key pair.
RSAPrivateKey privateKey = SimpleRSAEncryptionMaterials.getPrivateKeyFromPemPKCS1(PRIVATE_PKCS1_PEM);
RSAPublicKey publicKey = SimpleRSAEncryptionMaterials.getPublicKeyFromPemX509(PUBLIC_X509_PEM);
KeyPair keyPair = new KeyPair(publicKey, privateKey);

// Specify the description of the CMK. The description cannot be modified after it
is specified. You can specify only one description for a CMK.
// If all objects share a CMK, you can leave the description of the CMK empty. In this case, the CMK cannot be replaced.
// If the description of the CMK is empty, the client cannot determine which CMK to use for decryption.
// We recommend that you configure a description for each CMK and store the mapping relationship between the CMK and the description in the client. The server does not store the relationship.
Map<String, String> matDesc = new HashMap<String, String>();
matDesc.put("desc-key", "desc-value");
// Create RSA encryption materials.
SimpleRSAEncryptionMaterials encryptionMaterials = new SimpleRSAEncryptionMaterials(keyPair, matDesc);
// To download and decrypt objects encrypted by using other CMKs, add these CMKs and their descriptions to the encryption materials.
// encryptionMaterials.addKeyPairDescMaterial(<otherKeyPair>, <otherKeyPairMatDesc>);

// Create a client for client-side encryption.
OSSEncryptionClient ossEncryptionClient = new OSSEncryptionClientBuilder().build(endpoint, accessKeyId, accessKeySecret, encryptionMaterials);
try {
    // Encrypt the object that you want to upload.
    ossEncryptionClient.putObject(bucketName, objectName, new ByteArrayInputStream(content.getBytes()));
    // Start range download.
    int start = 17;
    int end = 35;
    GetObjectRequest getObjectRequest = new GetObjectRequest(bucketName, objectName);

    getObjectRequest.setRange(start, end);
    OSSObject ossObject = ossEncryptionClient.getObject(getObjectRequest);
    BufferedReader reader = new BufferedReader(new InputStreamReader(ossObject.getObjectContent()));
    StringBuffer buffer = new StringBuffer();
    String line;
    while ((line = reader.readLine()) != null) {
        buffer.append(line);
    }
}

```

```

        reader.close();
        // Display the downloaded data.
        System.out.println("Range-Get plain text:" + content.substring(start, end + 1))
;
        System.out.println("Range-Get decrypted text: " + buffer.toString());
    } catch (OSSException oe) {
        System.out.println("Caught an OSSException, which means your request made it to
OSS, "
            + "but was rejected with an error response for some reason.");
        System.out.println("Error Message:" + oe.getErrorMessage());
        System.out.println("Error Code:" + oe.getErrorCode());
        System.out.println("Request ID:" + oe.getRequestId());
        System.out.println("Host ID:" + oe.getHostId());
    } catch (ClientException ce) {
        System.out.println("Caught an ClientException, which means the client encounter
ed "
            + "a serious internal problem while trying to communicate with OSS, "
            + "such as not being able to access the network.");
        System.out.println("Error Message:" + ce.getMessage());
    } finally {
        if (ossEncryptionClient != null) {
            ossEncryptionClient.shutdown();
        }
    }
}
}
}

```

References

For the complete sample code that is used to perform client-side encryption, visit [GitHub](#).

2.14. Data security

Object Storage Service (OSS) SDK for Java uses MD5 verification and CRC-64 to ensure data integrity when you upload, download, and copy objects.

MD5 verification

If you configure Content-MD5 in an object upload request, OSS calculates the MD5 hash of the uploaded object. If the calculated MD5 hash is different from the MD5 hash configured in the upload request, `InvalidDigest` is returned. This allows OSS to ensure data integrity for object upload. If `InvalidDigest` is returned, you must upload the object again.

The following code provides an example on how to perform MD5 verification for object upload operations:

```

import com.aliyun.oss.*;
import com.aliyun.oss.common.utils.BinaryUtil;
import com.aliyun.oss.model.ObjectMetadata;
import java.io.ByteArrayInputStream;

public class Demo {
    public static void main(String[] args) throws Throwable {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify th
e actual endpoint.
    }
}

```

```

String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
// The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
String accessKeyId = "yourAccessKeyId";
String accessKeySecret = "yourAccessKeySecret";
// Specify the name of the bucket. Example: examplebucket.
String bucketName = "examplebucket";
// Specify the full path of the object. The full path cannot contain the name of the bucket.
String objectName = "exemplerdir/object";
// Create an OSSClient instance.
OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);

try {
    // Upload a string.
    String content = "Hello OSS";
    ObjectMetadata meta = new ObjectMetadata();
    // Configure MD5 verification.
    String md5 = BinaryUtil.toBase64String(BinaryUtil.calculateMd5(content.getBytes()));

    meta.setContentMD5(md5);
    ossClient.putObject(bucketName, objectName, new ByteArrayInputStream(content.getBytes()), meta);
} catch (OSSException oe) {
    System.out.println("Caught an OSSException, which means your request made it to OSS, "
        + "but was rejected with an error response for some reason.");
    System.out.println("Error Message:" + oe.getErrorMessage());
    System.out.println("Error Code:" + oe.getErrorCode());
    System.out.println("Request ID:" + oe.getRequestId());
    System.out.println("Host ID:" + oe.getHostId());
} catch (ClientException ce) {
    System.out.println("Caught an ClientException, which means the client encountered "
        + "a serious internal problem while trying to communicate with OSS, "
        + "such as not being able to access the network.");
    System.out.println("Error Message:" + ce.getMessage());
} finally {
    if (ossClient != null) {
        ossClient.shutdown();
    }
}
}

```

 **Note** MD5 verification can be used for putObject, getObject, appendObject, postObject, and uploadPart.

CRC-64

By default, CRC-64 is enabled to ensure data integrity when you upload, download, and copy an object.

Note

- CRC-64 can be used for putObject, getObject, appendObject, and uploadPart. By default, CRC-64 is enabled when you upload an object. If the CRC-64 checksum calculated on the client is different from the CRC-64 checksum returned by the OSS server, the InconsistentException error is returned.
- CRC-64 is not supported in range download.
- CRC-64 consumes CPU resources and slows down uploads and downloads.

• CRC-64 in object download

The following code provides an example on how to perform CRC-64 for object download operations:

```
import com.aliyun.oss.*;
import com.aliyun.oss.common.utils.IOUtils;
import com.aliyun.oss.internal.OSSHeaders;
import com.aliyun.oss.internal.OSSUtils;
import com.aliyun.oss.model.GetObjectRequest;
import com.aliyun.oss.model.OSSObject;
import java.io.BufferedReader;
import java.io.InputStreamReader;

public class Demo {
    public static void main(String[] args) throws Throwable {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify
        // the actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API oper-
        // ations. Using these credentials to perform operations in OSS is a high-risk operation. We
        // recommend that you use a RAM user to call API operations or perform routine O&M. To creat
        // e a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the name of the bucket. Example: examplebucket.
        String bucketName = "examplebucket";
        // Specify the full path of the object. The full path cannot contain the name of
        // the bucket.
        String objectName = "exampledir/object";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecr
        et);
        try {
            // Download the object in streaming mode.
            GetObjectRequest getObjectRequest = new GetObjectRequest(bucketName, objectNa
            me);

            OSSObject ossObject = ossClient.getObject(bucketName, objectName);
            // Read the object from the stream to obtain clientCRC.
            System.out.println("Object content:");
            BufferedReader reader = new BufferedReader(new InputStreamReader(ossObject.ge
            tObjectContent()));
            while (true) {
```



```

        while (true) {
            String line = reader.readLine();
            if (line == null) break;
            System.out.println("\n" + line);
        }
        // You must close the obtained stream after the object is read. Otherwise, co
        nnection leaks may occur. Consequently, no connections are available and an exception occ
        urs.
        reader.close();
        // Check whether CRC-64 is enabled on the client. By default, CRC-64 is enabl
        ed on the client.
        Boolean isCrcCheckEnabled = ((OSSClient)ossClient).getClientConfiguration().isCrcCheckEnabled();
        // Check whether the download quest is a range download request. CRC-64 cannot
        be used in range downloads.
        Boolean isRangGetRequest = getObjectRequest.getHeaders().get(OSSHeaders.RANGE)
        ) != null;
        // Check whether the CRC-64 value in the returned result is the same as that
        of the downloaded object.
        if (isCrcCheckEnabled && !isRangGetRequest) {
            Long clientCRC = IOUtils.getCRCValue(ossObject.getObjectContent());
            OSSUtils.checkChecksum(clientCRC, ossObject.getServerCRC(), ossObject.get
            RequestId());
        }
    } catch (OSSEException oe) {
        System.out.println("Caught an OSSEException, which means your request made it
        to OSS, "
            + "but was rejected with an error response for some reason.");
        System.out.println("Error Message:" + oe.getErrorMessage());
        System.out.println("Error Code:" + oe.getErrorCode());
        System.out.println("Request ID:" + oe.getRequestId());
        System.out.println("Host ID:" + oe.getHostId());
    } catch (ClientException ce) {
        System.out.println("Caught an ClientException, which means the client encount
        ered "
            + "a serious internal problem while trying to communicate with OSS, "
            + "such as not being able to access the network.");
        System.out.println("Error Message:" + ce.getMessage());
    } finally {
        if (ossClient != null) {
            ossClient.shutdown();
        }
    }
}
}
}

```

- CRC-64 in append upload

The following code provides an example on how to perform CRC-64 when you upload an object by using append upload:

```

import com.aliyun.oss.ClientException;
import com.aliyun.oss.OSS;
import com.aliyun.oss.OSSClientBuilder;
import com.aliyun.oss.OSSEException;
import com.aliyun.oss.model.*;

```

```

import com.aliyun.oss.model.*;
import java.io.ByteArrayInputStream;
public class Demo {
    public static void main(String[] args) throws Exception {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify
        the actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API oper
        ations. Using these credentials to perform operations in OSS is a high-risk operation. We
        recommend that you use a RAM user to call API operations or perform routine O&M. To creat
        e a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the name of the bucket. Example: examplebucket.
        String bucketName = "examplebucket";
        // Specify the full path of the object. Example: exampleobject.txt. The full path
        cannot contain the name of the bucket.
        String objectName = "exampleobject.txt";
        // Enter the content of the first append upload, such as Hello.
        String firstAppendContent = "Hello";
        // Enter the content of the second append upload, such as World.
        String secondAppendContent = "World";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecr
        et);
        try {
            // Perform the first append operation.
            AppendObjectRequest appendObjectRequest = new AppendObjectRequest(bucketName,
            objectName, new ByteArrayInputStream(firstAppendContent.getBytes()));
            appendObjectRequest.setPosition(0L);
            // Initialize setInitCRC. After setInitCRC is initialized, OSS SDK for Java p
            erforms CRC-64 on the uploaded content by default.
            appendObjectRequest.setInitCRC(0L);
            AppendObjectResult appendObjectResult = ossClient.appendObject(appendObjectRe
            quest);
            // Perform the second append operation.
            appendObjectRequest = new AppendObjectRequest(bucketName, objectName, new Byt
            eArrayInputStream(secondAppendContent.getBytes()));
            appendObjectRequest.setPosition(appendObjectResult.getNextPosition());
            // Set the CRC-64 value to that of the uploaded object. After setInitCRC is i
            nitialized, OSS SDK for Java performs CRC-64 on the uploaded content by default.
            appendObjectRequest.setInitCRC(appendObjectResult.getClientCRC());
            ossClient.appendObject(appendObjectRequest);
        } catch (OSSEException oe) {
            System.out.println("Caught an OSSEException, which means your request made it
            to OSS, "
                + "but was rejected with an error response for some reason.");
            System.out.println("Error Message:" + oe.getErrorMessage());
            System.out.println("Error Code:" + oe.getErrorCode());
            System.out.println("Request ID:" + oe.getRequestId());
            System.out.println("Host ID:" + oe.getHostId());
        } catch (ClientException ce) {
            System.out.println("Caught an ClientException, which means the client encount
            ered "
                + "a serious internal problem while trying to communicate with OSS, "

```

```

        + "such as not being able to access the network.");
        System.out.println("Error Message:" + ce.getMessage());
    } finally {
        if (ossClient != null) {
            ossClient.shutdown();
        }
    }
}
}
}

```

References

For more information about the complete sample code for MD5 verification and CRC-64, visit [GitHub](#).

2.15. Authorized access

This topic describes how to authorize temporary access to Object Storage Service (OSS) by using Security Token Service (STS) or a signed URL.

 **Note** To generate a signed URL for access over HTTPS, set the protocol in the endpoint to HTTPS.

Use STS for temporary access authorization

You can use Alibaba Cloud STS to authorize temporary access to OSS. STS is a web service that provides temporary access tokens for users. You can use STS to grant a set of temporary access credentials that have a custom validity period and custom permissions to a third-party application or a RAM user managed by you. For more information about STS, see [What is STS?](#)

STS provides the following benefits:

- You need only to generate an access token and send the access token to a third-party application, instead of exposing your AccessKey pair to the third-party application. You can define the access permissions and validity period of this token.
- The token automatically expires after the validity period. Therefore, you do not need to manually revoke the access permissions of a token.

 **Note** For more information about how to use STS, see [Use a temporary credential provided by STS to access OSS](#) in OSS User Guide. You can call the [AssumeRole](#) operation or use STS SDKs for various programming languages to obtain temporary access credentials. For more information, see [STS SDK overview](#). The temporary access credentials consist of an AccessKey pair and a security token. The AccessKey pair contains an AccessKey ID and an AccessKey secret. The minimum validity period of temporary access credentials is 900 seconds. The maximum validity period of temporary access credentials is the maximum session duration specified for the current role. For more information, see [Specify the maximum session duration for a RAM role](#).

The following code provides an example on how to generate a signed request by using STS credentials:

```

import com.aliyun.oss.*;
import com.aliyun.oss.model.GetObjectRequest;
import com.aliyun.oss.model.PutObjectRequest;

```

```

import java.io.File;
public class Demo {
    public static void main(String[] args) throws Throwable {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the security token obtained from STS.
        String securityToken = "yourSecurityToken";
        // Specify the bucket name. Example: examplebucket.
        //String bucketName = "examplebucket";
        // Specify the full path of the object. Example: exampleobject.txt. The full path of the object cannot contain the bucket name.
        //String objectName = "exampleobject.txt";
        //String pathName = "D:\\localpath\\examplefile.txt";
        // You can use the AccessKey pair and security token obtained from STS to create an OSSClient instance.
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret, securityToken);
        try {
            // Perform operations on OSS resources, such as uploading or downloading objects.

            // Upload an object. In this example, a local file is uploaded to OSS.
            // Specify the full path of the local file. By default, if you do not specify the full path, the local file is uploaded from the path of the project to which the sample program belongs.
            //PutObjectRequest putObjectRequest = new PutObjectRequest(bucketName, objectName, new File(pathName));
            //ossClient.putObject(putObjectRequest);
            // Download an object to your local computer as a local file. If a local file has the same name as the downloaded object, the local file is overwritten by the downloaded object. Otherwise, a file is created.
            // By default, if you do not specify the path of a local file, the downloaded object is saved to the path of the project to which the sample program belongs.
            //ossClient.getObject(new GetObjectRequest(bucketName, objectName), new File(pathName));
        } catch (OSSException oe) {
            System.out.println("Caught an OSSException, which means your request made it to OSS, "
                + "but was rejected with an error response for some reason.");
            System.out.println("Error Message:" + oe.getErrorMessage());
            System.out.println("Error Code:" + oe.getErrorCode());
            System.out.println("Request ID:" + oe.getRequestId());
            System.out.println("Host ID:" + oe.getHostId());
        } catch (ClientException ce) {
            System.out.println("Caught an ClientException, which means the client encountered "
                + "a serious internal problem while trying to communicate with OSS, "
                + "such as not being able to access the network.");
        }
    }
}

```

```

        // such as not being able to access the network. //
        System.out.println("Error Message:" + ce.getMessage());
    } finally {
        if (ossClient != null) {
            ossClient.shutdown();
        }
    }
}
}
}

```

Use a signed URL for temporary access authorization

This section provides examples on how to use a signed URL to authorize temporary access.

 **Notice** A validity period must be set for an STS temporary access credential and a signed URL. When you use an STS temporary account credential to generate a signed URL that is used to perform operations such as object upload and download, the minimum validity period takes precedence. For example, you can set the validity period of your STS temporary access credential to 1,200 seconds and the validity period of your signed URL generated by using this credential to 3,600 seconds. In this case, the signed URL cannot be used to upload objects after the STS temporary access credential expires, even if the signed URL is within its validity period.

- Generate a signed URL

You can generate a signed URL and provide the URL to a visitor for temporary access. When you generate a signed URL, you can specify the validity period of the URL to limit the period of access from visitors. By default, the validity period of a signed URL is 3,600 seconds. The maximum validity period of a signed URL is 32,400 seconds.

You can add signature information to a URL and provide the URL to a third-party user for authorized access. For more information, see [Add signatures to URLs](#).

 **Notice** If you use the following code to generate a signed URL that contains the plus sign (+), you may fail to access OSS by using the URL. In this case, you must replace + in the URL with %2B .

- Generate a signed URL that can be used to perform GET operations

You can generate one or more signed URLs that can be used to perform GET operations at a time based on your requirements.

- Generate a signed URL that can be used to perform GET operations

The following code provides an example on how to generate a signed URL that can be used to perform GET operations at a time:

```

import com.aliyun.oss.*;
import java.net.URL;
import java.util.Date;
public class Demo {
    public static void main(String[] args) throws Throwable {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify
        // your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API op

```

erations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.

```
String accessKeyId = "yourAccessKeyId";
String accessKeySecret = "yourAccessKeySecret";
// Specify the security token obtained from STS.
String securityToken = "yourSecurityToken";
// Specify the bucket name. Example: examplebucket.
String bucketName = "examplebucket";
// Specify the full path of the object. Example: exampleobject.txt. The full path of the object cannot contain the bucket name.
String objectName = "exampleobject.txt";
// You can use the AccessKey pair and security token obtained from STS to create an OSSClient instance.
// Create an OSSClient instance.
OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret, securityToken);
try {
    // Specify the validity period of a signed URL. Unit: milliseconds.
    Date expiration = new Date(new Date().getTime() + 3600 * 1000);
    // Generate a signed URL that can be used to perform GET operations. Visitors can enter the URL in a browser to access specified OSS resources.
    URL url = ossClient.generatePresignedUrl(bucketName, objectName, expiration);
    System.out.println(url);
} catch (OSSEException oe) {
    System.out.println("Caught an OSSEException, which means your request made it to OSS, "
        + "but was rejected with an error response for some reason.");
    System.out.println("Error Message:" + oe.getErrorMessage());
    System.out.println("Error Code:" + oe.getErrorCode());
    System.out.println("Request ID:" + oe.getRequestId());
    System.out.println("Host ID:" + oe.getHostId());
} catch (ClientException ce) {
    System.out.println("Caught an ClientException, which means the client encountered "
        + "a serious internal problem while trying to communicate with OSS, "
        + "such as not being able to access the network.");
    System.out.println("Error Message:" + ce.getMessage());
} finally {
    if (ossClient != null) {
        ossClient.shutdown();
    }
}
}
```

- Generate multiple signed URLs that can be used to perform GET operations

The following code provides an example on how to generate multiple signed URLs that can be used to perform GET operations at a time:

```
import com.aliyun.oss.*;
import java.net.URL;
```

```

import java.util.ArrayList;
import java.util.Date;
import java.util.List;
public class Demo {
    public static void main(String[] args) throws Throwable {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the security token obtained from STS.
        String securityToken = "yourSecurityToken";
        // Specify the bucket name. Example: examplebucket.
        String bucketName = "examplebucket";
        // Specify the full path of the object. Example: exampleobject.txt. The full path of the object cannot contain the bucket name.
        // Specify the full paths of the objects to obtain the signed URLs of these objects at a time.
        String objectNameList [] = {"exampleobject.txt","exampleimage.jpg"};
        // You can use the AccessKey pair and security token obtained from STS to create an OSSClient instance.
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret, securityToken);
        try {
            // Specify the validity period of the signed URLs. Unit: milliseconds.
            Date expiration = new Date(new Date().getTime() + 3600 * 1000);
            List<URL> urlList = new ArrayList<URL>();
            for(int i=0; i<objectNameList.length; i++){
                URL url = ossClient.generatePresignedUrl(bucketName, objectNameList[i], expiration);
                urlList.add(url);
            }
            // Display the signed URLs.
            for(URL url:urlList){
                System.out.println(url);
            }
        } catch (OSSException oe) {
            System.out.println("Caught an OSSException, which means your request made it to OSS, "
                + "but was rejected with an error response for some reason.");
            System.out.println("Error Message:" + oe.getErrorMessage());
            System.out.println("Error Code:" + oe.getErrorCode());
            System.out.println("Request ID:" + oe.getRequestId());
            System.out.println("Host ID:" + oe.getHostId());
        } catch (ClientException ce) {
            System.out.println("Caught an ClientException, which means the client encountered "
                + "a serious internal problem while trying to communicate with OSS, "

```

```

        + "such as not being able to access the network.");
        System.out.println("Error Message:" + ce.getMessage());
    } finally {
        if (ossClient != null) {
            ossClient.shutdown();
        }
    }
}
}
}

```

- Generate a signed URL that can be used to perform other HTTP operations

To authorize other users to temporarily perform operations such as object upload and deletion, you must generate a signed URL that can be used to perform specific HTTP operations. For example, you can generate a signed URL that can be used to perform PUT operations to authorize users to upload objects. You can generate one or more signed URLs that can be used to perform other HTTP operations at a time based on your requirements.

- Generate a signed URL that can be used to perform other HTTP operations

The following code provides an example on how to generate a single signed URL that can be used to perform other HTTP operations at a time:

```

import com.aliyun.oss.*;
import com.aliyun.oss.common.utils.HttpHeaders;
import com.aliyun.oss.model.GeneratePresignedUrlRequest;
import java.io.ByteArrayInputStream;
import java.net.URL;
import java.util.*;
import static com.aliyun.oss.internal.OSSHeaders.OSS_USER_METADATA_PREFIX;

public class Demo {
    public static void main(String[] args) throws Throwable {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify
        // your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API op
        // erations. Using these credentials to perform operations in OSS is a high-risk operation
        // . We recommend that you use a RAM user to call API operations or perform routine O&M. T
        // o create a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the security token obtained from STS.
        String securityToken = "yourSecurityToken";
        // Specify the bucket name. Example: examplebucket.
        String bucketName = "examplebucket";
        // Specify the full path of the object. Example: exampleobject.txt. The full pa
        // th of the object cannot contain the bucket name.
        String objectName = "exampleobject.txt";
        // You can use the AccessKey pair and security token obtained from STS to creat
        // e an OSSClient instance.
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySe
        cret, securityToken);
        try {
            GeneratePresignedUrlRequest request = new GeneratePresignedUrlRequest(bucke

```



```

tname, objectname, HttpMethod.PUT);
    // Specify the validity period of a signed URL. Unit: milliseconds.
    Date expiration = new Date(new Date().getTime() + 3600 * 1000);
    request.setExpiration(expiration);
    // Set ContentType.
    request.setContentType("text/plain");
    // Set user metadata.
    request.addUserMetadata("author", "aliy");
    // Generate a signed URL.
    URL signedUrl = ossClient.generatePresignedUrl(request);
    System.out.println(signedUrl);
    Map<String, String> requestHeaders = new HashMap<String, String>();
    // Set ContentType. Make sure that the value of ContentType must be the same
    // as the content type specified when you generate the signed URL.
    requestHeaders.put(HttpHeaders.CONTENT_TYPE, "text/plain");
    // Set user metadata.
    requestHeaders.put(OSS_USER_METADATA_PREFIX + "author", "aliy");
    // Use the signed URL to upload an object.
    ossClient.putObject(signedUrl, new ByteArrayInputStream("Hello OSS".getBytes()), -1, requestHeaders, true);
} catch (OSSEException oe) {
    System.out.println("Caught an OSSEException, which means your request made it
    to OSS, "
        + "but was rejected with an error response for some reason.");
    System.out.println("Error Message:" + oe.getErrorMessage());
    System.out.println("Error Code:" + oe.getErrorCode());
    System.out.println("Request ID:" + oe.getRequestId());
    System.out.println("Host ID:" + oe.getHostId());
} catch (ClientException ce) {
    System.out.println("Caught an ClientException, which means the client encountered "
        + "a serious internal problem while trying to communicate with OSS, "
        + "such as not being able to access the network.");
    System.out.println("Error Message:" + ce.getMessage());
} finally {
    if (ossClient != null) {
        ossClient.shutdown();
    }
}
}
}

```

- Generate multiple signed URLs that can be used to perform other HTTP operations

The following code provides an example on how to generate multiple signed URLs that can be used to perform other HTTP operations at a time:

```

import com.aliyun.oss.*;
import com.aliyun.oss.common.utils.HttpHeaders;
import com.aliyun.oss.model.GeneratePresignedUrlRequest;
import java.io.File;
import java.io.FileInputStream;
import java.io.FileNotFoundException;
import java.net.URL;

```

```

import java.util.*;
import static com.aliyun.oss.internal.OSSConstants.DEFAULT_OBJECT_CONTENT_TYPE;
import static com.aliyun.oss.internal.OSSHeaders.OSS_USER_METADATA_PREFIX;
public class Demo {
    public static void main(String[] args) throws Throwable {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify
        // your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API op-
        // erations. Using these credentials to perform operations in OSS is a high-risk operation
        // . We recommend that you use a RAM user to call API operations or perform routine O&M. To
        // create a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the security token obtained from STS.
        String securityToken = "yourSecurityToken";
        // Specify the bucket name. Example: examplebucket.
        String bucketName = "examplebucket";
        // Specify the full path of the object. Example: exampleobject.txt. The full pa-
        // th of the object cannot contain the bucket name.
        // Specify the full paths of the objects to obtain the signed URLs of these ob-
        // jects at a time.
        String objectNameList [] = {"exampleobject.txt", "exampleimage.jpg"};
        String uploadNameArray [] = {"D:\\localpath\\examplefile1.txt", "D:\\localpath\\
        examplefile2.jpg"};
        // You can use the AccessKey pair and security token obtained from STS to creat-
        // e an OSSClient instance.
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySe-
        cret, securityToken);
        try {
            // Specify the validity period of the signed URLs. Unit: milliseconds.
            Date expiration = new Date(new Date().getTime() + 3600 * 1000);
            for(int i=0; i<objectNameList.length; i++){
                GeneratePresignedUrlRequest request = new GeneratePresignedUrlRequest(b-
                ucketName, objectNameList[i], HttpMethod.PUT);
                request.setExpiration(expiration);
                // Set ContentType.
                request.setContentType(DEFAULT_OBJECT_CONTENT_TYPE);
                // Set user metadata.
                request.addUserMetadata("author", "aliy");
                // Generate signed URLs.
                URL signedUrl = ossClient.generatePresignedUrl(request);
                // Display the signed URLs.
                System.out.println(signedUrl);
                Map<String, String> requestHeaders = new HashMap<String, String>();
                requestHeaders.put(HttpHeaders.CONTENT_TYPE, DEFAULT_OBJECT_CONTENT_TYP-
                E);
                requestHeaders.put(OSS_USER_METADATA_PREFIX + "author", "aliy");
                // If you want to upload a string, use the following method:
                //ossClient.putObject(signedUrl, new ByteArrayInputStream("Hello OSS".g-
                etBytes()), -1, requestHeaders, true);
                // Use one of the signed URLs to upload an object.
                try {
                    ossClient.putObject(signedUrl, new FileInputStream(new File(uploadN

```

```

        ossClient.putObject(signedUrl, new FileInputStream(new File(uploadNameArray[i])), -1, requestHeaders, true);
    } catch (FileNotFoundException e) {
        e.printStackTrace();
    }
}
} catch (OSSException oe) {
    System.out.println("Caught an OSSException, which means your request made it to OSS, "
        + "but was rejected with an error response for some reason.");
    System.out.println("Error Message:" + oe.getErrorMessage());
    System.out.println("Error Code:" + oe.getErrorCode());
    System.out.println("Request ID:" + oe.getRequestId());
    System.out.println("Host ID:" + oe.getHostId());
} catch (ClientException ce) {
    System.out.println("Caught an ClientException, which means the client encountered "
        + "a serious internal problem while trying to communicate with OSS, "
        + "such as not being able to access the network.");
    System.out.println("Error Message:" + ce.getMessage());
} finally {
    if (ossClient != null) {
        ossClient.shutdown();
    }
}
}
}

```

Visitors can specify the `HttpMethod.PUT` parameter and use the signed URLs to upload objects.

- Generate one or more signed URLs that contain specified parameters

You can generate one or more signed URLs that contain specified parameters at a time based on your requirements.

- Generate a signed URL that contains specified parameters

The following code provides an example on how to generate a signed URL that contains specified parameters:

```

import com.aliyun.oss.*;
import com.aliyun.oss.model.GeneratePresignedUrlRequest;
import java.net.URL;
import java.util.*;
public class Demo {
    public static void main(String[] args) throws Throwable {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the security token obtained from STS
    }
}

```

```

// Specify the security token obtained from STS.
String securityToken = "yourSecurityToken";
// Specify the bucket name. Example: examplebucket.
String bucketName = "examplebucket";
// Specify the full path of the object. Example: exampleobject.txt. The full path of the object cannot contain the bucket name.
String objectName = "exampleobject.txt";
// You can use the AccessKey pair and security token obtained from STS to create an OSSClient instance.
// Create an OSSClient instance.
OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret, securityToken);
try {
    // Create a request.
    GeneratePresignedUrlRequest generatePresignedUrlRequest = new GeneratePresignedUrlRequest(bucketName, objectName);
    // Set HttpMethod to PUT.
    generatePresignedUrlRequest.setMethod(HttpMethod.PUT);
    // Add user metadata.
    generatePresignedUrlRequest.addUserMetadata("author", "baymax");
    // Set ContentType.
    generatePresignedUrlRequest.setContentType("application/txt");
    // Specify the validity period of a signed URL. Unit: milliseconds.
    Date expiration = new Date(new Date().getTime() + 3600 * 1000);
    generatePresignedUrlRequest.setExpiration(expiration);
    // Generate a signed URL.
    URL url = ossClient.generatePresignedUrl(generatePresignedUrlRequest);
    System.out.println(url);
} catch (OSSEException oe) {
    System.out.println("Caught an OSSEException, which means your request made it to OSS, "
        + "but was rejected with an error response for some reason.");
    System.out.println("Error Message:" + oe.getErrorMessage());
    System.out.println("Error Code:" + oe.getErrorCode());
    System.out.println("Request ID:" + oe.getRequestId());
    System.out.println("Host ID:" + oe.getHostId());
} catch (ClientException ce) {
    System.out.println("Caught an ClientException, which means the client encountered "
        + "a serious internal problem while trying to communicate with OSS, "
        + "such as not being able to access the network.");
    System.out.println("Error Message:" + ce.getMessage());
} finally {
    if (ossClient != null) {
        ossClient.shutdown();
    }
}
}

```

- Generate multiple signed URLs that contain specified parameters

The following code provides an example on how to generate multiple signed URLs that contain specified parameters:

```

import com.aliyun.oss.*;
import com.aliyun.oss.model.GeneratePresignedUrlRequest;
import java.net.URL;
import java.util.*;
public class Demo {
    public static void main(String[] args) throws Throwable {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the security token obtained from STS.
        String securityToken = "yourSecurityToken";
        // Specify the bucket name. Example: examplebucket.
        String bucketName = "examplebucket";
        // Specify the full path of the object. Example: exampleobject.txt. The full path of the object cannot contain the bucket name.
        // Specify the full paths of the objects to obtain the signed URLs of these objects at a time.
        String objectNameList [] = {"exampleobject.txt","exampleimage.jpg"};
        // You can use the AccessKey pair and security token obtained from STS to create an OSSClient instance.
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret, securityToken);
        try {
            // Specify the validity period of the signed URLs. Unit: milliseconds.
            Date expiration = new Date(new Date().getTime() + 3600 * 1000);
            for(int i=0; i<objectNameList.length; i++){
                // Create a request.
                GeneratePresignedUrlRequest generatePresignedUrlRequest = new GeneratePresignedUrlRequest(bucketName, objectNameList[i]);
                // Set HttpMethod to PUT.
                generatePresignedUrlRequest.setMethod(HttpMethod.PUT);
                // Add user metadata.
                generatePresignedUrlRequest.addUserMetadata("author", "baymax");
                // Set ContentType.
                generatePresignedUrlRequest.setContentType("application/txt");
                generatePresignedUrlRequest.setExpiration(expiration);
                // Generate signed URLs.
                URL url = ossClient.generatePresignedUrl(generatePresignedUrlRequest);
                // Display the signed URLs.
                System.out.println(url);
            }
        } catch (OSSException oe) {
            System.out.println("Caught an OSSException, which means your request made it to OSS, "
                + "but was rejected with an error response for some reason.");
            System.out.println("Error Message:" + oe.getErrorMessage());
            System.out.println("Error Code:" + oe.getErrorCode());
        }
    }
}

```

```

        System.out.println("Request ID:" + oe.getRequestId());
        System.out.println("Host ID:" + oe.getHostId());
    } catch (ClientException ce) {
        System.out.println("Caught an ClientException, which means the client encountered "
            + "a serious internal problem while trying to communicate with OSS, "
            + "such as not being able to access the network.");
        System.out.println("Error Message:" + ce.getMessage());
    } finally {
        if (ossClient != null) {
            ossClient.shutdown();
        }
    }
}
}

```

- Use a signed URL to upload or download an object
 - Use a signed URL to upload an object

The following code provides an example on how to use a signed URL to upload an object:

```

import com.aliyun.oss.*;
import com.aliyun.oss.common.utils.DateUtil;
import com.aliyun.oss.model.GeneratePresignedUrlRequest;
import com.aliyun.oss.model.PutObjectResult;
import java.io.File;
import java.io.FileInputStream;
import java.net.URL;
import java.util.*;

public class Demo {
    public static void main(String[] args) throws Throwable {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";

        // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the security token obtained from STS.
        String securityToken = "yourSecurityToken";
        // Specify the bucket name. Example: examplebucket.
        String bucketName = "examplebucket";
        // Specify the full path of the object. Example: exampleobject.txt. The full path of the object cannot contain the bucket name.
        String objectName = "exampleobject.txt";
        // Specify the full path of the local file. By default, if you do not specify the full path, the local file is uploaded from the path of the project to which the sample program belongs.
        String pathName = "D:\\\\localpath\\\\examplefile.txt";
        // You can use the AccessKey pair and security token obtained from STS to create an OSSClient instance.
    }
}

```

```

        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret, securityToken);
        try {
            // Specify the expiration time of a signed URL.
            Date expiration = null;
            expiration = DateUtil.parseRfc822Date("Wed, 18 May 2022 14:20:00 GMT");
            // Generate a signed URL.
            GeneratePresignedUrlRequest request = new GeneratePresignedUrlRequest(bucketName, objectName, HttpMethod.PUT);
            // Specify the expiration time of the request.
            request.setExpiration(expiration);
            // Set ContentType.
            request.setContentType("application/txt");
            // Add user metadata.
            request.addUserMetadata("author", "aliy");
            // Generate a signed URL that allows HTTP PUT requests.
            URL signedUrl = ossClient.generatePresignedUrl(request);
            System.out.println("signed url for putObject: " + signedUrl);
            // Use the signed URL to send a request.
            File f = new File(pathName);
            FileInputStream fin = null;
            fin = new FileInputStream(f);
            // Add headers to a PutObject request.
            Map<String, String> customHeaders = new HashMap<String, String>();
            customHeaders.put("Content-Type", "application/txt");
            customHeaders.put("x-oss-meta-author", "aliy");
            PutObjectResult result = ossClient.putObject(signedUrl, fin, f.length(), customHeaders);
        } catch (OSSException oe) {
            System.out.println("Caught an OSSException, which means your request made it to OSS, "
                + "but was rejected with an error response for some reason.");
            System.out.println("Error Message:" + oe.getErrorMessage());
            System.out.println("Error Code:" + oe.getErrorCode());
            System.out.println("Request ID:" + oe.getRequestId());
            System.out.println("Host ID:" + oe.getHostId());
        } catch (ClientException ce) {
            System.out.println("Caught an ClientException, which means the client encountered "
                + "a serious internal problem while trying to communicate with OSS, "
                + "such as not being able to access the network.");
            System.out.println("Error Message:" + ce.getMessage());
        } finally {
            if (ossClient != null) {
                ossClient.shutdown();
            }
        }
    }
}

```

- Use a signed URL to download an object

The following code provides an example on how to use a signed URL to download a specified object:

```
import com.aliyun.oss.*;
import com.aliyun.oss.common.utils.DateUtil;
import com.aliyun.oss.model.GeneratePresignedUrlRequest;
import com.aliyun.oss.model.OSSObject;
import java.net.URL;
import java.util.*;

public class Demo {
    public static void main(String[] args) throws Throwable {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify your actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the security token obtained from STS.
        String securityToken = "yourSecurityToken";
        // Specify the bucket name. Example: examplebucket.
        String bucketName = "examplebucket";
        // Specify the full path of the object. Example: exampleobject.txt. The full path of the object cannot contain the bucket name.
        String objectName = "exampleobject.txt";
        // You can use the AccessKey pair and security token obtained from STS to create an OSSClient instance.
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret, securityToken);
        try {
            // Specify the expiration time of a signed URL.
            Date expiration = null;
            expiration = DateUtil.parseRfc822Date("Wed, 18 May 2022 14:20:00 GMT");
            // Generate a signed URL.
            GeneratePresignedUrlRequest request = new GeneratePresignedUrlRequest(bucketName, objectName, HttpMethod.GET);
            // Specify the expiration time of the request.
            request.setExpiration(expiration);
            // Generate a signed URL that allows HTTP GET requests.
            URL signedUrl = ossClient.generatePresignedUrl(request);
            System.out.println("signed url for getObject: " + signedUrl);
            // Use the signed URL to send a request.
            Map<String, String> customHeaders = new HashMap<String, String>();
            // Add headers to a GetObject request.
            customHeaders.put("Range", "bytes=100-1000");
            OSSObject object = ossClient.getObject(signedUrl, customHeaders);
        } catch (OSSException oe) {
            System.out.println("Caught an OSSException, which means your request made it to OSS, "
                + "but was rejected with an error response for some reason.");
            System.out.println("Error Message:" + oe.getErrorMessage());
        }
    }
}
```



```

        System.out.println("Error Code:" + oe.getErrorCode());
        System.out.println("Request ID:" + oe.getRequestId());
        System.out.println("Host ID:" + oe.getHostId());
    } catch (ClientException ce) {
        System.out.println("Caught an ClientException, which means the client encountered "
            + "a serious internal problem while trying to communicate with OSS, "
            + "such as not being able to access the network.");
        System.out.println("Error Message:" + ce.getMessage());
    } finally {
        if (ossClient != null) {
            ossClient.shutdown();
        }
    }
}
}

```

2.16. IMG

Image Processing (IMG) provided by Object Storage Service (OSS) is a secure, cost-effective, and highly reliable image processing service that can process large numbers of images. After you upload source images to OSS, you can call RESTful API operations to process the images on a device that is connected over the Internet anytime, anywhere.

Use IMG parameters to process images

- Use a single IMG parameter to process an image

```

import com.aliyun.oss.*;
import com.aliyun.oss.model.GetObjectRequest;
import java.io.File;
public class Demo {
    public static void main(String[] args) throws Throwable {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify
        the actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to access OSS is a high-risk operation. We recommend that
        you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the name of the bucket. Example: examplebucket.
        String bucketName = "examplebucket";
        // Specify the full path of the object. The full path cannot contain the name of
        the bucket.
        String objectName = "exampleobject.jpg";
        // Specify the full path to which you want to save the processed image. Example:
        D:\\localpath\\example-resize.jpg. If a file with the same name already exists in the path, the processed image overwrites the file. Otherwise, a new file is created.
        String localPath = "D:\\localpath\\example-resize.jpg";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);
    }
}

```

```

et);
    try {
        // Resize the image to a height and width of 100 pixels.
        String style = "image/resize,m_fixed,w_100,h_100";
        GetObjectRequest request = new GetObjectRequest(bucketName, objectName);
        request.setProcess(style);
        // Name the processed image example-resize.jpg and save the image to your local computer.
        // By default, if you only specify the name of a local file such as example-resize.jpg without specifying the local path of the file, the processed image is saved to the local path of the project to which the sample program belongs.
        ossClient.getObject(request, new File(localPath));
    } catch (OSSException oe) {
        System.out.println("Caught an OSSException, which means your request made it to OSS, "
            + "but was rejected with an error response for some reason.");
        System.out.println("Error Message:" + oe.getErrorMessage());
        System.out.println("Error Code:" + oe.getErrorCode());
        System.out.println("Request ID:" + oe.getRequestId());
        System.out.println("Host ID:" + oe.getHostId());
    } catch (ClientException ce) {
        System.out.println("Caught an ClientException, which means the client encountered "
            + "a serious internal problem while trying to communicate with OSS, "
            + "such as not being able to access the network.");
        System.out.println("Error Message:" + ce.getMessage());
    } finally {
        if (ossClient != null) {
            ossClient.shutdown();
        }
    }
}
}

```

- Use different IMG parameters to process an image

```

import com.aliyun.oss.*;
import com.aliyun.oss.model.GetObjectRequest;
import java.io.File;
public class Demo {
    public static void main(String[] args) throws Throwable {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify the actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to access OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the name of the bucket. Example: examplebucket.
        String bucketName = "examplebucket";
        // Specify the full path of the object. The full path cannot contain the name of the bucket.
        String objectName = "exampleobject.png";
    }
}

```

```

String objectName = "exampleObject.jpg";
// Create an OSSClient instance.
OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);

try {
    // Resize the image to a height and width of 100 pixels.
    String style = "image/resize,m_fixed,w_100,h_100";
    GetObjectRequest request = new GetObjectRequest(bucketName, objectName);
    request.setProcess(style);
    // Name the processed image example-resize.jpg and save the image to your local computer.
    // Specify the full path to which you want to save the processed image. Example: D:\\localpath\\example-resize.jpg. If a file with the same name already exists in the path, the processed image overwrites the file. Otherwise, a new file is created.
    // By default, if you only specify the name of a local file such as example-resize.jpg without specifying the local path of the file, the processed image is saved to the local path of the project to which the sample program belongs.
    ossClient.getObject(request, new File("D:\\localpath\\example-resize.jpg"));
    // Crop the image to a height and width of 100 pixels by setting the coordinate pair to (100, 100).
    style = "image/crop,w_100,h_100,x_100,y_100";
    request = new GetObjectRequest(bucketName, objectName);
    request.setProcess(style);
    // Name the processed image example-crop.jpg and save the image to your local computer.
    ossClient.getObject(request, new File("D:\\localpath\\example-crop.jpg"));
    // Rotate the image 90 degrees.
    style = "image/rotate,90";
    request = new GetObjectRequest(bucketName, objectName);
    request.setProcess(style);
    // Name the processed image example-rotate.jpg and save the image to your local computer.
    ossClient.getObject(request, new File("D:\\localpath\\example-rotate.jpg"));
    // Add a text watermark to the image.
    // After the text watermark content is encoded in Base64, replace plus signs (+) in the encoded result with hyphens (-) and forward slashes (/) with underscores (_), and remove equal signs (=) at the end. Then, you can obtain the watermark string.
    // Specify Hello World as the content of the text watermark, and encode the text content. Then, you can obtain the SGVsbG8gV29ybGQ string.
    style = "image/watermark,text_SGVsbG8gV29ybGQ";
    request = new GetObjectRequest(bucketName, objectName);
    request.setProcess(style);
    // Name the processed image example-watermarktext.jpg and save the image to your local computer.
    ossClient.getObject(request, new File("D:\\localpath\\example-watermarktext.jpg"));
    // Add an image watermark to the image. Make sure that the watermark image is stored in the bucket that stores the source image.
    // After the full path of the watermark image is encoded in Base64, replace plus signs (+) in the encoded result with hyphens (-) and forward slashes (/) with underscores (_), and remove equal signs (=) at the end. Then, you can obtain the watermark string.
    // Specify panda.jpg as the full path of the watermark image, and encode the full path. Then, you can obtain the cGFuZGEuanBn string.
    style = "image/watermark,image_cGFuZGEuanBn";

```

```

        request = new GetObjectRequest(bucketName, objectName);
        request.setProcess(style);
        // Name the processed image example-watermarkimage.jpg and save the image to
        your local computer.
        ossClient.getObject(request, new File("D:\\localpath\\example-watermarkimage.
        jpg"));
    } catch (OSSException oe) {
        System.out.println("Caught an OSSException, which means your request made it
        to OSS, "
            + "but was rejected with an error response for some reason.");
        System.out.println("Error Message:" + oe.getErrorMessage());
        System.out.println("Error Code:" + oe.getErrorCode());
        System.out.println("Request ID:" + oe.getRequestId());
        System.out.println("Host ID:" + oe.getHostId());
    } catch (ClientException ce) {
        System.out.println("Caught an ClientException, which means the client encount
        ered "
            + "a serious internal problem while trying to communicate with OSS, "
            + "such as not being able to access the network.");
        System.out.println("Error Message:" + ce.getMessage());
    } finally {
        if (ossClient != null) {
            ossClient.shutdown();
        }
    }
}
}

```

- Use multiple IMG parameters to process an image at the same time

The following code provides an example on how to use multiple IMG parameters to process an image at the same time. Separate the IMG parameters with forward slashes (/).

```

import com.aliyun.oss.*;
import com.aliyun.oss.model.GetObjectRequest;
import java.io.File;
public class Demo {
    public static void main(String[] args) throws Throwable {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify
        the actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API oper
        ations. Using these credentials to access OSS is a high-risk operation. We recommend that
        you use a RAM user to call API operations or perform routine O&M. To create a RAM user, l
        og on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the name of the bucket. Example: examplebucket.
        String bucketName = "examplebucket";
        // Specify the full path of the object. The full path cannot contain the name of
        the bucket.
        String objectName = "exampleobject.jpg";
        // Specify the full path to which you want to save the processed image. Example:
        D:\\localpath\\example-new.jpg. If a file with the same name already exists in the path,
        the processed image overwrites the file. Otherwise, a new file is created.
    }
}

```

```

        String pathName = "D:\\localpath\\example-new.jpg";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);

        try {
            // Resize the image to a height and width of 100 pixels, and rotate the image 90 degrees.
            String style = "image/resize,m_fixed,w_100,h_100/rotate,90";
            GetObjectRequest request = new GetObjectRequest(bucketName, objectName);
            request.setProcess(style);
            // Name the processed image example-new.jpg and save the image to your local computer.
            // By default, if you only specify the name of a local file such as example-new.jpg without specifying the local path of the file, the processed image is saved to the local path of the project to which the sample program belongs.
            ossClient.getObject(request, new File("D:\\localpath\\example-new.jpg"));
        } catch (OSSException oe) {
            System.out.println("Caught an OSSException, which means your request made it to OSS, "
                + "but was rejected with an error response for some reason.");
            System.out.println("Error Message:" + oe.getErrorMessage());
            System.out.println("Error Code:" + oe.getErrorCode());
            System.out.println("Request ID:" + oe.getRequestId());
            System.out.println("Host ID:" + oe.getHostId());
        } catch (ClientException ce) {
            System.out.println("Caught an ClientException, which means the client encountered "
                + "a serious internal problem while trying to communicate with OSS, "
                + "such as not being able to access the network.");
            System.out.println("Error Message:" + ce.getMessage());
        } finally {
            if (ossClient != null) {
                ossClient.shutdown();
            }
        }
    }
}

```

Use an image style to process an image

You can create an image style in the OSS console and encapsulate multiple IMG parameters in the style. Then, you can use the style to process images. For more information, see [Image styles](#).

The following code provides an example on how to use an image style to process an image:

```

import com.aliyun.oss.*;
import com.aliyun.oss.model.GetObjectRequest;
import java.io.File;
public class Demo {
    public static void main(String[] args) throws Throwable {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify the actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to access OSS is a high-risk operation. We recommend that you
    }
}

```

tions. Using these credentials to access OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.

```
String accessKeyId = "yourAccessKeyId";
String accessKeySecret = "yourAccessKeySecret";
// Specify the name of the bucket. Example: examplebucket.
String bucketName = "examplebucket";
// Specify the full path of the object. The full path cannot contain the name of the bucket.
String objectName = "exampleobject.jpg";
// Specify the full path to which you want to save the processed image. Example: D:\\localpath\\example-new.jpg. If a file with the same name already exists in the path, the processed image overwrites the file. Otherwise, a new file is created.
String pathName = "D:\\localpath\\example-new.jpg";
// Create an OSSClient instance.
OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);

try {
    // Use the custom style to process the image.
    // Set yourCustomStyleName to the name of the image style that you create in the OSS console.
    String style = "style/yourCustomStyleName";
    GetObjectRequest request = new GetObjectRequest(bucketName, objectName);
    request.setProcess(style);
    // Name the processed image example-new.jpg and save the image to your local computer.
    // By default, if you only specify the name of a local file such as example-new.jpg without specifying the local path of the file, the processed image is saved to the local path of the project to which the sample program belongs.
    ossClient.getObject(request, new File(pathName));
} catch (OSSException oe) {
    System.out.println("Caught an OSSException, which means your request made it to OSS, "
        + "but was rejected with an error response for some reason.");
    System.out.println("Error Message:" + oe.getErrorMessage());
    System.out.println("Error Code:" + oe.getErrorCode());
    System.out.println("Request ID:" + oe.getRequestId());
    System.out.println("Host ID:" + oe.getHostId());
} catch (ClientException ce) {
    System.out.println("Caught an ClientException, which means the client encountered "
        + "a serious internal problem while trying to communicate with OSS, "
        + "such as not being able to access the network.");
    System.out.println("Error Message:" + ce.getMessage());
} finally {
    if (ossClient != null) {
        ossClient.shutdown();
    }
}
}
```

Save processed images

By default, IMG does not save processed images. You can call the `ImgSaveAs` operation to save the images to the bucket in which the source images are stored.

The following code provides an example on how to save a processed image:

```
import com.aliyun.oss.*;
import com.aliyun.oss.common.utils.BinaryUtil;
import com.aliyun.oss.common.utils.IOUtils;
import com.aliyun.oss.model.GenericResult;
import com.aliyun.oss.model.ProcessObjectRequest;
import java.util.Formatter;

public class Demo {
    public static void main(String[] args) throws Throwable {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify the actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to access OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the name of the bucket. Example: examplebucket.
        String bucketName = "examplebucket";
        // Specify the full path of the object. The full path cannot contain the name of the bucket.
        String sourceImage = "exampleimage.png";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);

        try {
            // Resize the image to a height and width of 100 pixels.
            StringBuilder sbStyle = new StringBuilder();
            Formatter styleFormatter = new Formatter(sbStyle);
            String styleType = "image/resize,m_fixed,w_100,h_100";
            // Name the processed image example-resize.png and save the image to the bucket in which the source image is stored.
            // Specify the full path of the object. The full path cannot contain the name of the bucket.
            String targetImage = "example-resize.png";
            styleFormatter.format("%s|sys/saveas,o_%s,b_%s", styleType,
                BinaryUtil.toBase64String(targetImage.getBytes()),
                BinaryUtil.toBase64String(bucketName.getBytes()));
            System.out.println(sbStyle.toString());
            ProcessObjectRequest request = new ProcessObjectRequest(bucketName, sourceImage, sbStyle.toString());
            GenericResult processResult = ossClient.processObject(request);
            String json = IOUtils.readStreamAsString(processResult.getResponse().getContent(), "UTF-8");
            processResult.getResponse().getContent().close();
            System.out.println(json);
        } catch (OSSException oe) {
            System.out.println("Caught an OSSException, which means your request made it to OSS, "
                + "but was rejected with an error response for some reason.");
        }
    }
}
```

```

        System.out.println("Error Message:" + oe.getErrorMessage());
        System.out.println("Error Code:" + oe.getErrorCode());
        System.out.println("Request ID:" + oe.getRequestId());
        System.out.println("Host ID:" + oe.getHostId());
    } catch (ClientException ce) {
        System.out.println("Caught an ClientException, which means the client encounter
ed "
            + "a serious internal problem while trying to communicate with OSS, "
            + "such as not being able to access the network.");
        System.out.println("Error Message:" + ce.getMessage());
    } finally {
        if (ossClient != null) {
            ossClient.shutdown();
        }
    }
}
}
}

```

Generate a signed URL that includes IMG parameters for an object

URLs of private objects must be signed. IMG parameters cannot be added to a signed URL. If you want to process a private object, add IMG parameters to the signature. The following code provides an example on how to add IMG parameters to the signature:

```

import com.aliyun.oss.*;
import com.aliyun.oss.model.GeneratePresignedUrlRequest;
import java.net.URL;
import java.util.Date;
public class Demo {
    public static void main(String[] args) throws Throwable {
        // In this example, the endpoint of the China (Hangzhou) region is used. Specify the
        // actual endpoint.
        String endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
        // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations.
        // Using these credentials to access OSS is a high-risk operation. We recommend that you
        // use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on
        // to the RAM console.
        String accessKeyId = "yourAccessKeyId";
        String accessKeySecret = "yourAccessKeySecret";
        // Specify the name of the bucket. Example: examplebucket.
        String bucketName = "examplebucket";
        // Specify the full path of the object. The full path cannot contain the name of the
        // bucket.
        String objectName = "exampleobject.jpg";
        // Create an OSSClient instance.
        OSS ossClient = new OSSClientBuilder().build(endpoint, accessKeyId, accessKeySecret);

        try {
            // Resize the image to a height and width of 100 pixels, and rotate the image 90
            // degrees.
            String style = "image/resize,m_fixed,w_100,h_100/rotate,90";
            // Set the validity period of the signed URL to 10 minutes.
            Date expiration = new Date(new Date().getTime() + 1000 * 60 * 10);
            GeneratePresignedUrlRequest req = new GeneratePresignedUrlRequest(bucketName, objectName,

```



```

        generatePresignedUrlRequest req = new generatePresignedUrlRequest(bucketName, o
bjectName, HttpMethod.GET);
        req.setExpiration(expiration);
        req.setProcess(style);
        URL signedUrl = ossClient.generatePresignedUrl(req);
        System.out.println(signedUrl);
    } catch (OSSException oe) {
        System.out.println("Caught an OSSException, which means your request made it to
OSS, "
            + "but was rejected with an error response for some reason.");
        System.out.println("Error Message:" + oe.getErrorMessage());
        System.out.println("Error Code:" + oe.getErrorCode());
        System.out.println("Request ID:" + oe.getRequestId());
        System.out.println("Host ID:" + oe.getHostId());
    } catch (ClientException ce) {
        System.out.println("Caught an ClientException, which means the client encounter
ed "
            + "a serious internal problem while trying to communicate with OSS, "
            + "such as not being able to access the network.");
        System.out.println("Error Message:" + ce.getMessage());
    } finally {
        if (ossClient != null) {
            ossClient.shutdown();
        }
    }
}
}

```

References

- For more information about the supported IMG parameters, see [Parameters](#).
- For more information about the complete sample code for IMG, visit [GitHub](#).

2.17. Error handling

OSS SDK for Java exceptions are classified into two types: `OSSException` and `ClientException`. Both are subclasses of `RuntimeException`.

Example of handling exceptions

The following code provides an example on how to handle exceptions:

```
try {
    // Perform operations on OSS such as object upload.
    ossClient.putObject(...);
} catch (OSSException oe) {
    System.out.println("Caught an OSSException, which means your request made it to OSS, "
        + "but was rejected with an error response for some reason." );
    System.out.println("Error Message: " + oe.getErrorMessage());
    System.out.println("Error Code:      " + oe.getErrorCode());
    System.out.println("Request ID:    " + oe.getRequestId());
    System.out.println("Host ID:      " + oe.getHostId());
} catch (ClientException ce) {
    System.out.println("Caught an ClientException, which means the client encountered "
        + "a serious internal problem while trying to communicate with OSS, "
        + "such as not being able to access the network." );
    System.out.println("Error Message: " + ce.getMessage());
} finally {
    if (ossClient != null) {
        ossClient.shutdown();
    }
}
```

ClientException

A `ClientException` indicates an exception that occurs when a client sends a request or transmit data to OSS. For example, a `ClientException` is returned when a request is sent under poor network conditions. A `ClientException` is also returned when an I/O exception occurs during object upload.

OSSException

`OSSException`: indicates a server exception that arises from the resolution of a server error message. A `ServiceException` includes the error code and message returned by OSS so that you can identify and resolve the error.

`OSSException` involves the following error information:

Parameter	Description
Code	The error code returned by OSS.
Message	The detailed error message returned by OSS.
RequestId	The UUID used to uniquely identify the request. If the problem persists, you can provide the request ID of a problem to the OSS development engineers for help.
HostId	The ID of the host in the accessed OSS cluster, which is the same as the host ID specified in the request.

Common OSS error codes

Error code	Description	HTTP status code
AccessDenied	The error message returned because access is denied.	403
BucketAlreadyExists	The error message returned when the bucket already exists.	409
BucketNotEmpty	The error message returned when the bucket is not empty.	409
EntityTooLarge	The error message returned because the entity is too large.	400
EntityTooSmall	The error message returned because the entity is too small.	400
FileGroupTooLarge	The error message returned because the object group is too large.	400
FilePartNotExist	The error message returned when the part does not exist.	400
FilePartStale	The error message returned when the part is expired.	400
InvalidArgument	The error message returned when the parameter format is invalid.	400
InvalidAccessKeyId	The error message returned because AccessKey ID does not exist.	403
InvalidBucketName	The error message returned when the bucket name is invalid.	400
InvalidDigest	The error message returned when the digest is invalid.	400
InvalidObjectName	The error message returned when the object name is invalid.	400
InvalidPart	The error message returned when the part is invalid.	400
InvalidPartOrder	The error message returned when the part order is invalid.	400
InvalidTargetBucketForLogging	The error message returned because the target bucket specified for logging is invalid.	400

Error code	Description	HTTP status code
InternalServerError	The error message returned when an internal OSS error occurs.	500
MalformedXML	The error message returned when the XML format is invalid.	400
MethodNotAllowed	The error message returned when the method is not supported.	405
MissingArgument	The error message returned when a parameter is missing.	411
MissingContentLength	The error message returned when the content length is missing.	411
NoSuchBucket	The error message returned when no such bucket exists.	404
NoSuchKey	The error message returned because no such object exists.	404
NoSuchUpload	The error message returned when no such multipart upload ID exists.	404
NotImplemented	The error message returned when the method cannot be implemented.	501
PreconditionFailed	The error message returned when a precondition error occurs.	412
RequestTimeTooSkewed	The error message returned when the time deviation of the OSS client and OSS server exceeds 15 minutes.	403
RequestTimeout	The error message returned because the request has timed out.	400
SignatureDoesNotMatch	The error message returned when a signature error occurs.	403
InvalidEncryptionAlgorithmError	The error message returned because the specified encryption algorithm based on entropy encoding is invalid.	400

2.18. FAQ

This topic lists the causes and solutions for common errors you may encounter when you use Object Storage Service (OSS) SDK for Java.

JAR conflicts

- Cause

If a similar output is displayed when you use OSS SDK for Java, your project has a JAR conflict.

```
Exception in thread "main" java.lang.NoClassDefFoundError: org/apache/http/ssl/TrustStrategy
    at com.aliyun.oss.OSSClient.<init>(OSSClient.java:268)
    at com.aliyun.oss.OSSClient.<init>(OSSClient.java:193)
    at com.aliyun.oss.demo.HelloOSS.main(HelloOSS.java:77)
Caused by: java.lang.ClassNotFoundException: org.apache.http.ssl.TrustStrategy
    at java.net.URLClassLoader$1.run(URLClassLoader.java:366)
    at java.net.URLClassLoader$1.run(URLClassLoader.java:355)
    at java.security.AccessController.doPrivileged(Native Method)
    at java.net.URLClassLoader.findClass(URLClassLoader.java:354)
    at java.lang.ClassLoader.loadClass(ClassLoader.java:425)
    at sun.misc.Launcher$AppClassLoader.loadClass(Launcher.java:308)
    at java.lang.ClassLoader.loadClass(ClassLoader.java:358)
    ... 3 more
```

Or

```
Exception in thread "main" java.lang.NoSuchFieldError: INSTANCE
    at org.apache.http.impl.io.DefaultHttpRequestWriterFactory.<init>(DefaultHttpRequestWriterFactory.java:52)
    at org.apache.http.impl.io.DefaultHttpRequestWriterFactory.<init>(DefaultHttpRequestWriterFactory.java:56)
    at org.apache.http.impl.io.DefaultHttpRequestWriterFactory.<clinit>(DefaultHttpRequestWriterFactory.java:46)
    at org.apache.http.impl.conn.ManagedHttpClientConnectionFactory.<init>(ManagedHttpClientConnectionFactory.java:82)
    at org.apache.http.impl.conn.ManagedHttpClientConnectionFactory.<init>(ManagedHttpClientConnectionFactory.java:95)
    at org.apache.http.impl.conn.ManagedHttpClientConnectionFactory.<init>(ManagedHttpClientConnectionFactory.java:104)
    at org.apache.http.impl.conn.ManagedHttpClientConnectionFactory.<clinit>(ManagedHttpClientConnectionFactory.java:62)
    at org.apache.http.impl.conn.PoolingHttpClientConnectionManager$InternalConnectionFactory.<init>(PoolingHttpClientConnectionManager.java:572)
    at org.apache.http.impl.conn.PoolingHttpClientConnectionManager.<init>(PoolingHttpClientConnectionManager.java:174)
    at org.apache.http.impl.conn.PoolingHttpClientConnectionManager.<init>(PoolingHttpClientConnectionManager.java:158)
    at org.apache.http.impl.conn.PoolingHttpClientConnectionManager.<init>(PoolingHttpClientConnectionManager.java:149)
    at org.apache.http.impl.conn.PoolingHttpClientConnectionManager.<init>(PoolingHttpClientConnectionManager.java:125)
    at com.aliyun.oss.common.comm.DefaultServiceClient.createHttpClientConnectionManager(DefaultServiceClient.java:237)
    at com.aliyun.oss.common.comm.DefaultServiceClient.<init>(DefaultServiceClient.java:78)
    at com.aliyun.oss.OSSClient.<init>(OSSClient.java:268)
    at com.aliyun.oss.OSSClient.<init>(OSSClient.java:193)
    at OSSManagerImpl.upload(OSSManagerImpl.java:42)
    at OSSManagerImpl.main(OSSManagerImpl.java:63)
```

The error occurs because of a version conflict between the Apache HttpClient or Commons HttpClient of your project and Apache HttpClient 4.4.1 of OSS SDK for Java. To view the JAR package and its version that is used in your project, run the `mvn dependency:tree` command in the directory of the project. The following figure shows that the project uses Apache HttpClient 4.3.

```
[INFO] --- maven-dependency-plugin:2.2:tree (default-cli) @ maven-demo ---
[INFO] com.aliyun.oss:maven-demo:jar:0.1.1-SNAPSHOT
[INFO] +- junit:junit:jar:4.10:test
[INFO] | \- org.hamcrest:hamcrest-core:jar:1.1:test
[INFO] +- org.apache.httpcomponents:httpClient:jar:4.3:compile
[INFO] | +- org.apache.httpcomponents:httpcore:jar:4.3:compile
[INFO] | +- commons-logging:commons-logging:jar:1.1.3:compile
[INFO] | \- commons-codec:commons-codec:jar:1.6:compile
[INFO] \- com.aliyun.oss:aliyun-sdk-oss:jar:2.2.1:compile
[INFO]     +- org.jdom:jdom:jar:1.1:compile
[INFO]     \- net.sf.json-lib:json-lib:jar:jdk15:2.4:compile
[INFO]         +- commons-beanutils:commons-beanutils:jar:1.8.0:compile
[INFO]         +- commons-collections:commons-collections:jar:3.2.1:compile
[INFO]         +- commons-lang:commons-lang:jar:2.5:compile
[INFO]         \- net.sf.ezmorph:ezmorph:jar:1.0.6:compile
```

- Solution

You can use one of the following methods to resolve the JAR package conflict:

- Use consistent versions of Apache HttpClient. If your project uses a version of Apache HttpClient that conflicts with Apache HttpClient 4.4.1, use Apache HttpClient 4.4.1 and delete all other Apache HttpClient version dependencies from the pom.xml file. If your project uses Commons HttpClient, conflicts may also occur. To resolve these conflicts, delete Commons HttpClient.
- Resolve dependency conflicts. If your project is dependent on multiple third-party packages and the third-party packages are dependent on different versions of Apache HttpClient, your project may contain dependency conflicts. Use exclusion to resolve these conflicts. For more information, visit [Maven Guides](#).

OSS SDK for Java is dependent on the following package versions. The solution to conflicts is similar to that for HttpClient.

```
[INFO] com.aliyun.oss:maven-demo:jar:0.1.1-SNAPSHOT
[INFO] +- junit:junit:jar:4.10:test
[INFO] | \- org.hamcrest:hamcrest-core:jar:1.1:test
[INFO] \- com.aliyun.oss:aliyun-sdk-oss:jar:2.2.1:compile
[INFO]     +- org.apache.httpcomponents:httpclient:jar:4.4.1:compile
[INFO]     | +- org.apache.httpcomponents:httpcore:jar:4.4.1:compile
[INFO]     | +- commons-logging:commons-logging:jar:1.2:compile
[INFO]     | \- commons-codec:commons-codec:jar:1.9:compile
[INFO]     +- org.jdom:jdom:jar:1.1:compile
[INFO]     \- net.sf.json-lib:json-lib:jar:jdk15:2.4:compile
[INFO]         +- commons-beanutils:commons-beanutils:jar:1.8.0:compile
[INFO]         +- commons-collections:commons-collections:jar:3.2.1:compile
[INFO]         +- commons-lang:commons-lang:jar:2.5:compile
[INFO]         \- net.sf.ezmorph:ezmorph:jar:1.0.6:compile
```

Missing packages

- Cause

If a similar output is displayed when you use OSS SDK for Java, your project may lack the necessary packages to compile or run OSS SDK for Java:

```
Exception in thread "main" java.lang.NoClassDefFoundError: org/apache/http/auth/Credentials
    at com.aliyun.oss.OSSClient.<init>(OSSClient.java:268)
    at com.aliyun.oss.OSSClient.<init>(OSSClient.java:193)
    at com.aliyun.oss.demo.HelloOSS.main(HelloOSS.java:76)
Caused by: java.lang.ClassNotFoundException: org.apache.http.auth.Credentials
    at java.net.URLClassLoader$1.run(URLClassLoader.java:366)
    at java.net.URLClassLoader$1.run(URLClassLoader.java:355)
    at java.security.AccessController.doPrivileged(Native Method)
    at java.net.URLClassLoader.findClass(URLClassLoader.java:354)
    at java.lang.ClassLoader.loadClass(ClassLoader.java:425)
    at sun.misc.Launcher$AppClassLoader.loadClass(Launcher.java:308)
    at java.lang.ClassLoader.loadClass(ClassLoader.java:358)
    ... 3 more
```

Or

```
Exception in thread "main" java.lang.NoClassDefFoundError: org/apache/http/protocol/HttpContext
    at com.aliyun.oss.OSSClient.<init>(OSSClient.java:268)
    at com.aliyun.oss.OSSClient.<init>(OSSClient.java:193)
    at com.aliyun.oss.demo.HelloOSS.main(HelloOSS.java:76)
Caused by: java.lang.ClassNotFoundException: org.apache.http.protocol.HttpContext
    at java.net.URLClassLoader$1.run(URLClassLoader.java:366)
    at java.net.URLClassLoader$1.run(URLClassLoader.java:355)
    at java.security.AccessController.doPrivileged(Native Method)
    at java.net.URLClassLoader.findClass(URLClassLoader.java:354)
    at java.lang.ClassLoader.loadClass(ClassLoader.java:425)
    at sun.misc.Launcher$AppClassLoader.loadClass(Launcher.java:308)
    at java.lang.ClassLoader.loadClass(ClassLoader.java:358)
    ... 3 more
```

Or

```
Exception in thread "main" java.lang.NoClassDefFoundError: org/jdom/input/SAXBuilder
    at com.aliyun.oss.internal.ResponseParsers.getXmlRootElement(ResponseParsers.java:645)
    at ... ..
    at com.aliyun.oss.OSSClient.doesBucketExist(OSSClient.java:471)
    at com.aliyun.oss.OSSClient.doesBucketExist(OSSClient.java:465)
    at com.aliyun.oss.demo.HelloOSS.main(HelloOSS.java:82)
Caused by: java.lang.ClassNotFoundException: org.jdom.input.SAXBuilder
    at java.net.URLClassLoader$1.run(URLClassLoader.java:366)
    at java.net.URLClassLoader$1.run(URLClassLoader.java:355)
    at java.security.AccessController.doPrivileged(Native Method)
    at java.net.URLClassLoader.findClass(URLClassLoader.java:354)
    at java.lang.ClassLoader.loadClass(ClassLoader.java:425)
    at sun.misc.Launcher$AppClassLoader.loadClass(Launcher.java:308)
    at java.lang.ClassLoader.loadClass(ClassLoader.java:358)
    ... 11 more
```

Dependencies of OSS SDK for Java:

- aliyun-sdk-oss-2.2.1.jar
- hamcrest-core-1.1.jar
- jdom-1.1.jar
- commons-codec-1.9.jar
- httpclient-4.4.1.jar
- commons-logging-1.2.jar
- httpcore-4.4.1.jar
- log4j-1.2.15.jar

All packages except the log4j-1.2.15.jar package are required for OSS SDK for Java. However, if you want to enable logging, you must also include the log4j-1.2.15.jar package.

- Solution

You can use one of the following methods to add OSS SDK for Java dependencies to your project:

- If your project is in Eclipse, import dependencies to Eclipse, as instructed in the Method 2 section of [Installation](#).
- If your project is in Ant, add OSS SDK for Java dependencies to the lib directory.
- If you directly use the `javac` or `java` file, run the `-classpath` or `-cp` command to specify the path where OSS SDK for Java dependencies are stored, or save SDK for Java dependencies to classpath.

Connection timeout

• Cause

If a similar output is displayed when you run the OSS SDK for Java program, possible causes are endpoint errors or unavailable networks:

```
com.aliyun.oss.ClientException: SocketException
    at com.aliyun.oss.common.utils.ExceptionFactory.createNetworkException(ExceptionFactory.java:71)
    at com.aliyun.oss.common.comm.DefaultServiceClient.sendRequestCore(DefaultServiceClient.java:116)
    at com.aliyun.oss.common.comm.ServiceClient.sendRequestImpl(ServiceClient.java:121)
    at com.aliyun.oss.common.comm.ServiceClient.sendRequest(ServiceClient.java:67)
    at com.aliyun.oss.internal.OSSOperation.send(OSSOperation.java:92)
    at com.aliyun.oss.internal.OSSOperation.doOperation(OSSOperation.java:140)
    at com.aliyun.oss.internal.OSSOperation.doOperation(OSSOperation.java:111)
    at com.aliyun.oss.internal.OSSBucketOperation.getBucketInfo(OSSBucketOperation.java:152)
    at com.aliyun.oss.OSSClient.getBucketInfo(OSSClient.java:1220)
    at com.aliyun.oss.OSSClient.getBucketInfo(OSSClient.java:1214)
    at com.aliyun.oss.demo.HelloOSS.main(HelloOSS.java:94)
Caused by: org.apache.http.conn.HttpHostConnectException: Connect to oss-test.oss-cn-hangzhou-internal.aliyuncs.com:80 [oss-test.oss-cn-hangzhou-internal.aliyuncs.com/10.84.135.99] failed: Connection timed out: connect
    at org.apache.http.impl.conn.DefaultHttpClientConnectionOperator.connect(DefaultHttpClientConnectionOperator.java:151)
    at org.apache.http.impl.conn.PoolingHttpClientConnectionManager.connect(PoolingHttpClientConnectionManager.java:353)
    at org.apache.http.impl.execchain.MainClientExec.establishRoute(MainClientExec.java:380)
    at org.apache.http.impl.execchain.MainClientExec.execute(MainClientExec.java:236)
    at org.apache.http.impl.execchain.ProtocolExec.execute(ProtocolExec.java:184)
    at org.apache.http.impl.execchain.RedirectExec.execute(RedirectExec.java:110)
    at org.apache.http.impl.client.InternalHttpClient.doExecute(InternalHttpClient.java:184)
    at org.apache.http.impl.client.CloseableHttpClient.execute(CloseableHttpClient.java:82)
    at com.aliyun.oss.common.comm.DefaultServiceClient.sendRequestCore(DefaultServiceClient.java:113)
    ... 9 more
```

• Solution

You can use [ossutil](#) to identify and troubleshoot problems.

What do I do if the "SignatureDoesNotMatch" error is returned?

- Cause 1

The AccessKey ID and AccessKey secret do not match.

For more information about how to obtain the AccessKey ID and AccessKey secret, see [Obtain an AccessKey pair](#).

- Cause 2

The signed URL is incorrectly used. The following code provides a wrong example on how to use a signed URL:

```
GeneratePresignedUrlRequest request = new GeneratePresignedUrlRequest(bucketName, object)
;
request.setExpiration( new Date(new Date().getTime() + 3600 * 1000));
request.addUserMetadata("author");
URL url = ossClient.generatePresignedUrl(request);
Map<String, String> header = new HashMap<String, String>();
header.put("author");
ossClient.putObject(url, new ByteArrayInputStream("Hello OSS".getBytes()), -1, header);
```

By default, if the Method parameter is not configured, the signed URL is generated to initiate a GET request. In the preceding example, the error is returned because the signed URL is used to initiate a PutObject request. In this case, you must set the Method parameter to PUT.

In addition, the user metadata headers in a PutObject request must start with the `x-oss-meta-` prefix. In the preceding example, the user metadata field must be changed to `x-oss-meta-author`.

Solution

Specify the Method parameter and modify the header in the request as follows:

```
request.addUserMetadata("author");
request.setMethod(HttpMethod.PUT);
URL url = ossClient.generatePresignedUrl(request);
Map<String, String> header = new HashMap<String, String>();
header.put("x-oss-meta-" + "author");
ossClient.putObject(url, new ByteArrayInputStream("Hello OSS".getBytes()), -1, header);
```

- Cause 3

- The version of OSS SDK for Java that you use is earlier than 3.7.0 and the HttpClient of version 4.5.9 and later is used in your project.
- The name of the uploaded object contains a plus sign (`+`). However, the httpClient of version 4.5.9 does not encode the plus sign (`+`) by using URL encoding. Therefore, this error is returned to indicate that the signatures on the client and server does not match.

```

320 POST /a HTTP/1.1 (application/json)
107 HTTP/1.1 200 (application/json)
453 PUT /%E4%B8%AD%E6%96%87%E6%B5%8B%E8%AF%95 HTTP/1.1
375 HTTP/1.1 200 OK
364 POST /p HTTP/1.1 (application/json)
1108 HTTP/1.1 200 (application/json)
1017 POST /a HTTP/1.1 (application/json)
107 HTTP/1.1 200 (application/json)
450 PUT /%E4%B8%AD%E6%96%87%E6%B5%8B%E8%AF%95 HTTP/1.1
1183 HTTP/1.1 403 Forbidden
325 POST /a HTTP/1.1 (application/json)
107 HTTP/1.1 200 (application/json)

```

Solution

- We recommend that you upgrade your OSS SDK for Java to version 3.11.1 or later to ensure the compatibility with the HttpClient of version 4.5.9.
- Remove the unnecessary dependency on HttpClient. The dependency on HttpClient is automatically imported when you import OSS SDK for Java. If HttpClient is imported by third-party libraries, see the solution described in [JAR conflicts](#).

• Cause 4

The HttpClient of version 4.5.10 or later is imported in your project and the request headers contain characters that are not supported by ISO/9959-1. For example, the headers whose names start with

x-oss-meta- contain Chinese characters.

```

String content = "123";
ObjectMetadata metadata = new ObjectMetadata();
metadata.setHeader("x-oss-meta-filename", "测试文件名称");
PutObjectRequest request = new PutObjectRequest(bucketName, objectName, new ByteArrayInputStream(content.getBytes()));
request.setMetadata(metadata);
ossClient.putObject(request);

```

Solution

- Follow the solution described in [JAR conflicts](#) to remove the HttpClient of version 4.5.10 or later.
- Make sure that the characters contained in request headers are supported by ISO/9959-1.

What do I do if the "Failed to parse the response result" error is returned?

```

com.aliyun.oss.OSSException: The request signature we calculated does not match the signature you provided. Check your key and signing method.
[ErrorCode]: SignatureDoesNotMatch
[RequestId]: 5FD8A90A4CE48334367729B9
[HostId]: oss-bucket-sdk-demo-hz.oss-cn-hangzhou.aliyuncs.com
[ResponseError]:
<?xml version="1.0" encoding="UTF-8"?>
<Error>
  <Code>SignatureDoesNotMatch</Code>
  <Message>The request signature we calculated does not match the signature you provided. Check your key and signing method.</Message>
  <RequestId>5FD8A90A4CE48334367729B9</RequestId>
  <HostId>oss-bucket-sdk-demo-hz.oss-cn-hangzhou.aliyuncs.com</HostId>
  <OSSAccessKeyId>LTAI4Pd2cecyHRAV1...</OSSAccessKeyId>
  <SignatureProvided>Bd/6DJ7QIcHheoUCoWT/3nJHV0w=</SignatureProvided>
  <StringToSign>GET
Tue, 15 Dec 2020 12:16:10 GMT
/oss-bucket-sdk-demo-hz/?bucketInfo</StringToSign>
  <StringToSignBytes>47 45 54 0A 0A 0A 54 75 65 2C 20 31 35 20 44 65 63 20 32 30 32 30 31 32 3A 31 36 3A 31 30 20 47 4D 54 0A 2F 6F 73 73 2D
  </StringToSignBytes>
  </Error>
at com.aliyun.oss.common.utils.ExceptionFactory.createOSSException(ExceptionFactory.java:100)
at com.aliyun.oss.internal.OSSErrorResponseHandler.handle(OSSErrorResponseHandler.java:72)
at com.aliyun.oss.common.comm.ServiceClient.handleResponse(ServiceClient.java:259)
at com.aliyun.oss.common.comm.ServiceClient.sendRequestImpl(ServiceClient.java:142)
at com.aliyun.oss.common.comm.ServiceClient.sendRequest(ServiceClient.java:71)
at com.aliyun.oss.internal.OSSOperation.send(OSSOperation.java:83)
at com.aliyun.oss.internal.OSSOperation.doOperation(OSSOperation.java:145)

```

• Cause

HTTP requests are intercepted by the software deployed on the client or hijacked by a router on the Internet.

Java 11 is installed and the dependencies on JAXB are not added to the pom.xml file.

- Solution

Initiate requests by using HTTPS instead of HTTP.

Add the dependencies on JAXB to the pom.xml file. For more information, see [Install OSS SDK for Java](#).

org.apache.http.NoHttpResponseException: The target server failed to respond

- Cause

If a similar output is displayed when you run the OSS SDK for Java program:

```
com.aliyun.oss.ClientException: unknown
    at com.aliyun.oss.common.utils.ExceptionFactory.createNetworkException(ExceptionFactory.java:68) ~[aliyun-sdk-oss-2.1.0.jar:na]
    at com.aliyun.oss.common.comm.DefaultServiceClient.sendRequestCore(DefaultServiceClient.java:115) ~[aliyun-sdk-oss-2.1.0.jar:na]
    at com.aliyun.oss.common.comm.ServiceClient.sendRequestImpl(ServiceClient.java:121) ~[aliyun-sdk-oss-2.1.0.jar:na]
    at com.aliyun.oss.common.comm.ServiceClient.sendRequest(ServiceClient.java:67) ~[aliyun-sdk-oss-2.1.0.jar:na]
    at com.aliyun.oss.internal.OSSOperation.send(OSSOperation.java:92) ~[aliyun-sdk-oss-2.1.0.jar:na]
    at com.aliyun.oss.internal.OSSOperation.doOperation(OSSOperation.java:111) ~[aliyun-sdk-oss-2.1.0.jar:na]
    at com.aliyun.oss.internal.OSSMultiPartOperation.doOperation(OSSMultiPartOperation.java:206) ~[aliyun-sdk-oss-2.1.0.jar:na]
    at com.aliyun.oss.internal.OSSMultiPartOperation.initiateMultipartUpload(OSSMultiPartOperation.java:765) ~[aliyun-sdk-oss-2.1.0.jar:na]
    at com.taobao.agoo.dump.Client.OSSTools.multipartUpload(OSSTools.java:79) ~[agoo-dump-client-2.0.0-SNAPSHOT.jar:na]
    at com.taobao.agoo.dump.biz.manager.TaskExecutorManager.UploadTask.run(TaskExecutorManager.java:114) ~[agoo-dump-biz-2.0.0-SNAPSHOT.jar:na]
    at java.util.concurrent.Executors$RunnableAdapter.call(Executors.java:471) [na:1.7.0_51]
    at java.util.concurrent.FutureTask.run(FutureTask.java:262) [na:1.7.0_51]
    at java.util.concurrent.ThreadPoolExecutor.runWorker(ThreadPoolExecutor.java:1145) [na:1.7.0_51]
    at java.util.concurrent.ThreadPoolExecutor$Worker.run(ThreadPoolExecutor.java:615) [na:1.7.0_51]
    at java.lang.Thread.run(Thread.java:744) [na:1.7.0_51]
caused by: org.apache.http.NoHttpResponseException: The target server failed to respond
    at org.apache.http.impl.conn.DefaultHttpResponseParser.parseHead(DefaultHttpResponseParser.java:143) ~[httpclient-4.4.jar:4.4]
    at org.apache.http.impl.conn.DefaultHttpResponseParser.parseHead(DefaultHttpResponseParser.java:57) ~[httpclient-4.4.jar:4.4]
    at org.apache.http.impl.io.AbstractMessageParser.parse(AbstractMessageParser.java:261) ~[httpcore-4.4.jar:4.4]
    at org.apache.http.impl.DefaultHttpClientConnection.receiveResponseHeader(DefaultHttpClientConnection.java:165) ~[httpcore-4.4.jar:4.4]
    at org.apache.http.impl.conn.CPooledProxy.receiveResponseHeader(CPooledProxy.java:167) ~[httpclient-4.4.jar:4.4]
    at org.apache.http.protocol.HttpRequestExecutor.doReceiveResponse(HttpRequestExecutor.java:272) ~[httpcore-4.4.jar:4.4]
```

The preceding error may occur if expired connections are used. This error occurs only when an SDK for Java version earlier than 2.1.2 is used.

- Solution

You can upgrade OSS SDK for Java to V2.1.2 or later.

What do I do if JVM contains a large number of org.apache.http.impl.conn.PoolingHttpClientConnectionManager instances?

- Cause

ossClient is not shut down.

- Solution

You can shut down the ossClient that is executed or use the single instance mode.

What do I do if OSS SDK for Java stops responding when OSS SDK for Java is called?

- Cause

OSS SDK for Java stops responding when OSS SDK for Java is called. Run the `jstack -l pid` command to view the stack. A similar error is displayed:

```

"main" prio=6 tid=0x000000000291e000 nid=0xc40 waiting on condition [0x0000000002dae000]
java.lang.Thread.State: WAITING (parking)
    at sun.misc.Unsafe.park(Native Method)
        - parking to wait for <0x000000007d85697f8> (a java.util.concurrent.locks.AbstractQueuedSynchronizer$ConditionObject)
    at java.util.concurrent.locks.LockSupport.park(LockSupport.java:186)
    at java.util.concurrent.locks.AbstractQueuedSynchronizer$ConditionObject.await(AbstractQueuedSynchronizer.java:2043)
    at org.apache.http.pool.PoolEntryFuture.await(PoolEntryFuture.java:138)
    at org.apache.http.pool.AbstractConnPool.getPoolEntryBlocking(AbstractConnPool.java:306)
    at org.apache.http.pool.AbstractConnPool.access$000(AbstractConnPool.java:64)
    at org.apache.http.pool.AbstractConnPool$2.getPoolEntry(AbstractConnPool.java:192)
    at org.apache.http.pool.AbstractConnPool$2.getPoolEntry(AbstractConnPool.java:185)
    at org.apache.http.pool.PoolEntryFuture.get(PoolEntryFuture.java:107)
    at org.apache.http.impl.conn.PoolingHttpClientConnectionManager.leaseConnection(PoolingHttpClientConnectionManager.java:276)
    at org.apache.http.impl.conn.PoolingHttpClientConnectionManager$1.get(PoolingHttpClientConnectionManager.java:263)
    at org.apache.http.impl.execchain.MainClientExec.execute(MainClientExec.java:190)
    at org.apache.http.impl.execchain.ProtocolExec.execute(ProtocolExec.java:184)
    at org.apache.http.impl.execchain.RedirectExec.execute(RedirectExec.java:110)
    at org.apache.http.impl.client.InternalHttpClient.doExecute(InternalHttpClient.java:184)
    at org.apache.http.impl.client.CloseableHttpClient.execute(CloseableHttpClient.java:82)
    at com.aliyun.oss.common.comm.DefaultServiceClient.sendRequestCore(DefaultServiceClient.java:113)
    at com.aliyun.oss.common.comm.ServiceClient.sendRequestImpl(ServiceClient.java:123)
    at com.aliyun.oss.common.comm.ServiceClient.sendRequest(ServiceClient.java:68)
    at com.aliyun.oss.internal.OSSOperation.send(OSSOperation.java:94)
    at com.aliyun.oss.internal.OSSOperation.doOperation(OSSOperation.java:146)
    at com.aliyun.oss.internal.OSSOperation.doOperation(OSSOperation.java:113)
    at com.aliyun.oss.internal.OSSObjectOperation.getObject(OSSObjectOperation.java:229)
    at com.aliyun.oss.OSSClient.getObject(OSSClient.java:629)
    at com.aliyun.oss.OSSClient.getObject(OSSClient.java:617)
    at samples.HelloOSS.main(HelloOSS.java:49)

```

The error is caused by connection leaks in the connection pool, which occur when `ossObject` is not shut down after usage.

- **Solution**

Check your program to ensure that no connection leaks occur. The following code provides an example on how to shut down `ossObject`:

```

// Read an object.
OSSObject ossObject = ossClient.getObject(bucketName, objectName);
// Perform operations on OSS.
// Shut down the ossObject.
ossObject.close();

```

Connection closure

- Cause

If a similar output is displayed when you use `ossClient.getObject`:

```
Exception in thread "main" org.apache.http.ConnectionClosedException: Premature end of Content-Length delimited message body (expected: 11990526; received: 202880)
    at org.apache.http.impl.io.ContentLengthInputStream.read(ContentLengthInputStream.java:180)
    at org.apache.http.impl.io.ContentLengthInputStream.read(ContentLengthInputStream.java:200)
    at org.apache.http.impl.io.ContentLengthInputStream.close(ContentLengthInputStream.java:103)
    at org.apache.http.impl.execchain.ResponseEntityProxy.streamClosed(ResponseEntityProxy.java:128)
    at org.apache.http.conn.EofSensorInputStream.checkClose(EofSensorInputStream.java:228)
    at org.apache.http.conn.EofSensorInputStream.close(EofSensorInputStream.java:174)
    at java.io.FilterInputStream.close(FilterInputStream.java:181)
    at java.io.FilterInputStream.close(FilterInputStream.java:181)
    at com.aliyun.oss.event.ProgressInputStream.close(ProgressInputStream.java:147)
    at java.io.FilterInputStream.close(FilterInputStream.java:181)
    at samples.HelloOSS.main(HelloOSS.java:39)
```

The error occurs because the interval between the two data reading attempts exceeds 1 minute. OSS closes the connection if no data is sent or received for more than 1 minute.

- Solution

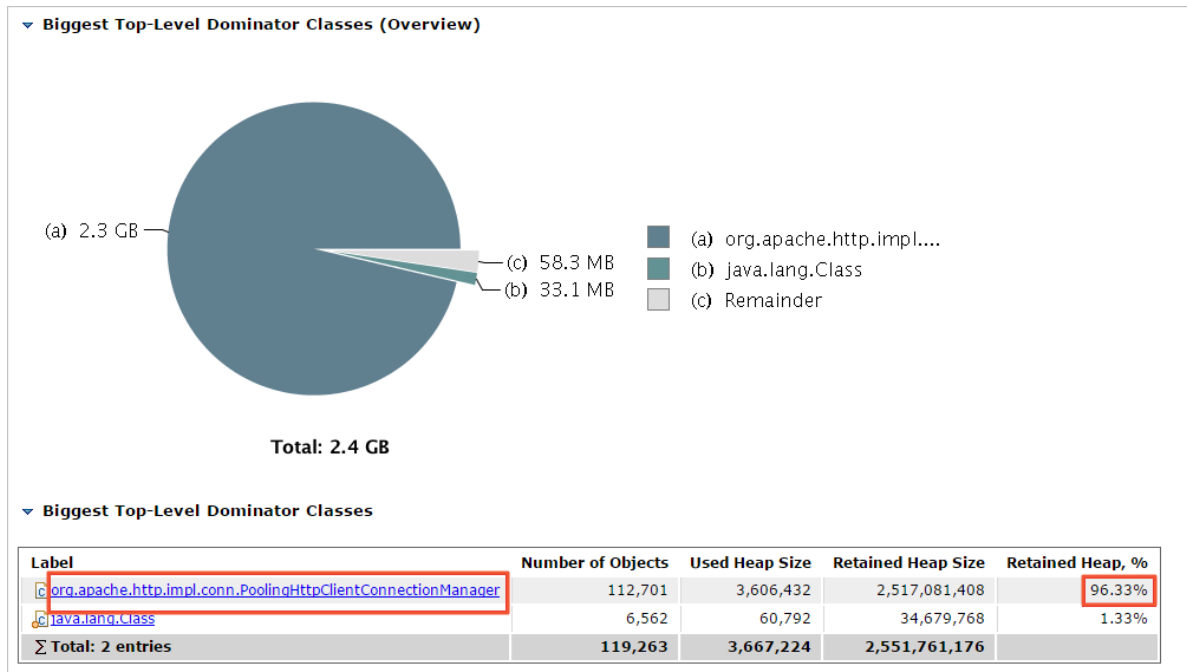
If you must read part of the data during unspecified time periods, we recommend that you use range download to avoid connection closure during data reading. For more information about how to ensure the idempotence, see [Range download](#).

Memory leaks

- Cause

Memory leaks occur when the OSS SDK for Java program runs for an extended period of time. This period can range from a few hours to several days based on the amount of access traffic. We recommend that you use the [Eclipse Memory Analyzer \(MAT\)](#) to analyze memory usage. For more information, visit [Use MAT for Heap Dump File Analysis](#).

If a similar analysis result is shown in the following figure, `PoolingHttpClientConnectionManager` takes up 96% of memory. The possible cause is that the new `OSSClient` command is run multiple times whereas `ossClient.shutdown` is not called. As a result, memory leaks occur.



- Solution

Make sure that `ossClient.shutdown` is called after the new `OSSClient` command is run.

What do I do if `InterruptedException` occurs when `ossClient.shutdown` is called?

- Cause

A similar error is displayed when `ossClient.shutdown` is called by using OSS SDK for Java earlier than V2.3.0:

```
java.lang.InterruptedException: sleep interrupted
    at java.lang.Thread.sleep(Native Method)
    at com.aliyun.oss.common.comm.IdleConnectionReaper.run(IdleConnectionReaper:76)
```

The error occurs because the backend thread (`IdleConnectionReaper`) of `ossClient` closes idle connections periodically. If `ossClient.shutdown` is called when `IdleConnectionReaper` is in Sleep mode, the preceding exception occurs.

- Solution

The following code provides an example on how to ignore this exception:

```
try {
    ossClient.shutdown();
} catch (Exception e) {
}
```

What do I do if the "SDK.ServerUnreachable: Specified endpoint or uri is not valid" exception occurs?


```
SunshineE/android/01040181-2018-05-07173747-860769-online-1.jpg</cloudUrl><fileSize>4096</fileSize><lossName/><loss
... skipping...
com.aliyuncs.exceptions.ClientException: SDK.ServerUnreachable : Specified endpoint or uri is not valid.
    at com.aliyuncs.DefaultAcsClient.doAction(DefaultAcsClient.java:201)
    at com.aliyuncs.DefaultAcsClient.doAction(DefaultAcsClient.java:151)
    at com.aliyuncs.DefaultAcsClient.doAction(DefaultAcsClient.java:59)
    at com.aliyuncs.DefaultAcsClient.getAcsResponse(DefaultAcsClient.java:103)
    at com.sunyard.insurance.oss.aliyun.GetSts.assumeRole(GetSts.java:50)
    at com.sunyard.insurance.oss.aliyun.GetSts.getSts(GetSts.java:60)
    at com.sunyard.insurance.ecm.web.action.oss.OssInfoAction.getUpOssInfo(OssInfoAction.java:259)
    at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
    at sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:39)
    at sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:25)
    at java.lang.reflect.Method.invoke(Method.java:505)
```

- Cause
 - Requests are sent from the client to STS with high concurrency
 - Timeout of connections to the server occurs
 - The STS SDK and the SDK core version are not the latest version.
- Solution
 - When ECS instances on the client or local PC are unable to support bursts of requests, you can reduce the amount of concurrent requests sent to STS.
 - If the connection to the server times out, you can check and analyze captured packets.
 - We recommend that you upgrade STS SDK or SDK core to the latest version.

NoSuchKey

```
case: java.lang.Exception: com.aliyun.oss.OSSException:
Not Found[ErrorCode]: NoSuchKey
```

- Cause

The error message returned because the required object does not exist.
- Solution

For more information, see [HTTP 404 status code](#).

SocketException

```
2018-10-10 16:21:11,127 ERROR com.qunhe.instdeco.plan.maxservice.convert.Converter
downloadOssObjecttaskId:LO63HXQKN4BMWM4BAU888888;
com.aliyun.oss.ClientException: SocketException
    at com.aliyun.oss.common.utils.ExceptionFactory.createNetworkException(ExceptionFactory.java:71)
    at com.aliyun.oss.common.comm.DefaultServiceClient.sendRequestCore(DefaultServiceClient.java:128)
    at com.aliyun.oss.common.comm.ServiceClient.sendRequestImpl(ServiceClient.java:123)
    at com.aliyun.oss.common.comm.ServiceClient.sendRequest(ServiceClient.java:68)
    at com.aliyun.oss.internal.OSSOperation.send(OSSOperation.java:94)
    at com.aliyun.oss.internal.OSSOperation.doOperation(OSSOperation.java:149)
    at com.aliyun.oss.internal.OSSOperation.doOperation(OSSOperation.java:113)
    at com.aliyun.oss.internal.OSSObjectOperation.getObject(OSSObjectOperation.java:273)
```

- Cause

The error is displayed because the socket failed to be specified during initialization. As a result, the request was not received by OSS.
- Solution

We recommend that you troubleshoot the exception based on the following aspects:

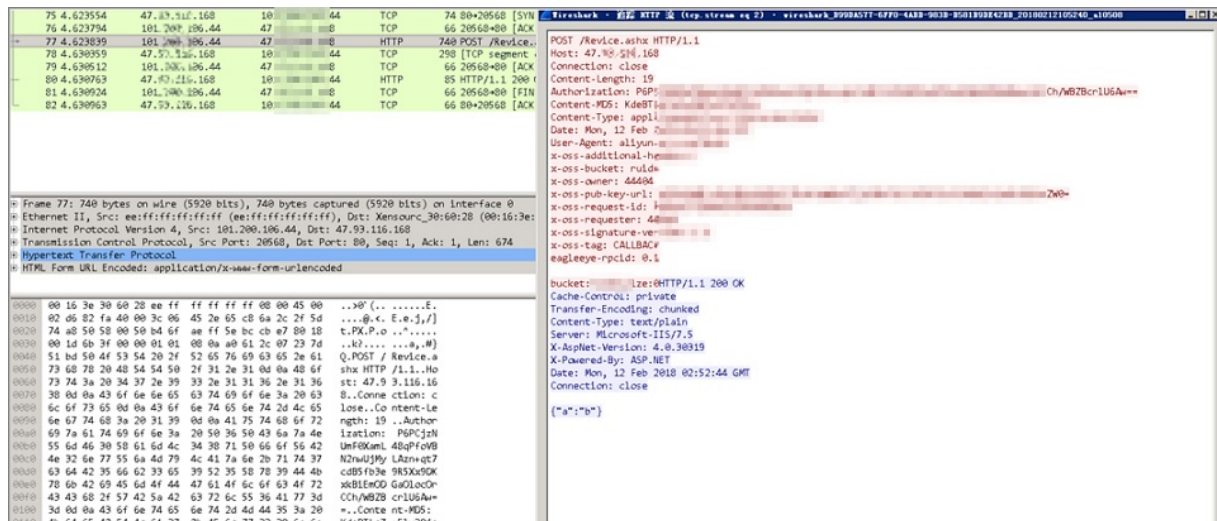
 - Check whether jitters occur.

- Check whether socket resources are used up by other processes.
- Check the maximum number of connections configured in the SDK. If the actual number of connections exceeds this connection limit, a socket exception occurs.

If the problem persists, we recommend that you use tcpdump or Wireshark to capture packets, reproduce the exception, and analyze the packets.

What do I do if the callback operation of OSS PostObject fails?

What do I do if the callback operation of OSS PostObject fails but the callback operation of PutObject succeeds? In most cases, if the JSON format is invalid or a callback fails, an error message is returned. In this case, you must test whether the callback operations of PUT and POST are triggered.



- Cause

The callback parameter follows the file parameter when you send the request.



- Solution

You can adjust the position of the callback parameter.

```

Content-Disposition: form-data; name="Signature"

gncf07Apde0vqaMwBto18XzoH0=
--9431149156168
Content-Disposition: form-data; name="callback"

eyJ3YXksYmFjaVYyYm90dHh0dHA6Ly80Ny45My4xMTYuMTY4L1l1dm
In0=
--9431149156168
Content-Disposition: form-data; name="file"; filename="1.txt"
Content-Type: text/plain

12345678910
--9431149156168--
HTTP/1.1 200 OK
Server: AliyunOSS
Date: Mon, 12 Feb 2018 06:39:22 GMT
Content-Type: application/json

```

The test result shows that the server captures the request.

No.	Time	Source	Destination	Protocol	Length	Info
157	7.634084	101.200.106.45	47.93.116.168	TCP	74	56822→80 [SYN] Seq=0 Win=14600 Len=0 MSS=1460 SACK_PERM=1 TSval=2704418748 TSecr=0 WS=512
158	7.634145	47.93.116.168	101.200.106.45	TCP	74	80→56822 [SYN] Seq=0 Win=14600 Len=0 MSS=1460 SACK_PERM=1 TSval=2704418748 TSecr=0 WS=512
159	7.634337	101.200.106.45	47.93.116.168	TCP	66	56822→80 [ACK] Seq=0 Win=0 Len=0
160	7.634342	101.200.106.45	47.93.116.168	HTTP	768	POST /Revice.ashx HTTP/1.1
161	7.640732	47.93.116.168	101.200.106.45	TCP	298	[TCP segment of data flow 0x1234567890: 47.93.116.168 → 101.200.106.45] Seq=1111111111 Len=298
162	7.640904	101.200.106.45	47.93.116.168	TCP	66	56822→80 [ACK] Seq=0 Win=0 Len=0
163	7.641145	47.93.116.168	101.200.106.45	HTTP	85	HTTP/1.1 200 OK
164	7.641343	101.200.106.45	47.93.116.168	TCP	66	56822→80 [FIN] Seq=0 Win=0 Len=0
165	7.641377	47.93.116.168	101.200.106.45	TCP	66	80→56822 [ACK] Seq=0 Win=0 Len=0

Frame 160: 768 bytes on wire (6144 bits), 768 bytes captured (6144 bits) on interface 0 Ethernet II, Src: e8:ff:ff:ff:ff:ff (e8:ff:ff:ff:ff:ff), Dst: Xensourc_30:60:2b (00:16:3e:30:60:2b) Internet Protocol Version 4, Src: 101.200.106.45, Dst: 47.93.116.168 Transmission Control Protocol, Src Port: 56822, Dst Port: 80, Seq: 1, Ack: 1, Len: 694 Hypertext Transfer Protocol HTML Form URL Encoded: application/x-www-form-urlencoded	POST /Revice.ashx HTTP/1.1 Host: 47.93.116.168 Connection: close Content-Length: 26 Authorization: kw8KVDJ0d9CHtXK1... Content-MD5: bX8GjAhE0UlnnG/q0vA... Content-Type: application/x-www-form-urlencoded Date: Mon, 12 Feb 2018 06:40:54 GMT User-Agent: aliyun-oss-callback X-oss-additional-headers: X-oss-bucket: rulede X-oss-owner: 44404 X-oss-pub-key-url: aHR8cHNSLy9nbWV... X-oss-request-id: 5AB136F600261B4G... X-oss-requester: 23279708537179640 X-oss-signature-version: 1.0 X-oss-tag: CALLBACK eagleeye-rpcid: 0.1 ("bucket"="rulede","size"=11)HTTP/1.1 200 OK Cache-Control: private Transfer-Encoding: chunked Content-Type: text/plain Server: Microsoft-IIS/7.5 X-AspNet-Version: 4.0.30319 X-Powered-By: ASP.NET Date: Mon, 12 Feb 2018 06:40:54 GMT Connection: close ("a"="b")
---	---

Connection pool shut down

```

Caused by: java.lang.IllegalStateException: Connection pool shut down
    at org.apache.http.util.Asserts.check(Asserts.java:34)
    at org.apache.http.pool.AbstractConnPool.lease(AbstractConnPool.java:184)
    at org.apache.http.impl.conn.PoolingHttpClientConnectionManager.requestConnection(PoolingHttpClientConnectionManager.java:251)
    at org.apache.http.impl.execchain.MainClientExec.execute(MainClientExec.java:175)
    at org.apache.http.impl.execchain.ProtocolExec.execute(ProtocolExec.java:184)
    at org.apache.http.impl.execchain.RedirectExec.execute(RedirectExec.java:110)
    at org.apache.http.impl.client.InternalHttpClient.doExecute(InternalHttpClient.java:184)
    at org.apache.http.impl.client.CloseableHttpClient.execute(CloseableHttpClient.java:82)
    at com.aliyun.oss.common.comm.DefaultServiceClient.sendRequestCore(DefaultServiceClient.java:124)
    at com.aliyun.oss.common.comm.ServiceClient.sendRequestImpl(ServiceClient.java:133)
    ... 8 more

```

- Cause

After `ossClient.shutdown()` is called, requests are sent by using `ossClient`.

- Solution

You can check the call logic. Make sure that no requests are sent by using `ossClient` after `ossClient.shutdown()` is called.

What do I do if "Request has expired" is returned when the request is generated by using `generatePresignedUrl` of OSS SDK for Java?

- Cause

An integer overflow occurs, which results in the Year 2038 problem.

An upload request is initiated when the expiration time specified for the URL is exceeded.

- Solution

If the integer overflows, we recommend that you set the expiration time for the URL in OSS SDK for Java to a value earlier than the year 2038.

If an upload request is initiated after the expiration time specified by the URL is exceeded, we recommend that you set a proper expiration time that is later than the time when you initiated the request.

What do I do if the "Invalid Response" or "Implementation of JAXB-API has not been found on module path or classpath" exception occurs?

- Cause

Java 9 or later is installed and the dependencies on JAXB are not added.

- Solution

For more information about how to add the dependencies on JAXB, see [Install OSS SDK for Java](#).

How do I enable or disable logging for OSS SDK for Java?

Apache Log4j defines logs of different levels, including OFF, FATAL, ERROR, WARN, INFO, DEBUG, TRACE, and ALL.

You can enable or disable logging for OSS SDK for Java by configuring the attributes of Log4j.

```
it > aliyun-oss-java-sdk-demo-mvn > conf > log4j.properties

log4j.appender.DRFA=org.apache.log4j.DailyRollingFileAppender
log4j.appender.DRFA.File=${ossdemo.log.dir}/${ossdemo.log.file}

# Rollver at midnight
log4j.appender.DRFA.DatePattern=.yyyy-MM-dd

# 30-day backup
#log4j.appender.DRFA.MaxBackupIndex=30
log4j.appender.DRFA.layout=org.apache.log4j.PatternLayout

# Pattern format: Date LogLevel LoggerName LogMessage
log4j.appender.DRFA.layout.ConversionPattern=%d{ISO8601} %p %c: %m%n
# Debugging Pattern format
#log4j.appender.DRFA.layout.ConversionPattern=%d{ISO8601} %-5p %c{2} (%F:%M(%L)) - %m%n

#
# console
# Add "console" to rootlogger above if you want to use this
#

log4j.appender.console=org.apache.log4j.ConsoleAppender
log4j.appender.console.target=System.err
log4j.appender.console.layout=org.apache.log4j.PatternLayout
log4j.appender.console.layout.ConversionPattern=%d{yy/MM/dd HH:mm:ss} %p %c{2}: %m%n

log4j.logger.org.apache.http=off

# oss log level
log4j.logger.com.aliyun.oss=DEBUG
```

Other errors

For more information about how to resolve other errors returned by OSS, see [Error responses](#).

3.PHP

3.1. Preface

This topic provides the sample code of Object Storage Service (OSS) SDK for PHP in various use cases.

Source code

For the source code of OSS SDK for PHP, visit [GitHub](#).

Sample code

OSS SDK for PHP provides a variety of sample code for your reference or use. The following table lists the sample code files provided by OSS SDK for PHP.

Sample code file	Content
Object.php	Operations related to objects, including uploading, downloading, and managing objects. For more information, see Upload objects , Download objects , and Manage objects .
MultipartUpload.php	Multipart upload
Signature.php	Authorized access
Callback.php	Upload callback
Image.php	Image processing
LiveChannel.php	Operations related to LiveChannel .
Bucket.php	Operations related to buckets, including Create buckets , List buckets , and Delete a bucket .
BucketLifecycle.php	Operations related to lifecycle management, such as configuring, querying, and deleting lifecycle rules for buckets. For more information, see Manage lifecycle rules .
BucketLogging.php	Operations related to logging, such as configuring, querying, and deleting logging configurations. For more information, see Logging .
BucketReferer.php	Operations related to hotlink protection on buckets, such as configuring, querying, and deleting hotlink protection configurations for buckets. For more information, see Hotlink protection .
BucketWebsite.php	Operations related to static website hosting, such as configuring, querying, and deleting static website hosting configurations for buckets. For more information, see Static website hosting .

Sample code file	Content
BucketCors.php	Operations related to cross-origin resource sharing (CORS), such as configuring, querying, and deleting CORS configurations. For more information, see CORS .

3.2. Installation

This topic describes how to install Object Storage Service (OSS) SDK for PHP.

Preparations

OSS SDK for PHP is applicable to PHP 5.3 and later. This topic uses PHP 5.6.22 in the example.

- Environment requirements

You must install PHP and the cURL extension:

- In Windows, you can go to [Compile and use Alibaba Cloud OSS PHP SDK in Windows](#) to install PHP and the cURL extension. If a message that indicates the specified module is unavailable appears, set `extension_dir` to `C:/Windows/System32/`.
- In Ubuntu, run the `sudo apt-get install php-curl` command to install the cURL extension of PHP by using apt-get.
- In CentOS, run the `sudo yum install php-curl` command to install the cURL extension of PHP by using YUM.

- PHP version and cURL extension checks

- You can run the `php -v` command to view the current PHP version.
- You can run the `php -m` command to check whether the cURL extension is installed.

Download OSS SDK for PHP

- [Download from GitHub](#)
- [Download previous versions](#)

For more information, see [API Documentation](#).

 **Note** We recommend that you use the SDK of the latest version. To download OSS SDK for PHP versions earlier than 2.0.0, click [Download](#).

Install OSS SDK for PHP

You can use one of the following methods to install OSS SDK for PHP:

- Composer

- Go to the root directory of your project and run the `composer require aliyuncs/oss-sdk-php` command or add the following dependency to the `composer.json` file:

```
"require": {
    "aliyuncs/oss-sdk-php": "~2.4"
}
```

- ii. Run the `composer install` command to install the dependency. After the dependency is installed, check whether your directory structure complies with the following structure:

```
.
├── app.php
├── composer.json
├── composer.lock
└── vendor
```

In the preceding directory structure, `app.php` indicates your application, and the `vendor/` directory contains the dependent library. You need to add the following dependency to `app.php`:

```
require_once __DIR__ . '/vendor/autoload.php';
```

Note

- o If `autoload.php` is imported to your project, you do not need to import it again after you add the dependency of OSS SDK for PHP.
- o If a network error occurs when you use Composer, you can use the [image source](#) of Composer China by running the `composer config -g repositories.packagist composer http://packagist.phpcomposer.com` command.

• PHAR

- i. Select the corresponding version and download the packaged PHAR file from [GitHub](#).
- ii. Import the PHAR file to your code:

```
require_once '/path/to/oss-sdk-php.phar';
```

• Source code

- i. Select the required version and download the ZIP package from [GitHub](#).
- ii. The decompressed root directory contains the `autoload.php` file. Import the file to the code:

```
require_once '/path/to/oss-sdk/autoload.php';
```

3.3. Initialization

OSSClient serves as the OSS Java client to manage OSS resources such as buckets and objects.

Create an OSSClient instance

To create an OSSClient instance, you need to specify an endpoint. For more information about endpoints, see [Regions and endpoints](#) and [Bind a custom domain name](#).

- Use an OSS domain to create an OSSClient instance

Run the following code to create an OSSClient instance with a domain assigned by OSS:

```
<? php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
// It is highly risky to log on with AccessKey of an Alibaba Cloud account because the account has permissions on all the APIs in OSS. We recommend that you log on as a RAM user to access APIs or perform routine operations and maintenance. To create a RAM account, log on to https://ram.console.aliyun.com.
$accessKeyId = "<yourAccessKeyId>";
$accessKeySecret = "<yourAccessKeySecret>";
// This example uses endpoint China (Hangzhou). Specify the actual endpoint based on your requirements.
$endpoint = "http://oss-cn-hangzhou.aliyuncs.com";
try {
    $ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint);
} catch (OssException $e) {
    print $e->getMessage();
}
```

- Use a custom domain (CNAME) to create an OSSClient instance

Run the following code to create an OSSClient instance with CNAME:


```
<? php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
// It is highly risky to log on with AccessKey of an Alibaba Cloud account because the account has permissions on all the APIs in OSS. We recommend that you log on as a RAM user to access APIs or perform routine operations and maintenance. To create a RAM account, log on to https://ram.console.aliyun.com.
$accessKeyId = "<yourAccessKeyId>";
$accessKeySecret = "<yourAccessKeySecret>";
$endpoint = "<yourEndpoint>";
try {
    // The true parameter indicates that the CNAME is enabled. CNAME indicates a custom domain bound to a bucket.
    $ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint, true);
} catch (OssException $e) {
    print $e->getMessage();
}
```

 **Note** The `ossClient.listBuckets` method is not available when CNAME is used.

- Use STS to create an `OSSClient` instance

Run the following code to create an `OSSClient` instance with Security Token Service (STS):

```
<? php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
// It is highly risky to log on with AccessKey of an Alibaba Cloud account because the account has permissions on all the APIs in OSS. We recommend that you log on as a RAM user to access APIs or perform routine operations and maintenance. To create a RAM account, log on to https://ram.console.aliyun.com.
$accessKeyId = "<yourAccessKeyId>";
$accessKeySecret = "<yourAccessKeySecret>";
// This example uses endpoint China (Hangzhou). Specify the actual endpoint based on your requirements.
$endpoint = "http://oss-cn-hangzhou.aliyuncs.com";
$securityToken = "<yourSecurityToken>";
try {
    $ossClient = new OssClient(
        $accessKeyId, $accessKeySecret, $endpoint, false, $securityToken);
} catch (OssException $e) {
    print $e->getMessage();
}
```

For more information, see [Authorized access](#).

- Use a proxy server to create an OSSClient instance

Run the following code to create an OSSClient instance with a proxy server, which is supported by PHP 5.3 and later.

```
<? php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
// It is highly risky to log on with AccessKey of an Alibaba Cloud account because the account has permissions on all the APIs in OSS. We recommend that you log on as a RAM user to access APIs or perform routine operations and maintenance. To create a RAM account, log on to https://ram.console.aliyun.com.
$accessKeyId = "<yourAccessKeyId>";
$accessKeySecret = "<yourAccessKeySecret>";
// The address of the proxy server, for example, http://<username>:<password>@<proxy ip>:<proxy port>
$requestProxy = "<yourRequestProxy>";
// This example uses endpoint China (Hangzhou). Specify the actual endpoint based on your requirements.
$endpoint = "http://oss-cn-hangzhou.aliyuncs.com>";
try {
    $ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint, false, $requestProxy);
} catch (OssException $e) {
    print $e->getMessage();
}
```

Configure network parameters

Run the following code to configure the network parameters of an OSSClient instance:

```
<? php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
// It is highly risky to log on with AccessKey of an Alibaba Cloud account because the account has permissions on all the APIs in OSS. We recommend that you log on as a RAM user to access APIs or perform routine operations and maintenance. To create a RAM account, log on to https://ram.console.aliyun.com.
$accessKeyId = "<yourAccessKeyId>";
$accessKeySecret = "<yourAccessKeySecret>";
// This example uses endpoint China (Hangzhou). Specify the actual endpoint based on your requirements.
$endpoint = "http://oss-cn-hangzhou.aliyuncs.com";
try {
    $ossClient = new OssClient(
        $accessKeyId, $accessKeySecret, $endpoint);
    // Set the timeout time in seconds for data transmission at the sockets layer. The default value is 5184000.
    $ossClient->setTimeout(3600);
    // Set the timeout time in seconds when connections are established. The default value is 10.
    $ossClient->setConnectTimeout(10);
} catch (OssException $e) {
    print $e->getMessage();
}
```

3.4. Quick start

This topic describes how to use Object Storage Service (OSS) SDK for PHP to perform routine operations, such as creating buckets, uploading objects, and downloading objects.

Create a bucket

A bucket is a global namespace in OSS. A bucket is a container for objects stored in OSS.

Note

The following code provides an example on how to create a bucket:

```
<?php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a Resource Access Management (RAM) user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
$accessKeyId = "<yourAccessKeyId>";
$accessKeySecret = "<yourAccessKeySecret>";
// The China (Hangzhou) region is used in this example as the endpoint. Specify the actual endpoint.
$endpoint = "http://oss-cn-hangzhou.aliyuncs.com";
// Specify the name of the bucket. Example: examplebucket.
$bucket = "<yourBucketName>";
try {
    $ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint);
    $ossClient->createBucket($bucket);
} catch (OssException $e) {
    print $e->getMessage();
}
```

Upload an object

The following code provides an example on how to upload objects to OSS by using stream upload:

```
<?php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
$accessKeyId = "<yourAccessKeyId>";
$accessKeySecret = "<yourAccessKeySecret>";
// The China (Hangzhou) region is used in this example as the endpoint. Specify the actual endpoint.
$endpoint = "http://oss-cn-hangzhou.aliyuncs.com";
// Specify the name of the bucket. Example: examplebucket.
$bucket = "<yourBucketName>";
// <yourObjectName> indicates the full path of the object to upload. The path must include the extension of the object name and exclude the bucket name. Example: abc/efg/123.jpg.
$object = "<yourObjectName>";
$content = "Hi, OSS.";
try {
    $ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint);
    $ossClient->putObject($bucket, $object, $content);
} catch (OssException $e) {
    print $e->getMessage();
}
```

For more information about how to upload objects, see [Overview](#).

Download an object

The following code provides an example on how to download an object:

```
<?php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
$accessKeyId = "<yourAccessKeyId>";
$accessKeySecret = "<yourAccessKeySecret>";
// The China (Hangzhou) region is used in this example as the endpoint. Specify the actual endpoint.
$endpoint = "http://oss-cn-hangzhou.aliyuncs.com";
// Specify the name of the bucket. Example: examplebucket.
$bucket = "<yourBucketName>";
// <yourObjectName> indicates the full path of the object to download from OSS. The path must include the extension of the object name and exclude the bucket name. Example: abc/efg/123.jpg.
$object = "<yourObjectName>";
try {
    $ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint);
    $content = $ossClient->getObject($bucket, $object);
    print("object content: " . $content);
} catch (OssException $e) {
    print $e->getMessage();
}
```

For more information about how to download objects, see [Overview](#).

List objects

The following code provides an example on how to list objects stored in a bucket named examplebucket. By default, a maximum of 100 objects are listed.

```
<?php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
$accessKeyId = "<yourAccessKeyId>";
$accessKeySecret = "<yourAccessKeySecret>";
// The China (Hangzhou) region is used in this example as the endpoint. Specify the actual endpoint.
$endpoint = "http://oss-cn-hangzhou.aliyuncs.com";
// Specify the name of the bucket. Example: examplebucket.
$bucket = "<yourBucketName>";
try {
    $ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint);
    $listObjectInfo = $ossClient->listObjects($bucket);
    $objectList = $listObjectInfo->getObjectList();
    if (!empty($objectList)) {
        foreach ($objectList as $objectInfo) {
            print($objectInfo->getKey() . "\t" . $objectInfo->getSize() . "\t" . $objectInfo->getLastModified() . "\n");
        }
    }
} catch (OssException $e) {
    print $e->getMessage();
}
```

For more information about how to list objects, see [List objects](#).

Delete an object

The following code provides an example on how to delete an object:


```
<?php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
$accessKeyId = "<yourAccessKeyId>";
$accessKeySecret = "<yourAccessKeySecret>";
// The China (Hangzhou) region is used in this example as the endpoint. Specify the actual endpoint.
$endpoint = "http://oss-cn-hangzhou.aliyuncs.com";
// Specify the name of the bucket. Example: examplebucket.
$bucket = "<yourBucketName>";
// <yourObjectName> indicates the full path of the object that you want to delete from OSS. The path must include the extension of the object name and exclude the bucket name. Example : abc/efg/123.jpg.
$object = "<yourObjectName>";
try {
    $ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint);
    $ossClient->deleteObject($bucket, $object);
} catch (OssException $e) {
    print $e->getMessage();
}
```

For more information about how to delete objects, see [Delete objects](#).

3.5. Buckets

3.5.1. Create buckets

A bucket is a container used to store objects in Object Storage Service (OSS). Each object is contained in a bucket. This topic describes how to create a bucket.

Sample code

The following code provides an example on how to create a bucket named examplebucket:

```

<?php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
$accessKeyId = "yourAccessKeyId";
$accessKeySecret = "yourAccessKeySecret";
// In this example, the endpoint of the China (Hangzhou) region is used. Specify the endpoint based on your business requirements.
$endpoint = "http://oss-cn-hangzhou.aliyuncs.com";
// Specify the name of the bucket. Example: examplebucket.
$bucket = "examplebucket";
try {
    $ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint);
    // Set the storage class of the bucket to Infrequent Access (IA). The default value is Standard.
    $options = array(
        OssClient::OSS_STORAGE => OssClient::OSS_STORAGE_IA
    );
    // Set the access control list (ACL) of the bucket to public-read. The default value is private.
    $ossClient->createBucket($bucket, OssClient::OSS_ACL_TYPE_PUBLIC_READ, $options);
} catch (OssException $e) {
    printf(__FUNCTION__ . ": FAILED\n");
    printf($e->getMessage() . "\n");
    return;
}
print(__FUNCTION__ . ": OK" . "\n");

```

References

- For more information about the complete sample code that is used to create a bucket, visit [GitHub](#).
- For more information about the API operation that you can call to create a bucket, see [PutBucket](#).

3.5.2. List buckets

Buckets are listed in alphabetical order. You can list all your buckets or buckets whose names contain the specified prefix.

List all buckets

The following code provides an example on how to list all buckets:

```
<?php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
$accessKeyId = "yourAccessKeyId";
$accessKeySecret = "yourAccessKeySecret";
// In this example, the endpoint of the China (Hangzhou) region is used. Specify your actual endpoint.
$endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
try{
    $ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint);
    // List all buckets that belong to the current account.
    $bucketListInfo = $ossClient->listBuckets();
} catch(OssException $e) {
    printf(__FUNCTION__ . ": FAILED\n");
    printf($e->getMessage() . "\n");
    return;
}
$bucketList = $bucketListInfo->getBucketList();
foreach($bucketList as $bucket) {
    print($bucket->getLocation() . "\t" . $bucket->getName() . "\t" . $bucket->getCreatedate() . "\n");
}
```

List buckets whose names contain the specified prefix

The following code provides an example on how to list buckets whose names contain the specified prefix:

```

<?php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
$accessKeyId = "yourAccessKeyId";
$accessKeySecret = "yourAccessKeySecret";
// In this example, the endpoint of the China (Hangzhou) region is used. Specify your actual endpoint.
$endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
$ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint);
// Specify the prefix.
$prefix = "yourPrefix";
try{
    $options = array(OssClient::OSS_QUERY_STRING => array(OssClient::OSS_PREFIX => $prefix));
    // List the buckets whose names contain the specified prefix.
    $bucketListInfo = $ossClient->listBuckets($options);
} catch(OssException $e) {
    printf(__FUNCTION__ . ": FAILED\n");
    printf($e->getMessage() . "\n");
    return;
}
$bucketList = $bucketListInfo->getBucketList();
foreach($bucketList as $bucket) {
    print($bucket->getLocation() . "\t" . $bucket->getName() . "\t" . $bucket->getCreatedate() . "\n");
}

```

List buckets whose names are alphabetically after the bucket specified by the marker parameter

The Marker parameter indicates a bucket name. Buckets whose names are alphabetically after the value of Marker are returned.

The following code provides an example on how to list buckets whose names are alphabetically after the bucket specified by Marker:

```

<?php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
$accessKeyId = "yourAccessKeyId";
$accessKeySecret = "yourAccessKeySecret";
// In this example, the endpoint of the China (Hangzhou) region is used. Specify your actual endpoint.
$endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
$ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint);
// Set the value of Marker.
$marker = "yourMarker";
// List buckets whose names are alphabetically after the bucket specified by Marker.
try{
    $options = array(OssClient::OSS_QUERY_STRING => array(OssClient::OSS_MARKER => $marker));
    $bucketListInfo = $ossClient->listBuckets($options);
} catch(OssException $e) {
    printf(__FUNCTION__ . ": FAILED\n");
    printf($e->getMessage() . "\n");
    return;
}
$bucketList = $bucketListInfo->getBucketList();
foreach($bucketList as $bucket) {
    print($bucket->getLocation() . "\t" . $bucket->getName() . "\t" . $bucket->getCreatedate() . "\n");
}

```

List a specified number of buckets

The MAX_KEYS parameter is used to specify the maximum number of the returned buckets. The value of MAX_KEYS cannot exceed 1000. By default, if MAX_KEYS is not specified, 100 buckets are returned.

The following code provides an example on how to list a specified number of buckets:

```
<?php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
$accessKeyId = "yourAccessKeyId";
$accessKeySecret = "yourAccessKeySecret";
// In this example, the endpoint of the China (Hangzhou) region is used. Specify your actual endpoint.
$endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
$ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint);
// Specify that 10 buckets are listed.
try{
    $options = array(OssClient::OSS_QUERY_STRING => array(OssClient::OSS_MAX_KEYS => 10));
    $bucketListInfo = $ossClient->listBuckets($options);
} catch(OssException $e) {
    printf(__FUNCTION__ . ": FAILED\n");
    printf($e->getMessage() . "\n");
    return;
}
$bucketList = $bucketListInfo->getBucketList();
foreach($bucketList as $bucket) {
    print($bucket->getLocation() . "\t" . $bucket->getName() . "\t" . $bucket->getCreateDate() . "\n");
}
```

References

- For more information about the complete sample code that is used to list buckets, visit [GitHub](#).
- For the API operation that you can call to list buckets, see [GetService \(List Buckets\)](#).

3.5.3. Determine whether buckets exist

A bucket is a container used to store objects in Object Storage Service (OSS). Every object is contained in a bucket. This topic describes how to determine whether a bucket exists.

```
<? php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log on to OSS, because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine operations and maintenance. To create a RAM user, log on to https://ram.console.aliyun.com.
$accessKeyId = "<yourAccessKeyId>";
$accessKeySecret = "<yourAccessKeySecret>";
// This example uses the endpoint of the China (Hangzhou) region. Specify the actual endpoint based on your requirements.
$endpoint = "http://oss-cn-hangzhou.aliyuncs.com";
// Specify the bucket name.
$bucket = "<yourBucketName>";
try {
    $ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint);
    $res = $ossClient->doesBucketExist($bucket);
} catch (OssException $e) {
    printf(__FUNCTION__ . ": FAILED\n");
    printf($e->getMessage() . "\n");
    return;
}
if ($res === true) {
    print(__FUNCTION__ . ": OK" . "\n");
} else {
    print(__FUNCTION__ . ": FAILED" . "\n");
}
```

3.5.4. Query the regions of buckets

A bucket is a container for objects stored in Object Storage Service (OSS). Every object is contained in a bucket. This topic describes how to query the region of a bucket.

Sample code

The following sample code shows how to query the region or location of a bucket:

```
<?php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS, because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
$accessKeyId = "yourAccessKeyId";
$accessKeySecret = "yourAccessKeySecret";
// Specify the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com.
$endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
// Specify the name of the bucket.
$bucket = "yourBucketName";
try {
    $ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint);
    $regions = $ossClient->getBucketLocation($bucket);
} catch (OssException $e) {
    printf(__FUNCTION__ . ": FAILED\n");
    printf($e->getMessage() . "\n");
    return;
}
print(__FUNCTION__ . ": OK" . "\n");
var_dump($regions);
```

References

- For more information about the complete sample code that is used to query the region of a bucket, visit [Git Hub](#).
- For more information about the API operation that you can call to query the region of a bucket, see [Get Bucket Location](#).

3.5.5. Query bucket information

A bucket is a container for objects stored in OSS. Every object is contained in a bucket. This topic describes how to query the information about a bucket.

The following code provides an example on how to query the information about a bucket:


```

<? php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to lo
g on to OSS because the account has permissions on all API operations. We recommend that yo
u use your RAM user's credentials to call API operations or perform routine operations and
maintenance. To create a RAM user, log on to the RAM console.
$accessKeyId = "<yourAccessKeyId>";
$accessKeySecret = "<yourAccessKeySecret>";
// The endpoint of the China (Hangzhou) region is used in this example. Specify the actual
endpoint.
$endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
$bucket= "<yourBucketName>";
$ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint, false);
try {
    // Query the information about the bucket, including the bucket name, region, created t
ime, storage class, public endpoint, and internal endpoint.
    $info = $ossClient->getBucketInfo($bucket);
    printf("bucket name:%s\n", $info->getName());
    printf("bucket location:%s\n", $info->getLocation());
    printf("bucket creation time:%s\n", $info->getCreateDate());
    printf("bucket storage class:%s\n", $info->getStorageClass());
    printf("bucket extranet endpoint:%s\n", $info->getExtranetEndpoint());
    printf("bucket intranet endpoint:%s\n", $info->getIntranetEndpoint());
} catch (OssException $e) {
    printf(__FUNCTION__ . ": FAILED\n");
    printf($e->getMessage() . "\n");
    return;
}
print(__FUNCTION__ . ": OK" . "\n");

```

For more information about how to query information about a bucket, see [GetBucketInfo](#).

3.5.6. Query the metadata of buckets

A bucket is a container for objects stored in OSS. Every object is contained in a bucket. This topic describes how to query the metadata of a bucket.

The following code provides an example on how to query the metadata of a bucket:

```

<? php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log
on to OSS, because the account has permissions on all API operations. We recommend that y
ou use a RAM user to call API operations or perform routine operations and maintenance. To
create a RAM user, log on to https://ram.console.aliyun.com.
$accessKeyId = "<yourAccessKeyId>";
$accessKeySecret = "<yourAccessKeySecret>";
// This example uses the endpoint of the China (Hangzhou) region. Specify the actual endpoi
nt based on your requirements.
$endpoint = "http://oss-cn-hangzhou.aliyuncs.com";
// Specify the bucket name.
$bucket= "<yourBucketName>";
try {
    $ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint);
    $metas = $ossClient->getBucketMeta($bucket);
} catch (OssException $e) {
    printf(__FUNCTION__ . ": FAILED\n");
    printf($e->getMessage() . "\n");
    return;
}
print(__FUNCTION__ . ": OK" . "\n");
var_dump($metas);

```

3.5.7. Manage bucket ACLs

A bucket is a container for objects stored in Object Storage Service (OSS). Every object is contained in a bucket. This topic describes how to configure and query the access control list (ACL) of a bucket.

Configure the ACL of a bucket

The following table describes the bucket ACLs.

ACL	Description	Method
private	Only the bucket owner and authorized users can perform read and write operations on objects in the bucket. Other users cannot access the objects in the bucket.	OssClient::OSS_ACL_TYPE_PRIVATE

ACL	Description	Method
public read	Only the bucket owner and authorized users of this bucket can perform read and write operations on objects in the bucket. Other users, including anonymous users, can only read objects in the bucket. Exercise caution when you set the bucket ACL to this value.	OssClient::OSS_ACL_TYPE_PUBLIC_READ
public read/write	All users, including anonymous users, can perform read and write operations on objects in the bucket. Exercise caution when you set the bucket ACL to this value.	OssClient::OSS_ACL_TYPE_PUBLIC_READ_WRITE

The following code provides an example on how to configure the ACL of a bucket:

```
<?php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
$accessKeyId = "<yourAccessKeyId>";
$accessKeySecret = "<yourAccessKeySecret>";
// In this example, the endpoint of the China (Hangzhou) region is used. Specify the endpoint based on your business requirements.
$endpoint = "http://oss-cn-hangzhou.aliyuncs.com";
// Specify the name of the bucket.
$bucket = "yourBucketName";
// Set the ACL of the bucket to private.
$acl = OssClient::OSS_ACL_TYPE_PRIVATE;
try {
    $ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint);
    $ossClient->putBucketAcl($bucket, $acl);
} catch (OssException $e) {
    printf(__FUNCTION__ . ": FAILED\n");
    printf($e->getMessage() . "\n");
    return;
}
print(__FUNCTION__ . ": OK" . "\n");
```

For more information about how to configure the ACL of a bucket, see [PutBucketAcl](#).

Query the ACL of a bucket

The following code provides an example on how to query the ACL of a bucket:

```
<?php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
$accessKeyId = "<yourAccessKeyId>";
$accessKeySecret = "<yourAccessKeySecret>";
// In this example, the endpoint of the China (Hangzhou) region is used. Specify the endpoint based on your business requirements.
$endpoint = "http://oss-cn-hangzhou.aliyuncs.com";
// Specify the name of the bucket.
$bucket = "<yourBucketName>";
try {
    $ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint);
    // Query the ACL of the bucket.
    $res = $ossClient->getBucketAcl($bucket);
} catch (OssException $e) {
    printf(__FUNCTION__ . ": FAILED\n");
    printf($e->getMessage() . "\n");
    return;
}
print(__FUNCTION__ . ": OK" . "\n");
print('acl: ' . $res);
```

For more information about how to query the ACL of a bucket, see [GetBucketAcl](#).

3.5.8. Delete a bucket

If you no longer use a bucket, delete the bucket to stop unnecessary charges.



Warning Deleted buckets cannot be restored. Exercise caution when you perform this operation.

Prerequisites

- All objects in the bucket are deleted.
 - For more information about how to manually delete a small number of objects, see [删除文件](#).

- To delete a large number of objects, we recommend that you configure lifecycle rules to batch delete the objects. For more information about how to delete a large number of objects, see [Configure lifecycle rules](#).

Notice

If versioning is enabled for the bucket that you want to delete, make sure that all versions of objects in the bucket are deleted. For more information about how to delete all versions of objects in a bucket, see [Configure versioning](#).

- Parts that are uploaded by multipart upload or resumable upload tasks in the bucket are deleted. For more information about how to delete parts in a bucket, see [Manage parts](#).
- All LiveChannels in the bucket are deleted. For more information about how to delete LiveChannels, see [DeleteLiveChannel](#).

Sample code

The following sample code shows how to delete a bucket named examplebucket:

```
<?php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
// The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to access OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
$accessKeyId = "yourAccessKeyId";
$accessKeySecret = "yourAccessKeySecret";
// Specify the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set endpoint to https://oss-cn-hangzhou.aliyuncs.com.
$endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
// Specify the name of the bucket. Example: examplebucket.
$bucket = "examplebucket";
try{
    $ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint);
    $ossClient->deleteBucket($bucket);
} catch(OssException $e) {
    printf(__FUNCTION__ . ": FAILED\n");
    printf($e->getMessage() . "\n");
    return;
}
print(__FUNCTION__ . ": OK" . "\n");
```

References

- For more information about the complete sample code that is used to delete a bucket, visit [GitHub](#).
- For more information about the API operation that you can call to delete a bucket, see [DeleteBucket](#).

3.5.9. Bucket tagging

You can use tags to identify buckets used for different purposes and manage these buckets.

Usage notes

The bucket tagging feature uses a key-value pair to identify a bucket. You can add tags to buckets that are used for different purposes and manage the buckets by tags.

- Only the bucket owner and users who are granted the `oss:PutBucketTagging` permission can add tags to the bucket or modify the tags of the bucket. If other users try to add tags to the bucket or modify the tags of the bucket, 403 Forbidden is returned with the `AccessDenied` error code.
- You can configure up to 20 tags for a bucket.
- The key and value of a tag must be UTF-8-encoded.
- The key of a tag can be up to 64 characters in length and cannot start with `http://`, `https://`, or `Aliyun`. In addition, the key of a tag cannot be empty.
- The value of a tag can be up to 128 characters in length and can be empty.

Configure tags for a bucket

The following code provides an example on how to configure tags for a bucket:

```
<?php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
use OSS\Model\TaggingConfig;
use OSS\Model\Tag;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access Object Storage Service (OSS) because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
$accessKeyId = "yourAccessKeyId";
$accessKeySecret = "yourAccessKeySecret";
// In this example, the endpoint of the China (Hangzhou) region is used. Specify your actual endpoint.
$endpoint = "http://oss-cn-hangzhou.aliyuncs.com";
// Specify the bucket name. Example: examplebucket.
$bucket = "examplebucket";
$ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint, false);
try {
    // Configure tags for the bucket.
    $config = new TaggingConfig();
    $config->addTag(new Tag("key1", "value1"));
    $config->addTag(new Tag("key2", "value2"));
    $ossClient->putBucketTags($bucket, $config);
} catch (OssException $e) {
    printf(__FUNCTION__ . ": FAILED\n");
    printf($e->getMessage() . "\n");
    return;
}
print(__FUNCTION__ . ": OK" . "\n");
```

Query tags of a bucket

The following code provides an example on how to query tags of a bucket:

```
<?php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
use OSS\Model\TaggingConfig;
use OSS\Model\Tag;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
$accessKeyId = "yourAccessKeyId";
$accessKeySecret = "yourAccessKeySecret";
// In this example, the endpoint of the China (Hangzhou) region is used. Specify your actual endpoint.
$endpoint = "http://oss-cn-hangzhou.aliyuncs.com";
// Specify the bucket name. Example: examplebucket.
$bucket = "examplebucket";
$ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint, false);
try {
    // Query tags of the bucket.
    $config = $ossClient->getBucketTags($bucket);
    print_r($config->getTags());
} catch (OssException $e) {
    printf(__FUNCTION__ . ": FAILED\n");
    printf($e->getMessage() . "\n");
    return;
}
print(__FUNCTION__ . ": OK" . "\n");
```

Delete tags of a bucket

The following code provides an example on how to delete tags of a bucket:


```

<?php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
use OSS\Model\Tag;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
$accessKeyId = "yourAccessKeyId";
$accessKeySecret = "yourAccessKeySecret";
// In this example, the endpoint of the China (Hangzhou) region is used. Specify your actual endpoint.
$endpoint = "http://oss-cn-hangzhou.aliyuncs.com";
// Specify the bucket name. Example: examplebucket.
$bucket = "examplebucket";
$ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint, false);
try {
    // Delete the specific tags configured for the bucket.
    $tags = array();
    $tags[] = new Tag("key1", "value1");
    $tags[] = new Tag("key2", "value2");
    $ossClient->deleteBucketTags($bucket, $tags);
    // Delete all tags configured for the bucket.
    $ossClient->deleteBucketTags($bucket);
} catch (OssException $e) {
    printf(__FUNCTION__ . ": FAILED\n");
    printf($e->getMessage() . "\n");
    return;
}
print(__FUNCTION__ . ": OK" . "\n");

```

References

- For more information about the API operation that you can call to configure tags for a bucket, see [PutBucketTags](#).
- For more information about the API operation that you can call to query tags of a bucket, see [GetBucketTags](#).
- For more information about the API operation that you can call to delete tags of a bucket, see [DeleteBucketTags](#).

3.5.10. Manage lifecycle rules

Object Storage Service (OSS) allows you to configure lifecycle rules to delete expired objects and parts or convert the storage class of expired objects to Infrequent Access (IA) or Archive to reduce your storage costs. This topic describes how to manage lifecycle rules for buckets.

Background information

Each lifecycle rule contains the following information:

- Rule ID. It uniquely identifies a rule in a bucket.
- Policy. You can use one of the following methods to configure a policy. Only a single method can be configured for each bucket.
 - Match by prefix: You can create multiple rules to configure different prefixes. Each prefix must be unique.
 - Match by bucket: If you select this method, only a single rule can be created.
- Expiration time. You can use one of the following configuration methods to configure the expiration time:
 - Specify a validity period for which to retain objects after they are last modified.
 - Specify an expiration date that objects that are last modified before this date expire.
- Status.

For the complete code used to manage lifecycle rules, visit [GitHub](#).

Configure lifecycle rules

```

<? php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
use OSS\Model\LifecycleConfig;
use OSS\Model\LifecycleRule;
use OSS\Model\LifecycleAction;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to lo
g on to OSS because the account has permissions on all API operations. We recommend that yo
u use your RAM user's credentials to call API operations or perform routine operations and
maintenance. To create a RAM user, log on to the RAM console.
$accessKeyId = "<yourAccessKeyId>";
$accessKeySecret = "<yourAccessKeySecret>";
// The endpoint of the China (Hangzhou) region is used in this example. Specify the actual
endpoint.
$endpoint = "http://oss-cn-hangzhou.aliyuncs.com";
$bucket= "<yourBucketName>";
// Specify the rule IDs and the object prefixes to match the rules.
$ruleId0 = "rule0";
$matchPrefix0 = "A0/";
$ruleId1 = "rule1";
$matchPrefix1 = "A1/";
$lifecycleConfig = new LifecycleConfig();
$actions = array();
// Specify that objects expire three days after they are last modified.
$actions[] = new LifecycleAction(OssClient::OSS_LIFECYCLE_EXPIRATION, OssClient::OSS_LIFECY
CLE_TIMING_DAYS, 3);
$lifecycleRule = new LifecycleRule($ruleId0, $matchPrefix0, "Enabled", $actions);
$lifecycleConfig->addRule($lifecycleRule);
$actions = array();
// Specify that the objects last modified before the specified date expire.
$actions[] = new LifecycleAction(OssClient::OSS_LIFECYCLE_EXPIRATION, OssClient::OSS_LIFECY
CLE_TIMING_DATE, '2022-10-12T00:00:00.000Z');
$lifecycleRule = new LifecycleRule($ruleId1, $matchPrefix1, "Enabled", $actions);
$lifecycleConfig->addRule($lifecycleRule);
try {
    $ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint);
    $ossClient->putBucketLifecycle($bucket, $lifecycleConfig);
} catch (OssException $e) {
    printf(__FUNCTION__ . ": FAILED\n");
    printf($e->getMessage() . "\n");
    return;
}
print(__FUNCTION__ . ": OK" . "\n");

```

View lifecycle rules

The following code provides an example on how to query the lifecycle rules configured for the bucket named `examplebucket`:

```
<? php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log on to OSS because the account has permissions on all API operations. We recommend that you use your RAM user's credentials to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
$accessKeyId = "<yourAccessKeyId>";
$accessKeySecret = "<yourAccessKeySecret>";
// The endpoint of the China (Hangzhou) region is used in this example. Specify the actual endpoint.
$endpoint = "http://oss-cn-hangzhou.aliyuncs.com";
$bucket = "<yourBucketName>";
$lifecycleConfig = null;
try{
    $ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint);
    $lifecycleConfig = $ossClient->getBucketLifecycle($bucket);
} catch(OssException $e) {
    printf(__FUNCTION__ . ": FAILED\n");
    printf($e->getMessage() . "\n");
    return;
}
print(__FUNCTION__ . ": OK" . "\n");
print($lifecycleConfig->serializeToXml() . "\n");
```

Delete lifecycle rules

The following code provides an example on how to delete lifecycle rules configured for the bucket named `examplebucket`:

```

<? php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log on to OSS because the account has permissions on all API operations. We recommend that you use your RAM user's credentials to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
$accessKeyId = "<yourAccessKeyId>";
$accessKeySecret = "<yourAccessKeySecret>";
// The endpoint of the China (Hangzhou) region is used in this example. Specify the actual endpoint.
$endpoint = "http://oss-cn-hangzhou.aliyuncs.com";
$bucket = "<yourBucketName>";
try{
    $ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint);
    $ossClient->deleteBucketLifecycle($bucket);
} catch(OssException $e) {
    printf(__FUNCTION__ . ": FAILED\n");
    printf($e->getMessage() . "\n");
    return;
}
print(__FUNCTION__ . ": OK" . "\n");

```

3.5.11. Bucket policy

You can use bucket policies to authorize other users to access your OSS resources. This topic describes how to configure, query, and delete a bucket policy.

Scenarios

Bucket policies provide resource-based authorization for users and apply to the following scenarios:

- Grant RAM users of other Alibaba Cloud accounts permissions to access your OSS resources.
You can grant RAM users of other Alibaba Cloud accounts permissions to access your OSS resources.
- Grant anonymous users from certain IP addresses permissions to access your OSS resources.

In some cases, you need to grant anonymous users from a specified IP address permissions to access your OSS resources. For example, you want to grant only users within your enterprise but not users in other regions permissions to access some confidential enterprise documents. In this case, you can configure a bucket policy that allows only access from specified IP addresses to authorize users more efficiently.

For more information about bucket policy configurations and use cases, see [Configure bucket policies to authorize other users to access OSS resources](#). For more information about the structure of a bucket policy, see [Policy structure and syntax](#).

Configure bucket policies

The following code provides an example on how to configure a bucket policy:

```
<? php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log on to OSS because the account has permissions on all API operations. We recommend that you use your RAM user's credentials to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
$accessKeyId = "<yourAccessKeyId>";
$accessKeySecret = "<yourAccessKeySecret>";
// The endpoint of the China (Hangzhou) region is used in this example. Specify the actual endpoint.
$endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
$bucket = "<yourBucketName>";
$ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint, false);
// Specify the bucket policy.
$policy = <<< BBBB
{
    "Version": "1",
    "Statement": [
        {
            "Action": [
                "oss:PutObject",
                "oss:GetObject"
            ],
            "Effect": "Allow",
            "Resource": ["acs:oss:*:*:*/*user1/*"]
        }
    ]
}
BBBB;
try {
    // Configure the bucket policy.
    $ossClient->putBucketPolicy($bucket, $policy);
} catch (OssException $e) {
    printf(__FUNCTION__ . ": FAILED\n");
    printf($e->getMessage() . "\n");
    return;
}
print(__FUNCTION__ . ": OK" . "\n");
```

For more information about how to configure bucket policies, see [PutBucketPolicy](#).

Query bucket policies

The following code provides an example on how to query the policies configured for a bucket:

```
<? php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log on to OSS because the account has permissions on all API operations. We recommend that you use your RAM user's credentials to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
$accessKeyId = "<yourAccessKeyId>";
$accessKeySecret = "<yourAccessKeySecret>";
// The endpoint of the China (Hangzhou) region is used in this example. Specify the actual endpoint.
$endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
$bucket = "<yourBucketName>";
$ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint, false);
try {
    // Query the bucket policies.
    $policy = $ossClient->getBucketPolicy($bucket);
    // Display the bucket policies.
    print($policy);
} catch (OssException $e) {
    printf(__FUNCTION__ . ": FAILED\n");
    printf($e->getMessage() . "\n");
    return;
}
print(__FUNCTION__ . ": OK" . "\n");
```

For more information about how to query bucket policies, see [GetBucketPolicy](#).

Delete bucket policies

The following code provides an example on how to delete the policies configured for a bucket:

```

<? php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to lo
g on to OSS because the account has permissions on all API operations. We recommend that yo
u use your RAM user's credentials to call API operations or perform routine operations and
maintenance. To create a RAM user, log on to the RAM console.
$accessKeyId = "<yourAccessKeyId>";
$accessKeySecret = "<yourAccessKeySecret>";
// The endpoint of the China (Hangzhou) region is used in this example. Specify the actual
endpoint.
$endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
$bucket= "<yourBucketName>";
$ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint, false);
try {
    // Delete the policies configured for the bucket.
    $ossClient->deleteBucketPolicy($bucket);
} catch (OssException $e) {
    printf(__FUNCTION__ . ": FAILED\n");
    printf($e->getMessage() . "\n");
    return;
}
print(__FUNCTION__ . ": OK" . "\n");

```

For more information about how to delete bucket policies, see [DeleteBucketPolicy](#).

3.5.12. Retention policy

You can configure a time-based retention policy for a bucket. A retention policy has a retention period that ranges from one day to 70 years. This topic describes how to create, query, and lock a retention policy.

Background information

OSS supports the Write Once Read Many (WORM) strategy that prevents an object from being deleted or overwritten for a specified period of time.

If a retention policy is not locked within 24 hours after it is created, the retention policy becomes invalid. After the retention policy configured for a bucket is locked, you can upload objects to or read objects from the bucket. However, you cannot delete objects in the bucket or the retention policy within the retention period of the policy. The retention period of the policy cannot be shortened but only be extended. For more information about retention policies, see [Retention policies](#).

Create a retention policy

The following code provides an example on how to create a retention policy:


```

<? php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log
on to OSS because the account has permissions on all API operations. We recommend that yo
u use your RAM user's credentials to call API operations or perform routine operations and
maintenance. To create a RAM user, log on to the RAM console.
$accessKeyId = "<yourAccessKeyId>";
$accessKeySecret = "<yourAccessKeySecret>";
// The endpoint of the China (Hangzhou) region is used in this example. Specify the actual
endpoint.
$endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
$bucket= "<yourBucketName>";
$ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint, false);
try {
    // Create a retention policy and set the retention period to 30 days.
    $wormId = $ossClient->initiateBucketWorm($bucket, 30);
    // Query the ID of the retention policy.
    print($wormId);
} catch (OssException $e) {
    printf(__FUNCTION__ . ": FAILED\n");
    printf($e->getMessage() . "\n");
    return;
}
print(__FUNCTION__ . ": OK" . "\n");

```

For more information about how to create a retention policy, see [InitiateBucketWorm](#).

Cancel an unlocked retention policy

The following code provides an example on how to cancel an unlocked retention policy:

```
<? php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log
on to OSS because the account has permissions on all API operations. We recommend that you
use your RAM user's credentials to call API operations or perform routine operations and
maintenance. To create a RAM user, log on to the RAM console.
$accessKeyId = "<yourAccessKeyId>";
$accessKeySecret = "<yourAccessKeySecret>";
// The endpoint of the China (Hangzhou) region is used in this example. Specify the actual
endpoint.
$endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
$bucket= "<yourBucketName>";
$ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint, false);
try {
    // Cancel the retention policy.
    $ossClient->abortBucketWorm($bucket);
} catch (OssException $e) {
    printf(__FUNCTION__ . ": FAILED\n");
    printf($e->getMessage() . "\n");
    return;
}
print(__FUNCTION__ . ": OK" . "\n");
```

For more information about how to cancel an unlocked retention policy, see [AbortBucketWorm](#).

Lock a retention policy

The following code provides an example on how to lock a retention policy:

```
<? php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log on to OSS because the account has permissions on all API operations. We recommend that you use your RAM user's credentials to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
$accessKeyId = "<yourAccessKeyId>";
$accessKeySecret = "<yourAccessKeySecret>";
// The endpoint of the China (Hangzhou) region is used in this example. Specify the actual endpoint.
$endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
$bucket= "<yourBucketName>";
$ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint, false);
try {
    // Create a retention policy and set the retention period to 30 days.
    $wormId = $ossClient->initiateBucketWorm($bucket, 30);
    // Lock the retention policy.
    $ossClient->completeBucketWorm($bucket, $wormId);
} catch (OssException $e) {
    printf(__FUNCTION__ . ": FAILED\n");
    printf($e->getMessage() . "\n");
    return;
}
print(__FUNCTION__ . ": OK" . "\n");
```

For more information about how to lock a retention policy, see [CompleteBucketWorm](#).

Query a retention policy

The following code provides an example on how to query a retention policy:

```
<? php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to lo
g on to OSS because the account has permissions on all API operations. We recommend that yo
u use your RAM user's credentials to call API operations or perform routine operations and
maintenance. To create a RAM user, log on to the RAM console.
$accessKeyId = "<yourAccessKeyId>";
$accessKeySecret = "<yourAccessKeySecret>";
// The endpoint of the China (Hangzhou) region is used in this example. Specify the actual
endpoint.
$endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
$bucket= "<yourBucketName>";
$ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint, false);
try {
    // Query the retention policy.
    $config = $ossClient->getBucketWorm($bucket);
    // Display the information about the retention policy.
    printf("WormId:%s\n", $config->getWormId());
    printf("State:%s\n", $config->getState());
    printf("Day:%d\n", $config->getDay());
} catch (OssException $e) {
    printf(__FUNCTION__ . ": FAILED\n");
    printf($e->getMessage() . "\n");
    return;
}
print(__FUNCTION__ . ": OK" . "\n");
```

For more information about how to query a retention policy, see [GetBucketWorm](#).

Extend the retention period of a retention policy

The following code provides an example on how to extend the retention period of a locked retention policy:

```

<? php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log
on to OSS because the account has permissions on all API operations. We recommend that yo
u use your RAM user's credentials to call API operations or perform routine operations and
maintenance. To create a RAM user, log on to the RAM console.
$accessKeyId = "<yourAccessKeyId>";
$accessKeySecret = "<yourAccessKeySecret>";
// The endpoint of the China (Hangzhou) region is used in this example. Specify the actual
endpoint.
$endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
$bucket= "<yourBucketName>";
$wormId = "<yourWormId>";
$ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint, false);
try {
    // Extend the retention period of the locked retention policy to 120 days.
    $ossClient->extendBucketWorm($bucket, $wormId, 120);
} catch (OssException $e) {
    printf(__FUNCTION__ . ": FAILED\n");
    printf($e->getMessage() . "\n");
    return;
}
print(__FUNCTION__ . ": OK" . "\n");

```

For more information about how to extend the retention period of a locked retention policy, see [ExtendBucketWorm](#).

3.5.13. Enable pay-by-requester

When pay-by-requester is enabled for a bucket in Object Storage Service (OSS), the requester is charged for requests and traffic fees instead of the bucket owner. The bucket owner is charged for only storage costs. You can enable pay-by-requester for a bucket to share data in the bucket without paying for the request and traffic fees incurred by the access to your bucket.

Enable pay-by-requester

The following code provides an example on how to enable pay-by-requester for a bucket:

```
<?php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
$accessKeyId = "yourAccessKeyId";
$accessKeySecret = "yourAccessKeySecret";
// In this example, the China (Hangzhou) region is used as the endpoint. Specify your actual endpoint.
$endpoint = "http://oss-cn-hangzhou.aliyuncs.com";
// Specify the name of the bucket. Example: examplebucket.
$bucket = "examplebucket";
$ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint, false);
try {
    // Enable pay-by-requester for the bucket.
    $ossClient->putBucketRequestPayment($bucket, "Requester");
} catch (OssException $e) {
    printf(__FUNCTION__ . ": FAILED\n");
    printf($e->getMessage() . "\n");
    return;
}
print(__FUNCTION__ . ": OK" . "\n");
```

Query the pay-by-requester configurations of a bucket

The following code provides an example on how to query the pay-by-requester configurations of a bucket:

```

<?php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
$accessKeyId = "yourAccessKeyId";
$accessKeySecret = "yourAccessKeySecret";
// In this example, the China (Hangzhou) region is used as the endpoint. Specify your actual endpoint.
$endpoint = "http://oss-cn-hangzhou.aliyuncs.com";
// Specify the name of the bucket. Example: examplebucket.
$bucket = "examplebucket";
$ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint, false);
try {
    // Query the pay-by-requester configurations of the bucket.
    $payer = $ossClient->getBucketRequestPayment($bucket);
    // Display the pay-by-requester configurations.
    print($payer);
} catch (OssException $e) {
    printf(__FUNCTION__ . ": FAILED\n");
    printf($e->getMessage() . "\n");
    return;
}
print(__FUNCTION__ . ": OK" . "\n");

```

Specify that third parties are charged for access to objects

If you specify that third parties are charged for access to objects, requesters must include the `x-oss-request-payer:requester` header in the HTTP requests to perform operations on your objects. If this header is not included, an error is returned.

The following code provides an example on how to specify that third parties are charged when they access objects by calling the `PutObject`, `GetObject`, and `DeleteObject` operations. You can use the same method to specify that third parties are charged when they call other operations to access your objects.

```

<?php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
$accessKeyId = "yourAccessKeyId";
$accessKeySecret = "yourAccessKeySecret";
// In this example, the China (Hangzhou) region is used as the endpoint. Specify your actual endpoint.
$endpoint = "http://oss-cn-hangzhou.aliyuncs.com";
// Specify the name of the bucket. Example: examplebucket.
$bucket = "examplebucket";
// Specify the full path of the object. The full path cannot contain the bucket name. Example: exampledir/exampleobject.txt.
$object = "exampledir/exampleobject.txt";
$ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint, false);
// Enable pay-by-requester for the bucket.
$options = array(
    OssClient::OSS_HEADERS => array(
        OssClient::OSS_REQUEST_PAYER => 'requester',
    ));
try {
    // Specify the payer when a third party calls the PutObject operation to access objects
    .
    $content = "hello";
    $ossClient->putObject($bucket, $object, $content, $options);
    // Specify the payer when a third party calls the GetObject operation to access objects
    .
    $ossClient->getObject($bucket, $object, $options);
    // Specify the payer when a third party calls the DeleteObject operation to access objects
    .
    $ossClient->deleteObject($bucket, $object, $options);
} catch (OssException $e) {
    printf(__FUNCTION__ . ": FAILED\n");
    printf($e->getMessage() . "\n");
    return;
}
print(__FUNCTION__ . ": OK" . "\n");

```

References

- For more information about the API operation that you can call to configure pay-by-requester for a bucket, see [PutBucketRequestPayment](#).
- For more information about the API operation that you can call to query the pay-by-requester

configurations of a bucket, see [GetBucketRequestPayment](#).

3.5.14. Hotlink protection

This topic describes how to use hotlink protection.

To prevent your data on Object Storage Service (OSS) from being leached, OSS supports hotlink protection through the Referer field settings in the HTTP header, including the following parameters:

- **Referer whitelist:** Used to allow access only for specified domains to OSS data.
- **Empty Referer:** Determines whether the Referer can be empty. If it is not allowed, only requests with the Referer field in their HTTP or HTTPS headers can access OSS data.

For more information about hotlink protection, see [Hotlink protection](#). For the complete sample code that is used to configure hotlink protection, visit [GitHub](#).

Configure hotlink protection

The following code provides an example on how to configure a Referer whitelist for a bucket:

```
<?php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
use OSS\Model\RefererConfig;
// The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console at https://ram.console.aliyun.com.
$accessKeyId = "<yourAccessKeyId>";
$accessKeySecret = "<yourAccessKeySecret>";
// In this example, the endpoint of the China (Hangzhou) region is used. Specify your actual endpoint.
$endpoint = "http://oss-cn-hangzhou.aliyuncs.com";
$bucket = "<yourBucketName>";
$refererConfig = new RefererConfig();
// Allow empty Referers.
$refererConfig->setAllowEmptyReferer(true);
// Add Referers to the Referer whitelist. You can use asterisks (*) and question marks (?) as wildcards in Referers.
$refererConfig->addReferer("example.com");
$refererConfig->addReferer("example.net");
try{
    $ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint);
    $ossClient->putBucketReferer($bucket, $refererConfig);
} catch(OssException $e) {
    printf(__FUNCTION__ . ": FAILED\n");
    printf($e->getMessage() . "\n");
    return;
}
print(__FUNCTION__ . ": OK" . "\n");
```

Obtain hotlink protection configurations

The following code provides an example on how to query a Referer whitelist of a bucket:

```

<?php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
use OSS\Model\RefererConfig;
// The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console at https://ram.console.aliyun.com.
$accessKeyId = "<yourAccessKeyId>";
$accessKeySecret = "<yourAccessKeySecret>";
// In this example, the endpoint of the China (Hangzhou) region is used. Specify your actual endpoint.
$endpoint = "http://oss-cn-hangzhou.aliyuncs.com";
$bucket = "<yourBucketName>";
$refererConfig = null;
try{
    $ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint);
    $refererConfig = $ossClient->getBucketReferer($bucket);
} catch(OssException $e) {
    printf(__FUNCTION__ . ": FAILED\n");
    printf($e->getMessage() . "\n");
    return;
}
print(__FUNCTION__ . ": OK" . "\n");
print($refererConfig->serializeToXml() . "\n");

```

Clear hotlink protection configurations

The following code provides an example on how to clear a Referer whitelist of a bucket:

```

<?php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
use OSS\Model\RefererConfig;
// The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console at https://ram.console.aliyun.com.
$accessKeyId = "<yourAccessKeyId>";
$accessKeySecret = "<yourAccessKeySecret>";
// In this example, the endpoint of the China (Hangzhou) region is used. Specify your actual endpoint.
$endpoint = "http://oss-cn-hangzhou.aliyuncs.com";
$bucket = "<yourBucketName>";
$refererConfig = new RefererConfig();
try{
    $ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint);
    // The hotlink protection configurations of a bucket cannot be directly cleared. You must configure a new hotlink protection rule that allows empty Referer fields to overwrite the existing hotlink protection configurations.
    $ossClient->putBucketReferer($bucket, $refererConfig);
} catch(OssException $e) {
    printf(__FUNCTION__ . ": FAILED\n");
    printf($e->getMessage() . "\n");
    return;
}
print(__FUNCTION__ . ": OK" . "\n");

```

3.5.15. Logging

A large number of logs are generated when Object Storage Service (OSS) resources are accessed. After you enable and configure logging for a bucket, OSS generates logs every hour in accordance with predefined naming conventions and then stores the logs as objects in a specified bucket.

Enable logging for a bucket

The following code provides an example on how to enable logging for a bucket:

```

<?php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
// The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
$accessKeyId = "yourAccessKeyId";
$accessKeySecret = "yourAccessKeySecret";
// Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com.
$endpoint = "yourEndpoint";
// Specify the name of the bucket for which you want to enable logging. Example: examplebucket.
$bucket = "examplebucket";
$options = array();
// Specify the destination bucket used to store the log objects.
$targetBucket = "destbucket";
// Specify the directory in which the log objects are stored. If you specify this parameter, log objects are stored in a specified directory of the destination bucket. If you do not specify this parameter, log objects are stored in the root directory of the destination bucket.
$targetPrefix = "log/";
try {
    $ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint);
    // Enable logging for the bucket.
    $ossClient->putBucketLogging($bucket, $targetBucket, $targetPrefix, $options);
} catch (OssException $e) {
    printf(__FUNCTION__ . ": FAILED\n");
    printf($e->getMessage() . "\n");
    return;
}
print(__FUNCTION__ . ": OK" . "\n");

```

View the logging configurations of a bucket

The following code provides an example on how to view the logging configurations of a bucket:

```

<?php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
// The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
$accessKeyId = "yourAccessKeyId";
$accessKeySecret = "yourAccessKeySecret";
// Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com.
$endpoint = "yourEndpoint";
// Specify the bucket name. Example: examplebucket.
$bucket = "examplebucket";
$loggingConfig = null;
$options = array();
try {
    $ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint);
    // View the logging configurations of the bucket.
    $loggingConfig = $ossClient->getBucketLogging($bucket, $options);
} catch (OssException $e) {
    printf(__FUNCTION__ . ": FAILED\n");
    printf($e->getMessage() . "\n");
    return;
}
print(__FUNCTION__ . ": OK" . "\n");
print($loggingConfig->serializeToXml() . "\n");

```

Disable logging for a bucket

The following code provides an example on how to disable logging for a bucket:

```

<?php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
// The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
$accessKeyId = "yourAccessKeyId";
$accessKeySecret = "yourAccessKeySecret";
// Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com.
$endpoint = "yourEndpoint";
// Specify the bucket name. Example: examplebucket.
$bucket = "examplebucket";
try {
    $ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint);
    // Disable logging for the bucket.
    $ossClient->deleteBucketLogging($bucket);
} catch (OssException $e) {
    printf(__FUNCTION__ . ": FAILED\n");
    printf($e->getMessage() . "\n");
    return;
}
print(__FUNCTION__ . ": OK" . "\n");

```

References

- For the complete sample code that is used to configure logging for a bucket, visit [GitHub](#).
- For more information about the API operation that you can call to enable logging for a bucket, see [PutBucketLogging](#).
- For more information about the API operation that you can call to view the logging configurations of a bucket, see [GetBucketLogging](#).
- For more information about the API operation that you can call to disable logging for a bucket, see [DeleteBucketLogging](#).

3.5.16. Static website hosting

You can enable static website hosting for buckets. After you host a static website on a bucket, you can access the bucket to visit the website. You are automatically redirected to the specified default homepage or default 404 page.

Configure static website hosting

The following code provides an example on how to configure static website hosting:

```

<?php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
use OSS\Model\WebsiteConfig;
// The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in Object Storage Service (OSS) is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
$accessKeyId = "yourAccessKeyId";
$accessKeySecret = "yourAccessKeySecret";
// In this example, the endpoint of the China (Hangzhou) region is used. Specify your actual endpoint.
$endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
// Specify the bucket name. Example: examplebucket.
$bucket = "examplebucket";
// Set the default homepage to index.html and the default 404 page to error.html for the static website hosted on the bucket.
$websiteConfig = new WebsiteConfig("index.html", "error.html");
try {
    $ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint);
    $ossClient->putBucketWebsite($bucket, $websiteConfig);
} catch (OssException $e) {
    printf(__FUNCTION__ . ": FAILED\n");
    printf($e->getMessage() . "\n");
    return;
}
print(__FUNCTION__ . ": OK" . "\n");

```

Query static website hosting configurations

The following code provides an example on how to query static website hosting configurations:


```
<?php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
// The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
$accessKeyId = "yourAccessKeyId";
$accessKeySecret = "yourAccessKeySecret";
// In this example, the endpoint of the China (Hangzhou) region is used. Specify your actual endpoint.
$endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
// Specify the bucket name. Example: examplebucket.
$bucket = "examplebucket";
$websiteConfig = null;
try{
    $ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint);
    // Query the static website hosting configurations.
    $websiteConfig = $ossClient->getBucketWebsite($bucket);
} catch(OssException $e) {
    printf(__FUNCTION__ . ": FAILED\n");
    printf($e->getMessage() . "\n");
    return;
}
print(__FUNCTION__ . ": OK" . "\n");
print($websiteConfig->serializeToXml() . "\n");
```

Delete static website hosting configurations

The following code provides an example on how to delete static website hosting configurations:

```

<?php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
// The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
$accessKeyId = "yourAccessKeyId";
$accessKeySecret = "yourAccessKeySecret";
// In this example, the endpoint of the China (Hangzhou) region is used. Specify your actual endpoint.
$endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
// Specify the bucket name. Example: examplebucket.
$bucket = "examplebucket";
try{
    $ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint);
    // Delete the static website hosting configurations.
    $ossClient->deleteBucketWebsite($bucket);
} catch(OssException $e) {
    printf(__FUNCTION__ . ": FAILED\n");
    printf($e->getMessage() . "\n");
    return;
}
print(__FUNCTION__ . ": OK" . "\n");

```

References

- For the complete sample code that is used to configure static website hosting, visit [GitHub](#).
- For more information about the API operation that you can call to configure static website hosting, see [PutBucketWebsite](#).
- For more information about the API operation that you can call to query static website hosting configurations, see [GetBucketWebsite](#).
- For more information about the API operation that you can call to delete static website hosting configurations, see [DeleteBucketWebsite](#).

3.5.17. CORS

Cross-origin resource sharing (CORS) allows web applications to access resources that belong to different origins. Object Storage Service (OSS) provides CORS API operations for cross-origin access control.

Configure CORS rules

The following code provides an example on how to configure CORS rules for a bucket named examplebucket:

```

<?php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
use OSS\Model\CorsConfig;
use OSS\Model\CorsRule;
// The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to access OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
$accessKeyId = "yourAccessKeyId";
$accessKeySecret = "yourAccessKeySecret";
// Set endpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set endpoint to https://oss-cn-hangzhou.aliyuncs.com.
$endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
// Specify the name of the bucket. Example: examplebucket.
$bucket = "examplebucket";
$corsConfig = new CorsConfig();
$rule = new CorsRule();
// Specify the response headers based on which to allow cross-origin requests. You can specify multiple allowed headers. Only one asterisk (*) can be used as the wildcard character for each allowed header.
// If you do not have special requirements, we recommend that you set AllowedHeader to an asterisk (*).
$rule->addAllowedHeader("*");
// Specify the response headers for allowed access requests from applications. You can specify multiple exposed headers. Exposed headers cannot contain asterisks (*).
$rule->addExposeHeader("x-oss-header");
// Specify the origins from which cross-origin requests are allowed. You can specify multiple allowed origins. Only one asterisk (*) can be used as the wildcard character for each allowed origin.
$rule->addAllowedOrigin("https://example.com:8080");
$rule->addAllowedOrigin("https://*.aliyun.com");
// If you set AllowedOrigin to an asterisk (*), requests from all origins are allowed.
//$rule->addAllowedOrigin("*");
// Specify the HTTP methods that cross-origin requests are allowed to use.
$rule->addAllowedMethod("POST");
// Specify the period of time in which the browser can cache the response to an OPTIONS preflight request for specific resources. Unit: seconds.
$rule->setMaxAgeSeconds(10);
// You can configure up to 10 CORS rules for each bucket.
$corsConfig->addRule($rule);
try{
    $ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint);
    // The existing rules are overwritten.
    $ossClient->putBucketCors($bucket, $corsConfig);
} catch(OssException $e) {
    printf(__FUNCTION__ . ": FAILED\n");
    printf($e->getMessage() . "\n");
}

```

```

        printf($e->getMessage() . "\n");
        return;
    }
    print(__FUNCTION__ . ": OK" . "\n");

```

Query CORS rules

The following code provides an example on how to query the CORS rules configured for a bucket named `examplebucket`:

```

<?php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
// The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to access OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
$accessKeyId = "yourAccessKeyId";
$accessKeySecret = "yourAccessKeySecret";
// Set endpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set endpoint to https://oss-cn-hangzhou.aliyuncs.com.
$endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
// Specify the name of the bucket. Example: examplebucket.
$bucket = "examplebucket";
$corsConfig = null;
try{
    $ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint);
    $corsConfig = $ossClient->getBucketCors($bucket);
} catch(OssException $e) {
    printf(__FUNCTION__ . ": FAILED\n");
    printf($e->getMessage() . "\n");
    return;
}
print(__FUNCTION__ . ": OK" . "\n");
print($corsConfig->serializeToXml() . "\n");

```

Delete CORS rules

The following code provides an example on how to delete all CORS rules configured for a bucket named `examplebucket`:

```
<?php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
// The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to access OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
$accessKeyId = "yourAccessKeyId";
$accessKeySecret = "yourAccessKeySecret";
// Set endpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set endpoint to https://oss-cn-hangzhou.aliyuncs.com.
$endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
// Specify the name of the bucket. Example: examplebucket.
$bucket = "examplebucket";
try{
    $ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint);
    $ossClient->deleteBucketCors($bucket);
} catch(OssException $e) {
    printf(__FUNCTION__ . ": FAILED\n");
    printf($e->getMessage() . "\n");
    return;
}
print(__FUNCTION__ . ": OK" . "\n");
```

References

- For more information about the complete sample code of CORS, visit [GitHub](#).
- For more information about the API operation that you can call to configure CORS rules, see [Put Bucket Cors](#).
- For more information about the API operation that you can call to query CORS rules, see [Get Bucket Cors](#).
- For more information about the API operation that you can call to delete CORS rules, see [Delete Bucket Cors](#).

3.5.18. Map custom domain names

This topic describes how to map a custom domain name to a bucket.

You can add a CNAME record to map the custom domain to a specified bucket. You can then use the custom domain to access Object Storage Service (OSS) resources. If your domain name is example.com, you can use the `http://img.example.com/x.jpg` format to access images. After you migrate data to OSS, you can map the custom domain and use the original IP address to access images.

For more information about CNAMEs, see [Map custom domain names](#) in OSS Developer Guide.

Add CNAME records

The following code provides an example on how to add a CNAME record for a bucket:

```
<?php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS, because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine operations and maintenance. To create a RAM user, visit https://ram.console.aliyun.com.
$accessKeyId = "<yourAccessKeyId>";
$accessKeySecret = "<yourAccessKeySecret>";
// The China (Hangzhou) region is used in this example as the endpoint. Specify the actual endpoint.
$endpoint = "http://oss-cn-hangzhou.aliyuncs.com";
$bucket = "<yourBucketName>";
// Specify a custom domain name.
$myDomain = '<yourDomainName>';
try {
    $ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint);
    // Add a CNAME record.
    $ossClient->addBucketCname($bucket, $myDomain);
} catch (OssException $e) {
    printf(__FUNCTION__ . ": FAILED\n");
    printf($e->getMessage() . "\n");
    return;
}
print(__FUNCTION__ . ": OK" . "\n");
```

View CNAME records

The following code provides an example on how to obtain an existing CNAME record of a bucket:

```
<?php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS, because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine operations and maintenance. To create a RAM user, visit https://ram.console.aliyun.com.
$accessKeyId = "<yourAccessKeyId>";
$accessKeySecret = "<yourAccessKeySecret>";
// The China (Hangzhou) region is used in this example as the endpoint. Specify the actual endpoint.
$endpoint = "http://oss-cn-hangzhou.aliyuncs.com";
$bucket = "<yourBucketName>";
try {
    $ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint);
    // View the CNAME record.
    $cnameConfig = $ossClient->getBucketCname($bucket);
} catch (OssException $e) {
    printf(__FUNCTION__ . ": FAILED\n");
    printf($e->getMessage() . "\n");
    return;
}
var_dump($cnameConfig);
print(__FUNCTION__ . ": OK" . "\n");
```

Delete CNAME records

The following code provides an example on how to delete a CNAME record:

```
<?php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS, because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine operations and maintenance. To create a RAM user, visit https://ram.console.aliyun.com.
$accessKeyId = "<yourAccessKeyId>";
$accessKeySecret = "<yourAccessKeySecret>";
// The China (Hangzhou) region is used in this example as the endpoint. Specify the actual endpoint.
$endpoint = "http://oss-cn-hangzhou.aliyuncs.com";
$bucket = "<yourBucketName>";
// Specify a custom domain name.
$myDomain = '<yourDomainName>';
try {
    $ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint);
    // Delete the CNAME record.
    $ossClient->deleteBucketCname($bucket, $myDomain);
} catch (OssException $e) {
    printf(__FUNCTION__ . ": FAILED\n");
    printf($e->getMessage() . "\n");
    return;
}
print(__FUNCTION__ . ": OK" . "\n");
```

3.6. Upload objects

3.6.1. Overview

In OSS, an object is the basic unit for data operations. OSS PHP SDK provides the following file upload methods:

- **Simple upload**: supports the upload of a file up to 5 GB in size.
- **Append upload**: supports the upload of a file up to 5 GB in size.
- **Multipart upload**: supports the upload of a file up to 48.8 TB in size. It applies to the upload of large files

 **Note** For more information about the usage scenarios of each upload method, see “Upload files” in OSS Developer Guide. After you complete the file upload, you can perform **upload callback**.

3.6.2. Simple upload

In simple upload, you can upload strings or local files. This topic describes how to upload strings and local files by using Object Storage Service (OSS) SDK for PHP.

Upload a string

The following code provides an example on how to upload a string to the `exampleobject.txt` object in the `exampledir` directory of the `examplebucket` bucket:

```
<?php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log on to OSS because the account has permissions on all API operations. We recommend that you use a Resource Access Management (RAM) user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
$accessKeyId = "yourAccessKeyId";
$accessKeySecret = "yourAccessKeySecret";
// Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com.
$endpoint = "yourEndpoint";
// Specify the name of the bucket to which you want to upload the string. Example: examplebucket.
$bucket = "examplebucket";
// Specify the full path of the object to which you want to upload the string. Example: exampledir/exampleobject.txt. The full path of the object cannot contain the bucket name.
$object = "exampledir/exampleobject.txt";
// Specify the string to upload.
$content = "Hello OSS";
try{
    $ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint);
    $ossClient->putObject($bucket, $object, $content);
} catch(OssException $e) {
    printf(__FUNCTION__ . ": FAILED\n");
    printf($e->getMessage() . "\n");
    return;
}
print(__FUNCTION__ . "OK" . "\n");
// You can configure headers in upload requests to specify the information of the uploaded string. For example, you can set the object access control list (ACL) to private and configure the user metadata of the object.
$options = array(
    OssClient::OSS_HEADERS => array(
        'x-oss-object-acl' => 'private',
        'x-oss-meta-info' => 'your info'
    ),
);
```

```
);  
try{  
    $ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint);  
    $ossClient->putObject($bucket, $object, $content, $options);  
} catch(OssException $e) {  
    printf(__FUNCTION__ . ": FAILED\n");  
    printf($e->getMessage() . "\n");  
    return;  
}  
print(__FUNCTION__ . "OK" . "\n");
```

Upload an object

The following code provides an example on how to upload a local file named `examplefile.txt` to the `exampledir` directory of the `examplebucket` bucket. After the local file is uploaded, the object in the `exampledir` directory is named `exampleobject.txt`.

```

<?php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log on to OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
$accessKeyId = "yourAccessKeyId";
$accessKeySecret = "yourAccessKeySecret";
// Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com.
$endpoint = "yourEndpoint";
// Specify the name of the bucket to which you want to upload the object. Example: examplebucket.
$bucket = "examplebucket";
// Specify the full path of the object. Example: exampledir/exampleobject.txt. The full path of the object cannot contain the bucket name.
$object = "exampledir/exampleobject.txt";
// <yourLocalFile> consists of a local file path and a file name with an extension. Example: /users/local/myfile.txt.
// Specify the full path of the local file to upload. Example: D:\\localpath\\examplefile.txt. By default, if you do not specify the local file, the local file is uploaded from the path of the project to which the sample program belongs.
$filePath = "D:\\localpath\\examplefile.txt";
try{
    $ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint);
    $ossClient->uploadFile($bucket, $object, $filePath);
} catch(OssException $e) {
    printf(__FUNCTION__ . ": FAILED\n");
    printf($e->getMessage() . "\n");
    return;
}
print(__FUNCTION__ . "OK" . "\n");

```

3.6.3. Append upload

You can call the AppendObject operation to append content to existing appendable objects.

Usage notes

- If the object to which you want to append content does not exist, an appendable object is created when you call the AppendObject operation.
- When you call the AppendObject operation to append content to an object, you may encounter a few situations: If the object is an appendable object, and the specified position from which the append operation starts is equal to the current object length, the content is appended to the end of

the object. If the specified position from which the append operation starts is not equal to the current object length, a `PositionNotEqualToLength` exception is thrown. If the object is not an appendable object, an `ObjectNotAppendable` exception is thrown.

- The `CopyObject` operation cannot be performed on appendable objects.

Upload strings by using append upload

The following code provides an example on how to use append upload to upload strings in sequence to the `srcexampleobject.txt` object in the `examplebucket` bucket:

```
<?php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access Object Storage Service (OSS) because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
$accessKeyId = "yourAccessKeyId";
$accessKeySecret = "yourAccessKeySecret";
// In this example, the endpoint of the China (Hangzhou) region is used. Specify the endpoint based on your business requirements. For more information about the regions and endpoints, see Regions and endpoints.
$endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
// Specify the name of the bucket. For more information about the naming conventions for buckets, see bucket.
$bucket = "examplebucket";
// Specify the full path of the object. The full path of the object cannot contain the bucket name. For more information about the naming conventions for objects, see object.
$object = "srcexampleobject.txt";
// Specify the strings that you want to upload in sequence. The strings are Hello OSS, Hi OSS, and OSS OK for the first, second, and third append upload.
$content_array = array('Hello OSS', 'Hi OSS', 'OSS OK');
$options = array(
    'headers' => array(
        // Specify the caching behavior of the web page when the object is downloaded.
        // 'Cache-Control' => 'no-cache',
        // Specify the name of the object when the object is downloaded.
        // 'Content-Disposition' => 'attachment;filename=oss_download.jpg',
        // Specify the content encoding format of the object.
        // 'Content-Encoding' => 'utf-8',
        // Specify the expiration time of the request.
        // 'Expires' => 'Fri, 31 Dec 2021 16:57:01 GMT',
        // Specify whether to overwrite the object with the same name in the append upload. In this example, this parameter is set to true, which indicates that the object with the same name cannot be overwritten.
        // 'x-oss-forbid-overwrite' => 'true',
        // Specify the server-side encryption method that is used to encrypt each part of the object to upload.
```

```

        // 'x-oss-server-side-encryption'=> 'AES256',
        // Specify the algorithm that is used to encrypt the object.
        // 'x-oss-server-side-data-encryption'=>'SM4',
        // Specify the storage class of the object.
        // 'x-oss-storage-class' => 'Standard',
        // Specify the access control list (ACL) of the object.
        // 'x-oss-object-acl' => 'private',
        ),
    );
}
try{
    $ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint);
    // Perform the first append upload. The position for the first append upload is 0, and
    the position for the next append upload is included in the response. The position from whic
    h the next append upload starts is the current length of the object.
    $position = $ossClient->appendObject($bucket, $object, $content_array[0], 0,$options);
    $position = $ossClient->appendObject($bucket, $object, $content_array[1], $position,$op
    tions);
    $position = $ossClient->appendObject($bucket, $object, $content_array[2], $position,$op
    tions);
} catch(OssException $e) {
    printf(__FUNCTION__ . ": FAILED\n");
    printf($e->getMessage() . "\n");
    return;
}
print(__FUNCTION__ . ": OK" . "\n");

```

Upload a local file by using append upload

The following code provides an example on how to use append upload to upload the content of the `examplefilea.txt`, `examplefileb.txt`, and `examplefilec.txt` files in sequence to the `exampleobject.txt` object in the `examplebucket` bucket:

```

<?php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
$accessKeyId = "yourAccessKeyId";
$accessKeySecret = "yourAccessKeySecret";
// In this example, the endpoint of the China (Hangzhou) region is used. Specify the endpoint based on your business requirements.
$endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
// Specify the name of the bucket.
$bucket = "examplebucket";
// Specify the full path of the object. The full path of the object cannot contain the bucket name.
$object = "exampleobject.txt";
// Specify the full paths of the local files to upload. By default, if you do not specify the full path of a local file, the local file is uploaded from the path of the project to which the sample program belongs.
$filePath = "D:\\localpath\\examplefilea.txt";
$filePath1 = "D:\\localpath\\examplefileb.txt";
$filePath2 = "D:\\localpath\\examplefilec.txt";
try{
    $ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint);
    // Perform the first append upload. The position for the first append upload is 0, and the position for the next append upload is included in the response. The position from which the next append upload starts is the current length of the object.
    $position = $ossClient->appendFile($bucket, $object, $filePath, 0);
    $position = $ossClient->appendFile($bucket, $object, $filePath1, $position);
    $position = $ossClient->appendFile($bucket, $object, $filePath2, $position);
} catch(OssException $e) {
    printf(__FUNCTION__ . ": FAILED\n");
    printf($e->getMessage() . "\n");
    return;
}
print(__FUNCTION__ . ": OK" . "\n");

```

References

For more information about the API operation that you can call to perform append upload, see [AppendObject](#).

3.6.4. Multipart upload

By using the multipart upload feature provided by OSS, you can split a large object into multiple parts and upload them separately. After all parts are uploaded, call the CompleteMultipartUpload operation to combine these parts into a single object to implement resumable upload.

Multipart upload process

To implement multipart upload, perform the following operations:

1. Initiate a multipart upload task.

Call `$ossClient->initiateMultipartUpload` to obtain a unique upload ID in OSS.

2. Upload the parts.

Call `$ossClient->uploadPart` to upload the parts.

Note

- For parts that are uploaded by a multipart upload task with a specific upload ID, part numbers identify their relative positions in an object. If you upload a part with the same part number as an existing part, the existing part is overwritten by the uploaded part.
- OSS includes the MD5 hash of part data in the ETag header and returns the MD5 hash to the user.
- OSS calculates the MD5 hash of uploaded data and compares the MD5 hash with the MD5 hash calculated by the SDK. If the two hashes are different, the InvalidDigest error code is returned.

3. Complete the multipart upload task.

Call `$ossClient->completeMultipartUpload` to combine all parts into a complete object.

For the complete code used to perform multipart upload, visit [GitHub](#).

Complete sample code of multipart upload

The following code provides a complete example that describes the process of multipart upload:

```
<? php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
use OSS\Core\OssUtil;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log on to OSS because the account has permissions on all API operations. We recommend that you use your RAM user's credentials to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
$accessKeyId = "<yourAccessKeyId>";
$accessKeySecret = "<yourAccessKeySecret>";
// The endpoint of the China (Hangzhou) region is used in this example. Specify the actual endpoint.
$endpoint = "http://oss-cn-hangzhou.aliyuncs.com";
$bucket = "yourBucketName";
```

```

$bucket= "<yourBucketName>";
$object = "<yourObjectName>";
$uploadFile = "<yourLocalFile>";
/**
 * Step 1: Initiate a multipart upload task and obtain the upload ID.
 */
try{
    $ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint);
    // Obtain the upload ID. The upload ID is the unique identifier for a multipart upload
    task. You can initiate related operations such as cancel or query a multipart upload task b
    ased on the upload ID.
    $uploadId = $ossClient->initiateMultipartUpload($bucket, $object);
} catch(OssException $e) {
    printf(__FUNCTION__ . ": initiateMultipartUpload FAILED\n");
    printf($e->getMessage() . "\n");
    return;
}
print(__FUNCTION__ . ": initiateMultipartUpload OK" . "\n");
/**
 * Step 2: Upload the parts.
 */
$partSize = 10 * 1024 * 1024;
$uploadFileSize = sprintf('%u',filesize($uploadFile));
$pieces = $ossClient->generateMultiuploadParts($uploadFileSize, $partSize);
$responseUploadPart = array();
$uploadPosition = 0;
$isCheckMd5 = true;
foreach ($pieces as $i => $piece) {
    $fromPos = $uploadPosition + (integer)$piece[$ossClient::OSS_SEEK_TO];
    $toPos = (integer)$piece[$ossClient::OSS_LENGTH] + $fromPos - 1;
    $uploadOptions = array(
        // Specify the local file you want to upload.
        $ossClient::OSS_FILE_UPLOAD => $uploadFile,
        // Configure part numbers.
        $ossClient::OSS_PART_NUM => ($i + 1),
        // Specify the position from which to start multipart upload.
        $ossClient::OSS_SEEK_TO => $fromPos,
        // Specify the object size.
        $ossClient::OSS_LENGTH => $toPos - $fromPos + 1,
        // Specify whether to enable MD5 verification. A value of true indicates that the M
        D5 verification is enabled.
        $ossClient::OSS_CHECK_MD5 => $isCheckMd5,
    );
    // Enable MD5 verification.
    if ($isCheckMd5) {
        $contentMd5 = OssUtil::getMd5SumForFile($uploadFile, $fromPos, $toPos);
        $uploadOptions[$ossClient::OSS_CONTENT_MD5] = $contentMd5;
    }
    try {
        // Upload the parts.
        $responseUploadPart[] = $ossClient->uploadPart($bucket, $object, $uploadId, $uploadOptions);
    } catch(OssException $e) {
        printf(__FUNCTION__ . ": initiateMultipartUpload, uploadPart - part#{ $i } FAILED\n");
    }
}

```



```

        printf($e->getMessage() . "\n");
        return;
    }
    printf(__FUNCTION__ . ": initiateMultipartUpload, uploadPart - part#{ $i } OK\n");
}
// $uploadParts is an array that consists of the ETags and the PartNumbers of each part.
$uploadParts = array();
foreach ($responseUploadPart as $i => $eTag) {
    $uploadParts[] = array(
        'PartNumber' => ($i + 1),
        'ETag' => $eTag,
    );
}
/**
 * Step 3: Complete the upload.
 */
try {
    // All valid $uploadParts are required for the completeMultipartUpload operation. OSS v
    erifies the validity of each part after OSS receives the submitted $uploadParts. After all
    parts are validated, OSS combines these parts into a complete object.
    $ossClient->completeMultipartUpload($bucket, $object, $uploadId, $uploadParts);
} catch (OssException $e) {
    printf(__FUNCTION__ . ": completeMultipartUpload FAILED\n");
    printf($e->getMessage() . "\n");
    return;
}
printf(__FUNCTION__ . ": completeMultipartUpload OK\n");

```

Upload a local file by using multipart upload

The following code provides an example on how to upload a local file to OSS by using multipart upload:

```
<? php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log on to OSS because the account has permissions on all API operations. We recommend that you use your RAM user's credentials to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
$accessKeyId = "<yourAccessKeyId>";
$accessKeySecret = "<yourAccessKeySecret>";
// The endpoint of the China (Hangzhou) region is used in this example. Specify the actual endpoint.
$endpoint = "http://oss-cn-hangzhou.aliyuncs.com";
$bucket = "<yourBucketName>";
$object = "<yourObjectName>";
$file = "<yourLocalFile>";
$options = array(
    OssClient::OSS_CHECK_MD5 => true,
    OssClient::OSS_PART_SIZE => 1,
);
try{
    $ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint);
    $ossClient->multiuploadFile($bucket, $object, $file, $options);
} catch(OssException $e) {
    printf(__FUNCTION__ . ": FAILED\n");
    printf($e->getMessage() . "\n");
    return;
}
print(__FUNCTION__ . ": OK" . "\n");
```

Upload a local folder by using multipart upload

The following code provides an example on how to upload a local folder, including all files contained in the folder, to OSS by using multipart upload:

```
<? php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log
on to OSS because the account has permissions on all API operations. We recommend that yo
u use your RAM user's credentials to call API operations or perform routine operations and
maintenance. To create a RAM user, log on to the RAM console.
$accessKeyId = "<yourAccessKeyId>";
$accessKeySecret = "<yourAccessKeySecret>";
// The endpoint of the China (Hangzhou) region is used in this example. Specify the actual
endpoint.
$endpoint = "http://oss-cn-hangzhou.aliyuncs.com";
$bucket= "<yourBucketName>";
$localDirectory = ".";
$prefix = "samples/codes";
try {
    $ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint);
    $ossClient->uploadDir($bucket, $prefix, $localDirectory);
} catch(OssException $e) {
    printf(__FUNCTION__ . ": FAILED\n");
    printf($e->getMessage() . "\n");
    return;
}
print(__FUNCTION__ . ": OK" . "\n");
```

Cancel a multipart upload task

You can call `$ossClient->abortMultipartUpload` to cancel a multipart upload task. If you cancel a multipart upload task, you cannot use the upload ID to upload any part. The uploaded parts are deleted.

The following code provides an example on how to cancel a multipart upload task:

```
<? php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to lo
g on to OSS because the account has permissions on all API operations. We recommend that yo
u use your RAM user's credentials to call API operations or perform routine operations and
maintenance. To create a RAM user, log on to the RAM console.
$accessKeyId = "<yourAccessKeyId>";
$accessKeySecret = "<yourAccessKeySecret>";
// The endpoint of the China (Hangzhou) region is used in this example. Specify the actual
endpoint.
$endpoint = "http://oss-cn-hangzhou.aliyuncs.com";
$bucket= "<yourBucketName>";
$object = "<yourObjectName>";
$upload_id = "<yourUploadId>";
try{
    $ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint);
    $ossClient->abortMultipartUpload($bucket, $object, $upload_id);
} catch(OssException $e) {
    printf(__FUNCTION__ . ": FAILED\n");
    printf($e->getMessage() . "\n");
    return;
}
print(__FUNCTION__ . ": OK" . "\n");
```

List uploaded parts

The following code provides an example on how to list uploaded parts:

```

<? php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to lo
g on to OSS because the account has permissions on all API operations. We recommend that yo
u use your RAM user's credentials to call API operations or perform routine operations and
maintenance. To create a RAM user, log on to the RAM console.
$accessKeyId = "<yourAccessKeyId>";
$accessKeySecret = "<yourAccessKeySecret>";
// The endpoint of the China (Hangzhou) region is used in this example. Specify the actual
endpoint.
$endpoint = "http://oss-cn-hangzhou.aliyuncs.com";
$bucket= "<yourBucketName>";
$object = "<yourObjectName>";
$uploadId = "<yourUploadId>";
try{
    $ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint);
    $listPartsInfo = $ossClient->listParts($bucket, $object, $uploadId);
    foreach ($listPartsInfo->getListPart() as $partInfo) {
        print($partInfo->getPartNumber() . "\t" . $partInfo->getSize() . "\t" . $partInfo->
getETag() . "\t" . $partInfo->getLastModified() . "\n");
    }
} catch(OssException $e) {
    printf(__FUNCTION__ . ": FAILED\n");
    printf($e->getMessage() . "\n");
    return;
}
print(__FUNCTION__ . ": OK" . "\n");

```

List multipart upload tasks

The following code provides an example on how to list multipart upload tasks:

```

<? php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log
on to OSS because the account has permissions on all API operations. We recommend that yo
u use your RAM user's credentials to call API operations or perform routine operations and
maintenance. To create a RAM user, log on to the RAM console.
$accessKeyId = "<yourAccessKeyId>";
$accessKeySecret = "<yourAccessKeySecret>";
// The endpoint of the China (Hangzhou) region is used in this example. Specify the actual
endpoint.
$endpoint = "http://oss-cn-hangzhou.aliyuncs.com";
$bucket= "<yourBucketName>";
$options = array(
    'delimiter' => '/',
    'max-uploads' => 100,
    'key-marker' => '',
    'prefix' => '',
    'upload-id-marker' => ''
);
try {
    $ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint);
    $listMultipartUploadInfo = $ossClient->listMultipartUploads($bucket, $options);
} catch(OssException $e) {
    printf(__FUNCTION__ . ": listMultipartUploads FAILED\n");
    printf($e->getMessage() . "\n");
    return;
}
printf(__FUNCTION__ . ": listMultipartUploads OK\n");
$listUploadInfo = $listMultipartUploadInfo->getUploads();
var_dump($listUploadInfo);

```

The following table describes the parameters you can configure for \$options.

Parameter	Description
delimiter	Groups objects by name. Objects whose names contain the same string from the prefix and the next occurrence of the delimiter are grouped as a single result element in CommonPrefixes.
key-marker	Lists all multipart upload tasks for objects whose names are alphabetically greater than the key-marker value. This parameter is used together with the upload-id-marker parameter to specify the position from which the next list begins.

Parameter	Description
max-uploads	Specifies the maximum number of multipart upload tasks to return each time. The default value is 1000. The maximum value is 1000.
prefix	Specifies the prefix that returned object names must contain.  Note Note that if you use a prefix for a query, the returned object names contain the prefix.
upload-id-marker	The upload ID of the multipart upload task after which the list begins. This parameter is used together with the key-marker parameter. If key-marker is not configured, upload-id-marker is invalid. If key-marker is configured, the query result includes: • All objects whose names are alphabetically greater than the key-marker value. • All objects whose names are the same as the key-marker parameter value and of which the uploadId parameter value is greater than the upload-id-marker parameter value.

3.6.5. Upload callback

After an object is uploaded, Object Storage Service (OSS) can start a callback process for the application server. To implement callback, you need only to send a request that contains relevant callback parameters to OSS. This topic describes how to configure upload callback for simple upload and multipart upload.

Configure upload callback for simple upload

The following code provides an example on how to configure upload callback for simple upload:

```

<?php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
$accessKeyId = "yourAccessKeyId";
$accessKeySecret = "yourAccessKeySecret";
// Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com.
$endpoint = "yourEndpoint";
// Specify the name of the bucket. Example: examplebucket.
$bucket = "examplebucket";
// Specify the full path of the object. Example: exampledir/exampleobject.txt. The full path of the object cannot contain the bucket name.
$object = "exampledir/exampleobject.txt";
$ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint);
// Configure callback for upload.
// Specify callbackUrl as the address of the callback server that receives the callback request. Examples: http://oss-demo.aliyuncs.com:23450 or http://127.0.0.1:9090.
// Optional. Set callbackHost to the value of the Host field included in the callback request header.
$url =
    '{
        "callbackUrl": "<yourCallbackServerUrl>",
        "callbackHost": "<yourCallbackServerHost>",
        "callbackBody": "bucket=${bucket}&object=${object}&etag=${etag}&size=${size}&mimeType=${mimeType}&imageInfo.height=${imageInfo.height}&imageInfo.width=${imageInfo.width}&imageInfo.format=${imageInfo.format}&my_var1=${x:var1}&my_var2=${x:var2}",
        "callbackBodyType": "application/x-www-form-urlencoded"
    }';
// Configure custom parameters for the callback request. Each custom parameter consists of a key and a value. The key must start with x:.
$var =
    '{
        "x:var1": "value1",
        "x:var2": "value2"
    }';
$options = array(OssClient::OSS_CALLBACK => $url,
    OssClient::OSS_CALLBACK_VAR => $var
);
$result = $ossClient->putObject($bucket, $object, file_get_contents(__FILE__), $options);
print_r($result['body']);
print_r($result['info']['http_code']);

```


Configure upload callback for multipart upload

The following code provides an example on how to configure upload callback for multipart upload:

```
<?php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
use OSS\Core\OssUtil;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
$accessKeyId = "yourAccessKeyId";
$accessKeySecret = "yourAccessKeySecret";
// Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com.
$endpoint = "yourEndpoint";
// Specify the name of the bucket. Example: examplebucket.
$bucket = "examplebucket";
// Specify the full path of the object. Example: exampledir/exampleobject.txt. The full path of the object cannot contain the bucket name.
$object = "exampledir/exampleobject.txt";
// Specify the full path of the local file. Example: D:\\localpath\\examplefile.txt. If you do not specify the full path of the local file, the local file is uploaded from the path of the project to which the sample program belongs.
$uploadFile = "D:\\localpath\\examplefile.txt";
/**
 * Step 1: Initiate a multipart upload task and obtain the upload ID.
 */
$ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint);
// An upload ID is returned. It is the unique identifier for a multipart upload task. You can initiate related operations such as canceling or querying a multipart upload task based on the upload ID.
$uploadId = $ossClient->initiateMultipartUpload($bucket, $object);
print(__FUNCTION__ . ": initiateMultipartUpload OK" . "\n");
/*
 * Step 2: Upload parts.
 */
$partSize = 10 * 1024 * 1024;
$uploadFileSize = sprintf('%u', filesize($uploadFile));
$pieces = $ossClient->generateMultiuploadParts($uploadFileSize, $partSize);
$responseUploadPart = array();
$uploadPosition = 0;
$isCheckMd5 = true;
foreach ($pieces as $i => $piece) {
    $fromPos = $uploadPosition + (integer)$piece[$ossClient::OSS_SEEK_TO];
    $toPos = (integer)$piece[$ossClient::OSS_LENGTH] + $fromPos - 1;
    $uploadOptions = array(

```

```

        $supOptions = array(
            $ossClient::OSS_FILE_UPLOAD => $uploadFile,
            $ossClient::OSS_PART_NUM => ($i + 1),
            $ossClient::OSS_SEEK_TO => $fromPos,
            $ossClient::OSS_LENGTH => $toPos - $fromPos + 1,
            $ossClient::OSS_CHECK_MD5 => $isCheckMd5,
        );
        // Perform the MD5 hash verification.
        if ($isCheckMd5) {
            $contentMd5 = OssUtil::getMd5SumForFile($uploadFile, $fromPos, $toPos);
            $supOptions[$ossClient::OSS_CONTENT_MD5] = $contentMd5;
        }
        try {
            // Upload parts.
            $responseUploadPart[] = $ossClient->uploadPart($bucket, $object, $uploadId, $supOptions);
        } catch (OssException $e) {
            printf(__FUNCTION__ . ": initiateMultipartUpload, uploadPart - part#{ $i } FAILED\n");
        }
        printf($e->getMessage() . "\n");
        return;
    }
    printf(__FUNCTION__ . ": initiateMultipartUpload, uploadPart - part#{ $i } OK\n");
}
// $uploadParts is an array that consists of the ETag and the PartNumber of each part.
$uploadParts = array();
foreach ($responseUploadPart as $i => $eTag) {
    $uploadParts[] = array(
        'PartNumber' => ($i + 1),
        'ETag' => $eTag,
    );
}
/**
 * Step 3: Complete the multipart upload task.
 */
// Specify callbackUrl as the address of the callback server that receives the callback request. Example: http://oss-demo.aliyuncs.com:23450 or http://127.0.0.1:9090.
// Optional. Set callbackHost to the value of the Host field included in the callback request header.
$json =
    '{
        "callbackUrl": "<yourCallbackServerUrl>",
        "callbackHost": "<yourCallbackServerHost>",
        "callbackBody": "{ \"mimeType\": \"${mimeType}\", \"size\": \"${size}\", \"x:var1\": \"${x:var1}\", \"x:var2\": \"${x:var2}\" }",
        "callbackBodyType": "application/json"
    }';
// Configure custom parameters for the callback request. Each custom parameter consists of a key and a value. The key must start with x:.
$var =
    '{
        "x:var1": "value1",
        "x:var2": "value2"
    }';
$options = array($ossClient::OSS_CALLBACK => $json,

```

```
    OssClient::OSS_CALLBACK_VAR => $var);  
    // You must provide all valid $uploadParts when you perform this operation. OSS verifies the  
    // validity of each part after it receives the submitted $uploadParts. After the verification  
    // of all parts is completed, OSS combines these parts together into a complete object.  
    $ossClient->completeMultipartUpload($bucket, $object, $uploadId, $uploadParts, $options);  
    printf(__FUNCTION__ . ": completeMultipartUpload OK\n");
```

References

- For more information about the complete sample code that is used to configure upload callback, visit [GitHub](#).
- For more information about the API operations that you can call to configure upload callback, see [Callback](#).

3.7. Download objects

3.7.1. Overview

This topic is the overview of object download in OSS PHP SDK.

For the complete code of object download, see [GitHub](#).

OSS PHP SDK provides the following download methods:

- [Download to your local file](#)
- [Download to local memory](#)
- [Range download](#)
- [Conditional download](#)

3.7.2. Download objects to local files

This topic describes how to download an object from a bucket to your local computer.

Sample code

The following code provides an example on how to download *exampleobject.txt* from the *testfolder* directory in examplebucket to examplefile.txt in the *D:\localpath* path:

```

<?php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
$accessKeyId = "yourAccessKeyId";
$accessKeySecret = "yourAccessKeySecret";
// Specify the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set Endpoint to https://oss-cn-hangzhou.aliyuncs.com.
$endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
// Specify the name of the bucket. Example: examplebucket.
$bucket = "examplebucket";
// Specify the full path of the object. The full path of the object cannot contain the bucket name. Example: testfolder/exampleobject.txt.
$object = "testfolder/exampleobject.txt";
// Download the object to D:\\localpath. After the object is downloaded, the local file is named examplefile.txt. If the specified local file exists, the object to download overwrites the file. Otherwise, a file is created.
// If the path for the object is not specified, the downloaded object is saved to the path of the project to which the sample program belongs.
$localfile = "D:\\localpath\\examplefile.txt";
$options = array(
    OssClient::OSS_FILE_DOWNLOAD => $localfile
);
// Use try-catch to catch exceptions. If an exception is caught, the download fails. If no exceptions are caught, the download succeeds.
try{
    $ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint);
    $ossClient->getObject($bucket, $object, $options);
} catch(OssException $e) {
    printf(__FUNCTION__ . ": FAILED\n");
    printf($e->getMessage() . "\n");
    return;
}
print(__FUNCTION__ . ": OK, please check localfile: 'upload-test-object-name.txt' . "\n");

```

References

- For more information about the complete sample code that is used to download objects to local files, visit [GitHub](#).
- For more information about the API operation that you can call to download objects to local files, see [GetObject](#).

3.7.3. Download to local memory

This topic describes how to use OSS SDK for PHP to download OSS objects to local memory.

The following code provides an example on how to download an OSS object to local memory and obtain the content of the object:

```
<? php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log on to OSS, because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine operations and maintenance. To create a RAM user, visit https://ram.console.aliyun.com.
$accessKeyId = "<yourAccessKeyId>";
$accessKeySecret = "<yourAccessKeySecret>";
// This example uses the public endpoint of OSS in the China (Hangzhou) region. Specify the actual endpoint based on your requirements.
$endpoint = "http://oss-cn-hangzhou.aliyuncs.com";
$bucket = "<yourBucketName>";
// Specify the object name.
$object = "<yourObjectName>";
try{
    $ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint);
    $content = $ossClient->getObject($bucket, $object);
} catch(OssException $e) {
    printf(__FUNCTION__ . ": FAILED\n");
    printf($e->getMessage() . "\n");
    return;
}
print($content);
print(__FUNCTION__ . ": OK" . "\n");
```

 **Note** If a power failure occurs, data stored in the memory is lost. To store the downloaded object in a local disk, see [Download to your local file](#).

3.7.4. Range download

If you only need part of the data in an object, you can use range downloads to download specified content.

Run the following code to download the content of an object in a range of [0, 4] to local memories:

```

<? php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
//It is highly risky to log on with AccessKey of an Alibaba Cloud account because the account has permissions on all the APIs in OSS. We recommend that you log on as a RAM user to access APIs or perform routine operations and maintenance. To create a RAM account, log on to https://ram.console.aliyun.com.
$accessKeyId = "<yourAccessKeyId>";
$accessKeySecret = "<yourAccessKeySecret>";
//This example uses the endpoint China East 1 (Hangzhou). Specify the actual endpoint based on your requirements.
$endpoint = "http://oss-cn-hangzhou.aliyuncs.com";
$bucket = "<yourBucketName>";
$object = "<yourObjectName>";
// Obtain data between the starting byte value of 0 and the maximum byte value of 4 inclusive (that is, a total of 5 bytes). If the specified value range is invalid, for example, the specified range includes a negative number, or the specified value is larger than the object size, the entire object is downloaded.
$options = array(OssClient::OSS_RANGE => '0-4');
try{
    $ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint);
    $content = $ossClient->getObject($bucket, $object, $options);
} catch(OssException $e) {
    printf(__FUNCTION__ . ": FAILED\n");
    printf($e->getMessage() . "\n");
    return;
}
print ($content);
print(__FUNCTION__ . ": OK" . "\n");

```

3.7.5. Conditional download

You can specify one or more conditions when you download objects. If the specified conditions are met, the object is downloaded. If the specified conditions are not met, an error is returned and the object is not downloaded. You can use conditional download to download an object from Object Storage Service (OSS) to your local memory or a local file.

Conditions

The following table describes conditions that OSS supports and you can specify when you use conditional download. You can use If-Modified-Since and If-Unmodified-Since as object download conditions at the same time. You can use If-Match and If-None-Match as object download conditions at the same time.

Parameter	Description	Method
-----------	-------------	--------

Parameter	Description	Method
If-Modified-Since	If the object is modified after the specified time, the object can be downloaded. Otherwise, the error code 304 Not Modified is returned.	OssClient::OSS_IF_MODIFIED_SINCE
If-Unmodified-Since	If the object is modified before or at the specified time, the object can be downloaded. Otherwise, the error code 412 Precondition Failed is returned.	OssClient::OSS_IF_UNMODIFIED_SINCE
If-Match	If the specified ETag matches that of an object, the object can be downloaded. Otherwise, the error 412 Precondition Failed is returned.  Note You can use \$ossClient->getObjectMeta to obtain the ETag of an object.	OssClient::OSS_IF_MATCH
If-None-Match	If the specified ETag does not match that of an object, the object can be downloaded. Otherwise, the error code 304 Not Modified is returned.	OssClient::OSS_IF_NONE_MATCH

Download an object to your local memory by using conditional download

The following code provides an example on how to specify conditions and download the object that matches the conditions to your local memory:

```

<?php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a Resource Access Management (RAM) user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
$accessKeyId = "<yourAccessKeyId>";
$accessKeySecret = "<yourAccessKeySecret>";
// The China (Hangzhou) region is used in this example as the endpoint. Specify the actual endpoint.
$endpoint = "http://oss-cn-hangzhou.aliyuncs.com";
// Specify the bucket name.
$bucket = "<yourBucketName>";
// Specify the object name. // Specify the full path of the object. The full path of the object does not contain the bucket name. Example: exampledir/exampleobject.txt.
$object = "<yourObjectName>";
try{
    $options = array(
        OssClient::OSS_HEADERS => array(
            // Set If-Modified-Since to 14:47:53 Friday, April 9, 2021 in GMT.
            OssClient::OSS_IF_MODIFIED_SINCE => "Fri, 9 Apr 2021 14:47:53 GMT",
            // Set If-Unmodified-Since to 14:47:53 Wednesday, October 13, 2021 in GMT.
            OssClient::OSS_IF_UNMODIFIED_SINCE => "Fri, 13 Oct 2021 14:47:53 GMT",
            // Set If-None-Match to the ETag of the object.
            OssClient::OSS_IF_NONE_MATCH => '"5B3C1A2E0563E1B002CC607C*****"',
            // Set If-Match to the ETag of the object.
            OssClient::OSS_IF_MATCH => '"fba9dede5f27731c9771645a3986*****"',
        )
    );
    $ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint);
    // Download the object to your local memory.
    $content = $ossClient->getObject($bucket, $object, $options);
} catch(OssException $e) {
    printf(__FUNCTION__ . ": FAILED\n");
    printf($e->getMessage() . "\n");
    return;
}
print ($content);
print(__FUNCTION__ . ": OK" . "\n");

```

Download an object to a local file by using conditional download

The following code provides an example on how to specify conditions and download the object that matches the conditions to a local file:


```

<?php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
$accessKeyId = "<yourAccessKeyId>";
$accessKeySecret = "<yourAccessKeySecret>";
// The China (Hangzhou) region is used in this example as the endpoint. Specify the actual endpoint.
$endpoint = "http://oss-cn-hangzhou.aliyuncs.com";
// Specify the bucket name.
$bucket = "<yourBucketName>";
// Specify the object name. // Specify the full path of the object. The full path of the object does not contain the bucket name. Example: exampledir/exampleobject.txt.
$object = "<yourObjectName>";
// Specify the full path of the local file. Example: D:\\localpath\\examplefile.txt.
$localfile = "<yourLocalFile>";
try{
    $options = array(
        OssClient::OSS_HEADERS => array(
            // Set If-Modified-Since to 14:47:53 Friday, April 9, 2021 in GMT.
            OssClient::OSS_IF_MODIFIED_SINCE => "Fri, 9 Apr 2021 14:47:53 GMT",
            // Set If-Unmodified-Since to 14:47:53 Wednesday, October 13, 2021 in GMT.
            OssClient::OSS_IF_UNMODIFIED_SINCE => "Fri, 13 Oct 2021 14:47:53 GMT",
            // Set If-None-Match to the ETag of the object.
            OssClient::OSS_IF_NONE_MATCH => '"5B3C1A2E0563E1B002CC607C****"',
            // Set If-Match to the ETag of the object.
            OssClient::OSS_IF_MATCH => '"fba9dede5f27731c9771645a3986****"',
            OssClient::OSS_FILE_DOWNLOAD => $localfile
        )
    );
    $ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint);
    $ossClient->getObject($bucket, $object, $options);
} catch(OssException $e) {
    printf(__FUNCTION__ . ": FAILED\n");
    printf($e->getMessage() . "\n");
    return;
}

```

3.8. Manage objects

3.8.1. Overview

You can perform the following operations on objects in a bucket with APIs:

- Determine whether an object exists
- Manage ACL for an object
- Manage Object Meta
- List objects
- Delete an object
- Copy an object
- Restore an archive object
- Manage a symbolic link
- Enable MD5 verification

3.8.2. Determine whether a specified object exists

Use the following code to determine whether a specified object exists:

```
<? php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
// It is highly risky to log on with AccessKey of an Alibaba Cloud account because the account has permissions on all the APIs in OSS. We recommend that you log on as a RAM user to access APIs or perform routine operations and maintenance. To create a RAM account, log on to https://ram.console.aliyun.com.
$accessKeyId = "<yourAccessKeyId>";
$accessKeySecret = "<yourAccessKeySecret>";
// This example uses endpoint China East 1 (Hangzhou). Specify the actual endpoint based on your requirements.
$endpoint = "http://oss-cn-hangzhou.aliyuncs.com";
$bucket = "<yourBucketName>";
$object = "<yourObjectName>";
try{
    $ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint);
    $exist = $ossClient->doesObjectExist($bucket, $object);
} catch(OssException $e) {
    printf(__FUNCTION__ . ": FAILED\n");
    printf($e->getMessage() . "\n");
    return;
}
print(__FUNCTION__ . ": OK" . "\n");
var_dump($exist);
```

3.8.3. Manage the ACL of an object

This topic describes how to manage the access control list (ACL) of an object.

The following table describes the ACLs that you can configure for an object.

ACL	Description	Value
Inherited from the bucket	The ACL of the object is the same as the ACL of the bucket in which the object is stored.	default
Private	Only the object owner and authorized users are granted the read and write permissions on the object.	private
Public read	Only the object owner and authorized users are granted the read and write permissions on the object. Other users are granted only the read permissions on the object. Exercise caution when you set the ACL of the object to this value.	public-read
Public read/write	All users are granted the read and write permissions on the object. Exercise caution when you set the ACL of the object to this value.	public-read-write

The ACL of the object takes precedence over the ACL of the bucket. For example, if the ACL of a bucket is private and the ACL of an object that is stored in the bucket is public, all users are granted the read and write permissions on the object. If the ACL of an object is not configured, the ACL of the object is the same as the ACL of the bucket in which the object is stored.

Configure the ACL of an object

The following code provides an example on how to configure the ACL of a specified object:

```
<?php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
// The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
$accessKeyId = "<yourAccessKeyId>";
$accessKeySecret = "<yourAccessKeySecret>";
// In this example, the endpoint of the China (Hangzhou) region is used. Specify your actual endpoint.
$endpoint = "http://oss-cn-hangzhou.aliyuncs.com";
$bucket = "<yourBucketName>";
$object = "<yourObjectName>";
// Configure the ACL of the object to public read. If you do not specify the ACL of the object, the ACL of the object inherits the ACL of the bucket in which the object is stored.
$acl = "public-read";
try {
    $ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint);
    $ossClient->putObjectAcl($bucket, $object, $acl);
} catch (OssException $e) {
    printf(__FUNCTION__ . ": FAILED\n");
    printf($e->getMessage() . "\n");
    return;
}
print(__FUNCTION__ . ": OK" . "\n");
```

Obtain the ACL of an object

The following code provides an example on how to query the ACL of a specified object:

```
<?php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
// The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
$accessKeyId = "<yourAccessKeyId>";
$accessKeySecret = "<yourAccessKeySecret>";
// In this example, the endpoint of the China (Hangzhou) region is used. Specify your actual endpoint.
$endpoint = "http://oss-cn-hangzhou.aliyuncs.com";
$bucket = "<yourBucketName>";
$object = "<yourObjectName>";
try {
    $ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint);
    $objectAcl = $ossClient->getObjectAcl($bucket, $object);
} catch (OssException $e) {
    printf(__FUNCTION__ . ": FAILED\n");
    printf($e->getMessage() . "\n");
    return;
}
print(__FUNCTION__ . ": OK" . "\n");
var_dump($objectAcl);
```

3.8.4. Manage object metadata

Object metadata includes HTTP headers and user metadata. This topic describes how to configure, modify, and query object metadata.

Configure object metadata

The following code provides an example on how to configure object metadata:

```

<?php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access Object Storage Service (OSS) because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
$accessKeyId = "yourAccessKeyId";
$accessKeySecret = "yourAccessKeySecret";
// Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com.
$endpoint = "yourEndpoint";
// Specify the bucket name. Example: examplebucket.
$bucket = "examplebucket";
// Specify the full path of the object. The full path cannot contain the bucket name. Example: exampledir/exampleobject.txt.
$object = "exampledir/exampleobject.txt";
$content = file_get_contents(__FILE__);
$options = array(
    OssClient::OSS_HEADERS => array(
        'Expires' => '2012-10-01 08:00:00',
        'Content-Disposition' => 'attachment; filename="xxxxxx"',
        'x-oss-meta-self-define-title' => 'user define meta info',
    ));
try{
    $ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint);
    $ossClient->putObject($bucket, $object, $content, $options);
} catch(OssException $e) {
    printf(__FUNCTION__ . ": FAILED\n");
    printf($e->getMessage() . "\n");
    return;
}
print(__FUNCTION__ . ": OK" . "\n");

```

Modify object metadata

The following code provides an example on how to modify object metadata:

```

<?php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
$accessKeyId = "yourAccessKeyId";
$accessKeySecret = "yourAccessKeySecret";
// In this example, the endpoint of the China (Hangzhou) region is used. Specify your actual endpoint.
$endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
// Specify the name of the source bucket. Example: srcexamplebucket.
$fromBucket = "srcexamplebucket";
// Specify the full path of the source object. The full path of the object cannot contain the bucket name. Example: srcdir/exampleobject.txt.
$fromObject = "<yourFromObjectName>";
// Specify the name of the destination bucket. The destination bucket must be located in the same region as the source bucket. Example: destexamplebucket.
$toBucket = destexamplebucket;
// Specify the full path of the destination object. The full path cannot contain the bucket name. Example: destdir/exampleobject.txt.
$toObject = destdir/exampleobject.txt;
$copyOptions = array(
    OssClient::OSS_HEADERS => array(
        'Expires' => '2018-10-01 08:00:00',
        'Content-Disposition' => 'attachment; filename="xxxxxx"',
        'x-oss-meta-location' => 'location',
        // Specify the method used to configure metadata for the destination object. In this example, the method is set to REPLACE, which indicates that the metadata of the destination object is replaced by the metadata specified in the request. If the method is set to COPY, the metadata of the source object is copied to the destination object.
        'x-oss-metadata-directive' => 'REPLACE',
    ),
);
try{
    $ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint);
    $ossClient->copyObject($fromBucket, $fromObject, $toBucket, $toObject, $copyOptions);
} catch(OssException $e) {
    printf(__FUNCTION__ . ": FAILED\n");
    printf($e->getMessage() . "\n");
    return;
}
print(__FUNCTION__ . ": OK" . "\n");

```

Query object metadata

The following code provides an example on how to query object metadata:

```
<?php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
$accessKeyId = "yourAccessKeyId";
$accessKeySecret = "yourAccessKeySecret";
// Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com.
$endpoint = "yourEndpoint";
// Specify the bucket name. Example: examplebucket.
$bucket = "examplebucket";
// Specify the full path of the object. The full path cannot contain the bucket name. Example: exampledir/exampleobject.txt.
$object = "exampledir/exampleobject.txt";
$ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint);
try {
    // Query all object metadata.
    $objectMeta = $ossClient->getObjectMeta($bucket, $object);
    print_r($objectMeta);
    // Query part of object metadata.
    $objectMeta = $ossClient->getSimplifiedObjectMeta($bucket, $object);
    print_r($objectMeta);
} catch (OssException $e) {
    printf(__FUNCTION__ . ": FAILED\n");
    printf($e->getMessage() . "\n");
    return;
}
print(__FUNCTION__ . ": OK" . "\n");
```

References

- For the complete sample code that is used to configure, modify, and query object metadata, visit [GitHub](#).
- For more information about the API operation that you can call to configure object metadata in simple upload, see [Put Object](#).
- For more information about the API operation that you can call to modify object metadata in a copy task, see [CopyObject](#).
- For more information about the API operation that you can call to query object metadata, see

[Get Object Meta.](#)

3.8.5. Convert the storage classes of objects

Object Storage Service (OSS) provides the following storage classes to cover a variety of data storage scenarios from hot data to cold data: Standard, Infrequent Access (IA), Archive, and Cold Archive. This topic describes how to call the CopyObject operation to convert the storage class of an object.

 **Note** For more information about storage classes, see [Overview](#) and [Convert storage classes](#) in the Developer Guide.

Covert a Standard or an IA object to an Archive object

The following code provides an example on how to convert a Standard or an IA object to an Archive object:

```

<?php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to lo
g on to OSS because the account has permissions on all API operations. We recommend that yo
u use a Resource Access Management (RAM) user to call API operations or perform routine ope
rations and maintenance. To create a RAM user, log on to the RAM console.
$accessKeyId = "<yourAccessKeyId>";
$accessKeySecret = "<yourAccessKeySecret>";
// The endpoint of the China (Hangzhou) region is used in this example. Specify the actual
endpoint.
$endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
// Specify the bucket name.
$bucket= "<yourBucketName>";
// Specify the full path of the object. The full path of the object does not contain bucket
names. Example: destfolder/exampleobject.txt.
$object = "<yourObjectName>";
$ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint);
try {
    // Specify the storage class that you want to convert to. In this example, set the stor
age class to Archive.
    $copyOptions = array(
        OssClient::OSS_HEADERS => array(
            'x-oss-storage-class' => 'Archive',
            'x-oss-metadata-directive' => 'REPLACE',
        ),
    );
    $ossClient->copyObject($bucket, $object, $bucket, $object, $copyOptions);
} catch (OssException $e) {
    printf(__FUNCTION__ . ": FAILED\n");
    printf($e->getMessage() . "\n");
    return;
}
print(__FUNCTION__ . ": OK" . "\n");

```

For more information about the naming conventions for buckets, see [bucket](#). For more information about the naming conventions for objects, see [object](#).

Covert an Archive object to an IA or a Standard object

The following code provides an example on how to convert an Archive object to an IA or a Standard object:

```

<?php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}

```

```

require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to lo
g on to OSS because the account has permissions on all API operations. We recommend that yo
u use a RAM user to call API operations or perform routine operations and maintenance. To c
reate a RAM user, log on to the RAM console.
$accessKeyId = "<yourAccessKeyId>";
$accessKeySecret = "<yourAccessKeySecret>";
// The endpoint of the China (Hangzhou) region is used in this example. Specify the actual
endpoint.
$endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
// Specify the bucket name.
$bucket = "<yourBucketName>";
// Specify the full path of the object. The full path of the object does not contain bucket
names. Example: destfolder/exampleobject.txt.
$object = "<yourObjectName>";
$ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint);
try {
    // Restore the Archive object.
    $ossClient->restoreObject($bucket, $object);
    // Wait until the object is restored.
    $waitTime = 100;
    while ($waitTime > 0) {
        $meta = $ossClient->getObjectMeta($bucket, $object);
        if (array_key_exists("x-oss-restore", $meta) &&
            strcmp($meta["x-oss-restore"], "ongoing-request=\"true\"") == 0) {
            print("In restore status, ". $meta["x-oss-restore"]. "\n");
            sleep(5);
        } else {
            break;
        }
        $waitTime -= 5;
    }
    // Specify the storage class to which you want to convert. In this example, you can set
the storage class to IA or Standard.
    $copyOptions = array(
        OssClient::OSS_HEADERS => array(
            'x-oss-storage-class' => 'IA',
            'x-oss-metadata-directive' => 'REPLACE',
        ),
    );
    $ossClient->copyObject($bucket, $object, $bucket, $object, $copyOptions);
} catch (OssException $e) {
    printf(__FUNCTION__ . ": FAILED\n");
    printf($e->getMessage() . "\n");
    return;
}
print(__FUNCTION__ . ": OK" . "\n");

```

3.8.6. Prevent objects from being overwritten by objects with the same names

By default, if you upload an object that has the same name as an existing object, the existing object is overwritten by the uploaded object. This topic describes how to set the `x-oss-forbid-overwrite` request header to prevent objects from being overwritten by objects with the same names in simple upload and multipart upload.

Simple upload

The following code provides an example on how to prevent objects from being overwritten by objects with the same names in simple upload:

```

<? php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to lo
g on to OSS because the account has permissions on all API operations. We recommend that yo
u use your RAM user's credentials to call API operations or perform routine operations and
maintenance. To create a RAM user, log on to the RAM console.
$accessKeyId = "<yourAccessKeyId>";
$accessKeySecret = "<yourAccessKeySecret>";
// The endpoint of the China (Hangzhou) region is used in this example. Specify the actual
endpoint.
$endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
$bucket= "<yourBucketName>";
$object = "<yourObjectName>";
$content = "Hello OSS";
$ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint);
try{
    // Specify whether the PutObject operation overwrites the object with the same name.
    // By default, if x-oss-forbid-overwrite is not specified, existing objects are overwrit
ten by the uploaded objects that have the same name.
    // If x-oss-forbid-overwrite is set to false, existing objects are overwritten by the up
loaded objects that have the same name.
    // If x-oss-forbid-overwrite is set to true, existing objects cannot be overwritten by o
bjects that have the same name. If you upload an object that has the same name as an existi
ng object, OSS returns FileAlreadyExists.
    $options = array(
        OssClient::OSS_HEADERS => array(
            'x-oss-forbid-overwrite' => 'true'
        ),
    );
    $ossClient->putObject($bucket, $object, $content, $options);
} catch(OssException $e) {
    printf(__FUNCTION__ . ": FAILED\n");
    printf($e->getMessage() . "\n");
    return;
}
print(__FUNCTION__ . ": OK" . "\n");

```

Multipart upload

The following code provides an example on how to prevent objects from being overwritten by objects with the same names in multipart upload:

```

<? php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;

```

```

if (!is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}

use OSS\OssClient;
use OSS\Core\OssException;
use OSS\Core\OssUtil;

// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log
// on to OSS because the account has permissions on all API operations. We recommend that you
// use your RAM user's credentials to call API operations or perform routine operations and
// maintenance. To create a RAM user, log on to the RAM console.
$accessKeyId = "<yourAccessKeyId>";
$accessKeySecret = "<yourAccessKeySecret>";
// The endpoint of the China (Hangzhou) region is used in this example. Specify the actual
// endpoint.
$endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
$bucket = "<yourBucketName>";
$object = "<yourObjectName>";
$uploadFile = "<yourLocalFile>";
$ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint);
try{
    // Specify whether existing objects are overwritten by objects with the same names when
    // the multipart upload task is initialized.
    // By default, if x-oss-forbid-overwrite is not specified, existing objects are overwri
    // tten by the uploaded objects that have the same name.
    // If x-oss-forbid-overwrite is set to false, existing objects are overwritten by the u
    // ploaded objects that have the same name.
    // If x-oss-forbid-overwrite is set to true, existing objects cannot be overwritten by
    // objects that have the same name. If you upload an object that has the same name as an exist
    // ing object, OSS returns FileAlreadyExists.
    $options = array(
        OssClient::OSS_HEADERS => array(
            'x-oss-forbid-overwrite' => 'true'
        ),
    );
    $uploadId = $ossClient->initiateMultipartUpload($bucket, $object, $options);
    // Upload the parts.
    $partSize = 1 * 1024 * 1024;
    $uploadFileSize = sprintf('%u', filesize($uploadFile));
    $pieces = $ossClient->generateMultiuploadParts($uploadFileSize, $partSize);
    $responseUploadPart = array();
    $uploadPosition = 0;
    $isCheckMd5 = true;
    foreach ($pieces as $i => $piece) {
        $fromPos = $uploadPosition + (integer)$piece[$ossClient::OSS_SEEK_TO];
        $toPos = (integer)$piece[$ossClient::OSS_LENGTH] + $fromPos - 1;
        $uploadOptions = array(
            $ossClient::OSS_FILE_UPLOAD => $uploadFile,
            $ossClient::OSS_PART_NUM => ($i + 1),
            $ossClient::OSS_SEEK_TO => $fromPos,
            $ossClient::OSS_LENGTH => $toPos - $fromPos + 1,
        );
        $responseUploadPart[] = $ossClient->uploadPart($bucket, $object, $uploadId, $uploadOptions);
    }
    $uploadParts = array();

```

```

        foreach ($responseUploadPart as $i => $eTag) {
            $uploadParts[] = array(
                'PartNumber' => ($i + 1),
                'ETag' => $eTag,
            );
        }
        // Specify whether existing objects are overwritten by objects with the same names when
        // the multipart upload task is complete.
        // By default, if x-oss-forbid-overwrite is not specified, existing objects are overwri
        // then by the uploaded objects that have the same name.
        // If x-oss-forbid-overwrite is set to false, existing objects are overwritten by the u
        // ploded objects that have the same name.
        // If x-oss-forbid-overwrite is set to true, existing objects cannot be overwritten by
        // objects that have the same name. If you upload an object that has the same name as an exist
        // ing object, OSS returns FileAlreadyExists.
        $options = array(
            OssClient::OSS_HEADERS => array(
                'x-oss-forbid-overwrite' => 'true'
            ),
        );
        $ossClient->completeMultipartUpload($bucket, $object, $uploadId, $uploadParts, $options
    );
} catch (OssException $e) {
    printf(__FUNCTION__ . ": FAILED\n");
    printf($e->getMessage() . "\n");
    return;
}
print(__FUNCTION__ . ": OK" . "\n");

```

3.8.7. List objects

This topic describes how to list all objects or objects that meet specific conditions in a bucket.

List all objects

The following sample code provides an example on how to list all objects in a bucket named `examplebucket`.

Note

- In the following sample code, `$options` is optional and can be used to list all objects in a bucket or directory.
- You can use the `listObjects` method to obtain a list of objects. For more information about how to obtain the attributes such as the size of an object, see [Manage object metadata](#).

```

<?php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS, because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
$accessKeyId = "yourAccessKeyId";
$accessKeySecret = "yourAccessKeySecret";
// Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if your bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com.
$endpoint = "yourEndpoint";
// Specify the name of the bucket. Example: examplebucket.
$bucket = "examplebucket";
$ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint);
$nextMarker = '';
while (true) {
    try {
        $options = array(
            'delimiter' => '',
            'marker' => $nextMarker,
        );
        $listObjectInfo = $ossClient->listObjects($bucket, $options);
    } catch (OssException $e) {
        printf(__FUNCTION__ . ": FAILED\n");
        printf($e->getMessage() . "\n");
        return;
    }
    // Obtain nextMarker to list the remaining objects. The list operation starts from the object next to the last object that is previously obtained by using the ListObjects method.

    $nextMarker = $listObjectInfo->getNextMarker();
    $listObject = $listObjectInfo->getObjectList(); // List the objects.
    if (!empty($listObject)) {
        print("objectList:\n");
        foreach ($listObject as $objectInfo) {
            print($objectInfo->getKey() . "\n");
        }
    }
    if ($listObjectInfo->getIsTruncated() !== "true") {
        break;
    }
}

```

List the objects that meet specific conditions

The following sample code provides an example on how to list objects that meet specific conditions in a bucket named `examplebucket`:

```
<?php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS, because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
$accessKeyId = "yourAccessKeyId";
$accessKeySecret = "yourAccessKeySecret";
// Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if your bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com.
$endpoint = "yourEndpoint";
// Specify the name of the bucket. Example: examplebucket.
$bucket = "examplebucket";
$ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint);
// Specify the prefix. Example: dir/.
$prefix = 'dir/';
// Specify the character that is used to group objects by name. Example: /.
$delimiter = '/';
// Specify the position from which the list operation starts.
$nextMarker = '';
// Specify the maximum number of objects that you want to list.
$maxkeys = 10;
$options = array(
    'delimiter' => $delimiter,
    'prefix' => $prefix,
    'max-keys' => $maxkeys,
    'marker' => $nextMarker,
);
try {
    $listObjectInfo = $ossClient->listObjects($bucket, $options);
} catch (OssException $e) {
    printf(__FUNCTION__ . ": FAILED\n");
    printf($e->getMessage() . "\n");
    return;
}
print(__FUNCTION__ . ": OK" . "\n");
$objectList = $listObjectInfo->getObjectList(); // List the objects.
$prefixList = $listObjectInfo->getPrefixList(); // List the directories. If the directory that matches the prefix does not contain subdirectories or the delimiter parameter is set to null, the directory list is not displayed.
if (!empty($objectList)) {
    print("objectList:\n");
    foreach ($objectList as $objectInfo) {
```

```
        print($objectInfo->getKey() . "\n");
    }
}
if (!empty($prefixList)) {
    print("prefixList: \n");
    foreach ($prefixList as $prefixInfo) {
        print($prefixInfo->getPrefix() . "\n");
    }
}
```

The following table describes the parameters included in `$options` in the preceding sample code.

Parameter	Required	Example	Description
delimiter	No	/	The character that is used to group objects by name. CommonPrefixes contains objects whose names share the same string that ranges from the specified prefix to the specified delimiter that appears for the first time.
prefix	No	dir/	The prefix that must be included in the names of the listed objects.
max-keys	No	100	The maximum number of objects that can be listed each time. The default value is 100, and the maximum value is 1000.
marker	No	exampleobject.txt	The position from which the list operation starts.

3.8.8. Delete objects

You can delete a single object, multiple specified objects, objects whose names contain a specified prefix, or a specified directory and all objects in the directory.

 **Warning** Deleted objects cannot be recovered. Exercise caution when you delete objects.

Delete a single object

The following code provides an example on how to delete an object named *exampleobject.txt* from a bucket named *examplebucket*:

```

<?php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
// The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in Object Storage Service (OSS) is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
$accessKeyId = "yourAccessKeyId";
$accessKeySecret = "yourAccessKeySecret";
// Specify the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com.
$endpoint = "yourEndpoint";
// Specify the bucket name. Example: examplebucket.
$bucket = "examplebucket";
// Specify the full path of the object. Example: exampledir/exampleobject.txt. The full path of the object cannot contain the bucket name.
$object = "exampledir/exampleobject.txt";
try{
    $ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint);
    $ossClient->deleteObject($bucket, $object);
} catch(OssException $e) {
    printf(__FUNCTION__ . ": FAILED\n");
    printf($e->getMessage() . "\n");
    return;
}
print(__FUNCTION__ . "OK" . "\n");

```

Delete multiple objects

You can delete up to 1,000 objects at a time. You can delete multiple specified objects, objects whose names contain a specified prefix, or a specified directory and all objects in the directory.

You can also configure lifecycle rules to automatically delete objects. For more information, see [Lifecycle rules based on the last modified time](#).

Delete multiple specified objects

The following code provides an example on how to delete multiple specified objects:

```

<?php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
// The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
$accessKeyId = "yourAccessKeyId";
$accessKeySecret = "yourAccessKeySecret";
// Specify the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com.
$endpoint = "yourEndpoint";
// Specify the bucket name. Example: examplebucket.
$bucket = "examplebucket";
try {
    $ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint, false);
    // Specify the full paths of the objects that you want to delete. The full paths of the objects cannot contain the bucket name.
    $objects = array();
    $objects[] = "exampleobjecta.txt";
    $objects[] = "exampledir/sampleobject.txt";
    $result = $ossClient->deleteObjects($bucket, $objects);
    foreach ($result as $info) {
        $obj = strval($info);
        printf("Delete ".$obj." : Success" . "\n");
    }
    printf("Delete Objects : OK" . "\n");
} catch (OssException $e) {
    printf("Delete Objects : Failed" . "\n");
    printf($e->getMessage() . "\n");
    return;
}

```

Delete multiple objects with a specified prefix or in a specified directory

The following code provides an example on how to delete multiple objects with a specified prefix and how to delete a specified directory and all objects in the directory.

 **Warning** If the value of `OSS_PREFIX` is an empty string or is set to `NULL` in the following code, all objects in the bucket are deleted. Exercise caution when you configure this value.

```

<?php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}

```

```

}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
// The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
$accessKeyId = "yourAccessKeyId";
$accessKeySecret = "yourAccessKeySecret";
// Specify the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com.
$endpoint = "yourEndpoint";
// Specify the bucket name. Example: examplebucket.
$bucket = "examplebucket";
try {
    $ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint, false);
    $option = array(
        OssClient::OSS_MARKER => null,
        // Specify the prefix of the objects that you want to delete. If you want to delete all objects whose names contain the src prefix, set the prefix to src. After you set the prefix to src, all non-directory objects whose names contain the src prefix, the src directory, and all objects in the directory are deleted.
        OssClient::OSS_PREFIX => "src",
        // If you want to delete only the src directory and all objects in the directory, set the prefix to src/.
        // OssClient::OSS_PREFIX => "src/",
    );
    $bool = true;
    while ($bool){
        // List and delete multiple objects.
        $result = $ossClient->listObjects($bucket,$option);
        $objects = array();
        if(count($result->getObjectList()) > 0){
            foreach ($result->getObjectList() as $key => $info){
                printf("key name:".$info->getKey().PHP_EOL);
                $objects[] = $info->getKey();
            }
            $delObjects = $ossClient->deleteObjects($bucket, $objects);
            foreach ($delObjects as $info){
                $obj = strval($info);
                printf("Delete ".$obj." : Success" . PHP_EOL);
            }
        }
        if($result->getIsTruncated() === 'true'){
            $option[OssClient::OSS_MARKER] = $result->getNextMarker();
        }else{
            $bool = false;
        }
    }
}
printf("Delete Objects : OK" . PHP_EOL);

```

```

} catch (OssException $e) {
    printf("Delete Objects : Failed" . PHP_EOL);
    printf($e->getMessage() . PHP_EOL);
    return;
}

```

 **Note** If an exception is thrown when you use `try{}catch{} to catch errors, no objects are deleted. In this case, you can use $e->getMessage() to obtain the error message and analyze the causes.`

References

- For the complete sample code that is used to delete a single object or multiple objects, visit [GitHub](#).
- For more information about the API operation that you can call to delete a single object, see [DeleteObject](#).
- For more information about the API operation that you can call to delete multiple objects, see [DeleteMultipleObjects](#).

3.8.9. Copy objects

This topic describes how to copy an object within a bucket or across buckets in the same region.

Precautions

- If you copy an object by using a RAM user, the RAM user must have the `oss:GetObject` read permission on the source object and the `oss:GetObject` and `oss:PutObject` read and write permissions on the destination bucket. For more information about how to attach a custom policy to a RAM user, see [Common examples of RAM policies](#).

-
-

Copy a small object

The following code provides an example on how to use `$ossClient->copyObject` to copy an object that is smaller than 1 GB in size:

```

<?php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
$accessKeyId = "yourAccessKeyId";
$accessKeySecret = "yourAccessKeySecret";

```

```
// In this example, the endpoint of the China (Hangzhou) region is used. Specify the endpoint based on your business requirements.
$endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
// Specify the name of the source bucket. Example: srcexamplebucket.
$from_bucket = "srcexamplebucket";
// Specify the full path of the source object. The full path of the object cannot contain the bucket name. Example: srcdir/exampleobject.txt.
$from_object = "srcdir/exampleobject.txt";
// Specify the name of the destination bucket. The destination bucket must be in the same region as the source bucket. Example: destexamplebucket.
// If you copy an object within a bucket, make sure that you specify the same bucket name for the source and destination buckets.
$to_bucket = "destexamplebucket";
// Specify the full path of the destination object. The full path of the object cannot contain the bucket name. Example: destdir/exampleobject.txt.
$to_object = "destdir/exampleobject.txt";
$options = array(
    'headers'=>array(
        // Specify whether the CopyObject operation overwrites the object with the same name. In this example, this parameter is set to true, which indicates that the object with the same name cannot be overwritten.
        // 'x-oss-forbid-overwrite' => 'true',
        // If the ETag value of the source object is the same as the ETag value that is specified in the request, OSS copies the object and returns 200 OK.
        // 'x-oss-copy-source-if-match' => '5B3C1A2E053D763E1B002CC****',
        // If the ETag value of the source object is different from the ETag value that is specified in the request, OSS copies the object and returns 200 OK.
        // 'x-oss-copy-source-if-none-match' => '5B3C1A2E053D763E1B002CC****',
        // If the time that is specified in the request is later than or equal to the actual modified time of the source object, OSS copies the object and returns 200 OK.
        // 'x-oss-copy-source-if-unmodified-since' => gmdate('2021-12-09T07:01:56.000Z'),
        // If the time that is specified in the request is earlier than the actual modified time of the source object, OSS copies the object and returns 200 OK.
        // 'x-oss-copy-source-if-modified-since' => gmdate('2021-12-09T07:01:56.000Z'),
        // Specify the method that is used to configure the metadata of the destination object. In this example, the method is set to COPY, which indicates that the metadata of the source object is copied to the destination object.
        // 'x-oss-metadata-directive' => 'COPY',
        // Specify the server-side encryption algorithm that is used to encrypt the destination object when OSS creates the object.
        // 'x-oss-server-side-encryption' => 'KMS',
        // Specify the CMK that is managed by KMS. This parameter takes effect only when you set x-oss-server-side-encryption to KMS.
        // 'x-oss-server-side-encryption-key-id' => '9468da86-3509-4f8d-a61e-6eab****',
        // Specify the access control list (ACL) of the destination object when OSS creates the object. In this example, the ACL is set to private, which indicates that only the owner of the object and authorized users have read and write permissions on the object.
        // 'x-oss-object-acl' => 'private',
        // Specify the storage class of the destination object. In this example, the storage class is set to Standard.
        // 'x-oss-storage-class' => 'Standard',
        // Specify tags for the destination object. You can specify multiple tags for the destination object at the same time.
        // 'x-oss-tagging' => 'k1=v1&k2=v2&k3=v3',
        // Specify the method that is used to configure the metadata of the destination object. In this example, the method is set to COPY, which indicates that the metadata of the source object is copied to the destination object.
```

```

        // Specify the method that is used to configure tags for the destination object. In this example, the method is set to COPY, which indicates that the tags of the source object are copied to the destination object.
        // 'x-oss-tagging-directive' => 'COPY',
    ),
);
try{
    $ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint);
    $ossClient->copyObject($from_bucket, $from_object, $to_bucket, $to_object);
} catch(OssException $e) {
    printf(__FUNCTION__ . ": FAILED\n");
    printf($e->getMessage() . "\n");
    return;
}
print(__FUNCTION__ . ": OK" . "\n");

```

Copy a large object

1. Use `$ossClient->initiateMultipartUpload` to initiate a multipart copy task.
2. Use `$ossClient->uploadPartCopy` to perform multipart copy. All parts except for the last part must be larger than 100 KB.
3. Use `$ossClient->completeMultipartUpload` to complete the multipart copy task.

The following code provides an example on how to use multipart copy to copy a large object:

```

<?php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
$accessKeyId = "yourAccessKeyId";
$accessKeySecret = "yourAccessKeySecret";
// In this example, the endpoint of the China (Hangzhou) region is used. Specify the endpoint based on your business requirements.
$endpoint = "http://oss-cn-hangzhou.aliyuncs.com";
// Specify the name of the source bucket.
$from_bucket = "yourSrcBucketName";
// Specify the full path of the source object. The full path of the object cannot contain the bucket name. Example: srcdir/exampleobject.txt.
$from_object = "yourSrcObjectName";
// Specify the name of the destination bucket. The destination bucket must be in the same region as the source bucket.
$to_bucket = "yourDestBucketName";
// Specify the full path of the destination object. The full path of the object cannot contain the bucket name. Example: destdir/exampleobject.txt.
$to_object = "yourDestObjectName";

```



```

$to_object = yourDestObjectName;
// Specify the part size. Unit: bytes.
$part_size = 256*1024*1024;
try{
    $ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint);
    $objectMeta = $ossClient->getObjectMeta($from_bucket, $from_object);
    $length = $objectMeta['content-length'] + 0;
    // Initiate a multipart copy task.
    $upload_id = $ossClient->initiateMultipartUpload($to_bucket, $to_object);
    // Start the multipart copy task.
    $pieces = $ossClient->generateMultiuploadParts($length, $part_size);
    $response_upload_part = array();
    $copyId = 1;
    $upload_position = 0;
    foreach ($pieces as $i => $piece) {
        $from_pos = $upload_position + (integer)$piece['seekTo'];
        $to_pos = (integer)$piece['length'] + $from_pos - 1;
        $up_options = array(
            'start' => $from_pos,
            'end' => $to_pos,
            'headers'=>array(
                // If the ETag value of the source object is the same as the ETag value that
                // is specified in the request, OSS copies the object and returns 200 OK.
                // 'x-oss-copy-source-if-match' => '5B3C1A2E053D763E1B002CC****',
                // If the ETag value of the source object is different from the ETag value
                // that is specified in the request, OSS copies the object and returns 200 OK.
                // 'x-oss-copy-source-if-none-match' => '5B3C1A2E053D763E1B002CC****',
                // If the time that is specified in the request is later than or equal to the
                // actual modified time of the source object, OSS copies the object and returns 200 OK.
                // 'x-oss-copy-source-if-unmodified-since' => gmdate('2021-12-09T07:01:56.000Z'),
                // If the time that is specified in the request is earlier than the actual
                // modified time of the source object, OSS copies the object and returns 200 OK.
                // 'x-oss-copy-source-if-modified-since' => gmdate('2021-12-09T07:01:56.000Z'),
            ),
        );
        $response_upload_part[] = $ossClient->uploadPartCopy($from_bucket, $from_object, $to_bucket, $to_object, $copyId, $upload_id, $up_options);
        $copyId = $copyId + 1;
    }
    // Complete the multipart copy task.
    $upload_parts = array();
    foreach ($response_upload_part as $i => $etag) {
        $upload_parts[] = array(
            'PartNumber' => ($i + 1),
            'ETag' => $etag,
        );
    }
    $result = $ossClient->completeMultipartUpload($to_bucket, $to_object, $upload_id, $upload_parts);
} catch (OssException $e) {
    printf(__FUNCTION__ . ": FAILED\n");
    printf($e->getMessage() . "\n");
    return;
}

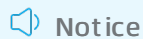
```

```
}  
print(__FUNCTION__ . ": OK" . "\n");
```

References

- Copy a small object
 - For more information about the complete sample code that is used to copy a small object, visit [GitHub](#).
 - For more information about the API operations that is called to copy a small object, see [CopyObject](#).
- Copy a large object
 - For more information about the complete sample code that is used to copy a large object, visit [GitHub](#).
 - For more information about the API operations that is called to copy a large object, see [UploadPartCopy](#).

3.8.10. Restore objects



Restoration

If you want to read an Archive object or a Cold Archive object, you must restore the object in advance. It takes several minutes to restore an Archive object and takes several hours to restore a Cold Archive object.

The following section describes the status of an Archive object or a Cold Archive object throughout the restoration process:

1. By default, an Archive or a Cold Archive object is in the frozen state before restoration.
2. After you submit a restore request, the object is in the restoring state.
3. After the server completes the restore task, the object enters the restored state and you can read the object.
 - Archive objects
For Archive objects, the restored state lasts 24 hours by default. During the 24 hours, if you call `RestoreObject` again, the restored state is extended by 24 hours. You can extend the restored state to seven days by calling `RestoreObject` six times during the first 24 hours of restored state. You can also configure the duration of the restored state in days by calling `RestoreObject` once. You can specify a duration of at most seven days.
 - Cold Archive objects
For Cold Archive objects, you can specify the duration of restored state and restoration priority. The duration of the restored state must be at least one day and at most 365 days. The time required to restore a Cold Archive object to the readable state is determined based on the restoration priority of the object:
 - Expedited: The object is restored within one hour.

- **Standard:** The object is restored within two to five hours. If the `JobParameters` element is not passed in, the default restoration priority is Standard.
- **Bulk:** The object is restored within five to twelve hours.

4. After the restored state expires, the object returns to the frozen state.

Restore Archive objects

The following code provides an example on how to restore an Archive object:

```
<?php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
$accessKeyId = "yourAccessKeyId";
$accessKeySecret = "yourAccessKeySecret";
// In this example, the endpoint of the China (Hangzhou) region is used. Specify your actual endpoint.
$endpoint = "http://oss-cn-hangzhou.aliyuncs.com";
// Specify the name of the bucket. Example: examplebucket.
$bucket = "examplebucket";
// Specify the full path of the object. The full path of the object cannot contain the bucket name. Example: srcexampleobject.txt.
$object = "srcexampleobject.txt";
try {
    $ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint);
    // Restore the object.
    $ossClient->restoreObject($bucket, $object);
} catch (OssException $e) {
    printf(__FUNCTION__ . ": FAILED\n");
    printf($e->getMessage() . "\n");
    return;
}
print(__FUNCTION__ . ": OK" . "\n");
```

Restore a Cold Archive object

The following code provides an example on how to restore a Cold Archive object:

```

<?php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
use OSS\Model\RestoreConfig;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
$accessKeyId = "yourAccessKeyId";
$accessKeySecret = "yourAccessKeySecret";
// In this example, the endpoint of the China (Hangzhou) region is used. Specify your actual endpoint.
$endpoint = "http://oss-cn-hangzhou.aliyuncs.com";
// Specify the name of the bucket. Example: examplebucket.
$bucket = "examplebucket";
// Specify the full path of the object. The full path of the object cannot contain the bucket name. Example: srcexampleobject.txt.
$object = "srcexampleobject.txt";
try {
    $ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint);
    // Specify the duration in which the object can remain in the restored state. Unit: days. Valid values: 1 to 7.
    $day = 3;
    // Specify the restoration priority of the object. In this example, this parameter is set to Expedited, which specifies that the object is restored within 1 hour.
    // If you want to restore the object within 2 to 5 hours, set this parameter to Standard. If you want to restore the object within 5 to 12 hours, set this parameter to Bulk.
    $tier = "Expedited";
    $config = new RestoreConfig($day, $tier);
    $options = array(
        OssClient::OSS_RESTORE_CONFIG => $config
    );
    $result = $ossClient->restoreObject($bucket, $object, $options);
} catch (OssException $e) {
    printf($e->getMessage() . "\n");
    return;
}

```

References

For more information about the API operation that is called to restore Archive and Cold Archive objects, see [RestoreObject](#).

3.8.11. Manage symbolic links

Symbolic links can be used to access objects that are frequently used in buckets. After you create a symbolic link for an object, you can use the symbolic link to access the object.

Create a symbolic link

The following code provides an example on how to create a symbolic link:

```
<?php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access Object Storage Service (OSS) because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
$accessKeyId = "yourAccessKeyId";
$accessKeySecret = "yourAccessKeySecret";
// In this example, the endpoint of the China (Hangzhou) region is used. Specify the endpoint based on your business requirements.
$endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
// Specify the name of the bucket. Example: examplebucket.
$bucket = "examplebucket";
// Specify the name of the object to which the symbolic link points. Example: exampleobject.txt.
$object = "exampleobject.txt";
// Specify the name of the symbolic link. Example: examplesymlink.txt.
$symlink = "examplesymlink.txt";
$options[OssClient::OSS_HEADERS] = array(
    // Specify whether to overwrite the object with the same name when the symbolic link is created. In this example, this parameter is set to true, which indicates that the object with the same name cannot be overwritten.
    'x-oss-forbid-overwrite'=>"true",
    // Specify the access control list (ACL) of the object. In this example, this parameter is set to private, which indicates that only the owner of the object and authorized users have read and write permissions on the object.
    'x-oss-object-acl' =>"private",
    // Specify the storage class of the object. In this example, this parameter is set to Standard.
    'x-oss-storage-class' =>"Standard"
);
try {
    $ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint,$options);
    $ossClient->putSymlink($bucket, $symlink, $object);
} catch (OssException $e) {
    printf($e->getMessage() . "\n");
    return;
}
```

Query the name of the object to which a symbolic link points

To query a symbolic link, you must have read permissions on the symbolic link. The following code provides an example on how to query a symbolic link and the name of the object to which the symbolic link points:

```
<?php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
$accessKeyId = "<yourAccessKeyId>";
$accessKeySecret = "<yourAccessKeySecret>";
// In this example, the endpoint of the China (Hangzhou) region is used. Specify the endpoint based on your business requirements.
$endpoint = "http://oss-cn-hangzhou.aliyuncs.com";
// Specify the name of the bucket.
$bucket = "examplebucket";
// Specify the name of the symbolic link that you want to query.
$symlink = "examplesymlink.txt";
try {
    $ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint);
    // Query the symbolic link and the name of the object to which the symbolic link points
    .
    $Symlinks = $ossClient->getSymlink($bucket, $symlink);
} catch (OssException $e) {
    printf(__FUNCTION__ . ": FAILED\n");
    printf($e->getMessage() . "\n");
    return;
}
print(__FUNCTION__ . ": OK" . "\n");
var_dump($Symlinks);
```

References

- For more information about the complete sample code of symbolic links, visit [GitHub](#).
- For more information about the API operation that you can call to create a symbolic link, see [Put Symlink](#).
- For more information about the API operation that you can call to query a symbolic link, see [Get Symlink](#).

3.8.12. Enable MD5 verification

MD5 verification is performed to ensure data integrity during transmission. MD5 verification brings performance loss. MD5 verification is disabled by default during file uploads.

Run the following code to enable MD5 verification during file uploads:

```
<? php
if (is_file(__DIR__ . '/../Autoload.php ')){
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
// It is highly risky to log on with AccessKey of an Alibaba Cloud account because the account has permissions on all the APIs in OSS. We recommend that you log on as a RAM user to access APIs or perform routine operations and maintenance. To create a RAM account, log on to https://ram.console.aliyun.com.
$accessKeyId = "<yourAccessKeyId>";
$accessKeySecret = "<yourAccessKeySecret>";
// This example uses endpoint China East 1 (Hangzhou). Specify the actual endpoint based on your requirements.
$endpoint = "http://oss-cn-hangzhou.aliyuncs.com";
$bucket = "<yourBucketName>";
$object = "<yourObjectName>";
$options = array(OssClient::OSS_CHECK_MD5 => true);
try{
    $ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint);
    $ossClient->uploadFile($bucket, $object, __FILE__, $options);
} catch(OssException $e) {
    printf(__FUNCTION__ . ": FAILED\n");
    printf($e->getMessage() . "\n");
    return;
}
print(__FUNCTION__ . ": OK" . "\n");
```

The `putObject`, `uploadFile`, `appendObject`, `appendFile` and `multiuploadFile` methods support MD5 verification.

3.9. Versioning

3.9.1. Versioning

The versioning state of a bucket applies to all objects in the bucket. Versioning allows you to restore objects in a bucket to any previous version and protects your data from being accidentally overwritten or deleted.

A bucket can be in one of the following versioning states: unversioned (default), enabled, or suspended. For more information about versioning, see [Overview](#) in OSS Developer Guide.



Notice If versioning is enabled for a bucket, the versioning state of the bucket cannot be set back to unversioned. However, you can set the versioning state of the bucket to suspended.

Set the versioning state of a bucket

The following code provides an example on how to set the versioning state of a bucket:

```
<? php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log on to OSS because the account has permissions on all API operations. We recommend that you use your RAM user's credentials to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
$accessKeyId = "<yourAccessKeyId>";
$accessKeySecret = "<yourAccessKeySecret>";
// The endpoint of the China (Hangzhou) region is used in this example. Specify the actual endpoint.
$endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
$bucket = "<yourBucketName>";
$ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint);
try {
    // Set the versioning state of the bucket to enabled or suspended.
    $ossClient->putBucketVersioning($bucket, "Enabled");
} catch (OssException $e) {
    printf(__FUNCTION__ . ": FAILED\n");
    printf($e->getMessage() . "\n");
    return;
}
print(__FUNCTION__ . ": OK" . "\n");
```

For more information about how to set the versioning state of a bucket, see [PutBucketVersioning](#).

Query the versioning state of a bucket

The following code provides an example on how to query the versioning state of a bucket:


```

<? php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to lo
g on to OSS because the account has permissions on all API operations. We recommend that yo
u use your RAM user's credentials to call API operations or perform routine operations and
maintenance. To create a RAM user, log on to the RAM console.
$accessKeyId = "<yourAccessKeyId>";
$accessKeySecret = "<yourAccessKeySecret>";
// The endpoint of the China (Hangzhou) region is used in this example. Specify the actual
endpoint.
$endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
$bucket= "<yourBucketName>";
$ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint);
try {
    // Query the versioning state of the bucket.
    $status = $ossClient->getBucketVersioning($bucket);
    print($status);
} catch (OssException $e) {
    printf(__FUNCTION__ . ": FAILED\n");
    printf($e->getMessage() . "\n");
    return;
}
print(__FUNCTION__ . ": OK" . "\n");

```

For more information about how to query the versioning state of a bucket, see [GetBucketVersioning](#).

Lists the versions of all objects in a bucket.

The following code provides an example on how to list the versions of all objects and delete markers in a bucket:

```

<? php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to lo
g on to OSS because the account has permissions on all API operations. We recommend that yo
u use your RAM user's credentials to call API operations or perform routine operations and
maintenance. To create a RAM user, log on to the RAM console.
$accessKeyId = "<yourAccessKeyId>";
$accessKeySecret = "<yourAccessKeySecret>";
// The endpoint of the China (Hangzhou) region is used in this example. Specifv the actual

```

```

endpoint.
$endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
$bucket= "<yourBucketName>";
$ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint);
try {
    $maxKey = 100;
    $nextKeyMarker = '';
    $nextVersionIdMarker = '';
    while (true) {
        $options = array(
            'delimiter' => '/',
            'key-marker' => $nextKeyMarker,
            'max-keys' => $maxKey,
            'version-id-marker' => $nextVersionIdMarker,
        );
        $result = $ossClient->listObjectVersions($bucket, $options);
        $nextKeyMarker = $result->getNextKeyMarker();
        $nextVersionIdMarker = $result->getNextVersionIdMarker();
        $objectList = $result->getObjectVersionList();
        $deleteMarkerList = $result->getDeleteMarkerList();
        // Display the versions of all objects in the bucket.
        if (! empty($objectList)) {
            print("objectList:\n");
            foreach ($objectList as $objectInfo) {
                print($objectInfo->getKey() . ",");
                print($objectInfo->getVersionId() . ",");
                print($objectInfo->getLastModified() . ",");
                print($objectInfo->getETag() . ",");
                print($objectInfo->getSize() . ",");
                print($objectInfo->getIsLatest() . "\n");
            }
        }
        // Display the version information about all delete markers in the bucket.
        if (! empty($deleteMarkerList)) {
            print("deleteMarkerList: \n");
            foreach ($deleteMarkerList as $deleteMarkerInfo) {
                print($deleteMarkerInfo->getKey() . ",");
                print($deleteMarkerInfo->getVersionId() . ",");
                print($deleteMarkerInfo->getLastModified() . ",");
                print($deleteMarkerInfo->getIsLatest() . "\n");
            }
        }
        if ($result->getIsTruncated() !== "true") {
            break;
        }
    }
} catch (OssException $e) {
    printf(__FUNCTION__ . ": FAILED\n");
    printf($e->getMessage() . "\n");
    return;
}
print(__FUNCTION__ . ": OK" . "\n");

```

For more information about how to list the versions of all objects and delete markers in a bucket, see [GetBucketVersions\(ListObjectVersions\)](#).

3.9.2. Upload objects

This topic describes how to upload objects to a versioned bucket.

Simple upload

If you initiate a PutObject request to upload an object to a versioned bucket, OSS generates a globally unique version ID for the uploaded object and includes the version ID in the response as the `x-oss-version-id` header.

If you initiate a PutObject request to upload an object to a bucket for which versioning is suspended, OSS generates a version ID null for the uploaded object. In this case, if you upload a new object with the same name as the existing object to the bucket, the existing object is overwritten.

The following code provides an example on how to upload an object to a versioned bucket by using simple upload:

```

<? php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to lo
g on to OSS because the account has permissions on all API operations. We recommend that yo
u use your RAM user's credentials to call API operations or perform routine operations and
maintenance. To create a RAM user, log on to the RAM console.
$accessKeyId = "<yourAccessKeyId>";
$accessKeySecret = "<yourAccessKeySecret>";
// The endpoint of the China (Hangzhou) region is used in this example. Specify the actual
endpoint.
$endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
$bucket= "<yourBucketName>";
// Specify the complete name of the object excluding the bucket name. Example: example/test
.txt.
$object = "<yourObjectName>";
$content = "hello world";
$ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint);
try {
    // Upload the object to the versioned bucket.
    $ret = $ossClient->putObject($bucket, $object, $content);
    // Display the version ID of the uploaded object.
    print("versionId:" . $ret[OssClient::OSS_HEADER_VERSION_ID]);
} catch (OssException $e) {
    printf(__FUNCTION__ . ": FAILED\n");
    printf($e->getMessage() . "\n");
    return;
}
print(__FUNCTION__ . ": OK" . "\n");

```

For more information about simple upload, see [PutObject](#).

Append upload

When you upload an object to a versioned bucket by using append upload, take note of the following items:

- You can perform the AppendObject operation only on an object whose current version is an append object.
- The AppendObject operation cannot be performed on an object whose current version is not an append object, such as a normal object or a delete marker.
- When you perform the AppendObject operation on an object whose current version is an append object, OSS does not generate a previous version for the append object.
- When you perform the PutObject or DeleteObject operation on an object whose current version is an append object, OSS stores the append object as a previous version and prevents the object from being further appended.

The following code provides an example on how to upload an object to a versioned bucket by using append upload:

```
<? php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log on to OSS because the account has permissions on all API operations. We recommend that you use your RAM user's credentials to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
$accessKeyId = "<yourAccessKeyId>";
$accessKeySecret = "<yourAccessKeySecret>";
// The endpoint of the China (Hangzhou) region is used in this example. Specify the actual endpoint.
$endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
$bucket = "<yourBucketName>";
$object = "<yourObjectName>";
// Specifies that the object content obtained after the first, second, and third append upload is Hello OSS, Hi OSS, and OSS OK.
$content_array = array('Hello OSS', 'Hi OSS', 'OSS OK');
try{
    $ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint);
    // Perform the first append upload. // If the object is appended for the first time, the append position is 0. The return value is the position for the next append. The position to start the next append is the total length of appended bytes and the append object.
    $position = $ossClient->appendObject($bucket, $object, $content_array[0], 0);
    $position = $ossClient->appendObject($bucket, $object, $content_array[1], $position);

    $position = $ossClient->appendObject($bucket, $object, $content_array[2], $position);
} catch(OssException $e) {
    printf(__FUNCTION__ . ": FAILED\n");
    printf($e->getMessage() . "\n");
    return;
}
print(__FUNCTION__ . ": OK" . "\n");
```

For more information about append upload, see [AppendObject](#).

Multipart upload

If you call the CompleteMultipartUpload operation to complete the MultipartUpload task for an object in a versioned bucket, OSS generates a unique version ID for the uploaded object and returns the version ID in the response as the `x-oss-version-id` header.

The following code provides an example on how to upload an object to a versioned bucket by using multipart upload:

```
<? php
if (is_file(__DIR__ . '/../autoload.php')) {
```

```

if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
use OSS\Core\OssUtil;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to lo
g on to OSS because the account has permissions on all API operations. We recommend that yo
u use your RAM user's credentials to call API operations or perform routine operations and
maintenance. To create a RAM user, log on to the RAM console.
$accessKeyId = "<yourAccessKeyId>";
$accessKeySecret = "<yourAccessKeySecret>";
// The endpoint of the China (Hangzhou) region is used in this example. Specify the actual
endpoint.
$endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
$bucket = "<yourBucketName>";
$object = "<yourObjectName>";
// Specify the complete path of the local file to upload.
$uploadFile = "<yourLocalFile>";
/**
 * Step 1: Initiate a multipart upload task and obtain the upload ID.
 */
try{
    $ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint);
    // Obtain the upload ID. // The upload ID uniquely identifies the multipart upload task
    . You can use the upload ID to cancel or query the multipart upload task.
    $uploadId = $ossClient->initiateMultipartUpload($bucket, $object);
} catch(OssException $e) {
    printf(__FUNCTION__ . ": initiateMultipartUpload FAILED\n");
    printf($e->getMessage() . "\n");
    return;
}
print(__FUNCTION__ . ": initiateMultipartUpload OK" . "\n");
/**
 * Step 2: Upload parts.
 */
$partSize = 10 * 1024 * 1024;
$uploadFileSize = sprintf('%u', filesize($uploadFile));
$pieces = $ossClient->generateMultiuploadParts($uploadFileSize, $partSize);
$responseUploadPart = array();
$uploadPosition = 0;
$isCheckMd5 = true;
foreach ($pieces as $i => $piece) {
    $fromPos = $uploadPosition + (integer)$piece[$ossClient::OSS_SEEK_TO];
    $toPos = (integer)$piece[$ossClient::OSS_LENGTH] + $fromPos - 1;
    $uploadOptions = array(
        // Upload the object.
        $ossClient::OSS_FILE_UPLOAD => $uploadFile,
        // Configure part numbers.
        $ossClient::OSS_PART_NUM => ($i + 1),
        // Specify the position from which the multipart upload task starts.
        $ossClient::OSS_SEEK_TO => $fromPos.

```

```

    // Specify the object size.
    $ossClient::OSS_LENGTH => $toPos - $fromPos + 1,
    // Specify whether to enable MD5 verification. A value of true indicates MD5 verification is enabled. A value of false indicates MD5 verification is disabled.
    $ossClient::OSS_CHECK_MD5 => $isCheckMd5,
);
// Enable MD5 verification.
if ($isCheckMd5) {
    $contentMd5 = OssUtil::getMd5SumForFile($uploadFile, $fromPos, $toPos);
    $uploadOptions[$ossClient::OSS_CONTENT_MD5] = $contentMd5;
}
try {
    // Upload the parts.
    $responseUploadPart[] = $ossClient->uploadPart($bucket, $object, $uploadId, $uploadOptions);
} catch (OssException $e) {
    printf(__FUNCTION__ . ": initiateMultipartUpload, uploadPart - part#{i} FAILED\n");
};

printf($e->getMessage() . "\n");
return;
}
printf(__FUNCTION__ . ": initiateMultipartUpload, uploadPart - part#{i} OK\n");
}
// $uploadParts is an array that consists of the ETag and the PartNumber of each part.
$uploadParts = array();
foreach ($responseUploadPart as $i => $eTag) {
    $uploadParts[] = array(
        'PartNumber' => ($i + 1),
        'ETag' => $eTag,
    );
}
/**
 * Step 3: Complete the multipart upload task.
 */
try {
    // All valid $uploadParts are required for the CompleteMultipartUpload operation. OSS verifies the validity of each part after it receives the submitted $uploadParts. After all parts are verified, OSS combines these parts into a complete object.
    $ret = $ossClient->completeMultipartUpload($bucket, $object, $uploadId, $uploadParts);
    // Display the version ID of the uploaded object.
    print("versionId:" . $ret[$ossClient::OSS_HEADER_VERSION_ID]);
} catch (OssException $e) {
    printf(__FUNCTION__ . ": completeMultipartUpload FAILED\n");
    printf($e->getMessage() . "\n");
    return;
}
printf(__FUNCTION__ . ": completeMultipartUpload OK\n");

```

For more information about multipart upload, see [InitiateMultipartUpload](#), [UploadPart](#), and [CompleteMultipartUpload](#).

3.9.3. Download objects

By default, when you perform the `GetObject` operation on an object in a versioned bucket, only the current version of the object is returned.

When you perform the `GetObject` operation on an object in a bucket, you may obtain one of the following results:

- If the current version of the object is a delete marker, OSS returns 404 Not Found.
- If the version ID of the object is specified in the request, OSS returns the specified version of the object. If the version ID is set to null in the request, OSS returns the version whose version ID is null.
- If the version ID specified in the request is the version ID of a delete marker, OSS returns 405 Method Not Allowed.

The following code provides an example on how to download an object:

```
<? php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log on to OSS because the account has permissions on all API operations. We recommend that you use your RAM user's credentials to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
$accessKeyId = "<yourAccessKeyId>";
$accessKeySecret = "<yourAccessKeySecret>";
// The endpoint of the China (Hangzhou) region is used in this example. Specify the actual endpoint.
$endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
$bucket = "<yourBucketName>";
// Specify the complete name of the object excluding the bucket name. Example: example/test.txt.
$object = "<yourObjectName>";
// Specify the version ID of the version you want to download.
$versionId = "<yourObjectVersionId>";
$ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint);
try{
    // Download the version with the specified version ID.
    $content = $ossClient->getObject($bucket, $object, array(OssClient::OSS_VERSION_ID => $versionId));
    print($content);
} catch(OssException $e) {
    printf(__FUNCTION__ . ": FAILED\n");
    printf($e->getMessage() . "\n");
    return;
}
print(__FUNCTION__ . ": OK" . "\n");
```

For more information about how to download an object, see [GetObject](#).

3.9.4. Copy objects

This topic describes how to copy objects in a versioned bucket. You can use `CopyObject` to copy an object that is up to 1 GB in size and use `UploadPartCopy` to copy an object larger than 1 GB.

Copy a small object

You can use `CopyObject` to copy an object that is up to 1 GB in size from a source bucket to a destination bucket in the same region.

- By default, the `x-oss-copy-source` header in a `CopyObject` request specifies the current version of the object to copy. If the current version of the object to copy is a delete marker, OSS returns 404 Not Found to indicate that the object does not exist. You can add a version ID in the `x-oss-copy-source` header to copy a specified version of the object. Delete markers cannot be copied.
- You can copy a previous version of an object to the same bucket. The copied previous version becomes the current version of the object. This way, you can restore an object to a specified previous version.
- If versioning is enabled for the destination bucket, OSS generates a unique version ID for the new object. The version ID is returned as the `x-oss-version-id` header in the response. If versioning is disabled or suspended for the destination bucket, OSS generates a version whose version ID is null for the destination object and overwrites the original version whose version ID is null.
- Append objects in a bucket for which versioning is enabled or suspended cannot be copied.

The following code provides an example on how to copy a small object:

```

<? php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to lo
g on to OSS because the account has permissions on all API operations. We recommend that yo
u use your RAM user's credentials to call API operations or perform routine operations and
maintenance. To create a RAM user, log on to the RAM console.
$accessKeyId = "<yourAccessKeyId>";
$accessKeySecret = "<yourAccessKeySecret>";
// The endpoint of the China (Hangzhou) region is used in this example. Specify the actual
endpoint.
$endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
$bucket= "<yourBucketName>";
// Specify the complete name of the object excluding the bucket name. Example: example/test
.txt.
$object = "<yourObjectName>";
$versionId = "<yourObjectVersionId>";
$to_bucket = $bucket;
$to_object = $object . '.copy';
$ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint);
try{
    // Copy the object with the specified version ID.
    $ret = $ossClient->copyObject($bucket, $object, $to_bucket, $to_object, array(OssClient
::OSS_VERSION_ID => $versionId));
    print("versionId:" . $ret['x-oss-version-id']);
} catch(OssException $e) {
    printf(__FUNCTION__ . ": FAILED\n");
    printf($e->getMessage() . "\n");
    return;
}
print(__FUNCTION__ . ": OK" . "\n");

```

For more information about how to copy small objects, see [CopyObject](#).

Copy a large object

To copy an object larger than 1 GB, you must split the object into parts and copy them sequentially by using UploadPart Copy.

By default, the UploadPart Copy operation uploads a part by copying data from the current version of an existing object. To copy data from a specific version of an object, include the versionId parameter in the `x-oss-copy-source` header in the request as a subcondition. For example, you can set `x-oss-copy-source` to `x-oss-copy-source : /SourceBucketName/SourceObjectName?versionId=111111`.

 **Note** The value of SourceObjectName must be URL-encoded. The version ID of the copied object version is returned as the x-oss-copy-source-version-id header value in the response.

If versionId is not specified in the request and the current version of the source object is a delete marker, OSS returns 404 Not Found. If versionId is specified in the request and the specified version of the source object is a delete marker, OSS returns 400 Bad Request.

The following code provides an example on how to use multipart copy to copy a large object:

```
<? php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to lo
g on to OSS because the account has permissions on all API operations. We recommend that yo
u use your RAM user's credentials to call API operations or perform routine operations and
maintenance. To create a RAM user, log on to the RAM console.
$accessKeyId = "<yourAccessKeyId>";
$accessKeySecret = "<yourAccessKeySecret>";
// The endpoint of the China (Hangzhou) region is used in this example. Specify the actual
endpoint.
$endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
$bucket = "<yourBucketName>";
$object = "<yourObjectName>";
$versionId = "<yourObjectVersionId>";
$to_bucket = $bucket;
$to_object = $object . '.copy';
$ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint);
/**
 * Step 1: Initiate a multipart upload task and obtain the upload ID.
 */
try {
    // Obtain the upload ID. The upload ID is the unique identifier for a multipart upload
    task. You can perform related operations such as cancel or query the multipart upload task
    based on the upload ID.
    $uploadId = $ossClient->initiateMultipartUpload($to_bucket, $to_object);
} catch (OssException $e) {
    printf(__FUNCTION__ . ": initiateMultipartUpload FAILED\n");
    printf($e->getMessage() . "\n");
    return;
}
print(__FUNCTION__ . ": initiateMultipartUpload OK" . "\n");
/*
 * Step 2: Upload parts.
 */
$partSize = 10 * 1024 * 1024;
try {
    $meta = $ossClient->getSimplifiedObjectMeta($bucket, $object);
    $uploadSize = $meta[strtolower('Content-Length')];
```

```

    } catch(OssException $e) {
        printf(__FUNCTION__ . ": getSimplifiedObjectMeta FAILED\n");
        printf($e->getMessage() . "\n");
        return;
    }
    $pieces = $ossClient->generateMultiuploadParts($uploadSize, $partSize);
    $responseUploadPart = array();
    $uploadPosition = 0;
    foreach ($pieces as $i => $piece) {
        $fromPos = $uploadPosition + (integer)$piece[$ossClient::OSS_SEEK_TO];
        $toPos = (integer)$piece[$ossClient::OSS_LENGTH] + $fromPos - 1;
        $upOptions = array(
            // Specify the position of the source object from which the multipart copy task starts.
            'start' => $fromPos,
            // Specify the position of the source object at which the multipart copy task ends.
            'end' => $toPos,
            // Specify the version ID of the source object you want to copy.
            OssClient::OSS_VERSION_ID => $versionId
        );
        try {
            // Upload the parts.
            $responseUploadPart[] = $ossClient->uploadPartCopy($bucket, $object, $to_bucket, $to_object, ($i + 1), $uploadId, $upOptions);
        } catch(OssException $e) {
            printf(__FUNCTION__ . ": initiateMultipartUpload, uploadPartCopy - part#{ $i } FAILED\n");
            printf($e->getMessage() . "\n");
            return;
        }
        printf(__FUNCTION__ . ": initiateMultipartUpload, uploadPartCopy - part#{ $i } OK\n");
    }
    // $uploadParts is an array that consists of the ETag and the PartNumber of each part.
    $uploadParts = array();
    foreach ($responseUploadPart as $i => $eTag) {
        $uploadParts[] = array(
            'PartNumber' => ($i + 1),
            'ETag' => $eTag,
        );
    }
    /**
     * Step 3: Complete multipart upload task.
     */
    try {
        // All valid $uploadParts are required for the CompleteMultipartUpload operation. OSS verifies the validity of each part after it receives the submitted $uploadParts. After all parts are verified, OSS combines these parts into a complete object.
        $ret = $ossClient->completeMultipartUpload($to_bucket, $to_object, $uploadId, $uploadParts);
        // Display the version ID of the copied object.
        print("versionId:" . $ret[OssClient::OSS_HEADER_VERSION_ID]);
    } catch(OssException $e) {
        printf(__FUNCTION__ . ": completeMultipartUpload FAILED\n");
    }
}

```

```
        printf($e->getMessage() . "\n");
        return;
    }
    printf(__FUNCTION__ . ": completeMultipartUpload OK\n");
```

For more information about multipart copy, see [UploadPartCopy](#).

3.9.5. List objects

This topic describes how to list objects in a versioned bucket, such as all objects, a specified number of objects, and objects whose names contain a specified prefix.

Lists the versions of all objects in a bucket.

The following code provides an example on how to list the versions of all objects including delete markers within the examplebucket bucket:

```

<?php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access Object Storage Service (OSS) because the account has permissions on all API operations. We recommend that you use a Resource Access Management (RAM) user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
$accessKeyId = "yourAccessKeyId";
$accessKeySecret = "yourAccessKeySecret";
// Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com.
$endpoint = "yourEndpoint";
// Specify the name of the bucket. Example: examplebucket.
$bucket = "examplebucket";
$ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint, false);
try{
    $option = array(
        OssClient::OSS_KEY_MARKER => null,
        OssClient::OSS_VERSION_ID_MARKER => null
    );
    $bool = true;
    while ($bool){
        $result = $ossClient->listObjectVersions($bucket,$option);
        // Display the information about the versions of the listed objects.
        foreach ($result->getObjectVersionList() as $key => $info){
            printf("key name: {$info->getKey()}\n");
            printf("versionid: {$info->getVersionId()}\n");
            printf("Is latest: {$info->getIsLatest()}\n\n");
        }
        // Display the version information of the listed delete markers.
        foreach ($result->getDeleteMarkerList() as $key => $info){
            printf("del_maker key name: {$info->getKey()}\n");
            printf("del_maker versionid: {$info->getVersionId()}\n");
            printf("del_maker Is latest: {$info->getIsLatest()}\n\n");
        }
        if($result->getIsTruncated() === 'true'){
            $option[OssClient::OSS_KEY_MARKER] = $result->getNextKeyMarker();
            $option[OssClient::OSS_VERSION_ID_MARKER] = $result->getNextVersionIdMarker();
        }else{
            $bool = false;
        }
    }
} catch(OssException $e) {
    printf($e->getMessage() . "\n");
    return;
}

```

List the versions of objects whose names contain a specified prefix

The following code provides an example on how to list the versions of objects whose names contain the "test" prefix in the bucket named examplebucket:

```
<?php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
$accessKeyId = "yourAccessKeyId";
$accessKeySecret = "yourAccessKeySecret";
// Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com.
$endpoint = "yourEndpoint";
// Specify the name of the bucket. Example: examplebucket.
$bucket = "examplebucket";
$ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint, false);
try{
    $option = array(
        OssClient::OSS_KEY_MARKER => null,
        OssClient::OSS_VERSION_ID_MARKER => null,
        // Specify that all versions of objects whose names contain the "test" prefix are listed.
        OssClient::OSS_PREFIX => "test"
    );
    $bool = true;
    while ($bool){
        $result = $ossClient->listObjectVersions($bucket,$option);
        // Display the information about the versions of the listed objects.
        foreach ($result->getObjectVersionList() as $key => $info){
            printf("key name: {$info->getKey()}\n");
            printf("versionid: {$info->getVersionId()}\n");
            printf("Is latest: {$info->getIsLatest()}\n\n");
        }
        // Display the version information of the listed delete markers.
        foreach ($result->getDeleteMarkerList() as $key => $info){
            printf("del_maker key name: {$info->getKey()}\n");
            printf("del_maker versionid: {$info->getVersionId()}\n");
            printf("del_maker Is latest: {$info->getIsLatest()}\n\n");
        }
        if($result->getIsTruncated() === 'true'){
            $option[OssClient::OSS_KEY_MARKER] = $result->getNextKeyMarker();
            $option[OssClient::OSS_VERSION_ID_MARKER] = $result->getNextVersionIdMarker();
        }else{
            $bool = false;
        }
    }
}
```

```
        $bool = false;
    }
} catch(OssException $e) {
    printf($e->getMessage() . "\n");
    return;
}
```

List a specified number of object versions

The following code provides an example on how to list a specified number of object versions in the examplebucket bucket:


```

<?php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
$accessKeyId = "yourAccessKeyId";
$accessKeySecret = "yourAccessKeySecret";
// Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com.
$endpoint = "yourEndpoint";
// Specify the name of the bucket. Example: examplebucket.
$bucket = "examplebucket";
$ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint, false);
try{
    $option = array(
        OssClient::OSS_KEY_MARKER => null,
        OssClient::OSS_VERSION_ID_MARKER => null,
        OssClient::OSS_MAX_KEYS => 200 // Specify that the maximum number of object versions that you can list is 200.
    );
    $result = $ossClient->listObjectVersions($bucket,$option);
    // Display the information about the versions of the listed objects.
    foreach ($result->getObjectVersionList() as $key => $info){
        printf("key name: ".$info->getKey().PHP_EOL);
        printf("versionid: ".$info->getVersionId().PHP_EOL);
        printf("Is latest: ".$info->getIsLatest().PHP_EOL.PHP_EOL);
    }
    // Display the version information of the listed delete markers.
    foreach ($result->getDeleteMarkerList() as $key => $info){
        printf("del_maker key name: ".$info->getKey().PHP_EOL);
        printf("del_maker versionid: ".$info->getVersionId().PHP_EOL);
        printf("del_maker Is latest: ".$info->getIsLatest().PHP_EOL.PHP_EOL);
    }
} catch(OssException $e) {
    printf($e->getMessage() . "\n");
    return;
}
}

```

List objects in a directory

OSS uses a flat structure for objects instead of a hierarchical structure to store objects. A directory is an object whose size is 0 and whose name ends with a forward slash (/). You can upload and download this object. By default, objects whose names end with a forward slash (/) are displayed as directories in the OSS console.

You can specify the delimiter and prefix parameters to list objects by directory.

- If you set prefix to a directory name in the request, objects and subdirectories whose names contain the prefix are listed.
- If you also set delimiter to a forward slash (/) in the request, objects and subdirectories whose names start with the specified prefix in the directory are listed. Each subdirectory is listed as a single result element in CommonPrefixes. The objects and directories in these subdirectories are not listed.
- List the versions of objects within the root directory of a bucket

The following code provides an example on how to list the versions of objects within the root directory of the bucket named examplebucket:

```
<?php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to
access OSS because the account has permissions on all API operations. We recommend that y
ou use a RAM user to call API operations or perform routine O&M. To create a RAM user, lo
g on to the RAM console.
$accessKeyId = "yourAccessKeyId";
$accessKeySecret = "yourAccessKeySecret";
// Set yourEndpoint to the endpoint of the region in which the bucket is located. For exa
mple, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https:
//oss-cn-hangzhou.aliyuncs.com.
$endpoint = "yourEndpoint";
// Specify the name of the bucket. Example: examplebucket.
$bucket = "examplebucket";
$ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint, false);
try{
    $option = array(
        OssClient::OSS_KEY_MARKER => null,
        OssClient::OSS_VERSION_ID_MARKER => null,
        OssClient::OSS_DELIMITER => "/",    // Specify that the delimiter is forward slas
hes (/).
    );
    $bool = true;
    while ($bool){
        $result = $ossClient->listObjectVersions($bucket,$option);
        // Display the information about the versions of the listed objects.
        foreach ($result->getObjectVersionList() as $key => $info){
            printf("key name: {$info->getKey()}\n");
            printf("versionid: {$info->getVersionId()}\n");
            printf("Is latest: {$info->getIsLatest()}\n\n");
        }
        // Display the version information of the listed delete markers.
        foreach ($result->getDeleteMarkerList() as $key => $info){
            printf("del_maker key name: {$info->getKey()}\n");
            printf("del_maker versionid: {$info->getVersionId()}\n");
        }
    }
}
```

```

        printf("del_maker Is latest: {$info->getIsLatest()}\n\n");
    }
    if($result->getIsTruncated() === 'true'){
        $option[OssClient::OSS_KEY_MARKER] = $result->getNextKeyMarker();
        $option[OssClient::OSS_VERSION_ID_MARKER] = $result->getNextVersionIdMarker()
;
    }else{
        $bool = false;
    }
}
} catch(OssException $e) {
    printf($e->getMessage() . "\n");
    return;
}
}

```

- List the versions of objects within a specified directory of a bucket

The following code provides an example on how to list the versions of objects within the test directory of the examplebucket bucket:

```

<?php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to
access OSS because the account has permissions on all API operations. We recommend that y
ou use a RAM user to call API operations or perform routine O&M. To create a RAM user, lo
g on to the RAM console.
$accessKeyId = "yourAccessKeyId";
$accessKeySecret = "yourAccessKeySecret";
// Set yourEndpoint to the endpoint of the region in which the bucket is located. For exa
mple, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https:
//oss-cn-hangzhou.aliyuncs.com.
$endpoint = "yourEndpoint";
// Specify the name of the bucket. Example: examplebucket.
$bucket= "examplebucket";
$ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint, false);
try{
    $option = array(
        OssClient::OSS_KEY_MARKER => null,
        OssClient::OSS_VERSION_ID_MARKER => null,
        OssClient::OSS_DELIMITER => "/",    // Specify that the delimiter is forward slas
hes (/).
        OssClient::OSS_PREFIX => "test/",    // Specify that the versions of objects withi
n the test directory are listed.
    );
    $bool = true;
    while ($bool){
        $result = $ossClient->listObjectVersions($bucket,$option);
        // Display the information about the versions of the listed objects.
    }
}

```

```

        foreach ($result->getObjectVersionList() as $key => $info){
            printf("key name: {$info->getKey()}\n");
            printf("versionid: {$info->getVersionId()}\n");
            printf("Is latest: {$info->getIsLatest()}\n\n");
        }
        // Display the version information of the listed delete markers.
        foreach ($result->getDeleteMarkerList() as $key => $info){
            printf("del_maker key name: {$info->getKey()}\n");
            printf("del_maker versionid: {$info->getVersionId()}\n");
            printf("del_maker Is latest: {$info->getIsLatest()}\n\n");
        }
        if($result->getIsTruncated() === 'true'){
            $option[OssClient::OSS_KEY_MARKER] = $result->getNextKeyMarker();
            $option[OssClient::OSS_VERSION_ID_MARKER] = $result->getNextVersionIdMarker();
        }
        ;
        }else{
            $bool = false;
        }
    }
} catch(OssException $e) {
    printf($e->getMessage() . "\n");
    return;
}

```

3.9.6. Delete objects

This topic describes how to delete a single object, multiple objects, or objects whose names contain a specified prefix from a versioned bucket.

Delete operations in a versioned bucket

- Do not specify a version ID (temporary deletion)

By default, if you do not specify the version ID of the object you want to delete in the request, OSS does not delete the current version of the object and instead add a delete marker to the object as the new current version. If you perform the `GetObject` operation on the object, OSS identifies the current version of the object as a delete marker and returns `404 Not Found`. In addition, the `x-oss-s-delete-marker` header and the `x-oss-version-id` header that indicates the version ID of the created delete marker are included in the response.

- Specify a version ID (permanent deletion)

If you specify the version ID of the object you want to delete in the request, OSS permanently deletes the version whose ID is specified by the `versionId` field in the `params` parameter. To delete a version whose ID is null, add `params['versionId'] = "null"` in `params` in the request. OSS identifies the string "null" as the ID of the version to delete and deletes the version whose ID is null.

Delete a single object

The following examples describe how to permanently or temporarily delete a single object from a versioned bucket.

- Permanently delete an object from a versioned bucket

The following code provides an example on how to permanently delete an object with a specified version ID from a versioned bucket:

```
<? php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to
log on to OSS because the account has permissions on all API operations. We recommend tha
t you use your RAM user's credentials to call API operations or perform routine operation
s and maintenance. To create a RAM user, log on to the RAM console.
$accessKeyId = "<yourAccessKeyId>";
$accessKeySecret = "<yourAccessKeySecret>";
// The endpoint of the China (Hangzhou) region is used in this example. Specify the actua
l endpoint.
$endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
$bucket= "<yourBucketName>";
// Specify the complete name of the object excluding the bucket name. Example: example/te
st.txt.
$object = "<yourObjectName>";
$versionId = "<yourObjectVersionId>";
$ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint);
try{
    // Delete the object with the specified version ID.
    $ossClient->deleteObject($bucket, $object, array(OssClient::OSS_VERSION_ID => $versio
nId));
} catch(OssException $e) {
    printf(__FUNCTION__ . ": FAILED\n");
    printf($e->getMessage() . "\n");
    return;
}
print(__FUNCTION__ . ": OK" . "\n");
```

- Temporarily delete an object from a versioned bucket

The following code provides an example on how to temporarily delete an object from a versioned bucket:

```

<? php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to
log on to OSS because the account has permissions on all API operations. We recommend tha
t you use your RAM user's credentials to call API operations or perform routine operation
s and maintenance. To create a RAM user, log on to the RAM console.
$accessKeyId = "<yourAccessKeyId>";
$accessKeySecret = "<yourAccessKeySecret>";
// The endpoint of the China (Hangzhou) region is used in this example. Specify the actua
l endpoint.
$endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
$bucket= "<yourBucketName>";
$object = "<yourObjectName>";
$ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint);
try{
    // Temporarily delete an object without specifying its version ID. A delete marker is
added to the object.
    $ret = $ossClient->deleteObject($bucket, $object);
    print("delete marker:" . $ret['x-oss-delete-marker'] . "\n");
    print("version id:" . $ret['x-oss-version-id']);
} catch(OssException $e) {
    printf(__FUNCTION__ . ": FAILED\n");
    printf($e->getMessage() . "\n");
    return;
}
print(__FUNCTION__ . ": OK" . "\n");

```

For more information about how to delete a single object, see [DeleteObject](#).

Delete multiple objects

The following examples describe how to permanently or temporarily delete multiple objects from a versioned bucket.

- Permanently delete multiple objects from a versioned bucket

The following code provides an example on how to permanently delete multiple objects with specified version IDs from a versioned bucket:

```

<? php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
use OSS\Model\DeleteObjectInfo;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to
log on to OSS because the account has permissions on all API operations. We recommend tha
t you use your RAM user's credentials to call API operations or perform routine operation
s and maintenance. To create a RAM user, log on to the RAM console.
$accessKeyId = "<yourAccessKeyId>";
$accessKeySecret = "<yourAccessKeySecret>";
// The endpoint of the China (Hangzhou) region is used in this example. Specify the actua
l endpoint.
$endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
$bucket= "<yourBucketName>";
$ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint);
try{
    // Delete the objects with the specified version IDs.
    $objects = array();
    $objects[] = new DeleteObjectInfo("<yourObject1Name>", "<object1VersionId>");
    $objects[] = new DeleteObjectInfo("<yourObject2Name>", "<object2VersionId>");
    $deletedObjectList = $ossClient->deleteObjectVersions($bucket, $objects);
    // Display the deleted objects.
    if (! empty($deletedObjectList)) {
        print("deletedObjectList:\n");
        foreach ($deletedObjectList as $deletedObjectInfo) {
            print($deletedObjectInfo->getKey() . ",");
            print($deletedObjectInfo->getVersionId() . ",");
            print($deletedObjectInfo->getDeleteMarker() . ",");
            print($deletedObjectInfo->getDeleteMarkerVersionId() . "\n");
        }
    }
} catch(OssException $e) {
    printf(__FUNCTION__ . ": FAILED\n");
    printf($e->getMessage() . "\n");
    return;
}
print(__FUNCTION__ . ": OK" . "\n");

```

- Temporarily delete multiple objects from a versioned bucket

The following code provides an example on how to temporarily delete multiple objects from a versioned bucket:

```

<? php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
use OSS\Model\DeleteObjectInfo;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to
log on to OSS because the account has permissions on all API operations. We recommend tha
t you use your RAM user's credentials to call API operations or perform routine operation
s and maintenance. To create a RAM user, log on to the RAM console.
$accessKeyId = "<yourAccessKeyId>";
$accessKeySecret = "<yourAccessKeySecret>";
// The endpoint of the China (Hangzhou) region is used in this example. Specify the actua
l endpoint.
$endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
$bucket= "<yourBucketName>";
$ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint);
try{
    // Temporarily delete multiple object without specifying their version IDs. Delete ma
rkers are added to the objects.
    $objects = array();
    $objects[] = new DeleteObjectInfo("<yourObject1Name>");
    $objects[] = new DeleteObjectInfo("<yourObject2Name>");
    $deletedObjectList = $ossClient->deleteObjectVersions($bucket, $objects);
    // Display the deleted objects.
    if (! empty($deletedObjectList)) {
        print("deletedObjectList:\n");
        foreach ($deletedObjectList as $deletedObjectInfo) {
            print($deletedObjectInfo->getKey() . ",");
            print($deletedObjectInfo->getVersionId() . ",");
            print($deletedObjectInfo->getDeleteMarker() . ",");
            print($deletedObjectInfo->getDeleteMarkerVersionId() . "\n");
        }
    }
} catch(OssException $e) {
    printf(__FUNCTION__ . ": FAILED\n");
    printf($e->getMessage() . "\n");
    return;
}
print(__FUNCTION__ . ": OK" . "\n");

```

For more information about how to delete multiple objects, see [DeleteMultipleObjects](#).

3.9.7. Restore objects

In a versioned bucket, the storage classes of different versions of an object can be different. By default, when you perform the `RestoreObject` operation on an object, the current version of the object is restored. You can specify a version ID in the request to restore the specified version of an object.

The following code provides an example on how to restore an object :

```
<? php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
use OSS\Model\DeleteObjectInfo;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log on to OSS because the account has permissions on all API operations. We recommend that you use your RAM user's credentials to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
$accessKeyId = "<yourAccessKeyId>";
$accessKeySecret = "<yourAccessKeySecret>";
// The endpoint of the China (Hangzhou) region is used in this example. Specify the actual endpoint.
$endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
$bucket = "<yourBucketName>";
// Specify the complete name of the object excluding the bucket name. Example: example/test.txt.
$object = "<yourObjectName>";
$versionId = "<yourObjectVersionId>";
$ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint);
try{
    // Restore the object with the specified version ID.
    $result = $ossClient->restoreObject($bucket, $object, array(OssClient::OSS_VERSION_ID => $versionId));
} catch(OssException $e) {
    printf(__FUNCTION__ . ": FAILED\n");
    printf($e->getMessage() . "\n");
    return;
}
print(__FUNCTION__ . ": OK" . "\n");
```

For more information about how to restore an object, see [RestoreObject](#).

3.9.8. Query object metadata

By default, when you perform the HeadObject operation on an object in a versioned bucket, only the metadata of the current version of the object is returned.

Note If the current version of the object is a delete marker, OSS returns 404 Not Found. If you specify a version ID in the request, OSS returns the metadata of the object with the specified version ID.

The following code provides an example on how to query object metadata:

```

<? php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to lo
g on to OSS because the account has permissions on all API operations. We recommend that yo
u use your RAM user's credentials to call API operations or perform routine operations and
maintenance. To create a RAM user, log on to the RAM console.
$accessKeyId = "<yourAccessKeyId>";
$accessKeySecret = "<yourAccessKeySecret>";
// The endpoint of the China (Hangzhou) region is used in this example. Specify the actual
endpoint.
$endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
$bucket= "<yourBucketName>";
$object = "<yourObjectName>";
$versionId = "<yourObjectVersionId>";
$ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint);
try {
    // Query all metadata of the object with the specified version ID.
    $objectMeta = $ossClient->getObjectMeta($bucket, $object, array(OssClient::OSS_VERSION_
ID => $versionId));
    print_r($objectMeta);
    // Query part of metadata of the object with the specified version ID.
    $objectMeta = $ossClient->getSimplifiedObjectMeta($bucket, $object, array(OssClient::OS
S_VERSION_ID => $versionId));
    print_r($objectMeta);
} catch (OssException $e) {
    printf(__FUNCTION__ . ": FAILED\n");
    printf($e->getMessage() . "\n");
    return;
}
print(__FUNCTION__ . ": OK" . "\n");

```

For more information about how to query object metadata, see [GetObjectMeta](#).

3.9.9. Manage object ACLs

OSS provides the following four access control lists (ACLs) for objects to control access to objects: public-read-write, public-read, private, and default. This topic describes how to manage the ACLs of objects in a versioned bucket.

Object ACLs

The following table describes the four ACLs of objects.

ACL	Description
public-read-write	Any users, including anonymous users can read and write the object.
public-read	The object is a public read resource. Only the owner or authorized users of the object can write the object. Other users, including anonymous users can only read the object.
private	The object is a private resource. Only the owner or authorized users of the object can read and write the object.
default	The ACL of an object is the same as that of the bucket in which the object is stored.

The ACL of an object takes precedence over the ACL of the bucket in which the object is stored. For example, if the ACL of a bucket is public read and the ACL of an object in the bucket is private, only the owner or authorized users of the object can read and write the object.

Configure object ACLs

By default, the `putObjectAcl` method is used to configure the ACL of the current version of an object. If the current version of the object is a delete marker, OSS returns 404 Not Found. You can specify a version ID in the request to configure the ACL of the specified version of an object.

The following code provides an example on how to configure the ACL of an object:

```

<? php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to lo
g on to OSS because the account has permissions on all API operations. We recommend that yo
u use your RAM user's credentials to call API operations or perform routine operations and
maintenance. To create a RAM user, log on to the RAM console.
$accessKeyId = "<yourAccessKeyId>";
$accessKeySecret = "<yourAccessKeySecret>";
// The endpoint of the China (Hangzhou) region is used in this example. Specify the actual
endpoint.
$endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
// Specify the bucket name.
$bucket= "<yourBucketName>";
// Specify the complete name of the object excluding the bucket name. Example: example/test
.txt.
$object = "<yourObjectName>";
// Specify the ID of the version for which you want to configure ACL.
$versionId = "<yourObjectVersionId>";
$ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint);
try {
    // Set the ACL of the specified version of the object to public-read.
    $ossClient->putObjectAcl($bucket, $object, 'public-read', array(OssClient::OSS_VERSION_
ID => $versionId));
} catch (OssException $e) {
    printf(__FUNCTION__ . ": FAILED\n");
    printf($e->getMessage() . "\n");
    return;
}
print(__FUNCTION__ . ": OK" . "\n");

```

For more information about how to configure object ACLs, see [PutObjectACL](#).

Query object ACLs

By default, the `getObjectAcl` method is used to query the ACL of the current version of an object. If the current version of the object is a delete marker, OSS returns 404 Not Found. You can specify a version ID in the request to query the ACL of the specified version of an object.

The following code provides an example on how to query the ACL of an object:

```

<? php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to lo
g on to OSS because the account has permissions on all API operations. We recommend that yo
u use your RAM user's credentials to call API operations or perform routine operations and
maintenance. To create a RAM user, log on to the RAM console.
$accessKeyId = "<yourAccessKeyId>";
$accessKeySecret = "<yourAccessKeySecret>";
// The endpoint of the China (Hangzhou) region is used in this example. Specify the actual
endpoint.
$endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
// Specify the bucket name.
$bucket= "<yourBucketName>";
// Specify the complete name of the object excluding the bucket name. Example: example/test
.txt.
$object = "<yourObjectName>";
// Specify the ID of the version that you want to query.
$versionId = "<yourObjectVersionId>";
$ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint);
try {
    // Query the ACL of the object.
    $acl = $ossClient->getObjectAcl($bucket, $object, array(OssClient::OSS_VERSION_ID => $v
ersionId));
    printf($acl . "\n");
} catch (OssException $e) {
    printf(__FUNCTION__ . ": FAILED\n");
    printf($e->getMessage() . "\n");
    return;
}
print(__FUNCTION__ . ": OK" . "\n");

```

For more information about how to query object ACLs, see [GetObjectACL](#)

3.9.10. Manage symbolic links

A symbolic link works in a similar manner as the short cut in Windows. You can configure a symbolic link that points to a frequently accessed object for quicker object access. In a versioned bucket, different versions of symbolic links can point to different objects.

Create a symbolic link

By default, when you create a symbolic link in a versioned bucket and point it to an object, the symbolic link points to the current version of the object. Symbolic links cannot be created for delete markers in versioned buckets.

The following code provides an example on how to create a symbolic link in a versioned bucket:

```

<? php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to lo
g on to OSS because the account has permissions on all API operations. We recommend that yo
u use your RAM user's credentials to call API operations or perform routine operations and
maintenance. To create a RAM user, log on to the RAM console.
$accessKeyId = "<yourAccessKeyId>";
$accessKeySecret = "<yourAccessKeySecret>";
// The endpoint of the China (Hangzhou) region is used in this example. Specify the actual
endpoint.
$endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
// Specify the bucket name.
$bucket= "<yourBucketName>";
// Specify the complete name of the object excluding the bucket name. Example: example/test
.txt.
$object = "<yourObjectName>";
// Specify the name of the symbolic link that you want to create.
$symLink = "<yourSymLink>";
$ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint);
try {
    // Create the symbolic link.
    $sym = $ossClient->putSymlink($bucket, $symlink, $object);
} catch (OssException $e) {
    printf(__FUNCTION__ . ": FAILED\n");
    printf($e->getMessage() . "\n");
    return;
}
print(__FUNCTION__ . ": OK" . "\n");

```

For more information about how to create a symbolic link, see [PutSymlink](#).

Query a symbolic link

By default, the `getSymlink` method is used to query the current version of a symbolic link. You can specify a version ID in the request to query the specified version of an symbolic link. If the current version of the queried symbolic link is a delete marker, OSS returns `404 Not Found` and includes the `x-oss-delete-marker = true` header and the version ID specified by the `x-oss-version-id` request header in the response.

The following code provides an example on how to query a symbolic link:

```

<? php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to lo
g on to OSS because the account has permissions on all API operations. We recommend that yo
u use your RAM user's credentials to call API operations or perform routine operations and
maintenance. To create a RAM user, log on to the RAM console.
$accessKeyId = "<yourAccessKeyId>";
$accessKeySecret = "<yourAccessKeySecret>";
// The endpoint of the China (Hangzhou) region is used in this example. Specify the actual
endpoint.
$endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
// Specify the bucket name.
$bucket= "<yourBucketName>";
// Specify the complete name of the object excluding the bucket name. Example: example/test
.txt.
$object = "<yourObjectName>";
// Specify the version ID of the version that you want to query.
$versionId = "<yourObjectVersionId>";
$ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint);
try {
    // Query the specified version of the symbolic link.
    $sym = $ossClient->getSymlink($bucket, $symlink, array(OssClient::OSS_VERSION_ID => $ve
rsionId));
    printf($sym['x-oss-version-id'] . "\n");
    printf($sym[OssClient::OSS_SYMLINK_TARGET] . "\n");
} catch (OssException $e) {
    printf(__FUNCTION__ . ": FAILED\n");
    printf($e->getMessage() . "\n");
    return;
}
print(__FUNCTION__ . ": OK" . "\n");

```

For more information about how to query a symbolic link, see [GetSymlink](#).

3.10. Object tagging

3.10.1. Configure object tagging

Object Storage Service (OSS) allows you to configure object tags to classify objects. Tags allow you to configure lifecycle rules and access control lists (ACLs) for objects that have the same tag.

Background information

When you configure object tagging, take note of the following items:

- You can add tags to an object when and after you upload the object. If the object already has tags,

the existing tags are overwritten. For more information about object tags, see [Put Object Tagging](#).

- To add tags to an object, you must have permissions to call the PutObjectTagging operation.

You can create a custom policy by using scripts to grant the permissions to RAM users. For more information, see [Attach a custom policy to a RAM user](#).

- The Last-Modified value of an object is not updated when the tags are changed.
- You can add up to 10 tags to an object. The tags added to an object must have unique keys.
- A tag key can be up to 128 characters in length. A tag value can be up to 256 characters in length.
- Tag keys and tag values are case-sensitive.
- The key and the value of a tag can contain letters, digits, spaces, and the following special characters:

+ - = . _ : /

 **Note** If you configure the tags in the HTTP header and the tags contain special characters, you can encode the keys and the values of the tags by using URL encoding.

Object tagging uses a key-value pair to identify objects. For more information about object tags, see [Object tagging](#).

Add tags to an object when you upload the object

The following examples describe how to add tags to an object when you upload the object by using simple upload, multipart upload, and append upload.

- Add tags to an object when you upload the object by using simple upload

The following code provides an example on how to add tags to an object when you upload the object by using simple upload:


```

<?php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to
access OSS because the account has permissions on all API operations. We recommend that y
ou use a RAM user to call API operations or perform routine O&M. To create a RAM user, lo
g on to the RAM console.
$accessKeyId = "yourAccessKeyId";
$accessKeySecret = "yourAccessKeySecret";
// Set yourEndpoint to the endpoint of the region in which the bucket is located. For exa
mple, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https:
//oss-cn-hangzhou.aliyuncs.com.
$endpoint = "yourEndpoint";
// Specify the name of the bucket. Example: examplebucket.
$bucket = "examplebucket";
// Specify the full path of the object. The full path of the object cannot contain the bu
cket name. Example: exampledir/exampleobject.txt.
$object = "exampledir/exampleobject.txt";
// Specify the string to upload.
$content = "hello world";
$ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint, false);
// Add tags to the object.
$options = array(
    OssClient::OSS_HEADERS => array(
        'x-oss-tagging' => 'key1=value1&key2=value2&key3=value3',
    ));
try {
    // Upload the object by using simple upload.
    $ossClient->putObject($bucket, $object, $content, $options);
} catch (OssException $e) {
    printf(__FUNCTION__ . ": FAILED\n");
    printf($e->getMessage() . "\n");
    return;
}
print(__FUNCTION__ . ": OK" . "\n");

```

- Add tags to an object when you upload the object by using multipart upload

The following code provides an example on how to add tags to an object when you upload the object by using multipart upload:

```

<?php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to
access OSS because the account has permissions on all API operations. We recommend that y
ou use a RAM user to call API operations or perform routine O&M. To create a RAM user, lo
g on to the RAM console.
$accessKeyId = "yourAccessKeyId";
$accessKeySecret = "yourAccessKeySecret";
// Set yourEndpoint to the endpoint of the region in which the bucket is located. For exa
mple, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https:
//oss-cn-hangzhou.aliyuncs.com.
$endpoint = "yourEndpoint";
// Specify the name of the bucket. Example: examplebucket.
$bucket = "examplebucket";
// Specify the full path of the object. The full path of the object cannot contain the bu
cket name. Example: exampledir/exampleobject.txt.
$object = "exampledir/exampleobject.txt";
// Specify the full path of the local file to upload. Example: D:\\localpath\\examplefile
.txt. By default, if you do not specify the local path of the file and specify only the n
ame of the file such as examplefile.txt, the local file is uploaded from the path of the
project to which the sample program belongs.
$file = "D:\\localpath\\examplefile.txt";
$ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint, false);
// Add tags to the object.
$options = array(
    OssClient::OSS_CHECK_MD5 => true,
    OssClient::OSS_PART_SIZE => 1,
    OssClient::OSS_HEADERS => array(
        'x-oss-tagging' => 'key1=value1&key2=value2&key3=value3',
    ),
);
try {
    // Upload the object by using multipart upload.
    $ossClient->multiuploadFile($bucket, $object, $file, $options);
} catch (OssException $e) {
    printf(__FUNCTION__ . ": FAILED\n");
    printf($e->getMessage() . "\n");
    return;
}
print(__FUNCTION__ . ": OK" . "\n");

```

- Add tags to an object when you upload the object by using append upload

The following code provides an example on how to add tags to an object when you upload the object by using append upload:

```

<?php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to
access OSS because the account has permissions on all API operations. We recommend that y
ou use a RAM user to call API operations or perform routine O&M. To create a RAM user, lo
g on to the RAM console.
$accessKeyId = "yourAccessKeyId";
$accessKeySecret = "yourAccessKeySecret";
// Set yourEndpoint to the endpoint of the region in which the bucket is located. For exa
mple, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https:
//oss-cn-hangzhou.aliyuncs.com.
$endpoint = "yourEndpoint";
// Specify the name of the bucket. Example: examplebucket.
$bucket = "examplebucket";
// Specify the full path of the object. The full path of the object cannot contain the bu
cket name. Example: exampledir/exampleobject.txt.
$object = "exampledir/exampleobject.txt";
// Specify the strings to upload in sequence.
$content_array = array('Hello OSS', 'Hi OSS');
$ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint, false);
// Add tags to the object.
$options = array(
    OssClient::OSS_HEADERS => array(
        'x-oss-tagging' => 'key1=value1&key2=value2',
    ));
try {
    // Upload the object by using append upload.
    $position = $ossClient->appendObject($bucket, $object, $content_array[0], 0, $options
);
    $position = $ossClient->appendObject($bucket, $object, $content_array[1], $position);
} catch (OssException $e) {
    printf(__FUNCTION__ . ": FAILED\n");
    printf($e->getMessage() . "\n");
    return;
}
print(__FUNCTION__ . ": OK" . "\n");

```

Add tags to or modify the tags of an existing object

If an existing object has no tags or the added tags of the object do not meet your requirements, you can add tags to or modify the tags of the existing object.

The following code provides an example on how to add tags to or modify the tags of an existing object:

```

<?php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
use OSS\Model\TaggingConfig;
use OSS\Model\Tag;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
$accessKeyId = "yourAccessKeyId";
$accessKeySecret = "yourAccessKeySecret";
// Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com.
$endpoint = "yourEndpoint";
// Specify the name of the bucket. Example: examplebucket.
$bucket = "examplebucket";
// Specify the full path of the object. The full path of the object cannot contain the bucket name. Example: exampledir/exampleobject.txt.
$object = "exampledir/exampleobject.txt";
$ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint, false);
// Add tags to the object.
$config = new TaggingConfig();
$config->addTag(new Tag("key1", "value1"));
$config->addTag(new Tag("key2", "value2"));
try {
    $ossClient->putObjectTagging($bucket, $object, $config);
} catch (OssException $e) {
    printf(__FUNCTION__ . ": FAILED\n");
    printf($e->getMessage() . "\n");
    return;
}
print(__FUNCTION__ . ": OK" . "\n");

```

Add tags to or modify the tags of a specified version of an object

If versioning is enabled for the bucket, you can add tags to a specified version of an object or modify the tags of the object by specifying the version ID of the object.

The following code provides an example on how to add tags to or modify the tags of a specified version of an object.

 **Note** For more information about how to obtain the version IDs of an object, see [List objects](#).

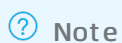
```

<?php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
use OSS\Model\TaggingConfig;
use OSS\Model\Tag;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
$accessKeyId = "yourAccessKeyId";
$accessKeySecret = "yourAccessKeySecret";
// Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com.
$endpoint = "yourEndpoint";
// Specify the name of the bucket. Example: examplebucket.
$bucket = "examplebucket";
// Specify the full path of the object. The full path of the object cannot contain the bucket name. Example: exampledir/exampleobject.txt.
$object = "exampledir/exampleobject.txt";
// Specify the version ID of the source object.
$options = array(
    'versionId'=>'CAEQMxiBgICAof2D0BYiIDJhMGE3N2MlYTl1NDQzOGY5NTkyNTI3MGYyMzMm****'
);
$ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint, false);
// Add tags to the object.
$config = new TaggingConfig();
$config->addTag(new Tag("key1", "value1"));
$config->addTag(new Tag("key2", "value2"));
try {
    $ossClient->putObjectTagging($bucket, $object, $config,$options);
    printf("putObjectTagging Success" . "\n");
} catch (OssException $e) {
    printf($e->getMessage() . "\n");
    return;
}

```

3.10.2. Query the tags of an object

After you configure tags for an object, you can query the tags of the object. If versioning is enabled for a bucket that stores the object whose tags you want to query, Object Storage Service (OSS) returns the tags of the current version of the object by default. To query the tags of a specified version of the object, you must specify the version ID of the object.



Note

Queries the tags of an object

If versioning is not enabled for the bucket that stores the object whose tags you want to query, you can query the tags of the object based on your requirements. If versioning is enabled for the bucket that stores the object whose tags you want to query, OSS returns the tags of the current version of the object by default.

The following code provides an example on how to query the tags of the `exampleobject.txt` object in the `exampledir` directory of the `examplebucket` bucket:

```
<?php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a Resource Access Management (RAM) user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
$accessKeyId = "yourAccessKeyId";
$accessKeySecret = "yourAccessKeySecret";
// Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com.
$endpoint = "yourEndpoint";
// Specify the name of the bucket. Example: examplebucket.
$bucket = "examplebucket";
// Specify the full path of the object. Example: exampledir/exampleobject.txt. The full path of the object cannot contain the bucket name.
$object = "exampledir/exampleobject.txt";
$ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint, false);
try {
    // Query the tags of the object.
    $config = $ossClient->getObjectTagging($bucket, $object);
    printf($object." tags are:". $config->serializeToXml(). "\n");
} catch (OssException $e) {
    printf($e->getMessage() . "\n");
    return;
}
```

Query the tags of a specified version of an object

If versioning is enabled for the bucket that stores the object whose tags you want to query, you can query the tags of a specified version of the object by specifying the version ID of the object.

The following code provides an example on how to query the tags of a specified version of the `exampleobject.txt` object in the `exampledir` directory of the `examplebucket` bucket:

 **Note** For more information about how to obtain the version IDs of an object, see [List objects](#).

```

<?php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
$accessKeyId = "yourAccessKeyId";
$accessKeySecret = "yourAccessKeySecret";
// Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com.
$endpoint = "yourEndpoint";
// Specify the name of the bucket. Example: examplebucket.
$bucket = "examplebucket";
// Specify the full path of the object. Example: exampledir/exampleobject.txt. The full path of the object cannot contain the bucket name.
$object = "exampledir/exampleobject.txt";
// Specify the version ID of the object.
$options = array(
    'versionId'=>'CAEQMxiBgICAof2D0BYiIDJhMGE3N2M1YTl1NDQzOGY5NTkyNTI3MGYyMzJm****'
);
$ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint, false);
try {
    // Query the tags of the specified version of the object.
    $config = $ossClient->getObjectTagging($bucket, $object,$options);
    printf($object." tags are:". $config->serializeToXml(). "\n");
} catch (OssException $e) {
    printf($e->getMessage() . "\n");
    return;
}

```

3.10.3. Remove the tags of an object

You can delete unnecessary tags of an object based on your requirements. If versioning is enabled for a bucket that stores the object whose tags you want to delete, Object Storage Service (OSS) deletes the tags of the current version of the object by default. To delete the tags of a specified version of the object, you must specify the version ID of the object.

Note

Remove the tags of an object

If versioning is not enabled for the bucket that stores the object whose tags you want to delete, you can delete the tags of the object based on your requirements. If versioning is enabled for the bucket that stores the object whose tags you want to delete, OSS deletes the tags of the current version of the object by default.

The following code provides an example on how to delete the tags of the `exampleobject.txt` object in the `exampledir` directory of the `examplebucket` bucket:

```
<?php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access Object Storage Service (OSS) because the account has permissions on all API operations. We recommend that you use a Resource Access Management (RAM) user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
$accessKeyId = "yourAccessKeyId";
$accessKeySecret = "yourAccessKeySecret";
// Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com.
$endpoint = "yourEndpoint";
// Specify the name of the bucket. Example: examplebucket.
$bucket = "examplebucket";
// Specify the full path of the object. Example: exampledir/exampleobject.txt. The full path of the object cannot contain the bucket name.
$object = "exampledir/exampleobject.txt";
$ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint, false);
try {
    // Remove the tags of the object.
    $ossClient->deleteObjectTagging($bucket, $object);
    printf("deleteObjectTagging Success" . "\n");
} catch (OssException $e) {
    printf($e->getMessage() . "\n");
    return;
}
```

Remove the tags of a specified version of an object

If versioning is enabled for the bucket that stores the object whose tags you want to delete, you can delete the tags of a specified version of the object by specifying the version ID of the object.

The following code provides an example on how to delete the tags of a specified version of the `exampleobject.txt` object in the `exampledir` directory of the `examplebucket` bucket:

 **Note** For more information about how to obtain the version IDs of an object, see [List objects](#).


```

<?php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
$accessKeyId = "yourAccessKeyId";
$accessKeySecret = "yourAccessKeySecret";
// Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com.
$endpoint = "yourEndpoint";
// Specify the name of the bucket. Example: examplebucket.
$bucket = "examplebucket";
// Specify the full path of the object. Example: exampledir/exampleobject.txt. The full path of the object cannot contain the bucket name.
$object = "exampledir/exampleobject.txt";
// Specify the version ID of the object. Example: CAEQMxiBgICAof2D0BYiIDJhMGE3N2M1YTI1NDQzOGY5NTkyNTI3MGYyMzJm****.
$options = array(
    'versionId'=>'CAEQMxiBgICAof2D0BYiIDJhMGE3N2M1YTI1NDQzOGY5NTkyNTI3MGYyMzJm****'
);
$ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint, false);
try {
    // Remove the tags of the object.
    $ossClient->deleteObjectTagging($bucket, $object,$options);
    printf("deleteObjectTagging Success" . "\n");
} catch (OssException $e) {
    printf($e->getMessage() . "\n");
    return;
}

```

3.11. Single-connection bandwidth throttling

This topic describes how to add parameters in an object upload and download request to set the limit of upload or download bandwidth. This way, you can ensure other applications have sufficient bandwidth.

Configure single-connection bandwidth throttling for simple upload and download

The following code provides an example on how to configure single-connection bandwidth throttling for simple upload and download:

```
<?php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
$accessKeyId = "yourAccessKeyId";
$accessKeySecret = "yourAccessKeySecret";
// Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com.
$endpoint = "yourEndpoint";
// Specify the name of the bucket. Example: examplebucket.
$bucket = "examplebucket";
// Specify the full path of the object. Example: exampledir/exampleobject.txt. The full path of the object cannot contain the bucket name.
$object = "exampledir/exampleobject.txt";
// Specify the content of the object.
$content = "hello world";
$ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint, false);
// Set the bandwidth limit to 100 KB/s, which is equal to 819,200 bit/s.
$options = array(
    OssClient::OSS_HEADERS => array(
        OssClient::OSS_TRAFFIC_LIMIT => 819200,
    ));
try {
    // Configure single-connection bandwidth throttling for the object upload.
    $ossClient->putObject($bucket, $object, $content, $options);
    // Configure single-connection bandwidth throttling for the object download.
    $ossClient->getObject($bucket, $object, $options);
} catch (OssException $e) {
    printf(__FUNCTION__ . ": FAILED\n");
    printf($e->getMessage() . "\n");
    return;
}
print(__FUNCTION__ . ": OK" . "\n");
```

Configure single-connection bandwidth throttling for object uploads and downloads that use signed URLs

The following code provides an example on how to configure single-connection bandwidth throttling when you use signed URLs to upload or download objects:

```

<?php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
$accessKeyId = "yourAccessKeyId";
$accessKeySecret = "yourAccessKeySecret";
// Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com.
$endpoint = "yourEndpoint";
// Specify the name of the bucket. Example: examplebucket.
$bucket = "examplebucket";
// Specify the full path of the object. Example: exampledir/exampleobject.txt. The full path of the object cannot contain the bucket name.
$object = "exampledir/exampleobject.txt";
$ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint, false);
// Set the bandwidth limit to 100 KB/s, which is equal to 819200 bit/s.
$options = array(
    OssClient::OSS_TRAFFIC_LIMIT => 819200,
);
// Generate a signed URL used to upload an object with single-connection bandwidth throttling configured and set the validity period of the URL to 60 seconds.
$timeout = 60;
$signedUrl = $ossClient->signUrl($bucket, $object, $timeout, "PUT", $options);
print($signedUrl);
// Generate a signed URL used to download an object with single-connection bandwidth throttling configured and set the validity period of the URL to 120 seconds.
$timeout = 120;
$signedUrl = $ossClient->signUrl($bucket, $object, $timeout, "GET", $options);
print($signedUrl);

```

3.12. Data encryption

3.12.1. Server-side encryption

OSS supports server-side encryption for uploaded data. When you upload data to OSS, OSS encrypts the uploaded data and then persistently stores the encrypted data. When you download the encrypted data, OSS automatically decrypts the data, returns the original data to you, and declares in the header of the response that the data had been encrypted on the server.

Encryption methods

OSS provides the following server-side encryption methods:

- Server-side encryption by using KMS-managed keys (SSE-KMS)

When you upload an object, you can use a specified CMK ID or the default CMK managed by KMS to encrypt the object. This method is cost-effective because you do not need to send user data to the KMS server over networks for encryption and decryption.

You are charged for calling API operations when you use CMKs to encrypt or decrypt data. For more information about the fees, see [KMS pricing](#).

- Server-side encryption by using OSS-managed keys (SSE-OSS)

When you upload an object, OSS encrypts the object on the server side by using AES-256 keys managed by OSS. OSS server-side encryption uses AES-256 to encrypt objects by using different data keys. AES-256 uses master keys that are regularly rotated to encrypt data keys.

Note

- An object can be encrypted by only one server-side encryption method at a time.
- If you configure server-side encryption for a bucket, you can still configure the encryption method for a single object when you upload or copy it. In this case, the encryption method configured for the object takes precedence. For more information, see [Put Object](#).

For more information about server-side encryption, see [Server-side encryption](#).

Configure server-side encryption for a bucket

The following code provides an example on how to configure the default encryption method for a bucket. After the method is configured, all objects that are uploaded to the bucket without encryption methods configured are encrypted by using the default encryption method.

The following code provides an example on how to configure the default encryption method for a bucket:

```

<? php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
use OSS\Model\ServerSideEncryptionConfig;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to lo
g on to OSS because the account has permissions on all API operations. We recommend that yo
u use your RAM user's credentials to call API operations or perform routine operations and
maintenance. To create a RAM user, log on to the RAM console.
$accessKeyId = "<yourAccessKeyId>";
$accessKeySecret = "<yourAccessKeySecret>";
// The endpoint of the China (Hangzhou) region is used in this example. Specify the actual
endpoint.
$endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
$bucket= "<yourBucketName>";
$ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint, false);
try {
    // Set the default server-side encryption method of the bucket to SSE-OSS.
    $config = new ServerSideEncryptionConfig("AES256");
    $ossClient->putBucketEncryption($bucket, $config);
    // Set the default server-side encryption method of the bucket to KMS without specifyin
g a CMK ID.
    $config = new ServerSideEncryptionConfig("KMS");
    $ossClient->putBucketEncryption($bucket, $config);
    // Set the default server-side encryption method of the bucket to KMS and specify a CMK
ID.
    $config = new ServerSideEncryptionConfig("KMS", "your kms id");
    $ossClient->putBucketEncryption($bucket, $config);
} catch (OssException $e) {
    printf(__FUNCTION__ . ": FAILED\n");
    printf($e->getMessage() . "\n");
    return;
}
print(__FUNCTION__ . ": OK" . "\n");

```

For more information about how to configure server-side encryption, see [PutBucketEncryption](#).

Query the server-side encryption configurations of a bucket

The following code provides an example on how to query the server-side encryption configurations of a bucket:

```
<? php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
use OSS\Model\ServerSideEncryptionConfig;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log on to OSS because the account has permissions on all API operations. We recommend that you use your RAM user's credentials to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
$accessKeyId = "<yourAccessKeyId>";
$accessKeySecret = "<yourAccessKeySecret>";
// The endpoint of the China (Hangzhou) region is used in this example. Specify the actual endpoint.
$endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
$bucket = "<yourBucketName>";
$ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint, false);
try {
    // Query the server-side encryption configurations of the bucket.
    $config = $ossClient->getBucketEncryption($bucket);
    // Display the server-side encryption configurations of the bucket.
    print($config->getSSEAlgorithm());
    print($config->getKMSTMasterKeyID());
} catch (OssException $e) {
    printf(__FUNCTION__ . ": FAILED\n");
    printf($e->getMessage() . "\n");
    return;
}
print(__FUNCTION__ . ": OK" . "\n");
```

For more information about how to query the server-side encryption configurations of a bucket, see [GetBucketEncryption](#).

Delete the server-side encryption configurations of a bucket

The following code provides an example on how to delete the server-side encryption configurations of a bucket:

```
<? php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log on to OSS because the account has permissions on all API operations. We recommend that you use your RAM user's credentials to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
$accessKeyId = "<yourAccessKeyId>";
$accessKeySecret = "<yourAccessKeySecret>";
// The endpoint of the China (Hangzhou) region is used in this example. Specify the actual endpoint.
$endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
$bucket = "<yourBucketName>";
$ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint, false);
try {
    // Delete the server-side encryption configurations of the bucket.
    $ossClient->deleteBucketEncryption($bucket);
} catch (OssException $e) {
    printf(__FUNCTION__ . ": FAILED\n");
    printf($e->getMessage() . "\n");
    return;
}
print(__FUNCTION__ . ": OK" . "\n");
```

For more information about how to delete the server-side encryption configurations of a bucket, see [DeleteBucketEncryption](#).

3.13. Authorized access

This topic describes how to authorize temporary access to Object Storage Service (OSS) by using Security Token Service (STS) or a signed URL.

 **Notice** A validity period must be set for an STS temporary access credential and a signed URL. When you use an STS temporary account credential to generate a signed URL that is used to perform operations such as object upload and download, the minimum validity period takes precedence. For example, you can set the validity period of your STS temporary access credential to 1,200 seconds and the validity period of your signed URL generated by using this credential to 3,600 seconds. In this case, the signed URL cannot be used to upload objects after the STS temporary access credential expires, even if the signed URL is within its validity period.

Use STS for temporary access authorization

You can use Alibaba Cloud STS to authorize temporary access to OSS. STS is a web service that provides temporary access tokens for users. You can use STS to grant a set of temporary access credentials that have a custom validity period and custom permissions to a third-party application or a RAM user managed by you. For more information about STS, see [What is STS?](#)

STS provides the following benefits:

- You need only to generate an access token and send the access token to a third-party application, instead of exposing your AccessKey pair to the third-party application. You can define the access permissions and validity period of this token.
- The token automatically expires after the validity period. Therefore, you do not need to manually revoke the access permissions of a token.

 **Note** For more information about how to use STS, see [Use a temporary credential provided by STS to access OSS](#) in OSS User Guide. You can call the [AssumeRole](#) operation or use STS SDKs for various programming languages to obtain temporary access credentials. For more information, see [STS SDK overview](#). The temporary access credentials consist of an AccessKey pair and a security token. The AccessKey pair contains an AccessKey ID and an AccessKey secret. The minimum validity period of temporary access credentials is 900 seconds. The maximum validity period of temporary access credentials is the maximum session duration specified for the current role. For more information, see [Specify the maximum session duration for a RAM role](#).

The following sample code provides an example on how to upload a string to an object by using a temporary access credential that is obtained from STS.

 **Note** For more information about how to obtain temporary access credentials from STS, see [STS SDK for PHP](#).


```

<?php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
// Specify the temporary AccessKey ID and AccessKey secret that are obtained from STS.
$accessKeyId = "yourAccessKeyId";
$accessKeySecret = "yourAccessKeySecret";
// Specify the security token that is obtained from STS.
$securityToken = "yourSecurityToken";
// Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if your bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com.
$endpoint = "yourEndpoint";
// Specify the bucket name.
$bucket = "examplebucket";
// Specify the full path of the object. The full path does not contain the bucket name.
$object = "exampleobject.txt";
// Specify the string that you want to upload.
$content = "Hello OSS";
try {
    $ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint, false, $securityToken);
    // Upload the string to the object by using the temporary access credential that is obtained from STS.
    $ossClient->putObject($bucket, $object, $content);
} catch (OssException $e) {
    print $e->getMessage();
}

```

Use a signed URL for temporary access authorization

You can generate a signed URL and provide the URL to a visitor for temporary access. When you generate a signed URL, you can specify the validity period of the URL to limit the period of access from visitors. By default, the validity period of a signed URL is 3,600 seconds. The maximum validity period of a signed URL is 32,400 seconds.

For more information about the complete code, visit [GitHub](#).

- Generate a signed URL to upload a string to an object

The following sample code provides an example on how to generate a signed URL for a Put Object request and use the signed URL to upload a string to an object:

```

<?php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}

```

```

use OSS\OssClient;
use OSS\Core\OssException;
use OSS\Http\RequestCore;
use OSS\Http\ResponseCore;
// Specify the temporary AccessKey ID and AccessKey secret that are obtained from STS.
$accessKeyId = "yourAccessKeyId";
$accessKeySecret = "yourAccessKeySecret";
// Specify the security token that is obtained from STS.
$securityToken = "yourSecurityToken";
// Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if your bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com.
$endpoint = "yourEndpoint";
// Specify the bucket name.
$bucket = "examplebucket";
// Specify the full path of the object. The full path does not contain the bucket name.
$object = "exampleobject.txt";
// Set the validity period of the URL to 3,600 seconds.
$timeout = 3600;
try {
    $ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint, false, $securityToken);
    // Generate a signed URL.
    $signedUrl = $ossClient->signUrl($bucket, $object, $timeout, "PUT");
} catch (OssException $e) {
    printf(__FUNCTION__ . ": FAILED\n");
    printf($e->getMessage() . "\n");
    return;
}
print(__FUNCTION__ . ": signedUrl: " . $signedUrl . "\n");
// Use the signed URL to upload a string to the object.
// Specify the string that you want to upload.
$content = "Hello OSS";
$request = new RequestCore($signedUrl);
// Set the method to access the signed URL to PUT.
$request->set_method('PUT');
$request->add_header('Content-Type', '');
$request->add_header('Content-Length', strlen($content));
$request->set_body($content);
$request->send_request();
$res = new ResponseCore($request->get_response_header(),
    $request->get_response_body(), $request->get_response_code());
if ($res->isOk()) {
    print(__FUNCTION__ . ": OK" . "\n");
} else {
    print(__FUNCTION__ . ": FAILED" . "\n");
};

```

- Generate a signed URL to download an object

The following sample code provides an example on how to generate a signed URL for a Get Object request and use the signed URL to download an object:

```

<?php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
use OSS\Core\OssException;
use OSS\Http\RequestCore;
use OSS\Http\ResponseCore;
// Specify the temporary AccessKey ID and AccessKey secret that are obtained from STS.
$accessKeyId = "yourAccessKeyId";
$accessKeySecret = "yourAccessKeySecret";
// Specify the security token that is obtained from STS.
$securityToken = "yourSecurityToken";
// Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if your bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com.
$endpoint = "yourEndpoint";
// Specify the bucket name.
$bucket = "examplebucket";
// Specify the full path of the object. The full path does not contain the bucket name.
$object = "exampleobject.txt";
// Set the validity period of the URL to 3,600 seconds.
$timeout = 3600;
try {
    $ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint, false, $securityToken);
    // Generate the signed URL for the GetObject request.
    $signedUrl = $ossClient->signUrl($bucket, $object, $timeout);
} catch (OssException $e) {
    printf(__FUNCTION__ . ": FAILED\n");
    printf($e->getMessage() . "\n");
    return;
}
print(__FUNCTION__ . ": signedUrl: " . $signedUrl . "\n");
// You can run code to access the signed URL, or enter the URL in the address bar of a browser to access the signed URL.
$request = new RequestCore($signedUrl);
// Set the default method to access the signed URL to GET.
$request->set_method('GET');
$request->add_header('Content-Type', '');
$request->send_request();
$res = new ResponseCore($request->get_response_header(), $request->get_response_body(), $request->get_response_code());
if ($res->isOk()) {
    print(__FUNCTION__ . ": OK" . "\n");
} else {
    print(__FUNCTION__ . ": FAILED" . "\n");
};

```

3.14. IMG

Image Processing (IMG) provided by Object Storage Service (OSS) is a secure, cost-effective, and highly reliable image processing service that can process large numbers of images. After you upload source images to OSS, you can call RESTful API operations to process the images on a device that is connected over the Internet anytime, anywhere.

 **Note** For more information about IMG parameters, see [Parameters](#). For the complete IMG code, visit [GitHub](#).

Use IMG parameters to process images

- The following code provides an example on how to use a single IMG parameter to process an image and save the image to your local computer:

```

<?php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to
access OSS because the account has permissions on all API operations. We recommend that y
ou use a Resource Access Management (RAM) user to call API operations or perform routine
O&M. To create a RAM user, log on to the RAM console.
$accessKeyId = "yourAccessKeyId";
$accessKeySecret = "yourAccessKeySecret";
// Set yourEndpoint to the endpoint of the region in which the bucket is located. For exa
mple, if your bucket is located in the China (Hangzhou) region, set yourEndpoint to https
://oss-cn-hangzhou.aliyuncs.com.
$endpoint = "yourEndpoint";
// Specify the name of the bucket. Example: examplebucket.
$bucket = "examplebucket";
// Specify the full path of the object. Example: exampledir/exampleobject.jpg. The full p
ath of the object cannot contain the bucket name.
$object = "exampledir/exampleobject.jpg";
// Specify the full local path to which you want to save the image. Example: D:\\localpat
h\\example-resize.jpg. If the specified local file exists, it is overwritten by the proces
sed image. Otherwise, the local file is created.
// By default, if you set this parameter to the name of a local file such as example-resi
ze.jpg without specifying the local path of the file, the processed object is saved to th
e local path of the project to which the sample program belongs.
$download_file = "D:\\localpath\\example-resize.jpg";
$ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint);
// If the image does not exist in the required bucket, you must upload the image to the r
equired bucket.
// $ossClient->uploadFile($bucket, $object, "D:\\localpath\\exampleobject.jpg");
// Resize the image to the height and width of 100 pixels.
$options = array(
    OssClient::OSS_FILE_DOWNLOAD => $download_file,
    OssClient::OSS_PROCESS => "image/resize,m_fixed,h_100,w_100" );
// Name the processed image example-resize.jpg and save the image to your local computer.

$ossClient->getObject($bucket, $object, $options);
// After the image is processed, you can delete the source image from the bucket if you n
o longer need the image.
// $ossClient->deleteObject($bucket, $object);

```

- Use multiple IMG parameters to process an image and save the image to your local computer

The following code provides an example on how to use multiple IMG parameters to process an image. Separate these parameters with forward slashes (/).

```

<?php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to
access OSS because the account has permissions on all API operations. We recommend that y
ou use a RAM user to call API operations or perform routine O&M. To create a RAM user, lo
g on to the RAM console.
$accessKeyId = "yourAccessKeyId";
$accessKeySecret = "yourAccessKeySecret";
// Set yourEndpoint to the endpoint of the region in which the bucket is located. For exa
mple, if your bucket is located in the China (Hangzhou) region, set yourEndpoint to https
://oss-cn-hangzhou.aliyuncs.com.
$endpoint = "yourEndpoint";
// Specify the name of the bucket. Example: examplebucket.
$bucket = "examplebucket";
// Specify the full path of the object. Example: exampledir/exampleobject.jpg. The full p
ath of the object cannot contain the bucket name.
$object = "exampledir/exampleobject.jpg";
// Specify the local path to which you want to save the image. Example: D:\\localpath\\ex
ample-new.jpg. If the specified local file exists, it is overwritten by the processed ima
ge. Otherwise, the local file is created.
// By default, if you set this parameter to the name of a local file such as example-new.
jpg without specifying the local path of the file, the processed object is saved to the l
ocal path of the project to which the sample program belongs.
$download_file = "D:\\localpath\\example-new.jpg";
$ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint, false);
// If the image does not exist in the required bucket, you must upload the image to the r
equired bucket.
// $ossClient->uploadFile($bucket, $object, "D:\\localpath\\exampleobject.jpg");
// After you resize the image to the height and width of 100 pixels, rotate the image 90
degrees.
$style = "image/resize,m_fixed,w_100,h_100/rotate,90";
$options = array(
    OssClient::OSS_FILE_DOWNLOAD => $download_file,
    OssClient::OSS_PROCESS => $style);
// Name the processed image example-new.jpg and save the image to your local computer.
$ossClient->getObject($bucket, $object, $options);
// After the image is processed, you can delete the source image if you no longer use the
image from the bucket.
// $ossClient->deleteObject($bucket, $object);

```

Use image style to process images

You can create an image style in the OSS console and encapsulate multiple IMG parameters in the style. Then, you can use the style to process images. For more information, see [Image styles](#).

The following code provides an example on how to use an image style to process an image:

```

<?php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
$accessKeyId = "yourAccessKeyId";
$accessKeySecret = "yourAccessKeySecret";
// Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if your bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com.
$endpoint = "yourEndpoint";
// Specify the name of the bucket. Example: examplebucket.
$bucket = "examplebucket";
// Specify the full path of the object. Example: exampledir/exampleobject.jpg. The full path of the object cannot contain the bucket name.
$object = "exampledir/exampleobject.jpg";
// Specify the local path to which you want to save the image. Example: D:\\localpath\\example-new.jpg. If the specified local file exists, it is overwritten by the processed image. Otherwise, the local file is created.
// By default, if you set this parameter to the name of a local file such as example-new.jpg without specifying the local path of the file, the processed object is saved to the local path of the project to which the sample program belongs.
$download_file = "D:\\localpath\\example-new.jpg";
$ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint);
// If the image does not exist in the required bucket, you must upload the image to the required bucket.
// $ossClient->uploadFile($bucket, $object, "D:\\localpath\\exampleobject.jpg");
// Use the custom style to process the image.
// Set yourCustomStyleName to the name of the image style you create in the OSS console.
$style = "style/yourCustomStyleName";
$options = array(
    OssClient::OSS_FILE_DOWNLOAD => $download_file,
    OssClient::OSS_PROCESS => $style);
// Name the processed image example-new.jpg and save the image to your local computer.
$ossClient->getObject($bucket, $object, $options);
// After the image is processed, you can delete the source image if you no longer use the image from the bucket.
// $ossClient->deleteObject($bucket, $object);

```

Save processed images

By default, IMG does not save processed images. You can call the `ImgSaveAs` operation to save the images to the bucket in which the source images are stored.

The following code provides an example on how to save a processed image:

```

<?php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
$accessKeyId = "yourAccessKeyId";
$accessKeySecret = "yourAccessKeySecret";
// Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if your bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com.
$endpoint = "yourEndpoint";
// Specify the name of the bucket. Example: examplebucket.
$bucket = "examplebucket";
// Specify the full path of the source object. Example: exampledir/exampleobject.jpg. The full path of the object cannot contain the bucket name.
$object = "exampledir/exampleobject.jpg";
// Specify the full path that is used to store the processed image. Example: example-new.jpg.
$save_object = "example-new.jpg";
function base64url_encode($data)
{
    return rtrim(strtr(base64_encode($data), '+/', '-_'), '=');
}
$ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint, false);
// If the image does not exist in the required bucket, you must upload the image to the required bucket.
// $ossClient->uploadFile($bucket, $object, "D:\\localpath\\exampleobject.jpg");
// After you resize the image to the height and width of 100 pixels, rotate the image 90 degrees.
$style = "image/resize,m_fixed,w_100,h_100/rotate,90";
$process = $style.
    '|sys/saveas'.
    ',o_'.base64url_encode($save_object).
    ',b_'.base64url_encode($bucket);
// Name the processed image example-new.png and save the image to the current bucket.
$result = $ossClient->processObject($bucket, $object, $process);
// Display the processed result.
print($result);

```

Generate a signed object URL that includes IMG parameters

URLs of private objects must be signed. IMG parameters cannot be added to a signed URL. If you want to process a private object, add IMG parameters to the signature. The following code provides an example on how to add IMG parameters to the signature:


```
<?php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
$accessKeyId = "yourAccessKeyId";
$accessKeySecret = "yourAccessKeySecret";
// Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if your bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com.
$endpoint = "yourEndpoint";
// Specify the name of the bucket. Example: examplebucket.
$bucket = "examplebucket";
// Specify the full path of the object. Example: exampledir/exampleobject.jpg. The full path of the object cannot contain the bucket name.
$object = "exampledir/exampleobject.jpg";
$ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint);
// Generate a signed URL that contains the IMG parameters. The validity period of the URL is 3,600 seconds. The signed URL can be directly accessed by using a browser.
$timeout = 3600;
$options = array(
    // Resize the image to the height and width of 100 pixels.
    OssClient::OSS_PROCESS => "image/resize,m_fixed,h_100,w_100" );
$signedUrl = $ossClient->signUrl($bucket, $object, $timeout, "GET", $options);
print("rtmp url: \n" . $signedUrl);
```

The following code provides an example on how to generate a signed object URL that contains the watermark parameter:

```

<?php
if (is_file(__DIR__ . '/../autoload.php')) {
    require_once __DIR__ . '/../autoload.php';
}
if (is_file(__DIR__ . '/../vendor/autoload.php')) {
    require_once __DIR__ . '/../vendor/autoload.php';
}
use OSS\OssClient;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
$accessKeyId = "yourAccessKeyId";
$accessKeySecret = "yourAccessKeySecret";
// Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if your bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com.
$endpoint = "yourEndpoint";
// Specify the name of the bucket. Example: examplebucket.
$bucket = "examplebucket";
// Specify the full path of the object. Example: exampledir/exampleobject.jpg. The full path of the object cannot contain the bucket name.
$object = "exampledir/exampleobject.jpg";
$ossClient = new OssClient($accessKeyId, $accessKeySecret, $endpoint);
// Generate a signed URL that contains the watermark parameter. The validity period of the URL is 3,600 seconds. The signed URL can be directly accessed by using a browser.
$timeout = 3600;
function base64url_encode($data)
{
    return rtrim(strtr(base64_encode($data), '+/', '-_'), '=');
}
// Specify the content of the watermark such as Hello World or specify the full path of a watermark image such as panda.jpg.
// When you add a watermark image to the image, make sure that the watermark image is stored in the same bucket as the image.
$content = "Hello World";
$string = base64url_encode($content);
$options = array(
    // Add the watermark to the image to process.
    OssClient::OSS_PROCESS => "image/watermark,text_".$string
);
$signedUrl = $ossClient->signUrl($bucket, $object, $timeout, "GET", $options);
print("rtmp url: \n" . $signedUrl);

```

3.15. Exception handling

OSS PHP SDK exceptions (OssException) include various errors, such as invalid parameters and file missing. You can use the getMessage method to obtain error messages.

For more information about OssException, see [GitHub](#).

Example for handling exceptions

The following code shows how to handle exceptions when creating a existing bucket and prints error messages.

```
try {
    $ossClient->createBucket($bucket);
} catch (OssException $e) {
    print("Exception:" . $e->getMessage() . "\n");
}
```

You can also obtain the following information:

Parameter	Description
HTTPStatus	HTTP status code. You can get the status code using the getHTTPStatus method.
ErrorCode	Specifies the error code returned by OSS. You can get the error code using the getErrorCode method.
ErrorMessage	Specifies the error messages returned by OSS. You can get the error messages using the getErrorMessage method.
RequestId	Specifies the UUID. It is unique and used to identify a request. If a problem persists, send the RequestId to OSS developers for help. You can get the RequestId using the getRequestId method.
Details	Specifies detailed error messages returned by OSS. You can get the detailed error messages using the getDetails method.

Common OSS error codes

Error code	Description	HTTP status code
AccessDenied	Access denied	403
BucketAlreadyExists	The bucket already exists.	409
BucketNotEmpty	The bucket is not empty.	409
EntityTooLarge	The entity size exceeds the maximum limit.	400
EntityTooSmall	The entity size is below the minimum limit.	400
FileGroupTooLarge	The total file group size exceeds the maximum limit.	400
FilePartNotExist	No such part exists.	400

Error code	Description	HTTP status code
FilePartStale	The part has expired.	400
InvalidArgument	The parameter format is invalid.	400
InvalidAccessKeyId	No such AccessKeyId exists.	403
InvalidBucketName	The bucket name is invalid.	400
InvalidDigest	The digest is invalid.	400
InvalidObjectName	The object name is invalid.	400
InvalidPart	The part is invalid.	400
InvalidPartOrder	The part order is invalid.	400
InvalidTargetBucketForLogging	The buckets that store log files are invalid.	400
InternalError	Internal OSS error	500
MalformedXML	The XML format is invalid.	400
MethodNotAllowed	The method is not allowed.	405
MissingArgument	Parameters are not configured.	411
MissingContentLength	The content length is not configured.	411
NoSuchBucket	No such bucket exists.	404
NoSuchKey	No such object exists.	404
NoSuchUpload	No such uploadId exists.	404
NotImplemented	The methods are not implemented.	501
PreconditionFailed	Precondition error	412
RequestTimeTooSkewed	The local time set for OSSClient is deviated from the time set for the OSS server by over 15 minutes.	403
RequestTimeout	Request timeout	400
SignatureDoesNotMatch	Signature mismatch	403

Error code	Description	HTTP status code
InvalidEncryptionAlgorithmError	The specified encryption algorithm (the entropy encoding type) is invalid.	400

4. Node.js

4.1. Installation

This topic describes how to install Object Storage Service (OSS) SDK for Node.js.

Prerequisites

- Alibaba Cloud OSS is activated.
- The AccessKey ID and AccessKey secret of a RAM user are created.

Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log on to OSS because the account has permissions on all API operations. We recommend that you use the AccessKey pair of a RAM user. If you deploy your service on the server, you can use a RAM user or STS credentials to call API operations or perform routine operations. If you deploy your service on the client, use STS credentials to call API operations. For more information, see [Resource Access Management](#).

Background information

OSS SDK for Node.js is built based on the [Node.js](#) environment.

Demos in documents on the official website are based on OSS SDK for Node.js V6.x. If your OSS SDK for Node.js version is earlier than 6.x, see [5.x](#). To upgrade your OSS SDK for Node.js version to 6.x, see [Upgrading Notes](#).

For more information about the source code of OSS SDK for Node.js, see [GitHub](#). For more information, see [documents](#).

Install OSS SDK for Node.js

OSS supports Node.js 8.0.0 and later. To use OSS SDK for Node.js in Node.js versions earlier than 8.0.0, use OSS SDK for Node.js V4.x.

Run `npm install ali-oss --save` to install the SDK package. For more information, visit [npm](#).

If you encounter network issues when you use npm, we recommend that you use [cnpm](#), which is the npm image provided by Taobao.

Use OSS SDK for Node.js

You can use OSS SDK for Node.js in synchronous and asynchronous modes. `new OSS()` is used to create OSS clients in synchronous and asynchronous modes.

- Synchronous mode: The `async` and `await` methods are used to synchronize asynchronous operations.
- Asynchronous mode: Perform asynchronous operations in a way similar to callback. The API operation returns a [Promise](#) object for a request. The `.then()` method is used to process the returned result, and the `.catch()` method is used to handle errors.

The following code provides examples on how to upload an object in synchronous mode and download an object in asynchronous mode:

- Synchronous mode

```

let client = new OSS(...);
async function put () {
  try {
    // Set 'object' to the name of the object uploaded to OSS. The name cannot contain bucket names.
    // Set 'localfile' to the full path of the local file from which the object is uploaded to OSS.
    let r1 = await client.put('object','localfile');
    console.log('put success: %j', r1);
    let r2 = await client.get('object');
    console.log('get success: %j', r2);
  } catch(e) {
    console.error('error: %j', e);
  }
}
await put();

```

- Asynchronous mode

```

let client = new OSS(...);
// Set 'object' to the name of the object downloaded from OSS. The name is a full path that cannot contain bucket names.
// Set 'localfile' to the full path of the local file to which the object is downloaded.
client.put('object', 'localfile').then(function (r1) {
  console.log('put success: %j', r1);
  return client.get('object');
}).then(function (r2) {
  console.log('get success: %j', r2);
}).catch(function (err) {
  console.error('error: %j', e);
});

```

4.2. Initialization

This topic describes how to initialize Object Storage Service (OSS) SDK for Node.js.

Create an object named `app.js` and write the following content to the object:

```

let OSS = require('ali-oss');
let client = new OSS({
  // Set yourRegion to the endpoint of the region in which the bucket is located. For example, if your bucket is located in the China (Hangzhou) region, set yourRegion to oss-cn-hangzhou.
  region: 'yourRegion',
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a Resource Access Management (RAM) user to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
  accessKeyId: 'yourAccessKeyId',
  accessKeySecret: 'yourAccessKeySecret',
});

```

The `region` parameter indicates the region in which you activate OSS. Example: `oss-cn-hangzhou`. For the complete list of regions, see [Endpoints and regions](#).

-  **Note** You can also use the following parameters to specify an endpoint based on your actual scenario:
- To access internal nodes, specify `region` and `internal` and set `internal` to `true`.
 - To access OSS over HTTPS, specify `region` and `secure` and set `secure` to `true`.
 - To access OSS by using custom domain names, specify `cname` and `endpoint` and set `cname` to `true` and `endpoint` to the custom domain name that you mapped to the bucket where the required OSS data is stored.
 - If you set `endpoint` to a value such as `http://oss-cn-hangzhou.aliyuncs.com`, `region` is ignored. `endpoint` can be set to an IP address, and you can specify HTTPS.
 - If `bucket` is not configured, you must first call `useBucket` when you manage objects stored in the bucket. You need to call this operation only once.
 - To specify a timeout period for API calls to access OSS, use `timeout`. The default value of `timeout` is 60000. Unit: milliseconds.

4.3. Configuration items

This topic describes how to use configuration items when you initialize Object Storage Service (OSS) SDK for Node.js.

The following table describes the configuration items (options).

Configuration item	Type	Description
<code>accessKeyId</code>	String	The AccessKey ID that is created in the Alibaba Cloud Management Console.
<code>accessKeySecret</code>	String	The AccessKey secret that is created in the Alibaba Cloud Management Console.
<code>stsToken</code>	String	The temporary Security Token Service (STS) token used to access OSS. For more information, see Use STS for temporary access authorization .
<code>bucket</code>	String	The bucket that is created by using the OSS console or by calling the <code>PutBucket</code> operation.
<code>endpoint</code>	String	The endpoint of the OSS bucket.
<code>region</code>	String	The region where your bucket is located. Default value: <code>oss-cn-hangzhou</code> .

Configuration item	Type	Description
internal	Boolean	Specifies whether to use the Alibaba Cloud internal network to access OSS. Default value: false. For example, set this parameter to true if you use an Elastic Compute Service (ECS) instance to access OSS. The ECS instance accesses OSS by using an internal endpoint, which reduces costs.
cname	Boolean	Specifies whether CNAME can be used to access OSS. Default value: false. If you set this parameter to true, you must map a CNAME to your bucket before you use the CNAME to access the bucket.
isRequestPay	Boolean	Specifies whether the pay-by-requester mode is enabled for your bucket. Default value: false.
secure	Boolean	Specifies whether HTTPS is used to access OSS. A value of true indicates that HTTPS is used to access OSS. A value of false indicates that HTTP is used to access OSS. For more information, see FAQ .
timeout	String Number	The timeout period. Default value: 60000. Unit: milliseconds.

The following code provides an example on how to configure parameters when you use OSS SDK for Node.js:

```
var oss = require('ali-oss');
var client = new oss({
  // Set yourRegion to the endpoint of the region where your bucket is located. For example,
  // if your bucket is located in the China (Hangzhou) region, set yourRegion to oss-cn-hangzhou
  .
  region: 'yourRegion',
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to
  // access OSS because the account has permissions on all API operations. We recommend that you
  // use a Resource Access Management (RAM) user to call API operations or perform routine O&M.
  // To create a RAM user, log on to the RAM console.
  accessKeyId: 'yourAccessKeyId',
  accessKeySecret: 'yourAccessKeySecret',
  // Specify the bucket name.
  bucket: 'examplebucket'
});
```

4.4. Quick Start

This topic describes how to manage Object Storage Service (OSS) in the Node.js environment, such as view buckets and upload objects.

 **Note** To facilitate modification, a file named `app.js` is created in this topic. The following code for each operation is described in sync mode. For more information about how to create clients in the sample code, see [Initialization](#).

Query the list of buckets

Add the following content to the end of the `app.js` file and call `listBuckets` to query buckets:

```
async function listBuckets () {
  try {
    let result = await client.listBuckets();
    console.log(result)
  }
  catch(err) {
    console.log(err)
  }
}
listBuckets();
```

You can run the `node app.js` script and view the response.

For more information about bucket-related operations, visit [GitHub](#).

Query objects in a bucket

Modify the `app.js` file, and call `list` to query the objects in a bucket:

```
client.useBucket('examplebucket');
async function list () {
  try {
    let result = await client.list({
      'max-keys': 5
    })
    console.log(result)
  } catch (err) {
    console.log (err)
  }
}
list();
```

You can run the `node app.js` script and view the response.

Upload an object

Modify the `app.js` file and call `put` to upload a single object:

```
client.useBucket('examplebucket');
async function put () {
  try {
    let result = await client.put('exampleobject.txt', 'D:\\localpath\\examplefile.txt');
    console.log(result);
  } catch (err) {
    console.log (err);
  }
}
put();
```

Download an object

Modify the `app.js` file, and call `get` to download a single object:

```
client.useBucket('examplebucket');
async function get () {
  try {
    let result = await client.get('exampleobject.txt');
    console.log(result);
  } catch (err) {
    console.log (err);
  }
}
get();
```

Delete an object

Modify the `app.js` file and call `delete` to delete an object:

```
client.useBucket('examplebucket');
async function delete () {
  try {
    let result = await client.delete('exampleobject.txt');
    console.log(result);
  } catch (err) {
    console.log (err);
  }
}
delete();
```

For more information about object-related operations, visit [GitHub](#).

4.5. Buckets

4.5.1. Create a bucket

A bucket is a container for objects stored in Object Storage Service (OSS). Every object is contained in a bucket. This topic describes how to create a bucket.

Sample code

The following code provides an example on how to create a bucket named `examplebucket`:

```
const OSS = require('ali-oss');
const client = new OSS({
  // Set yourregion to the region in which the bucket is located. For example, if your bucket
  // is located in the China (Hangzhou) region, set yourregion to oss-cn-hangzhou.
  region: 'yourregion',
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to
  // access OSS because the account has permissions on all API operations. We recommend that you
  // use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on
  // to the RAM console.
  accessKeyId: 'yourAccessKeyId',
  accessKeySecret: 'yourAccessKeySecret'
});
// Create the bucket.
async function putBucket() {
  try {
    const options = {
      storageClass: 'Standard', // By default, the storage class of a bucket is Standard. To
      // set the storage class of the bucket to Archive, set storageClass to Archive.
      acl: 'private', // By default, the ACL of a bucket is private. To set the ACL of the
      // bucket to public read, set acl to public-read.
      dataRedundancyType: 'LRS' // By default, the data redundancy type of a bucket is locally
      // redundant storage (LRS). To set the data redundancy type of the bucket to zone-redundant
      // storage (ZRS), set dataRedundancyType to ZRS.
    };
    // Specify the bucket name.
    const result = await client.putBucket('examplebucket', options);
    console.log(result);
  } catch (err) {
    console.log(err);
  }
}
putBucket();
```

References

For more information about the API operation that you can call to create a bucket, see [PutBucket](#).

4.5.2. List buckets

A bucket is a container that stores objects. You can list all buckets or only buckets that meet specified conditions.

List all buckets

```
const OSS = require('ali-oss');
const client = new OSS({
  // The endpoint of the China (Hangzhou) region is used in this example. Specify the actual endpoint.
  region: 'yourRegion',
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log on to OSS because the account has permissions on all API operations. We recommend that you use your RAM user's credentials to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
  accessKeyId: 'youAccessKeyId',
  accessKeySecret: 'yourAccessKeySecret'
});
async function listBuckets() {
  try {
    const result = await client.listBuckets();
    console.log(result);
  } catch (err) {
    console.log(err);
  }
}
listBuckets();
```

List buckets whose names contain a specified prefix

```
const OSS = require('ali-oss');
const client = new OSS({
  // The endpoint of the China (Hangzhou) region is used in this example. Specify the actual endpoint.
  region: 'yourRegion',
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log on to OSS because the account has permissions on all API operations. We recommend that you use your RAM user's credentials to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
  accessKeyId: 'youAccessKeyId',
  accessKeySecret: 'yourAccessKeySecret'
});
async function listBuckets() {
  try {
    const result = await client.listBuckets({
      prefix: 'prefix' // Specify the prefix that is contained in the buckets to list.
    });
    console.log(result);
  } catch (err) {
    console.log(err);
  }
}
listBuckets();
```

List buckets that follow a specified marker

```
const OSS = require('ali-oss');
const client = new OSS({
  // The endpoint of the China (Hangzhou) region is used in this example. Specify the actual endpoint.
  region: 'yourRegion',
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log on to OSS because the account has permissions on all API operations. We recommend that you use your RAM user's credentials to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
  accessKeyId: 'yourAccessKeyId',
  accessKeySecret: 'yourAccessKeySecret'
});
async function listBuckets() {
  try {
    const result = await client.listBuckets({
      marker: 'marker' // List buckets that follow a specified marker.
    });
    console.log(result);
  } catch (err) {
    console.log(err);
  }
}
listBuckets();
```

List a specified number of buckets

```
const OSS = require('ali-oss');
const client = new OSS({
  // The endpoint of the China (Hangzhou) region is used in this example. Specify the actual endpoint.
  region: 'yourRegion',
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log on to OSS because the account has permissions on all API operations. We recommend that you use your RAM user's credentials to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
  accessKeyId: 'yourAccessKeyId',
  accessKeySecret: 'yourAccessKeySecret'
});
async function listBuckets() {
  try {
    const result = await client.listBuckets({
      'max-keys': 20 // Set the number of buckets to list each time to 20.
    });
    console.log(result);
  } catch (err) {
    console.log(err);
  }
}
listBuckets();
```

4.5.3. Determine whether a bucket exists

A bucket is a container used to store objects in Object Storage Service (OSS). Every object is contained in a bucket. This topic describes how to determine whether a bucket exists.

```
const OSS = require('ali-oss')
const client = new OSS({
  // The endpoint of the China (Hangzhou) region is used in this example. Specify the actual endpoint.
  region: '<Your region>'
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log on to OSS because the account has permissions on all API operations. We recommend that you use your RAM user's credentials to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
  accessKeyId: '<Your AccessKeyId>',
  accessKeySecret: '<Your AccessKeySecret>',
});
async function bucketisExist() {
  try {
    const result = await client.getBucketInfo('<Your bucket>')
    console.log('bucketInfo: ', result.bucket)
  } catch (error) {
    // The specified bucket does not exist.
    if (error.name === 'NoSuchBucketError') {
      console.log
    } else {
      console.log(error)
    }
  }
}
bucketisExist()
```

4.5.4. Query the regions of buckets

A bucket is a container for objects stored in Object Storage Service (OSS). Every object is contained in a bucket. This topic describes how to query the region of a bucket.

```
const OSS = require('ali-oss')
const client = new OSS({
  // The endpoint of the China (Hangzhou) region is used in this example. Specify the actual endpoint.
  region: '<Your region>'
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log on to OSS because the account has permissions on all API operations. We recommend that you use your RAM user's credentials to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
  accessKeyId: '<Your AccessKeyId>',
  accessKeySecret: '<Your AccessKeySecret>',
  bucket: '<Your bucket name>'
});
async function getLocation() {
  const result = await client.getBucketInfo
  console.log(result.bucket.Location)
}
getLocation()
```

4.5.5. Query bucket information

A bucket is a container for objects stored in Object Storage Service (OSS). Every object is contained in a bucket. This topic describes how to query the information about a bucket.

The following code provides an example on how to query the information about a bucket named `examplebucket`, which includes the region, storage class, and versioning status of the bucket:


```
const OSS = require('ali-oss');
const client = new OSS({
  // Set yourRegion to the endpoint of the region in which the bucket is located. For example, if your bucket is located in the China (Hangzhou) region, set yourRegion to oss-cn-hangzhou.
  region: 'yourRegion',
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access Object Storage Service (OSS) because the account has permissions on all API operations. We recommend that you use a Resource Access Management (RAM) user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
  accessKeyId: 'yourAccessKeyId',
  accessKeySecret: 'yourAccessKeySecret'
});
async function getBucketInfo() {
  // Specify the name of the bucket. Example: examplebucket.
  const bucket = 'examplebucket'
  const result = await client.getBucketInfo(bucket)
  // Query the region of the bucket.
  console.log(result.bucket.Location)
  // Query the name of the bucket.
  console.log(result.bucket.Name)
  // Query the ID of the bucket owner.
  console.log(result.bucket.Owner.ID)
  // Query the name of the bucket owner, which is currently the same as the ID of the bucket owner.
  console.log(result.bucket.Owner.DisplayName)
  // Query the creation time of the bucket.
  console.log(result.bucket.CreationDate)
  // Query the storage class of the bucket.
  console.log(result.bucket.StorageClass)
  // Query the versioning status of the bucket.
  console.log(result.bucket.Versioning)
}
getBucketInfo()
```

For more information about how to query bucket information, see [GetBucketInfo](#).

4.5.6. Manage bucket ACLs

You can configure the access control list (ACL) when you create a bucket, or modify the ACL for a created bucket. This topic describes how to set and query the ACL of a bucket.

Set the ACL of a bucket

The following table describes the ACL of a bucket.

ACL	Description	Value
-----	-------------	-------

ACL	Description	Value
Private	Only the owner or authorized users of this bucket can read and write objects in the bucket. Other users, including anonymous users cannot access the objects in the bucket without authorization.	private
Public read	Only the owner or authorized users of this bucket can write objects in the bucket. Other users, including anonymous users can only read objects from the bucket. We recommend that you exercise caution when you set the bucket ACL to public read.	public-read
Public read/write	Any users, including anonymous users can read and write objects in the bucket. We recommend that you exercise caution when you set the bucket ACL to public read/write.	public-read-write

- Set the ACL when you create a bucket

The following code provides an example on how to set the ACL when you create a bucket:

```
const OSS = require('ali-oss');

const client = new OSS({
  // The endpoint of the China (Hangzhou) region is used in this example. Specify the actual endpoint.
  region: '<Your region>',
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log on to OSS because the account has permissions on all API operations. We recommend that you use your RAM user's credentials to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
  accessKeyId: '<Your AccessKeyId>',
  accessKeySecret: '<Your AccessKeySecret>',
  bucket: '<Your bucket name>',
});
// The following code provides an example on how to set the ACL to public-read when you create a bucket.
async function putBucket() {
  const acl = 'public-read';
  try {
    await client.putBucket('<Your Bucket Name>', { acl });
  } catch (error) {
    console.log(error)
  }
}

putBucket()
```

- Modify the ACL after you create a bucket

The following code provides an example on how to modify the ACL after you create a bucket:

```
const OSS = require('ali-oss');

const client = new OSS({
  // The endpoint of the China (Hangzhou) region is used in this example. Specify the actual endpoint.
  region: '<Your region>',
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log on to OSS because the account has permissions on all API operations. We recommend that you use your RAM user's credentials to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
  accessKeyId: '<Your AccessKeyId>',
  accessKeySecret: '<Your AccessKeySecret>',
  bucket: '<Your bucket name>',
});

async function putBucketACL() {
  // The following code provides an example on how to change the ACL to private after you create a bucket.
  const acl = 'private'
  try {
    await client.putBucketACL('<Your Bucket Name>', acl)
  } catch (error) {
    console.log(error)
  }
}

putBucketACL()
```

Query the ACL of a bucket

The following code provides an example on how to query the ACL of a bucket:

```
const OSS = require('ali-oss');

const client = new OSS({
  // The endpoint of the China (Hangzhou) region is used in this example. Specify the actual endpoint.
  region: '<Your region>',
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log on to OSS because the account has permissions on all API operations. We recommend that you use your RAM user's credentials to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
  accessKeyId: '<Your AccessKeyId>',
  accessKeySecret: '<Your AccessKeySecret>',
  bucket: '<Your bucket name>',
});

// Query the ACL of the bucket.
async function getBucketAcl() {
  const result = await client.getBucketACL('<Your Bucket Name>')
  console.log('acl: ', result.acl)
}

getBucketAcl()
```

4.5.7. Delete buckets

A bucket is a container for objects stored in OSS. Every object is contained in a bucket. This topic describes how to delete a bucket.

Note

- Before you delete a bucket, you must delete all objects in the bucket, LiveChannel objects, and fragments generated by multipart uploads. For more information about deleting a LiveChannel object, see [LiveChannel](#).
- To delete a bucket in which incomplete multipart upload tasks exist, you must first call `listMultipartUploads` and `abortMultipartUpload` to cancel the tasks.

The following code provides an example on how to call `deleteBucket` to delete a specified bucket:

```
let OSS = require('ali-oss');
let client = new OSS({
  region: '<Your region>',
  accessKeyId: '<Your AccessKeyId>',
  accessKeySecret: '<Your AccessKeySecret>'
});
async function deleteBucket() {
  try {
    const result = await client.deleteBucket('your bucket name');
    console.log(result);
  } catch (err) {
    console.log(err);
  }
}
deleteBucket();
```

For more information about how to delete a bucket, see [DeleteBucket](#).

4.5.8. Pay-by-requester mode

When pay-by-requester is enabled for a bucket in Object Storage Service (OSS), the requester is charged the requests and traffic fees instead of the bucket owner. The bucket owner is charged only the storage fees. You can enable pay-by-requester for a bucket to share data in the bucket without paying for the request and traffic fees incurred by the access to your bucket.

Configure the pay-by-requester mode

The following code provides an example on how to enable pay-by-requester for a bucket:

```
const OSS = require('ali-oss')
const client = new OSS({
  region: '<Your region>',
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to
  // log on to OSS, because the account has permissions on all API operations. We recommend that
  // you use a RAM user to call API operations or perform routine operations and maintenance. To
  // create a RAM user, log on to https://ram.console.aliyun.com.
  accessKeyId: '<Your AccessKeyId>',
  accessKeySecret: '<Your AccessKeySecret>',
});
async function setBucketRequestPayment(bucket, Payer) {
  try {
    // Set bucket to the name of the bucket to configure the pay-by-requester mode.
    // Set Payer to Requester or BucketOwner.
    // If Payer is set to Requester, the pay-by-requester mode is enabled for the bucket. The
    // requester is charged for the traffic and request fees that are generated when reading data
    // in the bucket.
    // If Payer is set to BucketOwner, the pay-by-requester mode is disabled for the bucket.
    // This is the default configuration for the bucket. In this case, the bucket owner is charged
    // for the request fees.
    let result = await client.putBucketRequestPayment(bucket, Payer);
    console.log(result);
  } catch (e) {
    console.log(e);
  }
}
setBucketRequestPayment('bucketName', 'Requester')
```

For more information about how to configure the pay-by-requester mode for a bucket, see [PutBucketRequestPayment](#).

Query pay-by-requester configurations

The following code provides an example on how to query the pay-by-requester configurations of a bucket:

```
const OSS = require('ali-oss')
const client = new OSS({
  region: '<Your region>',
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to
  // log on to OSS, because the account has permissions on all API operations. We recommend that
  // you use a RAM user to call API operations or perform routine operations and maintenance. To
  // create a RAM user, log on to https://ram.console.aliyun.com.
  accessKeyId: '<Your AccessKeyId>',
  accessKeySecret: '<Your AccessKeySecret>',
});
async function getBucketRequestPayment(bucket) {
  try {
    // Query the pay-by-requester configurations.
    let result = await client.getBucketRequestPayment(bucket);
    console.log(result.payer);
  } catch (e) {
    console.log(e);
  }
}
getBucketRequestPayment('bucketName')
```

For more information about how to query the pay-by-requester configurations of a bucket, see [GetBucketRequestPayment](#).

Specify a third party to pay for access to objects

The following code provides an example on how to specify a third party to pay for access to objects when you call the PutObject, GetObject, and DeleteObject operations:


```
const OSS = require('ali-oss')
const bucket = 'bucket-name'
const payer = 'Requester'
const client = new OSS({
  region: '<Your region>',
  // Security risks may arise if you log on with the AccessKey pair of an Alibaba Cloud account because the account has permissions on all API operations in OSS. We recommend that you use a RAM user to call API operations or perform routine operations and maintenance. To create a RAM user, log on to https://ram.console.aliyun.com.
  accessKeyId: '<Your AccessKeyId>',
  accessKeySecret: '<Your AccessKeySecret>',
  bucket: bucket
});

async function put() {
  await client.putBucketRequestPayment(bucket, payer)
  // Specify the payer when you call the PutObject operation.
  await client.put('fileName', 'fileContent', {
    headers: {
      'x-oss-request-payer': 'requester'
    }
  })
}

async function get () {
  await client.putBucketRequestPayment(bucket, payer)
  // Specify the payer when you call the GetObject operation.
  await client.get('fileName', {
    headers: {
      'x-oss-request-payer': 'requester'
    }
  })
}

async function del() {
  await client.putBucketRequestPayment(bucket, payer)
  // Specify the payer when you call the DeleteObject operation.
  await client.delete('fileName', {
    headers: {
      'x-oss-request-payer': 'requester'
    }
  })
}
```

4.5.9. Manage lifecycle rules

Object Storage Service (OSS) allows you to configure lifecycle rules to delete expired objects and parts or convert the storage class of expired objects to Infrequent Access (IA) or Archive to reduce your storage costs. This topic describes how to manage lifecycle rules for buckets.

Configure lifecycle rules

The following code provides an example on how to use `putBucketLifecycle` to configure lifecycle rules for a bucket:

```

const OSS = require('ali-oss')
const client = new OSS({
  // The endpoint of the China (Hangzhou) region is used in this example. Specify the actual endpoint.
  region: '<Your region>'
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log on to OSS because the account has permissions on all API operations. We recommend that you use your RAM user's credentials to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
  accessKeyId: '<Your AccessKeyId>',
  accessKeySecret: '<Your AccessKeySecret>',
  bucket: '<Your bucket name>'
});

async function putBucketLifecycle(lifecycle) {
  try {
    const result = await client.putBucketLifecycle('<bucket-name>', [
      lifecycle
    ]);
    console.log(result);
  } catch (e) {
    console.log(e);
  }
}

const lifecycle1 = {
  id: 'rule1',
  status: 'Enabled',
  prefix: 'foo/',
  expiration: {
    days: 3 // Specify that the current versions of objects expire three days after they are last modified.
  }
}

putBucketLifecycle(lifecycle1)

const lifecycle2 = {
  id: 'rule2',
  status: 'Enabled',
  prefix: 'foo/',
  expiration: {
    createdBeforeDate: '2020-02-18T00:00:00.000Z' // Specify that the objects last modified before the specified date expire.
  },
}

putBucketLifecycle(lifecycle2)

const lifecycle3 = {
  id: 'rule3',
  status: 'Enabled',
  prefix: 'foo/',
  abortMultipartUpload: {
    days: 3 // Specify the parts expire three days after they are last modified.
  },
}

putBucketLifecycle(lifecycle3)

const lifecycle4 = {
  id: 'rule4',

```

```
    status: 'Enabled',
    prefix: 'foo/',
    abortMultipartUpload: {
      createdBeforeDate: '2020-02-18T00:00:00.000Z' // Specify that parts last modified before the specified date expire.
    },
  }
}
putBucketLifecycle(lifecycle4)
const lifecycle5 = {
  id: 'rule5',
  status: 'Enabled',
  prefix: 'foo/',
  transition: {
    // Specify that the storage class of the current versions of objects is converted to Archive 20 days after they are last modified.
    days: 20,
    storageClass: 'Archive'
  },
  expiration: {
    days: 21 // Specify that the current versions of objects expire 21 days after they are last modified.
  },
}
putBucketLifecycle(lifecycle5)
const lifecycle6 = {
  id: 'rule6',
  status: 'Enabled',
  prefix: 'foo/',
  transition: {
    // Specify that the storage class of the objects that are last modified before the specified date is converted to Archive.
    createdBeforeDate: '2020-02-18T00:00:00.000Z', // Specify the date for conversion that is earlier than that for expiration.
    storageClass: 'Archive'
  },
  expiration: {
    createdBeforeDate: '2020-02-19T00:00:00.000Z' // Specify that the objects last modified before the specified date expire.
  },
}
putBucketLifecycle(lifecycle6)
const lifecycle7 = {
  id: 'rule7',
  status: 'Enabled',
  prefix: 'foo/',
  noncurrentVersionExpiration: {
    noncurrentDays: 1 // Specify that objects expire one day after they become previous versions.
  },
}
putBucketLifecycle(lifecycle7)
const lifecycle8 = {
  id: 'rule8',
  status: 'Enabled',
```

```
    prefix: 'foo/',
    expiredObjectDeleteMarker: true // Specify that delete markers are automatically removed
    when they expire.
  }
  putBucketLifecycle(lifecycle8)
  const lifecycle9 = {
    id: 'rule9',
    status: 'Enabled',
    prefix: 'foo/',
    // Specify that the storage class of previous versions of objects is converted to IA 10 d
    ays after they become previous versions.
    noncurrentVersionTransition: {
      noncurrentDays: '10',
      storageClass: 'IA'
    }
  }
  putBucketLifecycle(lifecycle9)
  const lifecycle10 = {
    id: 'rule10',
    status: 'Enabled',
    prefix: 'foo/',
    // Specify that the storage class of previous versions of objects is converted to IA 10 d
    ays after they become previous versions.
    noncurrentVersionTransition: {
      noncurrentDays: '10',
      storageClass: 'IA'
    },
    // Specify the tags for objects to match the rules.
    tag: [{
      key: 'key1',
      value: 'value1'
    },
    {
      key: 'key2',
      value: 'value2'
    }
  ]
  }
  putBucketLifecycle(lifecycle10)
```

View lifecycle rules

The following code provides an example on how to use `getBucketLifecycle` to view the lifecycle rules configured for a bucket:

```
const OSS = require('ali-oss')
const client = new OSS({
  region: '<Your region>'
  accessKeyId: '<Your AccessKeyId>',
  accessKeySecret: '<Your AccessKeySecret>',
  bucket: '<Your bucket name>'
});
async function getBucketLifecycle () {
  try {
    let result = await client.getBucketLifecycle('<bucket-name>');
    console.log(result.rules); // Query the lifecycle rules.
    rules.forEach(rule => {
      console.log(rule.id) // View the rule IDs.
      console.log(rule.status) // View the status of the rules.
      console.log(rule.tags) // View the tags configured in the lifecycle rules.
      console.log(rule.expiration.days) // View the validity period configurations.
      console.log(rule.expiration.createdBeforeDate) // View the expiration date configurations.
      // View the configurations used to delete expired parts.
      console.log(rule.abortMultipartUpload.days || rule.abortMultipartUpload.createdBeforeDate)
      // View the configurations used to convert storage classes.
      console.log(rule.transition.days || rule.transition.createdBeforeDate) // View the conversion date configurations.
      console.log(rule.transition.storageClass) // View the configurations used to convert storage classes.
      // View the configurations used to remove delete markers.
      console.log(rule.transition.expiredObjectDeleteMarker)
      // View the lifecycle rules to convert the storage class of the previous versions of the objects.
      console.log(rule.noncurrentVersionTransition.noncurrentDays) // View the conversion date configurations for previous versions of objects.
      console.log(rule.noncurrentVersionTransition.storageClass) // View the configurations used to convert the storage classes for previous versions.
      // View the expiration configurations for the previous versions of the objects.
      console.log(rule.noncurrentVersionExpiration.noncurrentDays)
    })
  } catch (e) {
    console.log(e);
  }
}
getBucketLifecycle();
```

Delete lifecycle rules

The following code provides an example on how to use `deleteBucketLifecycle` to delete the lifecycle rules configured for a bucket:

```
let OSS = require('ali-oss')
let client = new OSS({
  region: '<Your region>'
  accessKeyId: '<Your AccessKeyId>',
  accessKeySecret: '<Your AccessKeySecret>',
  bucket: '<Your bucket name>'
});
async function deleteBucketLifecycle () {
  try {
    let result = await client.deleteBucketLifecycle('bucket-name');
    console.log(result);
  } catch (e) {
    console.log(e);
  }
}
deleteBucketLifecycle();
```

4.5.10. Bucket policy

This topic describes how to configure, query, and delete policies for a bucket.

Background information

Bucket policies provide resource-based authorization for users. Bucket policies apply to the following scenarios:

For more information about bucket policy configurations and use cases, see [Configure bucket policies to authorize other users to access OSS resources](#). For more information about bucket policy syntax, see [Policy structure and syntax](#).

Configure a bucket policy

The following sample code shows how to configure a policy for a bucket:

```
const OSS = require('ali-oss')
const client = new OSS({
  bucket: '<Your BucketName>',
  // This example uses the endpoint of the China (Hangzhou) region. Specify the actual endpoint based on your requirements.
  region: '<Your Region>',
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log on to OSS, because the account has permissions on all API operations. We recommend that you use your RAM user's credentials to call API operations or perform routine operations and maintenance. To create your RAM user, log on to https://ram.console.aliyun.com.
  accessKeyId: '<Your AccessKeyId>',
  accessKeySecret: '<Your AccessKeySecret>',
});
// Configure the bucket policy.
const policy = {
  Version: '1',
  Statement: [
    {
      Action: ['oss:PutObject', 'oss:GetObject'],
      Effect: 'Deny',
      Principal: ['1234567890'],
      Resource: ['acs:oss:*:1234567890:*/*']
    }
  ]
};
async function putPolicy() {
  const result = await client.putBucketPolicy('<Your Bucket Name>', policy);
  console.log(result)
}
putPolicy()
```

Query a bucket policy

The following sample code shows how to query the policies configured for a bucket:

```
const OSS = require('ali-oss')
const client = new OSS({
  bucket: '<Your BucketName>',
  // This example uses the endpoint of the China (Hangzhou) region. Specify the actual endpoint based on your requirements.
  region: '<Your Region>',
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log on to OSS, because the account has permissions on all API operations. We recommend that you use your RAM user's credentials to call API operations or perform routine operations and maintenance. To create your RAM user, log on to https://ram.console.aliyun.com.
  accessKeyId: '<Your AccessKeyId>',
  accessKeySecret: '<Your AccessKeySecret>',
});
// Query the bucket policy.
async function getPolicy() {
  const result = await client.getBucketPolicy('<Your Bucket Name>');
  console.log(result.policy)
}
getPolicy()
```

Delete a bucket policy

The following sample code shows how to delete the policies configured for a bucket:

```
const OSS = require('ali-oss')
const client = new OSS({
  bucket: '<Your BucketName>',
  // This example uses the endpoint of the China (Hangzhou) region. Specify the actual endpoint based on your requirements.
  region: '<Your Region>',
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log on to OSS, because the account has permissions on all API operations. We recommend that you use your RAM user's credentials to call API operations or perform routine operations and maintenance. To create your RAM user, log on to https://ram.console.aliyun.com.
  accessKeyId: '<Your AccessKeyId>',
  accessKeySecret: '<Your AccessKeySecret>',
});
// Delete the bucket policy.
async function deletePolicy() {
  const result = await client.deleteBucketPolicy('<Your Bucket Name>');
  console.log(result)
}
deletePolicy()
```

4.5.11. Retention policy

You can configure a time-based retention policy for a bucket. A retention policy has a retention period that ranges from one day to 70 years. This topic describes how to create, query, and lock a retention policy.

Background information

OSS supports the Write Once Read Many (WORM) strategy that prevents an object from being deleted or overwritten for a specified period of time.

If a retention policy is not locked within 24 hours after it is created, the retention policy becomes invalid. After the retention policy configured for a bucket is locked, you can upload objects to or read objects from the bucket. However, you cannot delete objects in the bucket or the retention policy within the retention period of the policy. The retention period of the policy cannot be shortened but only be extended. For more information about retention policies, see [Retention policies](#).

Create a retention policy

The following code provides an example on how to create a retention policy:

```
const OSS = require('ali-oss');
const client = new OSS({
  // The endpoint of the China (Hangzhou) region is used in this example. Specify the actual endpoint.
  region: '<Your region>',
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log on to OSS because the account has permissions on all API operations. We recommend that you use your RAM user's credentials to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
  accessKeyId: '<Your AccessKeyId>',
  accessKeySecret: '<Your AccessKeySecret>'
});
// Create the retention policy.
async function initiateBucketWorm() {
  const bucket = '<Your Bucket Name>'
  // Specify the retention period of the retention policy.
  const days = '<Retention Days>'
  const res = await client.initiateBucketWorm(bucket, days)
  console.log(res.wormId)
}
initiateBucketWorm()
```

For more information about how to create a retention policy, see [InitiateBucketWorm](#).

Cancel an unlocked retention policy

The following code provides an example on how to cancel an unlocked retention policy:

```
const OSS = require('ali-oss');
const client = new OSS({
  // The endpoint of the China (Hangzhou) region is used in this example. Specify the actual endpoint.
  region: '<Your region>',
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log on to OSS because the account has permissions on all API operations. We recommend that you use your RAM user's credentials to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
  accessKeyId: '<Your AccessKeyId>',
  accessKeySecret: '<Your AccessKeySecret>'
});
// Cancel the retention policy.
async function abortBucketWorm() {
  const bucket = '<Your Bucket Name>'
  try {
    await client.abortBucketWorm(bucket)
    console.log('abort success')
  } catch(err) {
    console.log('err: ', err)
  }
}
abortBucketWorm()
```

For more information about how to cancel an unlocked retention policy, see [AbortBucketWorm](#).

Lock a retention policy

The following code provides an example on how to lock a retention policy:

```
const OSS = require('ali-oss');
const client = new OSS({
  // The endpoint of the China (Hangzhou) region is used in this example. Specify the actual endpoint.
  region: '<Your region>',
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log on to OSS because the account has permissions on all API operations. We recommend that you use your RAM user's credentials to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
  accessKeyId: '<Your AccessKeyId>',
  accessKeySecret: '<Your AccessKeySecret>'
});
// Lock the retention policy.
async function completeBucketWorm() {
  const bucket = '<Your Bucket Name>'
  const wormId = '<Your Worm Id>'
  try {
    await client.completeBucketWorm(bucket, wormId)
  } catch(err) {
    console.log(err)
  }
}
completeBucketWorm()
```

For more information about how to lock a retention policy, see [CompleteBucketWorm](#).

Query a retention policy

The following code provides an example on how to query a retention policy:

```
const OSS = require('ali-oss');
const client = new OSS({
  // The endpoint of the China (Hangzhou) region is used in this example. Specify the actual endpoint.
  region: '<Your region>',
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log on to OSS because the account has permissions on all API operations. We recommend that you use your RAM user's credentials to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
  accessKeyId: '<Your AccessKeyId>',
  accessKeySecret: '<Your AccessKeySecret>'
});
// Query the retention policy.
async function getBucketWorm() {
  const bucket = '<Your Bucket Name>'
  try {
    const res = await client.getBucketWorm(bucket)
    // Query the ID of the retention policy.
    console.log(res.wormId)
    // Query the status of the retention policy. InProgress indicates that the retention policy is not locked. Locked indicates that the retention policy is locked.
    console.log(res.state)
    // Query the retention period of the retention policy.
    console.log(res.days)
  } catch(err) {
    console.log(err)
  }
}
getBucketWorm()
```

For more information about how to query a retention policy, see [GetBucketWorm](#).

Extend the retention period of a retention policy

The following code provides an example on how to extend the retention period of a locked retention policy:

```
const OSS = require('ali-oss');
const client = new OSS({
  // The endpoint of the China (Hangzhou) region is used in this example. Specify the actual endpoint.
  region: '<Your region>',
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log on to OSS because the account has permissions on all API operations. We recommend that you use your RAM user's credentials to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
  accessKeyId: '<Your AccessKeyId>',
  accessKeySecret: '<Your AccessKeySecret>'
});
// Extend the retention period of the locked retention policy.
async function extendBucketWorm() {
  const bucket = '<Your Bucket Name>'
  const wormId = '<Your Worm Id>'
  const days = '<Retention Days>'
  try {
    const res = await client.extendBucketWorm(bucket, wormId, days)
    console.log(res)
  } catch(err) {
    console.log(err)
  }
}
extendBucketWorm()
```

For more information about how to extend the retention period of a locked retention policy, see [ExtendBucketWorm](#).

4.5.12. Bucket tagging

Configure bucket tagging

You can run the following code to add tags to a bucket:

```
const OSS = require('ali-oss')
const client = new OSS({
  // This example uses the endpoint of the China (Hangzhou) region. Specify the actual endpoint based on your requirements.
  region: '<Your region>',
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log on to OSS, because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine operations and maintenance. To create a RAM user, log on to https://ram.console.aliyun.com.
  accessKeyId: '<Your AccessKeyId>',
  accessKeySecret: '<Your AccessKeySecret>',
});
// Configure bucket tagging.
async function putBucketTags(bucketName, tag) {
  try {
    let result = await client.putBucketTags(bucketName, tag);
    console.log(result);
  } catch (e) {
    console.log(e);
  }
}
const tag = { a: '1', b: '2' };
putBucketTags('bucketName', tag)
```

Query bucket tags

You can run the following code to query the tags to a bucket:

```
const OSS = require('ali-oss')
const client = new OSS({
  //This example uses the endpoint of the China (Hangzhou) region. Specify the actual endpoint based on your requirements.
  region: '<Your region>',
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log on to OSS, because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine operations and maintenance. To create a RAM user, log on to https://ram.console.aliyun.com.
  accessKeyId: '<Your AccessKeyId>',
  accessKeySecret: '<Your AccessKeySecret>',
});
//Query bucket tags.
async function getBucketTags(bucketName) {
  try {
    let result = await client.getBucketTags(bucketName);
    console.log(result);
  } catch (e) {
    console.log(e);
  }
}
getBucketTags('bucketName')
```

Lists buckets that have specific tags

The following code provides an example on how to list buckets that have a specified tag:

```
const OSS = require('ali-oss')
const client = new OSS({
  //This example uses the endpoint of the China (Hangzhou) region. Specify the actual endpoint based on your requirements.
  region: '<Your region>',
  //Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log on to OSS, because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine operations and maintenance. To create a RAM user, log on to https://ram.console.aliyun.com.
  accessKeyId: '<Your AccessKeyId>',
  accessKeySecret: '<Your AccessKeySecret>',
});
// List buckets that have specific tags.
async function listBucketsTags() {
  try {
    const params = {
      // Add the tag-key and tag-value fields to the params parameter of the listBuckets API operation.
      'tag-key': '<yourTagKey>',
      'tag-value': '<yourTagValue>'
    }
    const result = await client.listBuckets(params);
    console.log(result);
  } catch (err) {
    console.log(err);
  }
}
listBucketTags('bucketName')
```

Delete bucket tagging configurations

```
const OSS = require('ali-oss')
const client = new OSS({
  //This example uses the endpoint of the China (Hangzhou) region. Specify the actual endpoint based on your requirements.
  region: '<Your region>',
  //Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log on to OSS, because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine operations and maintenance. To create a RAM user, log on to https://ram.console.aliyun.com.
  accessKeyId: '<Your AccessKeyId>',
  accessKeySecret: '<Your AccessKeySecret>',
});
// Delete bucket tagging configurations.
async function deleteBucketTags(bucketName) {
  try {
    let result = await client.deleteBucketTags(bucketName);
    console.log(result);
  } catch (e) {
    console.log(e);
  }
}
deleteBucketTags('bucketName')
```

4.5.13. Bucket inventory

This topic describes how to add inventory rules for a bucket and how to query, list, and delete inventory rules configured for a bucket.

Notice

Ensure that you have permissions to perform the preceding operations. By default, the bucket owner has permissions to perform these operations. If you do not have permissions to perform these operations, apply for the permissions from the bucket owner.

Add inventory rules

Take note of the following items when you add inventory rules for a bucket:

- You can add up to 1,000 inventory rules for a bucket.
- The inventory list must be stored in a bucket within the same region as the bucket for which the inventory is configured.

The following code provides an example on how to add inventory rules for a bucket:

```
const OSS = require('ali-oss');

const client = new OSS({
  bucket: '<Your BucketName>',
  // The endpoint of the China (Hangzhou) region is used in this example. Specify the actual endpoint.
  region: '<oss-cn-hangzhou>',
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to
```

```

log on to OSS because the account has permissions on all API operations. We recommend that
you use your RAM user's credentials to call API operations or perform routine operations an
d maintenance. To create a RAM user, log on to the RAM console.
    accessKeyId: '<Your AccessKeyId>',
    accessKeySecret: '<Your AccessKeySecret>'
  });

const inventory = {
  // Specify the ID of the inventory rule.
  id: 'default',
  // Specify whether inventory is enabled for the bucket. Valid values: true and false.
  isEnabled: false,
  // (Optional) Specify the rule used to filter the objects to include in inventory lists.
  // In this example, prefix is used to filter the objects to include in inventory lists.
  prefix: 'tst',
  OSSBucketDestination: {
    // Specify the format of the inventory lists.
    format: 'CSV',
    // Specify the account ID of the owner of the destination bucket where the generated inv
    entory lists are stored.
    accountId: '<Your AccountId>',
    // Specify the role used to access the destination bucket where the generated inventory
    lists are stored.
    rolename: 'AliyunOSSRole',
    // Specify the name of the destination bucket where the generated inventory lists are s
    tored.
    bucket: '<Your BucketName>',
    // (Optional) Specify the folder where the generated inventory lists are stored.
    prefix: '<Your Prefix>',
    // To encrypt inventory lists by using keys managed by OSS, use the following code.
    // encryption: {'SSE-OSS': ''},
    // To encrypt inventory lists by using CMKs managed by KMS, use the following code.
    /*
    encryption: {
      'SSE-KMS': {
        keyId: 'test-kms-id';
      };
    },
    */
  },
  // Specify the frequency at which inventory lists are generated. A value of Weekly indica
  tes that an inventory list is generated each week. A value of Daily indicates that an inven
  tory list is generated each day.
  frequency: 'Daily',
  // Specify that all versions of the objects are included in inventory lists. If the value
  of includedObjectVersions is set to Current, only the current versions of the objects are i
  ncluded in inventory lists.
  includedObjectVersions: 'All',
  optionalFields: {
    // (Optional) Specify object attributes that are included in inventory lists.
    field: ["Size", "LastModifiedDate", "ETag", "StorageClass", "IsMultipartUploaded", "Enc
    ryptionStatus"]
  },
}

```



```
async function putInventory(){
    // Specify the name of the bucket for which you want to add inventory rules.
    const bucket = '<Your BucketName>';
    try {
        await client.putBucketInventory(bucket, inventory);
        console.log('The inventory rule is added.')
    } catch(err) {
        console.log('Failed to add the inventory rule: ', err);
    }
}

putInventory()
```

For more information about how to add inventory rules for a bucket, see [PutBucketInventory](#).

Query an inventory rule

The following code provides an example on how to query an inventory rule configured for a bucket:

```
const OSS = require('ali-oss');

const client = new OSS({
    bucket: '<Your BucketName>',
    // The endpoint of the China (Hangzhou) region is used in this example. Specify the actual endpoint.
    region: '<oss-cn-hangzhou>',
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log on to OSS because the account has permissions on all API operations. We recommend that you use your RAM user's credentials to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
    accessKeyId: '<Your AccessKeyId>',
    accessKeySecret: '<Your AccessKeySecret>'
});

async function getBucketInventoryById() {
    // Specify the name of the bucket from which you want to query the inventory rule.
    const bucket = '<Your BucketName>';
    try {
        // Query the inventory rule configured for the bucket.
        const result = await client.getBucketInventory(bucket, 'inventoryid');
        console.log(result.inventory);
    } catch (err) {
        console.log(err)
    }
}

getBucketInventoryById();
```

For more information about how to query an inventory rule configured for a bucket, see [GetBucketInventory](#).

List inventory rules

 Note

You can query up to 100 inventory rules by sending a request. To query more than 100 inventory rules, you must send multiple requests and use the token returned for each request as the parameter for the next request.

The following code provides an example on how to list all inventory rules configured for a bucket at a time:

```
const OSS = require('ali-oss');

const client = new OSS({
  bucket: '<Your BucketName>',
  // The endpoint of the China (Hangzhou) region is used in this example. Specify the actual endpoint.
  region: '<oss-cn-hangzhou>',
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log on to OSS because the account has permissions on all API operations. We recommend that you use your RAM user's credentials to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
  accessKeyId: '<Your AccessKeyId>',
  accessKeySecret: '<Your AccessKeySecret>'
});

async function listBucketInventory() {
  const bucket = '<Your Bucket Name>';
  // List the inventory rules configured for the bucket.
  const result = await client.listBucketInventory(bucket);
  console.log(result.inventoryList);
  // By default, 100 inventory rules are listed each time. If more the 100 inventory rules are configured for the bucket, the results are listed in multiple pages. You must use the nextContinuationToken returned for the request to send the next request to list the next page.
  const { nextContinuationToken } = result;
  const resultNext = await client.listBucketInventory(bucket, nextContinuationToken);
  console.log(resultNext.inventoryList);
}

listBucketInventory();
```

For more information about how to list multiple inventory rules at a time, see [ListBucketInventory](#).

Delete an inventory rule

The following code provides an example on how to delete an inventory rule configured for a bucket:

```
const OSS = require('ali-oss');

const client = new OSS({
  bucket: '<Your BucketName>',
  // The endpoint of the China (Hangzhou) region is used in this example. Specify the actual endpoint.
  region: '<oss-cn-hangzhou>',
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log on to OSS because the account has permissions on all API operations. We recommend that you use your RAM user's credentials to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
  accessKeyId: '<Your AccessKeyId>',
  accessKeySecret: '<Your AccessKeySecret>'
});

async function deleteBucketInventoryById() {
  const bucket = '<Your Bucket Name>';
  // Delete the inventory rule with the specified ID.
  const inventory_id = '<Your InventoryId>';
  try {
    await client.deleteBucketInventory(bucket, inventory_id);
  } catch (err) {
    console.log(err)
  }
}

deleteBucketInventoryById();
```

For more information about how to delete an inventory rule configured for a bucket, see [DeleteBucketInventory](#).

4.5.14. CORS

Cross-origin resource sharing (CORS) allows web applications to access resources that belong to different regions. Object Storage Service (OSS) provides CORS operations for cross-origin access control.

Configure CORS rules

The following sample code shows how to configure CORS rules for a specific bucket:

```
const OSS = require('ali-oss');
const client = new OSS({
  // Set yourRegion to the endpoint of the region in which the bucket is located. For example, if your bucket is located in the China (Hangzhou) region, set yourRegion to oss-cn-hangzhou.
  region: 'yourRegion',
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
  accessKeyId: 'yourAccessKeyId',
  accessKeySecret: 'yourAccessKeySecret',
  // Specify the name of the bucket for which you want to configure CORS rules.
  bucket: 'yourBucket'
});
const rules = [{
  // Specify the origin of allowed cross-origin requests. You can set the origin to an asterisk (*) wildcard to allow requests from all regions.
  allowedOrigin: 'http://example.com',
  // Specify the method of allowed cross-origin requests, including GET, PUT, DELETE, POST, and HEAD.
  allowedMethod: 'GET',
  // Specify the header included in the responses to allowed cross-origin requests. We recommend that you use an asterisk (*) wildcard unless otherwise specified.
  allowedHeader: '*',
  // Specify the response header that can be obtained by web applications, such as an XMLHttpRequest object in JavaScript. The asterisk (*) wildcard is not supported.
  exposeHeader: 'Content-Length',
  // Specify the time period during which the browser can cache the response to a preflight OPTIONS request for a specific resource. Unit: seconds.
  maxAgeSeconds: '30'
}];
// You can configure up to 10 CORS rules. If a new rule that is the same as an existing rule is configured, the existing rule is overwritten.
const putResult = await client.putBucketCORS(bucket, rules);
```

Query CORS rules

The following sample code shows how to query the CORS rules of a specified bucket:

```
const OSS = require('ali-oss');
const client = new OSS({
  // Set yourRegion to the endpoint of the region in which the bucket is located. For example, if your bucket is located in the China (Hangzhou) region, set yourRegion to oss-cn-hangzhou.
  region: 'yourRegion',
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
  accessKeyId: 'yourAccessKeyId',
  accessKeySecret: 'yourAccessKeySecret'
});
// Specify the name of the bucket whose CORS rules you want to query.
const getResult = await client.getBucketCORS(bucket);
assert.equal(getResult.res.status, 200);
assert.deepEqual(getResult.rules, [{
  allowedOrigin: 'https://example.com',
  allowedMethod: 'GET',
  allowedHeader: '*',
  exposeHeader: 'Content-Length',
  maxAgeSeconds: '30'
}]);
```

Delete CORS rules

The following sample code shows how to delete the CORS rules configured for a specific bucket:

```
const OSS = require('ali-oss');
const client = new OSS({
  // Set yourRegion to the endpoint of the region in which the bucket is located. For example, if your bucket is located in the China (Hangzhou) region, set yourRegion to oss-cn-hangzhou.
  region: 'yourRegion',
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
  accessKeyId: 'yourAccessKeyId',
  accessKeySecret: 'yourAccessKeySecret'
});
// Specify the name of the bucket whose CORS rules you want to delete.
client.deleteBucketCORS('bucketName').then((res) => {
  console.log(res);
}).catch(e => {
  console.log(e)
})
```

References

- For more information about the complete sample code of CORS, visit [GitHub](#).
- For more information about the API operation that you can call to configure CORS rules, see

[PutBucketCors](#).

- For more information about the API operation that you can call to query CORS rules, see [GetBucketCors](#).
- For more information about the API operation that you can call to delete CORS rules, see [DeleteBucketCors](#).

4.5.15. Access log

You can enable access log to record bucket access to log files, which are stored in a specified bucket

Log format:

```
<TargetPrefix><SourceBucket>YYYY-mm-DD-HH-MM-SS-UniqueString
```

For more information about access log, see the "[Log storage](#)."

Enable access log

Use `putBucketLogging` to enable access log of the bucket.

```
let OSS = require('ali-oss')
let client = new OSS({
  region: '<Your region>',
  accessKeyId: '<Your AccessKeyId>',
  accessKeySecret: '<Your AccessKeySecret>',
  bucket: '<Your bucket name>'
});
async function putBucketLogging () {
  try {
    let result = await client.putBucketLogging('bucket-name', 'logs/');
    console.log(result)
  } catch (e) {
    console.log(e)
  }
}
putBucketLogging();
```

View configurations of access log

Use `getBucketLogging` to view configurations of access log.

```
let OSS = require('ali-oss')
let client = new OSS({
  region: '<Your region>'
  accessKeyId: '<Your AccessKeyId>',
  accessKeySecret: '<Your AccessKeySecret>',
  bucket: '<Your bucket name>'
});
async function getBucketLogging() {
  try {
    let result = await client.getBucketLogging('bucket-name');
    console.log(result);
  } catch (e) {
    console.log(e);
  }
}
getBucketLogging();
```

Disable access log

Use `deleteBucketLogging` to disable access log of the bucket.

```
let OSS = require('ali-oss')
let client = new OSS({
  region: '<Your region>'
  accessKeyId: '<Your AccessKeyId>',
  accessKeySecret: '<Your AccessKeySecret>',
  bucket: '<Your bucket name>'
});
async function deleteBucketLogging () {
  try {
    let result = await client.deleteBucketLogging('bucket-name');
    console.log(result);
  } catch (e) {
    console.log(e);
  }
}
deleteBucketLogging();
```

4.5.16. Hotlink protection

This topic describes how to use hot link protection.

To prevent your data stored in Object Storage Service (OSS) from unauthorized access, you can configure a Referrer whitelist for your bucket by specifying the following parameters:

- Referrer Whitelist: specifies that only specified domain names are allowed to access your resources.
- Allow Empty Referrer: determines whether requests that contain an empty Referrer field are allowed. If you specify that an empty Referrer field is not allowed, only HTTP and HTTPS requests that contain an allowed Referrer field can access your OSS resources.

For more information about hot link protection, see [Hotlink protection](#) in OSS Developer Guide.

Configure a Referer whitelist

The following code provides an example on how to call `putBucketReferer` to configure the Referer whitelist for a bucket:

```
let OSS = require('ali-oss')
let client = new OSS({
  region: '<Your region>'
  accessKeyId: '<Your AccessKeyId>',
  accessKeySecret: '<Your AccessKeySecret>',
  bucket: '<Your bucket name>'
});
async function putBucketReferer () {
  try {
    let result = await client.putBucketReferer('bucket-name', true, [
      'example.com',
      '*.example.com'
    ]);
    console.log(result);
  } catch (e) {
    console.log(e);
  }
}
putBucketReferer();
```

Query a Referer whitelist

The following code provides an example on how to call `GetBucketReferer` to query a Referer whitelist that is configured for a bucket:

```
let OSS = require('ali-oss')
let client = new OSS({
  region: '<Your region>'
  accessKeyId: '<Your AccessKeyId>',
  accessKeySecret: '<Your AccessKeySecret>',
  bucket: '<Your bucket name>'
});
async function getBucketReferer () {
  try {
    let result = await client.getBucketReferer('bucket-name');
    console.log(result);
  } catch (e) {
    console.log(e);
  }
}
getBucketReferer();
```

Clear a Referer whitelist

The following code provides an example on how to call `deleteBucketReferer` to clear a Referer whitelist:


```
let OSS = require('ali-oss')
let client = new OSS({
  region: '<Your region>'
  accessKeyId: '<Your AccessKeyId>',
  accessKeySecret: '<Your AccessKeySecret>',
  bucket: '<Your bucket name>'
});
async function deleteBucketReferer () {
  try {
    let result = await client.deleteBucketReferer('bucket-name');
    console.log(result);
  } catch (e) {
    console.log(e);
  }
}
deleteBucketReferer();
```

4.5.17. Static website hosting (mirroring-based back-to-origin)

Static websites are websites in which all web pages consist only of static content, including scripts such as JavaScript code that can be run on the client. You can use the static website hosting feature to host your static website in an Object Storage Service (OSS) bucket and use the domain name of the bucket to access the website.

Static website hosting

Static websites are websites in which all web pages consist only of static content, including scripts such as JavaScript code that can be run on the client. You can use the static website hosting feature to host your static website in an Object Storage Service (OSS) bucket and use the domain name of the bucket to access the website.

- The following code provides an example on how to configure static website hosting:

```
const OSS = require('ali-oss')
const client = new OSS({
  // Set yourRegion to the region in which the bucket is located. For example, if your bucket is located in the China (Hangzhou) region, set yourRegion to oss-cn-hangzhou.
  region: 'yourRegion',
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
  accessKeyId: 'yourAccessKeyId',
  accessKeySecret: 'yourAccessKeySecret',
});
// Configure static website hosting.
async function putBucketWebsite () {
  try {
    // Specify the name of the bucket. Example: examplebucket.
    const result = await client.putBucketWebsite('examplebucket', {
      // Specify the default homepage of the static website hosted on the bucket.
      index: 'index.html',
      // Specify the default 404 page of the static website hosted on the bucket.
      error: 'error.html'
    });
    console.log(result);
  } catch (e) {
    console.log(e);
  }
}
putBucketWebsite();
```

Configure static website hosting

- The following code provides an example on how to query static website hosting configurations:

```

const OSS = require('ali-oss')
const client = new OSS({
  // Set yourRegion to the region in which the bucket is located. For example, if your bucket is located in the China (Hangzhou) region, set yourRegion to oss-cn-hangzhou.
  region: 'yourRegion',
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
  accessKeyId: 'yourAccessKeyId',
  accessKeySecret: 'yourAccessKeySecret',
});
// Query the default homepage and default 404 page of the static website hosted on the bucket.
async function getBucketWebsite () {
  try {
    // Specify the name of the bucket. Example: examplebucket.
    const result = await client.getBucketWebsite('examplebucket');
    console.log(result);
  } catch (e) {
    console.log(e);
  }
}
getBucketWebsite();

```

Query static website hosting configurations

- The following code provides an example on how to delete static website hosting configurations:

```

const OSS = require('ali-oss')
const client = new OSS({
  // Set yourRegion to the region in which the bucket is located. For example, if your bucket is located in the China (Hangzhou) region, set yourRegion to oss-cn-hangzhou.
  region: 'yourRegion',
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
  accessKeyId: 'yourAccessKeyId',
  accessKeySecret: 'yourAccessKeySecret',
});
// Delete the static website hosting configurations.
async function deleteBucketWebsite () {
  try {
    // Specify the name of the bucket. Example: examplebucket.
    const result = await client.deleteBucketWebsite('examplebucket');
    console.log(result);
  } catch (e) {
    console.log(e);
  }
}
deleteBucketWebsite();

```

Delete static website hosting configurations

Mirroring-based back-to-origin

- When a requester attempts to access an object in the bucket that you specified and the object does not exist, you can specify the URL of the object in the origin and back-to-origin conditions to obtain the object. For example, a bucket named `examplebucket` resides in the China (Hangzhou) region. When a requester attempts to access an object in the `examplefolder` directory of the root directory of the bucket and the object does not exist, the requester is redirected to `https://www.example.com/` to access the required object that is stored in the `examplefolder` directory.

The following code provides an example on how to configure mirroring-based back-to-origin rules in the preceding scenario:

```
const OSS = require('ali-oss')
const path = require("path")
const client = new OSS({
  // Set yourRegion to the region in which the bucket is located. For example, if your bucket is located in the China (Hangzhou) region, set yourRegion to oss-cn-hangzhou.
  region: 'yourRegion',
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
  accessKeyId: 'yourAccessKeyId',
  accessKeySecret: 'yourAccessKeySecret',
});
async function putBucketWebsite() {
  try {
    // Specify the name of the bucket. Example: examplebucket.
    const result = await client.putBucketWebsite("examplebucket", {
      // Specify the default homepage of the static website hosted on the bucket.
      index: "index.html",
      // Specify the default 404 page of the static website hosted on the bucket.
      error: "error.html",
      // Specify whether to redirect the requester to the default homepage in the subdirectory when the subdirectory is accessed.
      supportSubDir: true,
      // Specify the operation that you want to perform when the default homepage is specified, the name of the accessed object does not contain a forward slash (/) as a suffix, and the object does not exist. This parameter takes effect only when SupportSubDir is set to true.
      // type: 0,
      routingRules: [
        { RuleNumber: 1,
          // Specify the prefix of object names. Only the objects whose names contain the specified prefix match the rule.
          Condition: { KeyPrefixEquals: "examplefolder/" ,
            // Specify the HTTP status code. The rule is matched only when the specified object is accessed and HTTP status code 404 is returned.
            HttpStatusCodeReturnedEquals: 404
          },
          // Specify the redirect type.
          Redirect: { RedirectType: "Mirror",
            // Specify whether to include parameters of the original request
```

```

// Specify whether to include parameters of the original request
in the redirect request when the system uses the redirection rule or the mirroring-based
back-to-origin rule.
    PassQueryString: true,
    // Specify the origin URL for the mirroring-based back-to-origin
rule.
    MirrorURL: 'http://example.com/examplefolder',
    // Specify the HTTP status codes from the origin when you want t
o pass through the HTTP status codes and the body to the client. Examples of valid values
: 4xx and 5xx. Separate multiple HTTP status codes with commas (,).
    TransparentMirrorResponseCodes: "404,405",
    // This parameter is used in the same manner as the PassQueryStr
ing parameter and is assigned a higher priority level than the PassQueryString parameter.
This parameter takes effect only when the value of the RedirectType parameter is Mirror.
    MirrorPassQueryString: true,
    // Specify whether to redirect the requester to the address that
is specified by the Location parameter if the origin returns a 3xx HTTP status code. This
parameter takes effect only when the value of the RedirectType parameter is Mirror. If yo
u set MirrorFollowRedirect to true, OSS redirects the requester to the address that is sp
ecified by the Location parameter.
    MirrorFollowRedirect: true,
    // Specify whether to check the MD5 hash of the body of the resp
onse that is returned by the origin.
    MirrorCheckMd5: false,
    // Specify the headers that are included in the request when you
use a mirroring-based back-to-origin rule.
    // Specify whether to pass through all request headers to the or
igin.
    MirrorHeaders: { PassAll: true,
        // Configure the Pass parameter to specify the h
eaders that you want to pass through to the origin.
        Pass: 'myheader-key1',
        Pass: 'myheader-key2',
        // Configure the Remove parameter to specify the
headers that cannot be passed through to the origin.
        Remove: 'myheader-key3',
        Remove: 'myheader-key4' }
    }
}
});
console.log(result);
} catch (e) {
    console.log(e);
}
}
putBucketWebsite();

```

Configure mirroring-based back-to-origin

- The following code provides an example on how to query the mirroring-based back-to-origin configurations of a bucket:

```
const OSS = require('ali-oss')
const path = require("path")
const client = new OSS({
  // Set yourRegion to the region in which the bucket is located. For example, if your bucket is located in the China (Hangzhou) region, set yourRegion to oss-cn-hangzhou.
  region: 'yourRegion',
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
  accessKeyId: 'yourAccessKeyId',
  accessKeySecret: 'yourAccessKeySecret',
});
// Query mirroring-based back-to-origin configurations.
async function getBucketWebsite () {
  try {
    // Specify the name of the bucket. Example: examplebucket.
    const result = await client.getBucketWebsite('examplebucket');
    console.log(result);
  } catch (e) {
    console.log(e);
  }
}
getBucketWebsite();
```

Query mirroring-based back-to-origin configurations

- The following code provides an example on how to delete the mirroring-based back-to-origin configurations of a bucket:

```
const OSS = require('ali-oss')
const path = require("path")
const client = new OSS({
  // Set yourRegion to the region in which the bucket is located. For example, if your bucket is located in the China (Hangzhou) region, set yourRegion to oss-cn-hangzhou.
  region: 'yourRegion',
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
  accessKeyId: 'yourAccessKeyId',
  accessKeySecret: 'yourAccessKeySecret',
});
// Delete the mirroring-based back-to-origin configurations of the bucket.
async function deleteBucketWebsite() {
  try {
    // Specify the name of the bucket. Example: examplebucket.
    const result = await client.deleteBucketWebsite('examplebucket');
    console.log(result);
  } catch (e) {
    console.log(e);
  }
}
deleteBucketWebsite();
```

Delete mirroring-based back-to-origin configurations

References

- For more information about the API operation that you can call to configure static website hosting or mirroring-based back-to-origin, see [PutBucketWebsite](#).
- For more information about the API operation that you can call to query static website hosting or mirroring-based back-to-origin configurations, see [GetBucketWebsite](#).
- For more information about the API operation that you can call to delete static website hosting or mirroring-based back-to-origin configurations, see [DeleteBucketWebsite](#).

4.5.18. Use custom domain names

After you add a CNAME record to map a custom domain name to a specified bucket, you can use the custom domain name to access Object Storage Service (OSS) resources. This topic describes how to use a custom domain name to access a bucket.

For example, you have a website whose domain name is example.com. Users access the images on your website by using URLs in the following format: `http://img.example.com/x.jpg`. After you migrate the data of your website to an OSS bucket, you can map the domain name to the bucket to access the images by using the same URLs. For more information, see [Map custom domain names](#) in OSS Developer Guide.

When you use OSS SDK for Node.js, you can use a custom domain name as the endpoint of a bucket to access the bucket. In this case, you must set `cname` to true.

```
let OSS = require('ali-oss')
let client = new OSS({
  endpoint: 'http://img.example.com', // Use the custom domain name as the endpoint to access the bucket.
  accessKeyId: 'yourAccessKeyId',
  accessKeySecret: 'yourAccessKeySecret',
  cname: true
});
client.useBucket('examplebucket')
```

 **Notice** If a custom domain name is mapped to a bucket, the domain name cannot be used to call the `list_buckets` operation.

4.6. Upload objects

4.6.1. Overview

Objects are the basic unit for data operations in OSS. OSS SDK for Node.js provides various methods to upload objects:

You can use OSS SDK for Node.js to upload objects by using the following method

- **Upload from local files**: supports the upload of a file up to 5 GB in size.
- **Upload from cache**: supports the upload of a file up to 5 GB in size.
- **Stream upload**: supports the upload of a file up to 5 GB in size.
- **Multipart upload**: supports the upload of a file up to 48.8 TB in size. This method is suitable for the upload of large files.
- **Append upload**: supports the upload of a file up to 5 GB in size. You can use `AppendObject` to append content to append objects that have been uploaded.
- **Resumable upload**: supports the upload of an object that is up to 48.8 TB in size. This method is suitable for the upload of large files. Resumable upload supports concurrent upload. You can define the size of each part.

4.6.2. Upload a local file

This topic describes how to call the `put` operation to upload a local file to Object Storage Service (OSS).

Sample code

The following code provides an example on how to upload a local file named `examplefile.txt` to a bucket named `examplebucket`. After the local file is uploaded, the name of the object stored in OSS is `exampleobject.txt`.


```

const OSS = require('ali-oss')
const path = require("path")
const client = new OSS({
  // Set yourregion to the endpoint of the region in which the bucket is located. For example, if your bucket is located in the China (Hangzhou) region, set yourregion to oss-cn-hangzhou.
  region: 'yourregion',
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
  accessKeyId: 'yourAccessKeyId',
  accessKeySecret: 'yourAccessKeySecret',
  // Specify the bucket name. For more information about the naming conventions for buckets, see Terms.
  bucket: 'examplebucket',
});
const headers = {
  // Specify the caching behavior of the web page when the object is downloaded.
  // 'Cache-Control': 'no-cache',
  // Specify the name of the object when the object is downloaded.
  // 'Content-Disposition': 'oss_download.txt',
  // Specify the content encoding format of the object when the object is downloaded.
  // 'Content-Encoding': 'UTF-8',
  // Specify the expiration time.
  // 'Expires': 'Wed, 08 Jul 2022 16:57:01 GMT',
  // Specify the storage class of the object.
  // 'x-oss-storage-class': 'Standard',
  // Specify the access control list (ACL) of the object.
  // 'x-oss-object-acl': 'private',
  // Set tags for the object. You can set multiple tags at a time.
  // 'x-oss-tagging': 'Tag1=1&Tag2=2',
  // Specify whether the CopyObject operation overwrites the object with the same name. In this example, this parameter is set to true, which indicates that the object with the same name cannot be overwritten.
  // 'x-oss-forbid-overwrite': 'true',
};
async function put () {
  try {
    // Specify the full paths of the object and the local file. The full path of the OSS object cannot contain the bucket name.
    // If the path of the local file is not specified, the local file is uploaded from the path of the project to which the sample program belongs.
    const result = await client.put('exampleobject.txt', path.normalize('D:\\localpath\\examplefile.txt'));
    // const result = await client.put('exampleobject.txt', path.normalize('D:\\localpath\\examplefile.txt'), { headers });
    console.log(result);
  } catch (e) {
    console.log(e);
  }
}
put();

```

References

- For more information about the complete sample code of simple upload, visit [GitHub](#).
- For more information about the API operation that is called to perform simple upload, see [Put Object](#).

4.6.3. Upload from local memory

This document describes how to upload content from local memory to OSS.

 **Note** The following sample code uses the catch syntax. Learn ES6 Promise and ES6 Async/Await on your own. For more information about how to use the SDK, see [Installation](#).

You can use `put` to upload content from local memory to OSS:

```
let OSS = require('ali-oss');
let client = new OSS({
  region: '<Your region>',
  accessKeyId: '<Your AccessKeyId>',
  accessKeySecret: '<Your AccessKeySecret>',
  bucket: 'Your bucket name'
});
async function putBuffer () {
  try {
    let result = await client.put('object-name', new Buffer('hello world'));
    console.log(result);
  } catch (e) {
    console.log(e);
  }
}
putBuffer();
```

4.6.4. Stream upload

This topic describes how to upload data streams such as file streams and network streams to objects in Object Storage Service (OSS) buckets.

Upload a file stream

The following code provides an example on how to upload a file stream to the exampleobject.txt object in the exampledir directory of the examplebucket bucket:

```

const OSS = require('ali-oss');
const fs = require('fs');
const client = new OSS({
  // Set yourRegion to the endpoint of the region in which the bucket is located. For example, if your bucket is located in the China (Hangzhou) region, set yourRegion to oss-cn-hangzhou.
  region: 'yourRegion',
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS, because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
  accessKeyId: 'yourAccessKeyId',
  accessKeySecret: 'yourAccessKeySecret',
  // Specify the name of the bucket. Example: examplebucket.
  bucket: 'examplebucket',
});
async function putStream () {
  try {
    // Use chunked encoding. When you call putStream, the OSS SDK initiates an HTTP PUT request for chunked encoding.
    // Specify the full path of the local file, and data stream is read from the local file.
    // If the path of the local file is not specified, the local file is uploaded from the path of the project to which the sample program belongs.
    let stream = fs.createReadStream('D:\\localpath\\examplefile.txt');
    // Specify the full path of the object. Example: exampledir/exampleobject.txt. The full path of the object cannot contain the bucket name.
    let result = await client.putStream('exampledir/exampleobject.txt', stream);
    console.log(result);
    // Do not use chunked encoding. If contentLength is specified in options, chunked encoding is not used.
    let stream = fs.createReadStream('D:\\localpath\\examplefile.txt');
    let size = fs.statSync('D:\\localpath\\examplefile.txt').size;
    let result = await client.putStream(
      // You can set stream to specify the types of Readable Stream, such as file stream and network stream.
      'exampledir/exampleobject.txt', stream, {contentLength: size});
    console.log(result);
  } catch (e) {
    console.log(e)
  }
}
putStream();

```

Upload a network stream

The following code provides an example on how to upload a network stream to the exampleobject.txt object in the exampledir directory of the examplebucket bucket:

```
const OSS = require("ali-oss");
const fs = require("fs");
const urlLib = require("urlLib");
const client = new OSS({
  // Set yourRegion to the endpoint of the region in which the bucket is located. For example, if your bucket is located in the China (Hangzhou) region, set yourRegion to oss-cn-hangzhou.
  region: 'yourRegion',
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS, because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
  accessKeyId: 'yourAccessKeyId',
  accessKeySecret: 'yourAccessKeySecret',
  // Specify the name of the bucket. Example: examplebucket.
  bucket: 'examplebucket',
});
// Specify the network stream URL.
const url = "https://gosspublic.alicdn.com/aliyun-oss-sdk-6.16.0.min.js";
// Import the duplex stream.
// You can set stream to specify the types of Readable Stream, such as file stream and network stream.
const Duplex = require("stream").Duplex;
// Instantiate the duplex stream.
let stream = new Duplex();
urlLib.request(url, (err, data, res) => {
  if (!err) {
    // Receive data from the duplex stream.
    stream.push(data);
    stream.push(null);
    client
      // Specify the full path of the object. Example: exampledir/exampleobject.txt. The full path of the object cannot contain the bucket name.
      .putStream("exampledir/exampleobject.txt", stream)
      .then((r) => console.log(r))
      .catch((e) => console.log(e));
  }
});
```

References

For more information about the complete sample code for stream upload, visit [GitHub](#).

4.6.5. Multipart upload

By using the multipart upload feature provided by OSS, you can split a large object into multiple parts and upload them separately. After all parts are uploaded, call the CompleteMultipartUpload operation to combine these parts into a single object to implement resumable upload.

Background information

You can call the `MultipartUpload` operation to upload a large object. In multipart upload, an object is split into multiple parts to be uploaded individually. If some parts fail to be uploaded, Object Storage Service (OSS) records the upload progress. When you resume the upload, you need only to upload the parts that fail to be uploaded.

Notice

If the `ConnectionTimeoutError` error occurs when you call the `MultipartUpload` method, you must handle the error. You can reduce the size of each part, extend the timeout period, or resend the request. You can also capture `ConnectionTimeoutError` to analyze the specific cause and fix the error. For more information, see [Network connection timeout handling](#).

The following table describes the parameters that you can configure when you upload objects by using multipart upload.

Type	Parameter	Description
Required parameters	name {String}	The full path of the object that you want to upload. The path cannot contain the bucket name.
	file {String File}	The path of the local file or HTML5 file that you want to upload.
[options] {Object} Optional parameters	[checkpoint] {Object}	The checkpoint file that records the result of the multipart upload task. You must set this parameter if you want to perform resumable upload by using multipart upload. The checkpoint file stores the progress information about a multipart upload task. If a part fails to be uploaded, the task can be continued based on the progress recorded in the checkpoint file. After the local file is uploaded, the checkpoint file is deleted.
	[parallel] {Number}	The number of parts that can be concurrently uploaded. Default value: 5. Do not modify the value of this parameter unless necessary.
	[partSize] {Number}	The size of each part that you want to upload. Valid values: 100 KB to 5 GB. Default value: 1 MB (1 x 1024 x 1024). Do not modify the value of this parameter unless necessary.
	[progress] {Function}	The callback function that is used to query the progress of the upload task. The callback function can be an async function. The callback function includes the following parameters: - percentage {Number}: the progress of the upload task in percentage. Valid values: decimals between 0 and 1. - checkpoint {Object}: the checkpoint file that records the result of the multipart upload task. - res {Object}: the response returned when a part is uploaded.

Type	Parameter	Description
	[meta] {Object}	The user metadata header, which is prefixed by <code>x-oss-meta</code> .
	[mime] {String}	The Content-Type request header.
	[headers] {Object}	Other headers. For more information, see RFC 2616. Examples: • Cache-Control: specifies the caching behavior by using specific commands in HTTP requests or responses. Example: <code>Cache-Control: public, no-cache</code>. • Content-Disposition: specifies whether the returned content is displayed as a web page or downloaded as an attachment and stored in your on-premises machine. Example: <code>Content-Disposition: somename</code>. • Content-Encoding: specifies the method that is used to compress data of the specified media type. Example: <code>Content-Encoding: gzip</code>. • Expires: specifies the validity period of the request. Unit: milliseconds.

Complete sample code

 **Notice** OSS SDK for Node.js does not support MD5 verification in multipart upload tasks. We recommend that you call the CRC64 library to determine whether to perform CRC64 checks after the multipart upload task is completed.

The following code provides an example on how to call the `multipartUpload` method to perform multipart upload:

```
const OSS = require('ali-oss');
const path = require("path");
const client = new OSS({
  // Specify the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the region to oss-cn-hangzhou.
  region: 'yourRegion',
  // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
  accessKeyId: 'yourAccessKeyId',
  accessKeySecret: 'yourAccessKeySecret',
  // Specify the bucket name. Example: examplebucket.
  bucket: 'examplebucket',
});
const progress = (p, _checkpoint) => {
  // Record the upload progress of the object.
  console.log(p);
}
```

```

    console.log(p);
    // Record the checkpoint information about the multipart upload task.
    console.log(_checkpoint);
};
const headers = {
    // Specify the caching behavior of the web page when the object is downloaded.
    'Cache-Control': 'no-cache',
    // Specify the name of the object when the object is downloaded.
    'Content-Disposition': 'example.txt',
    // Specify the content encoding format of the object when the object is downloaded.
    'Content-Encoding': 'utf-8',
    // Specify the validity period of the request. Unit: milliseconds.
    'Expires': '1000',
    // Specify the storage class of the object.
    'x-oss-storage-class': 'Standard',
    // Specify tags for the object. You can specify multiple tags for the object at the same
    time.
    'x-oss-tagging': 'Tag1=1&Tag2=2',
    // Specify whether to overwrite existing objects with the same names as the objects to up
    load when the multipart upload task is initialized. In this example, this parameter is set
    to true, which indicates that existing objects cannot be overwritten by objects with the sa
    me names.
    'x-oss-forbid-overwrite': 'true'
}
// Start the multipart upload task.
async function multipartUpload() {
    try {
        // Specify the full path of the object. Example: exampledir/exampleobject.txt. Then, sp
        ecify the full path of the local file. Example: D:\\localpath\\examplefile.txt. The full pa
        th of the object cannot contain the bucket name.
        // By default, if you set this parameter to the name of a local file such as examplefil
        e.txt without specifying the local path, the local file is uploaded from the local path of
        the project to which the sample program belongs.
        const result = await client.multipartUpload('exampledir/exampleobject.txt', path.normal
        ize('D:\\localpath\\examplefile.txt'), {
            progress,
            // headers,
            // Configure the meta parameter to specify metadata for the object. You can call the
            HeadObject operation to query the object metadata.
            meta: {
                year: 2020,
                people: 'test',
            },
        });
        console.log(result);
        // Specify the full path of the object. Example: exampledir/exampleobject.txt. The full
        path of the object cannot contain the bucket name.
        const head = await client.head('exampledir/exampleobject.txt');
        console.log(head);
    } catch (e) {
        // Handle timeout exceptions.
        if (e.code === 'ConnectionTimeoutError') {
            console.log('TimeoutError');
            // do ConnectionTimeoutError operation
        }
    }
}

```

```
,
    console.log(e);
  }
}
multipartUpload();
```

The `multipartUpload` method that is called in the preceding code contains the `.initMultipartUpload`, `.uploadPart`, and `.completeMultipartUpload` operations. If you want to perform multipart upload step by step, call the preceding operations in sequence.

Cancel a multipart upload task

You can call the `client.abortMultipartUpload` method to cancel a multipart upload task. If a multipart upload task is cancelled, the upload ID cannot be used to upload parts. The uploaded parts are deleted.

The following code provides an example on how to cancel a multipart upload task:

```
const OSS = require('ali-oss');
const client = new OSS({
  // Specify the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the region to oss-cn-hangzhou.
  region: 'yourRegion',
  // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
  accessKeyId: 'yourAccessKeyId',
  accessKeySecret: 'yourAccessKeySecret',
  // Specify the bucket name. Example: examplebucket.
  bucket: 'examplebucket',
});

async function abortMultipartUpload() {
  // Specify the full path of the object. Example: exampledir/exampleobject.txt. The full path of the object cannot contain the bucket name.
  const name = 'exampledir/exampleobject.txt';
  // Specify the upload ID. You can obtain the upload ID from the information that is returned by the multipartUpload method.
  const uploadId = '0004B999EF518A1FE585B0C9360D****';
  const result = await client.abortMultipartUpload(name, uploadId);
  console.log(result);
}

abortMultipartUpload();
```

List multipart upload tasks

You can call the `client.listUploads` method to list all ongoing multipart upload tasks. Ongoing multipart upload tasks include tasks that have been initiated but are not completed and tasks that have been canceled.


```
const OSS = require('ali-oss');
const client = new OSS({
  // Specify the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the region to oss-cn-hangzhou.
  region: 'yourRegion',
  // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
  accessKeyId: 'yourAccessKeyId',
  accessKeySecret: 'yourAccessKeySecret',
  // Specify the bucket name. Example: examplebucket.
  bucket: 'examplebucket',
});
async function listUploads(query = {}) {
  // You can configure parameters such as prefix, marker, delimiter, upload-id-marker, and max-uploads for query.
  const result = await client.listUploads(query);
  result.uploads.forEach(upload => {
    // Specify the upload IDs of the multipart upload tasks.
    console.log(upload.uploadId);
    // Combine all parts for each multipart upload task into a complete object and specify the full path of the object.
    console.log(upload.name);
  });
}
const query = {
  // Specify the maximum number of multipart upload tasks to return for the current list operation. The default value and the maximum value of max-uploads are both 1000.
  'max-uploads': 1000,
};
listUploads(query);
```

List uploaded parts

You can call the `client.listParts` method to list all parts that are uploaded by using the specified upload ID in the multipart upload task.

- List uploaded parts by using simple list

The following code provides an example on how to list uploaded parts by using simple list:

```
const OSS = require('ali-oss');
const client = new OSS({
  // Specify the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the region to oss-cn-hangzhou.
  region: 'yourRegion',
  // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
  accessKeyId: 'yourAccessKeyId',
  accessKeySecret: 'yourAccessKeySecret',
  // Specify the bucket name. Example: examplebucket.
  bucket: 'examplebucket',
});
async function listParts() {
  const query = {
    // Specify the maximum number of parts to return for the current list operation. The default value and the maximum value of max-parts are both 1000.
    'max-parts': 1000,
  };
  // Specify the full path of the object such as exampledir/exampleobject.txt and the upload ID. The full path of the object cannot contain the bucket name.
  const result = await client.listParts('exampledir/exampleobject.txt', '0004B999EF518A1FE585B0C9360D****', query);
  result.parts.forEach(part => {
    console.log(part.PartNumber);
    console.log(part.LastModified);
    console.log(part.ETag);
    console.log(part.Size);
  });
}
listParts();
```

- List all uploaded parts

By default, you can list up to 1,000 parts at the same time by using `client.listParts`. To list more than 1,000 uploaded parts, use the following code:

```
const OSS = require('ali-oss');
const client = new OSS({
  // Specify the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the region to oss-cn-hangzhou.
  region: 'yourRegion',
  // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
  accessKeyId: 'yourAccessKeyId',
  accessKeySecret: 'yourAccessKeySecret',
  // Specify the bucket name. Example: examplebucket.
  bucket: 'examplebucket',
});
async function listParts() {
  const query = {
    // Specify the maximum number of parts to return for the current list operation. The default value and the maximum value of max-parts are both 1000.
    'max-parts': 1000,
  };
  let result;
  do {
    // Specify the full path of the object such as exampledir/exampleobject.txt and the upload ID. The full path of the object cannot contain the bucket name.
    result = await client.listParts('exampledir/exampleobject.txt', '0004B999EF518A1FE585B0C9360D****', query);
    // Specify the position from which the list operation starts. Example: 2. Only the parts whose part numbers are greater than the value of this parameter are listed.
    query['2'] = result.nextPartNumberMarker;
    result.parts.forEach(part => {
      console.log(part.PartNumber);
      console.log(part.LastModified);
      console.log(part.ETag);
      console.log(part.Size);
    });
  } while (result.isTruncated === "true");
}
listParts();
```

References

- For the complete sample code that is used to perform multipart upload, visit [GitHub](#).
- The `multipartUpload` method that is used by OSS SDK for Node.js to perform multipart upload contains the following API operations:
 - The API operation that you can call to initiate a multipart upload task. For more information, see [InitiateMultipartUpload](#).
 - The API operation that you can call to upload data by part. For more information, see [UploadPart](#).
 - The API operation that you can call to complete a multipart upload task. For more information, see [CompleteMultipartUpload](#).
- For more information about the API operation that you can call to cancel a multipart upload task, see [AbortMultipartUpload](#).

- For more information about the API operation that you can call to list uploaded parts, see [ListParts](#).
- For more information about the API operation that you can call to list ongoing multipart upload tasks, see [ListMultipartUploads](#). Ongoing multipart upload tasks include tasks that have been initiated but are not completed and tasks that have been canceled.

4.6.6. Append upload

You can call the AppendObject operation to append content to existing appendable objects.

Usage notes

- If the object to which you want to append content does not exist, an appendable object is created when you call the AppendObject operation.
- When you call the AppendObject operation to append content to an object, you may encounter a few situations: If the object is an appendable object, and the specified position from which the append operation starts is equal to the current object length, the content is appended to the end of the object. If the specified position from which the append operation starts is not equal to the current object length, a `PositionNotEqualToLength` exception is thrown. If the object is not an appendable object, an `ObjectNotAppendable` exception is thrown.
- The CopyObject operation cannot be performed on appendable objects.

Sample code

The following code provides an example on how to perform append upload:

```
const OSS = require('ali-oss')
const client = new OSS({
  // Set yourRegion to the region in which the bucket is located. For example, if your bucket is located in the China (Hangzhou) region, set yourRegion to oss-cn-hangzhou.
  region: 'yourRegion',
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access Object Storage Service (OSS) because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
  accessKeyId: 'yourAccessKeyId',
  accessKeySecret: 'yourAccessKeySecret',
  // Specify the name of the bucket. Example: examplebucket.
  bucket: 'examplebucket',
});

const headers = {
  // Specify the caching behavior of the web page when the object is downloaded.
  'Cache-Control': 'no-cache',
  // Specify the name of the object when the object is downloaded.
  'Content-Disposition': 'oss_download.txt',
  // Specify the access control list (ACL) of the object.
  'x-oss-object-acl': 'private',
  // Specify the storage class of the object.
  'x-oss-storage-class': 'Standard',
  // Specify the request header that is used to check whether the content of the received message is the same as the content of the sent message.
  // To obtain the value of the Content-MD5 header, calculate a 128-bit number based on the content of the message, except the headers. Then, encode the number in Base64.
  'Content-MD5': 'ohhnqLBJFiKkPSB0leNaUA==',
}
```

```
// Specify the expiration time of the request.
'Expires': 'Fri, 31 Dec 2021 16:57:01 GMT',
// Specify the server-side encryption method. In this example, the method is set to serve
r-side encryption by using OSS-managed keys (SSE-OSS).
'x-oss-server-side-encryption': 'AES256',
// Specify the content encoding format of the object.
'Content-Encoding': 'utf-8',
};
const meta = {
  'name': 'eliot',
  'age': '18',
}
async function append () {
  // Perform the first append upload. The position from which the next append operation sta
rts is included in the response.
  // objectName specifies the full path of the uploaded object without the bucket name. Exa
mple: destfolder/examplefile.txt.
  // localFile specifies the full path of the local file, including the suffix. Example: /u
sers/local/examplefile.txt.
  const result = await client.append('objectName', 'localFile'
  // Specify custom headers and user metadata.
  //,{ headers, meta}
  )
  // Start the second append. The position from which the next append operation starts is t
he current length of the object, which is specified by Content-Length.
  result = await client.append('objectName', 'localFile', {
    position: result.nextAppendPosition
  })
}
append();
```

References

For more information about the API operation that you can call to perform append upload, see [AppendObject](#).

4.6.7. Resumable upload

You can use resumable upload to split an object into several parts and upload them simultaneously. After all parts are uploaded, you can combine the uploaded parts into a complete object and complete the resumable upload task.

Note

For more information, see [Multipart upload and resumable upload](#). You can configure lifecycle rules to regularly delete parts that are no longer used. For more information, see [Manage parts](#).

The progress parameter is provided to obtain the upload progress in callback. The SDK uses the upload progress and the checkpoint information as parameters in callback requests. In resumable upload, checkpoint information is recorded. When the upload task fails because of errors, you can pass the checkpoint information to multipartUpload as a parameter to continue the upload.

```
let OSS = require('ali-oss');
let client = new OSS({
  region: '<Your region>',
  accessKeyId: '<Your AccessKeyId>',
  accessKeySecret: '<Your AccessKeySecret>',
  bucket: '<Your bucket name>',
});
let checkpoint;
async function resumeUpload() {
  // retry 5 times
  for (let i = 0; i < 5; i++) {
    try {
      const result = await client.multipartUpload('object-name', filePath, {
        checkpoint,
        async progress(percentage, cpt) {
          checkpoint = cpt;
        },
      });
      console.log(result);
      break; // break if success
    } catch (e) {
      console.log(e);
    }
  }
}
resumeUpload();
```

In the preceding code, the checkpoint information is recorded as a variable. If the program crashes, the checkpoint information is lost. We recommend that you save the checkpoint information in a file. This way, the program can obtain the checkpoint information from the file after the program restarts.

4.6.8. Upload callback

This topic describes how to use upload callback.

 **Note** For the complete code used to implement upload callback, visit [GitHub](#). For more information about upload callback, see [Upload callback](#) in Object Storage Service (OSS) Developer Guide and [Callback](#) in API Reference.

The following code provides an example on how to use upload callback when you upload a local file named `examplefile.txt` to a bucket named `examplebucket`. After the local file is uploaded, the name of the object in OSS is `exampleobject.txt`.

```

const oss = require('ali-oss');
var path = require('path');
const client = oss({
  // Set yourregion to the endpoint of the region where the bucket is located. For example,
  // if your bucket is located in the China (Hangzhou) region, set yourregion to oss-cn-hangzhou
  .
  region: 'yourregion',
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to
  // log on to OSS because the account has permissions on all API operations. We recommend that
  // you use a RAM user to call API operations or perform routine operations and maintenance. To
  // create a RAM user, log on to the RAM console.
  accessKeyId: 'yourAccessKeyId',
  accessKeySecret: 'yourAccessKeySecret',
  // Specify the bucket name.
  bucket: 'examplebucket'
});
const options = {
  callback: {
    // Set the address of the server to which the callback request is sent. Example: http://
    // oss-demo.aliyuncs.com:23450.
    url: 'http://oss-demo.aliyuncs.com:23450',
    // Optional. Set the value of the Host field included in the callback request header.
    // host: 'yourCallbackHost',
    // Set the value of the body field included in the callback request.
    body: 'bucket=${bucket}&object=${object}&var1=${x:var1}',
    // Set Content-Type for the callback request.
    contentType: 'application/x-www-form-urlencoded',
    // Configure the custom parameters for the callback request.
    customValue: {
      var1: 'value1',
      var2: 'value2'
    }
  }
}
async function put () {
  try {
    // Specify the full paths of the object and the local file. The full path of the object
    // cannot contain bucket names.
    // By default, if the path of the local file is not specified, the local file is upload
    // ed from the path of the project to which the sample program belongs.
    let result = await client.put('exampleobject.txt', path.normalize('/localpath/examplefi
    le.txt'), options);
    console.log(result);
  } catch (e) {
    console.log(e);
  }
}
put();

```

4.7. Download objects

4.7.1. Overview

The OSS Node.js SDK provides the following methods to download an object:

- [Download objects as files](#)
- [Download to local memory](#)
- [Streaming download](#)
- [Range download](#)
- [Conditional download](#)

4.7.2. Download objects as files

This topic describes how to download objects from a bucket.

The following code provides an example on how to download an object named `exampleobject.txt` from the `testfolder` directory in a bucket named `examplebucket` to the `D:\localpath` path in a local disk.

```
let OSS = require('ali-oss');
let client = new OSS({
  // Set yourregion to the endpoint of the region where the bucket is located. For example,
  // if your bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-c
  // n-hangzhou.aliyuncs.com.
  region: 'yourRegion',
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to
  // log on to OSS because the account has permissions on all API operations. We recommend that
  // you use your RAM user's credentials to call API operations or perform routine operations an
  // d maintenance. To create a RAM user, log on to the RAM console.
  accessKeyId: 'yourAccessKeyId',
  accessKeySecret: 'yourAccessKeySecret',
  // Specify the bucket name.
  bucket: 'examplebucket'
});
async function get () {
  try {
    // Specify a name for the object and the full path to where the object is downloaded. T
    // he full path of the object cannot contain bucket names.
    // If a local file with the same name exists, the downloaded object overwrites the loca
    // l file.
    // If the path for the object is not specified, the downloaded object is saved to the p
    // ath of the project to which the sample program belongs.
    let result = await client.get('exampleobject.txt', 'D:\\localpath\\examplefile.txt');
    console.log(result);
  } catch (e) {
    console.log(e);
  }
}
get();
```

4.7.3. Download to local memory

This topic describes how to download an object to local memory.

You can use `get` to download an object from OSS to local memory:

```
let OSS = require('ali-oss');
let client = new OSS({
  region: '<Your region>',
  accessKeyId: '<Your AccessKeyId>',
  accessKeySecret: '<Your AccessKeySecret>',
  bucket: 'Your bucket name'
});
async function getBuffer () {
  try {
    let result = await client.get('object-name');
    console.log(result.content);
  } catch (e) {
    console.log(e);
  }
}
getBuffer();
```

4.7.4. Streaming download

If a large object takes a long time to download, you can use streaming download to download the object in increments until the entire object is downloaded.

The following code provides an example on how to stream download an object named `exampleobject.txt` from a bucket named `examplebucket` to the `D:\localpath` path. After the object is downloaded, the local file is named `examplefile.txt`.

 **Note** When you use `getStream` to download an object, `Readable Stream` is returned and used to stream the object content.

```
const OSS = require('ali-oss');
const fs = require('fs');
const client = new OSS({
  // Set yourRegion to the endpoint of the region where the bucket is located. For example,
  // if your bucket is located in the China (Hangzhou) region, set yourRegion to oss-cn-hangzhou
  //
  region: 'yourRegion',
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to
  // access Object Storage Service (OSS) because the account has permissions on all API operatio
  // ns. We recommend that you use a Resource Access Management (RAM) user to call API operatio
  // ns or perform routine operations and maintenance. To create a RAM user, log on to the RAM co
  // nsole.
  accessKeyId: 'yourAccessKeyId',
  accessKeySecret: 'yourAccessKeySecret',
  // Specify the name of the bucket.
  bucket: 'examplebucket',
});
async function getStream () {
  try {
    // Specify the full path of the object. The full path of the object cannot contain buck
    // et names.
    const result = await client.getStream('exampleobject.txt');
    console.log(result);
    // Specify the full path of the local file. If the specified local file exists, the dow
    // nloaded object replaces the local file. Otherwise, the local file is created.
    // If the path of the local file is not specified, the downloaded object is saved to th
    // e path of the project to which the sample program belongs.
    const writeStream = fs.createWriteStream('D:\\localpath\\examplefile.txt');
    result.stream.pipe(writeStream);
  } catch (e) {
    console.log(e);
  }
}
getStream()
```

4.7.5. Range download

If you need only part of the data in an object, you can use range download to download data within a specified range.

Specify a valid range to download data

If the start and end values of the range that you specify are within the size of the object, the content within the specified range is downloaded. For example, if the object from which you want to download data is 1,000 bytes in size, the valid range is from byte 0 to byte 999.

The following code provides an example on how to specify a valid range to download data:

```
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log on to OSS because the account has permissions on all API operations. We recommend that you use your RAM user's credentials to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
const accessKeyId = "yourAccessKeyId";
const accessKeySecret = "yourAccessKeySecret";
// Specify the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com.
const endpoint = "yourEndpoint";
// Specify the bucket name.
const bucket = "yourBucketName";
async function main() {
    let start = 1, end = 900;
    // yourObjectName indicates the complete path of the object excluding the bucket name. Example: destfolder/examplefile.txt.
    // Obtain the data that is within the range from byte 0 to byte 900, which includes a total of 900 bytes.
    // If the start or end value of the specified range is not within the valid range, the entire object is downloaded.
    let result = await client.get("<yourObjectName>", {
        headers: {
            Range: `bytes=${start}-${end}`,
        },
    });
    console.log(result.content.toString());
};
main();
```

For more information about the naming conventions for buckets, see [bucket](#). For more information about the naming conventions for objects, see [object](#).

Specify compatible behaviors to download data by range

You can add the `x-oss-range-behavior:standard` header in the request to modify the download behavior when the specified range is not within the valid range.

The following code provides an example on how to specify compatible behaviors to download data by range:

```
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log on to OSS because the account has permissions on all API operations. We recommend that you use your RAM user's credentials to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
const accessKeyId = "yourAccessKeyId";
const accessKeySecret = "yourAccessKeySecret";
// Specify the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com.
const endpoint = "yourEndpoint";
// Specify the bucket name.
const bucket = "yourBucketName";
async function main() {
  // Upload an object named exampleobject.txt, which is 10 bytes in size.
  let buf = Buffer.from("abcdefghij");
  await client.put("exampleobject.txt", buf);
  let result = await client.get("exampleobject.txt", {
    // Set Range to bytes=5-15. In this case, the end value of the range is not within the valid range. Therefore, OSS returns the HTTP status code 206 and the content from byte 6 to byte 10.
    headers: {
      Range: "bytes=5-15",
      "x-oss-range-behavior": "standard",
    },
  });
  console.log(result.content.toString() === 'fghij')
  console.log(result.res.status === 206)
  try{
    await client.get("exampleobject.txt", {
      // Set Range to bytes=15-25. In this case, the start value of the range is not within the valid range. Therefore, OSS returns the HTTP status code 416 and the error code InvalidRange.
      headers: {
        Range: "bytes=15-25",
        "x-oss-range-behavior": "standard",
      },
    })
  }catch(e) {
    console.log(e.status === 416);
    console.log(e.name === 'InvalidRangeError')
  }
}
main();
```

4.7.6. Conditional download

You can specify conditions when you download objects. If the specified conditions are met, the objects can be downloaded. If the specified conditions are not met, an error is returned and the objects cannot be downloaded.

Download objects that are modified earlier than the specified time

The following code provides an example on how to download objects that are modified earlier than the specified time:

```
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log on to OSS because the account has permissions on all API operations. We recommend that you use your RAM user's credentials to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
const accessKeyId = "yourAccessKeyId";
const accessKeySecret = "yourAccessKeySecret";
// Specify the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com.
const endpoint = "yourEndpoint";
// Specify the bucket name.
const bucket = "yourBucketName";
async function main() {
  // Upload an object named exampleobject.txt.
  await client.put('exampleobject.txt', Buffer.from('contenttest'))
  // Set the If-Modified-Since header in the request. If the value of this header is earlier than the time when the uploaded object is last modified, the object is downloaded.
  let result = await client.get("exampleobject.txt", {
    headers: {
      "If-Modified-Since": new Date("1970-01-01").toGMTString(),
    },
  });
  console.log(result.content.toString() === "contenttest");
  console.log(result.res.status === 200);
  // If the value of the If-Modified-Since header is equal to or later than the time when the uploaded object is last modified, OSS returns 304 Not Modified.
  result = await client.get("exampleobject.txt", {
    headers: {
      "If-Modified-Since": new Date().toGMTString(),
    },
  });
  console.log(result.content.toString() === "");
  console.log(result.res.status === 304);
  console.log(e.code === "Not Modified");
}
main();
```

For more information about the naming conventions for buckets, see [bucket](#). For more information about the naming conventions for objects, see [object](#).

Download objects that are modified no earlier than the specified time

The following code provides an example on how to download objects that are modified no earlier than the specified time:

```
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log on to OSS because the account has permissions on all API operations. We recommend that you use your RAM user's credentials to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
const accessKeyId = "yourAccessKeyId";
const accessKeySecret = "yourAccessKeySecret";
// Specify the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com.
const endpoint = "yourEndpoint";
// Specify the bucket name.
const bucket = "yourBucketName";
async function main() {
  // Upload an object named exampleobject.txt.
  await client.put('exampleobject.txt', Buffer.from('contenttest'))
  // Set the If-Unmodified-Since header in the request. If the value of this header is equal to or later than the time when the uploaded object is last modified, the object is downloaded.
  let result = await client.get("exampleobject.txt", {
    headers: {
      "If-Unmodified-Since": new Date().toGMTString(),
    },
  });
  console.log(result.content.toString() === "contenttest");
  console.log(result.res.status === 200);
  // If the value of the If-Unmodified-Since header is earlier than the time when the uploaded object is last modified, OSS returns 412 Precondition Failed.
  try {
    await client.get("exampleobject.txt", {
      headers: {
        "If-Unmodified-Since": new Date("1970-01-01").toGMTString(),
      },
    });
  } catch (e) {
    console.log(e.status === 412);
    console.log(e.code === "PreconditionFailed");
  }
  main();
}
```

Download objects whose ETags match the specified ETag values

The ETag of an object can be used to check whether the object content changes. The following code provides an example on how to download objects whose ETags match the specified ETag values:

```
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log on to OSS because the account has permissions on all API operations. We recommend that you use your RAM user's credentials to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
const accessKeyId = "yourAccessKeyId";
const accessKeySecret = "yourAccessKeySecret";
// Specify the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com.
const endpoint = "yourEndpoint";
// Specify the bucket name.
const bucket = "yourBucketName";
async function main() {
    // Upload an object named exampleobject.txt and obtain the ETag of the object.
    let {res: {headers: {etag}}} = await client.put('exampleobject.txt', Buffer.from('contenttest'))
}
// Set the If-Match header to an ETag value. If the specified ETag value matches the ETag of the object, the object is downloaded.
let result = await client.get("exampleobject.txt", {
    headers: {
        "If-Match": etag,
    },
});
console.log(result.content.toString() === "contenttest");
console.log(result.res.status === 200);
// If the specified ETag value does not match the ETag of the object, OSS returns 412 Precondition Failed.
try {
    await client.get("exampleobject.txt", {
        headers: {
            "If-Match": etag,
        },
    });
} catch (e) {
    console.log(e.status === 412);
    console.log(e.code === "PreconditionFailed");
}
main();
```

Download objects whose ETags do not match the specified ETag values

The following code provides an example on how to download objects whose ETags do not match the specified ETag values:

```
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log on to OSS because the account has permissions on all API operations. We recommend that you use your RAM user's credentials to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
const accessKeyId = "yourAccessKeyId";
const accessKeySecret = "yourAccessKeySecret";
// Specify the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com.
const endpoint = "yourEndpoint";
// Specify the bucket name.
const bucket = "yourBucketName";
async function main() {
  // Upload an object named exampleobject.txt and obtain the ETag of the object.
  let {res: {headers: {etag}}} = await client.put('exampleobject.txt', Buffer.from('contenttest'))
}
// Set the If-Match header to an ETag value. If the specified ETag value does not match the ETag of the object, the object is downloaded.
let result = await client.get("exampleobject.txt", {
  headers: {
    "If-None-Match": etag,
  },
});
console.log(result.content.toString() === "contenttest");
console.log(result.res.status === 200);
// If the specified ETag value matches the ETag of the object, OSS returns 304 Not Modified
result = await client.get("exampleobject.txt", {
  headers: {
    "If-None-Match": etag,
  },
});
console.log(result.content.toString() === "");
console.log(result.res.status === 304);
console.log(e.code === "Not Modified");
}
main();
```

4.8. Manage objects

4.8.1. Overview

You can use APIs to perform the following operations on objects in a bucket:

- [Determine whether an object exists](#)
- [Manage Object Meta](#)
- [Manage the ACL of an object](#)
- [List objects](#)
- [Copy an object](#)

- [Delete objects](#)
- [Restore objects](#)

4.8.2. Determine whether an object exists

Before you perform an operation on an object in a bucket, you must determine whether the object exists. This topic describes how to determine whether an object exists.

The following code provides an example on how to determine whether an object exists:

```
const OSS = require('ali-oss');
const client = new OSS({
  bucket: '<Your BucketName>',
  // The endpoint of the China (Hangzhou) region is used in this example. Specify the actual endpoint.
  region: '<Your Region>',
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log on to OSS because the account has permissions on all API operations. We recommend that you use your RAM user's credentials to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
  accessKeyId: '<Your AccessKeyId>',
  accessKeySecret: '<Your AccessKeySecret>'
});
async function isExistObject(name, options = {}) {
  try {
    await client.head(name, options);
    console.log('The object exists.')
  } catch (error) {
    if (error.code === 'NoSuchKey') {
      console.log('The object does not exist.')
    }
  }
}
// Determine whether an object exists in an unversioned bucket.
// Specify the complete name of the object excluding the bucket name. Example: example/test.txt.
const name = '<Your ObjectName>'
isExistObject(name)
// Determine whether an object with a specified version ID exists in a versioned bucket.
const options = {
  // Specify the version ID of the object.
  versionId: '<Your Object VersionId>'
}
isExistObject(name, options)
```

4.8.3. Manage Object Meta

Object Meta contains HTTP headers and User Meta. For more information, see the "[Object Meta](#)" section in OSS Developer Guide.

When you upload a file to OSS, you can specify the information of certain properties in addition to the content. The property information is called Object Meta. Object Meta is stored together with its object during uploads, and returned with the object when the object is downloaded.

Object Meta is stored in HTTP headers during object uploads or downloads. Therefore, Object Meta cannot contain complex characters in accordance with HTTP.

 **Note** Object Meta can only contain simple characters in the ASCII format and cannot contain line breaks. The total size of Object Meta cannot exceed 8 KB.

You can set meta to specify Object Meta when you use `put` , `putStream` , or `multipartUpload` .

```
let OSS = require('ali-oss')
let client = new OSS({
  region: '<Your region>',
  accessKeyId: '<Your AccessKeyId>',
  accessKeySecret: '<Your AccessKeySecret>',
  bucket: 'Your bucket name'
});
async function put () {
  try {
    let result = await client.put('object-name', 'local-file', {
      meta: {
        year: 2016,
        people: 'mary'
      }
    });
    console.log(result);
  } catch (e) {
    console.log(e);
  }
}
put();
```

You can also use `putMeta` to update Object Meta:

```
let OSS = require('ali-oss')
let client = new OSS({
  region: '<Your region>',
  accessKeyId: '<Your AccessKeyId>',
  accessKeySecret: '<Your AccessKeySecret>',
  bucket: 'Your bucket name'
});
async function putMeta () {
  try {
    let result = await client.putMeta('object-name', {
      meta: {
        year: 2015,
        people: 'mary'
      }
    });
    console.log(result);
  } catch (e) {
    console.log(e);
  }
}
putMeta();
```

4.8.4. Manage the ACL of an object

ACL	Description	Value
Inherited from the bucket	The ACL of an object is the same as that of the bucket to which the object belongs.	default
Private	Only the bucket owner and authorized users have the read and write permissions on the object.	private
Public read	Only the bucket owner and authorized users have the read and write permissions on the object. Other users have only read permissions on the object. Exercise caution when you grant this permission.	public-read
Public read/write	All users have read and write permissions on the object. Exercise caution when you grant this permission.	public-read-write

The ACL of objects take precedence over that of buckets. For example, if the ACL of a bucket is private, while the object ACL is public read-write, all users can read and write the object. If an object is not configured with an ACL, its ACL is the same as that of its bucket by default.

Configure the ACL for an object

You can run the following code to configure an ACL for an object:

```
const oss = require('ali-oss');
const store = oss({
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log on to OSS, because the account has permissions on all API operations. We recommend that you log on to the OSS console as a RAM user to call API operations or perform routine operations and maintenance. To create a RAM user, log on to https://ram.console.aliyun.com.
  accessKeyId: '<your access key>',
  accessKeySecret: '<your access secret>',
  bucket: '<your bucket name>',
  // Obtain the region where the current bucket is located.
  region: 'oss-cn-hangzhou'
});
// Configure the ACL of the specified object to public read.
await store.putACL('<object name>', 'public-read');
```

Obtain the ACL of an object

You can run the following code to obtain the ACL for an object:

```
const oss = require('ali-oss');
const store = oss({
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log on to OSS, because the account has permissions on all API operations. We recommend that you log on to the OSS console as a RAM user to call API operations or perform routine operations and maintenance. To create a RAM user, log on to https://ram.console.aliyun.com.
  accessKeyId: '<your access key>',
  accessKeySecret: '<your access secret>',
  bucket: '<your bucket name>',
  // Obtain the region where the current bucket is located.
  region: 'oss-cn-hangzhou'
});
// Obtain the ACL of an object.
const result = await store.getACL('<object name>');
console.log(result.acl);
```

4.8.5. List objects

This topic describes how to list all objects in a bucket, a specified number of objects in a bucket, and objects whose names contain a specified prefix in a bucket.

Methods

You can call the `list` or `listV2` method to list up to 1,000 objects in a bucket at a time. You can configure parameters to list objects in different ways. For example, you can configure parameters to list objects from a specified start position, list objects and subfolders in a specified folder, and list objects by page. The `list` and `listV2` methods are different in the following aspects:

- By default, when you use the `list` method to list objects, the information about the object owners is returned.

- When you use `listV2` to list objects, you must configure the `fetch-owner` parameter to specify whether to include the information about the object owners in the returned results.

 **Note** We recommend that you use the `listV2` method to list objects in versioned buckets.

The following tables describe the parameters that you can configure when you use the `list` and `listV2` methods to list objects.

- Use the `list` method to list objects

The following table describes the parameters that you can configure when you use the `list` method to list objects.

Parameter	Type	Description
<code>prefix</code>	string	The prefix that the names of listed objects must contain.
<code>delimiter</code>	string	The character used to group objects by name.
<code>marker</code>	string	The name of the object from which the list operation begins. If this parameter is specified, objects whose names are alphabetically greater than the marker parameter value are returned.
<code>max-keys</code>	number string	The maximum number of returned objects.
<code>encoding-type</code>	'url' ''	Specifies that the object names in the response are URL-encoded.

- Use the `listV2` method to list objects

The following table describes the parameters that you can configure when you use the `listV2` method to list objects.

Parameter	Type	Description
<code>prefix</code>	string	The prefix that the names of listed objects must contain.
<code>continuation-token</code>	string	The position from which the object list is obtained.
<code>delimiter</code>	string	The character used to group objects by name.
<code>max-keys</code>	number string	The maximum number of returned objects.
<code>start-after</code>	string	The name of the object from which the list operation begins. If this parameter is specified, objects whose names are alphabetically greater than the start-after parameter value are returned.
<code>fetch-owner</code>	boolean	Specifies whether to include the information about object owners in the response.

Parameter	Type	Description
encoding-type	'url' ''	Specifies that the object names in the response are URL-encoded.

Simple list

The following code provides an example on how to list objects in a specified bucket. By default, up to 100 objects are listed by an operation.

- Use the list method to list objects

```
let OSS = require('ali-oss');
let client = new OSS({
  // The endpoint of the China (Hangzhou) region is used in this example. Specify the actual endpoint.
  region: 'yourRegion',
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log on to OSS because the account has permissions on all API operations. We recommend that you use your RAM user's credentials to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
  accessKeyId: 'yourAccessKeyId',
  accessKeySecret: 'yourAccessKeySecret',
  // Specify the bucket name.
  bucket: 'yourBucketName',
});
async function list () {
  // By default, if no parameter is specified, up to 100 objects can be returned.
  let result = await client.list();
  console.log(result);
}
list();
```

- Use the listV2 method to list objects

```
let OSS = require('ali-oss');
let client = new OSS({
  // The endpoint of the China (Hangzhou) region is used in this example. Specify the actual endpoint.
  region: 'yourRegion',
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log on to OSS because the account has permissions on all API operations. We recommend that you use your RAM user's credentials to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
  accessKeyId: 'yourAccessKeyId',
  accessKeySecret: 'yourAccessKeySecret',
  // Specify the bucket name.
  bucket: 'yourBucketName',
});
async function list () {
  // By default, if no parameter is specified, up to 100 objects can be returned.
  let result = await client.listV2();
  console.log(result);
}
list();
```

List a specified number of objects

The following code provides an example on how to list a specified number of objects in a bucket by configuring the max-keys parameter.

- Use the list method to list objects

```
let OSS = require('ali-oss');
let client = new OSS({
  // The endpoint of the China (Hangzhou) region is used in this example. Specify the actual endpoint.
  region: 'yourRegion',
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log on to OSS because the account has permissions on all API operations. We recommend that you use your RAM user's credentials to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
  accessKeyId: 'yourAccessKeyId',
  accessKeySecret: 'yourAccessKeySecret',
  // Specify the bucket name.
  bucket: 'yourBucketName',
});
async function list () {
  let result = await client.list({
    // Specify that the first 10 objects in alphabetical order are returned.
    "max-keys": 10
  });
  console.log(result);
}
list();
```

- Use the listV2 method to list objects

```
let OSS = require('ali-oss');
let client = new OSS({
  // The endpoint of the China (Hangzhou) region is used in this example. Specify the actual endpoint.
  region: 'yourRegion',
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log on to OSS because the account has permissions on all API operations. We recommend that you use your RAM user's credentials to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
  accessKeyId: 'yourAccessKeyId',
  accessKeySecret: 'yourAccessKeySecret',
  // Specify the bucket name.
  bucket: 'yourBucketName',
});
async function list () {
  let result = await client.listV2({
    // Specify that the first 10 objects in alphabetical order are returned.
    "max-keys": 10
  });
  console.log(result);
}
list();
```

List objects whose names contain a specified prefix

The following code provides an example on how to list objects whose names contain a specified prefix in a bucket by configuring the prefix parameter.

- Use the list method to list objects

```
let OSS = require('ali-oss');
let client = new OSS({
  // The endpoint of the China (Hangzhou) region is used in this example. Specify the actual endpoint.
  region: 'yourRegion',
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log on to OSS because the account has permissions on all API operations. We recommend that you use your RAM user's credentials to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
  accessKeyId: 'yourAccessKeyId',
  accessKeySecret: 'yourAccessKeySecret',
  // Specify the bucket name.
  bucket: 'yourBucketName',
});
async function list () {
  let result = await client.list({
    // List 10 objects.
    "max-keys": 10,
    // List objects whose names contain the prefix foo/.
    prefix: 'foo/'
  });
  console.log(result);
}
list();
```


- Use the listV2 method to list objects

```
let OSS = require('ali-oss');
let client = new OSS({
  // The endpoint of the China (Hangzhou) region is used in this example. Specify the actual endpoint.
  region: 'yourRegion',
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log on to OSS because the account has permissions on all API operations. We recommend that you use your RAM user's credentials to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
  accessKeyId: 'yourAccessKeyId',
  accessKeySecret: 'yourAccessKeySecret',
  // Specify the bucket name.
  bucket: 'yourBucketName',
});
async function list () {
  let result = await client.listV2({
    // List 10 objects.
    "max-keys": 10,
    // List objects whose names contain the prefix foo/.
    prefix: 'foo/'
  });
  console.log(result);
}
list();
```

List objects whose names are alphabetically greater than the specified object name

The following code provides an example on how to list objects whose names are alphabetically greater than the string specified by the marker or startAfter parameter.

- Use the list method to list objects

The name of the object from which the list operation begins is specified by the marker parameter.

```
let OSS = require('ali-oss');
let client = new OSS({
  // The endpoint of the China (Hangzhou) region is used in this example. Specify the actual endpoint.
  region: 'yourRegion',
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log on to OSS because the account has permissions on all API operations. We recommend that you use your RAM user's credentials to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
  accessKeyId: 'yourAccessKeyId',
  accessKeySecret: 'yourAccessKeySecret',
  // Specify the bucket name.
  bucket: 'yourBucketName',
});
// List the objects whose names are alphabetically greater than the object name test. By default, up to 100 objects are listed.
const marker = 'test'
async function list () {
  let result = await client.list({
    marker
  });
  console.log(result);
}
list();
```

- Use the listV2 method to list objects

The name of the object from which the list operation begins is specified by the startAfter parameter.

```
let OSS = require('ali-oss');
let client = new OSS({
  // The endpoint of the China (Hangzhou) region is used in this example. Specify the actual endpoint.
  region: 'yourRegion',
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log on to OSS because the account has permissions on all API operations. We recommend that you use your RAM user's credentials to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
  accessKeyId: 'yourAccessKeyId',
  accessKeySecret: 'yourAccessKeySecret',
  // Specify the bucket name.
  bucket: 'yourBucketName',
});
async function list () {
  let result = await client.listV2({
    // List the objects and subfolders whose names are alphabetically greater than a/b in the a/ folder.
    delimiter: '/',
    prefix: 'a/',
    'start-after': 'a/b'
  });
  console.log(result.objects, result.prefixes);
}
list();
```

List all objects by page

The following code provides an example on how to list all objects in a specified bucket by page. You can configure the max-keys parameter to specify the maximum number of objects that can be listed on each page.

- Use the list method to list objects

```
let OSS = require('ali-oss');
let client = new OSS({
  // The endpoint of the China (Hangzhou) region is used in this example. Specify the actual endpoint.
  region: 'yourRegion',
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log on to OSS because the account has permissions on all API operations. We recommend that you use your RAM user's credentials to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
  accessKeyId: 'yourAccessKeyId',
  accessKeySecret: 'yourAccessKeySecret',
  // Specify the bucket name.
  bucket: 'yourBucketName',
});
let marker = null;
// List up to 20 objects on each page.
const maxKeys = 20;
async function list () {
  do {
    let result = await client.list({
      marker,
      'max-keys': maxKeys
    });
    marker = result.nextMarker;
    console.log(result);
  }while(marker)
}
list();
```

- Use the listV2 method to list objects

```
let OSS = require('ali-oss');
let client = new OSS({
  // The endpoint of the China (Hangzhou) region is used in this example. Specify the actual endpoint.
  region: 'yourRegion',
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log on to OSS because the account has permissions on all API operations. We recommend that you use your RAM user's credentials to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
  accessKeyId: 'yourAccessKeyId',
  accessKeySecret: 'yourAccessKeySecret',
  // Specify the bucket name.
  bucket: 'yourBucketName',
});
async function list () {
  let continuationToken = null;
  // List up to 20 objects on each page.
  const maxKeys = 20;
  do {
    let result = await client.listV2({
      'continuation-token': continuationToken,
      'max-keys': maxKeys
    });
    continuationToken = result.nextContinuationToken;
    console.log(result);
  }while(continuationToken)
}
list();
```

List objects and their owner information

The following code provides an example on how to list objects and their owner information:

```
let OSS = require('ali-oss');
let client = new OSS({
  // The endpoint of the China (Hangzhou) region is used in this example. Specify the actual endpoint.
  region: 'yourRegion',
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log on to OSS because the account has permissions on all API operations. We recommend that you use your RAM user's credentials to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
  accessKeyId: 'yourAccessKeyId',
  accessKeySecret: 'yourAccessKeySecret',
  // Specify the bucket name.
  bucket: 'yourBucketName',
});
// By default, the information about the object owners is not listed. To include the object owner information in the returned results, you must set the fetch-owner parameter to true.
async function list () {
  let result = await client.listV2({
    'fetch-owner': true
  });
  console.log(result.objects);
}
list();
```

List objects and subfolders in a specified folder

OSS does not use a hierarchical structure for objects, but instead uses a flat structure. All elements are stored as objects in buckets. A folder is an object whose size is 0 and whose name ends with a forward slash (/). You can upload and download this object. By default, objects whose names end with a forward slash (/) is displayed as a folder in the OSS console.

You can use the `delimiter` and `prefix` parameters to list objects by folder.

- If you set `prefix` to a folder name in the request, objects and subfolders whose names contain the prefix are listed.
- If you also set `delimiter` to a forward slash (/) in the request, the objects and subfolders whose names start with the specified prefix in the folder are listed. Each subfolder is listed as a single result element in `commonPrefixes`. The objects and folders in these subfolders are not listed.

The following objects are stored in the bucket:

```
foo/x
foo/y
foo/bar/a
foo/bar/b
foo/hello/C/1
foo/hello/C/2
...
foo/hello/C/9999
```

The following code provides an example on how to use the `list` or `listV2` method to list objects and subfolders in a specified folder.

- Use the `list` method to list objects

```
let OSS = require('ali-oss');
let client = new OSS({
  // The endpoint of the China (Hangzhou) region is used in this example. Specify the actual endpoint.
  region: 'yourRegion',
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log on to OSS because the account has permissions on all API operations. We recommend that you use your RAM user's credentials to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
  accessKeyId: 'yourAccessKeyId',
  accessKeySecret: 'yourAccessKeySecret',
  // Specify the bucket name.
  bucket: 'yourBucketName',
});
// Call the listDir function and configure different prefixes to list the required objects.
async function listDir(dir) {
  const result = await client.list({
    prefix: dir,
    // Set the delimiter to a forward slash (/).
    delimiter: '/'
  });
  result.prefixes.forEach(subDir => {
    console.log('SubDir: %s', subDir);
  });
  result.objects.forEach(obj => {
    console.log('Object: %s', obj.name);
  });
}
listDir('foo/');
// SubDir: foo/bar/
// SubDir: foo/hello/
// Object: foo/x
// Object: foo/y
listDir('foo/bar/');
// Object: foo/bar/a
// Object: foo/bar/b
listDir('foo/hello/C/');
// Object: foo/hello/C/1
// Object: foo/hello/C/2
// ...
// Object: foo/hello/C/9999
```

- Use the listV2 method

```

let OSS = require('ali-oss');
let client = new OSS({
  // The endpoint of the China (Hangzhou) region is used in this example. Specify the actual endpoint.
  region: 'yourRegion',
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log on to OSS because the account has permissions on all API operations. We recommend that you use your RAM user's credentials to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
  accessKeyId: 'yourAccessKeyId',
  accessKeySecret: 'yourAccessKeySecret',
  // Specify the bucket name.
  bucket: 'yourBucketName',
});
// Call the listV2Dir function and configure different prefixes to list the required objects.
async function listV2Dir(dir) {
  const result = await client.listV2({
    prefix: dir,
    // Set the delimiter to a forward slash (/).
    delimiter: '/'
  });
  result.prefixes.forEach(subDir => {
    console.log('SubDir: %s', subDir);
  });
  result.objects.forEach(obj => {
    console.log('Object: %s', obj.name);
  });
}
listDir('foo/');
// SubDir: foo/bar/
// SubDir: foo/hello/
// Object: foo/x
// Object: foo/y
listDir('foo/bar/');
// Object: foo/bar/a
// Object: foo/bar/b
listDir('foo/hello/C/');
// Object: foo/hello/C/1
// Object: foo/hello/C/2
// ...
// Object: foo/hello/C/9999

```

4.8.6. Copy objects

Copy a small object

You can use the copy method to copy objects within the same bucket or between different buckets in the same region. The object size cannot exceed 1 GB.

- Copy objects within the same bucket

The following code provides an example on how to copy objects within the same bucket:

```

const OSS = require('ali-oss');
const client = new OSS({
  // Set yourRegion to the region in which the bucket is located. For example, if your bucket is located in the China (Hangzhou) region, set yourRegion to oss-cn-hangzhou.
  region: 'yourRegion',
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access Object Storage Service (OSS) because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
  accessKeyId: 'yourAccessKeyId',
  accessKeySecret: 'yourAccessKeySecret',
  // Specify the bucket name. Example: examplebucket.
  bucket: 'examplebucket',
  // Specify whether to enable HTTPS. If you set secure to true, HTTPS is enabled.
  // secure: true
})
// Copy the objects within the same bucket.
// Specify the full paths of the source object and the destination object. The full paths of the objects cannot contain the bucket name.
client.copy('destexampleobject.txt', 'srcexampleobject.txt',
// Specify HTTP headers and metadata for the destination object.
//{
  // Configure the headers parameter to specify HTTP headers for the destination object. If you do not configure the headers parameter, the HTTP headers of the destination object are the same as the HTTP headers of the source object. The HTTP headers of the source object are copied.
  // headers:{
    // 'Cache-Control': 'no-cache',
    // If the ETag value of the source object is the same as the ETag value that is specified in the request, OSS copies the object and returns 200 OK.
    // 'if-match': '5B3C1A2E053D763E1B002CC607C5****',
    // If the ETag value of the source object is different from the ETag value that is specified in the request, OSS copies the object and returns 200 OK.
    // 'if-none-match': '5B3C1A2E053D763E1B002CC607C5****',
    // If the time that is specified in the request is earlier than the time when the object is modified, OSS copies the object and returns 200 OK.
    // 'if-modified-since': '2021-12-09T07:01:56.000Z',
    // If the time that is specified in the request is later than the time when the object is modified, OSS copies the object and returns 200 OK.
    // 'if-unmodified-since': '2021-12-09T07:01:56.000Z',
    // Specify the access control list (ACL) of the destination object. In this example, the ACL is set to private, which indicates that only the object owner and authorized users have read and write permissions on the object.
    // 'x-oss-object-acl': 'private',
    // Specify the tag for the destination object. You can specify multiple tags for the destination object at the same time.
    // 'x-oss-tagging': 'Tag1=1&Tag2=2',
    // Specify whether the CopyObject operation overwrites the object with the same name. If you set x-oss-forbid-overwrite to true, the CopyObject operation cannot overwrite the object with the same name.
    // 'x-oss-forbid-overwrite': 'true',
  //},
  // Configure the meta parameter to specify metadata for the destination object. If you do not configure the meta parameter, the metadata of the destination object is the same

```



```

as the metadata of the source object. The metadata of the source object is copied.
    //meta:{
        // location: 'hangzhou',
        // year: 2015,
        // people: 'mary'
    //}
//}
).then((res) => {
    console.log(res);
}).catch(e => {
    console.log(e);
})

```

- Copy objects between different buckets in the same region

The following code provides an example on how to copy objects between two different buckets in the same region:

```

const OSS = require('ali-oss');
const client = new OSS({
    // Set yourRegion to the region in which the bucket is located. For example, if your bucket
    // is located in the China (Hangzhou) region, set yourRegion to oss-cn-hangzhou.
    region: 'yourRegion',
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to
    // access OSS because the account has permissions on all API operations. We recommend that
    // you use a RAM user to call API operations or perform routine operations and maintenance.
    // To create a RAM user, log on to the RAM console.
    accessKeyId: 'yourAccessKeyId',
    accessKeySecret: 'yourAccessKeySecret',
    // Specify the name of the destination bucket.
    bucket: 'destexamplebucket',
});
// Specify destobject.txt as the name of the destination object and srcobject.txt as the
// name of the source object, and specify the bucket in which the source object is stored.
client.copy('destobject.txt', 'srcobject.txt', 'srcbucket', {
    // Configure the headers parameter to specify HTTP headers for the destination object.
    // If you do not configure the headers parameter, the HTTP headers of the destination object
    // are the same as the HTTP headers of the source object. The HTTP headers of the source
    // object are copied.
    // headers:{
    //     // 'Cache-Control': 'no-cache',
    // },
    // Configure the meta parameter to specify metadata for the destination object. If you
    // do not configure the meta parameter, the metadata of the destination object is the same
    // as the metadata of the source object. The metadata of the source object is copied.
    // meta:{
    //     // location: 'hangzhou',
    //     // year: 2015,
    //     // people: 'mary'
    // }
})
.then(r => console.log(r));

```

Copy a large object

You can use the `multipartUploadCopy` method to copy an object whose size is greater than 1 GB in parts.

The following code provides an example on how to copy large objects in parts between two different buckets in the same region:

```
const OSS = require('ali-oss');
const store = new OSS({
  // Set yourRegion to the region in which the bucket is located. For example, if your bucket is located in the China (Hangzhou) region, set yourRegion to oss-cn-hangzhou.
  region: 'yourRegion',
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
  accessKeyId: 'yourAccessKeyId',
  accessKeySecret: 'yourAccessKeySecret',
  // Specify the name of the destination bucket.
  bucket: 'destexamplebucket',
});
const copyheaders = {
  // If the ETag value of the source object is the same as the ETag value that is specified in the request, OSS copies the object. Otherwise, OSS returns the HTTP 412 status code (PreconditionFailed).
  'x-oss-copy-source-if-match': '5B3C1A2E053D763E1B002CC607C5****',
  // If the ETag value of the source object is different from the ETag value that is specified in the request, OSS copies the object and returns 200 OK. Otherwise, OSS returns the HTTP 304 status code (NotModified).
  'x-oss-copy-source-if-none-match': '5B3C1A2E053D763E1B002CC607C5****',
  // If the time that is specified in the request is later than or the same as the time when the object is modified, OSS copies the object and returns 200 OK. Otherwise, OSS returns the HTTP 412 status code (PreconditionFailed).
  'x-oss-copy-source-if-unmodified-since': '2022-12-09T07:01:56.000Z',
  // If the time that is specified in the request is earlier than the time when the object is modified, OSS copies the object and returns 200 OK. Otherwise, OSS returns the HTTP 304 status code (NotModified).
  'x-oss-copy-source-if-modified-since': '2022-12-09T07:01:56.000Z'
}
const headers = {
  // Specify the caching behavior of the web page when the object is downloaded.
  'Cache-Control': 'no-cache',
  // Specify the name of the object when the object is downloaded.
  'Content-Disposition': 'somename',
  // Specify the content encoding format of the object when the object is downloaded.
  'Content-Encoding': 'utf-8',
  // Specify the validity period of the request. Unit: milliseconds.
  'Expires': '1000'
}
let savedCpt;
// Specify the full path of the destination object. Example: destexampleobject.txt. The path cannot contain the bucket name.
store.multipartUploadCopy('destexampleobject.txt'
, {
  // Specify the full path of the source object. Example: srcexampleobject.txt. The path cannot contain the bucket name.
```

```

// Do not contain the bucket name.
    sourceKey: 'srcexampleobject.txt',
    // Specify the name of the source bucket. Example: sourcebucket.
    sourceBucketName: 'sourcebucket'
  }).then(r=>console.log(r)).catch(e=>console.log(e))
// Specify the full path of the destination object. Example: destexampleobject.txt. The path cannot contain the bucket name.
store.multipartUploadCopy('destexampleobject.txt'
, {
  // Specify the full path of the source object. Example: srcexampleobject.txt. The path cannot contain the bucket name.
  sourceKey: 'srcexampleobject.txt',
  // Specify the name of the source bucket. Example: sourcebucket.
  sourceBucketName: 'sourcebucket'
});
// Specify the maximum number of parts that can be uploaded in parallel.
parallel: 4,
// Specify the available size for each part.
partSize: 1024 * 1024,
progress: function (p, cpt, res) {
  console.log(p);
  savedCpt = cpt;
  console.log(cpt);
  console.log(res.headers['x-oss-request-id']);
},
//headers,
//copyheaders,
}).then(r=>console.log(r)).catch(e=>console.log(e))
// Specify the full path of the destination object. Example: destexampleobject.txt. The path cannot contain the bucket name.
store.multipartUploadCopy('destexampleobject.txt'
, {
  // Specify the full path of the source object. Example: srcexampleobject.txt. The path cannot contain the bucket name.
  sourceKey: 'srcexampleobject.txt',
  // Specify the name of the source bucket. Example: sourcebucket.
  sourceBucketName: 'sourcebucket'
});
checkpoint: savedCpt,
progress: function (p, cpt, res) {
  console.log(p);
  console.log(cpt);
  console.log(res.headers['x-oss-request-id']);
}
});

```

References

- Copy a small object
 - For more information about the complete sample code that is used to copy a small object, visit [GitHub](#).
 - For more information about the API operation that you can call to copy a small object, see [CopyObject](#).

- Copy a large object
 - For more information about the complete sample code that is used to copy a large object, visit [GitHub](#).
 - For more information about the API operation that you can call to copy a large object, see [UploadPartCopy](#).

4.8.7. Disable overwrite for objects with the same name

By default, if you upload an object that has the same name as an existing object, the existing object is overwritten by the uploaded object. This topic describes how to configure the `x-oss-forbid-overwrite` request header to prevent objects from being overwritten by objects with the same names when you copy objects or perform simple upload or multipart upload.

Simple upload

The following code provides an example on how to prevent objects from being overwritten by objects that have the same names in simple upload:

```
const OSS = require("ali-oss")
const client = new OSS({
  region: '<Your region>',
  accessKeyId: '<Your AccessKeyId>',
  accessKeySecret: '<Your AccessKeySecret>',
  bucket: '<Your bucket name>',
});

async function put () {
  try {
    // Set object-name to the object name such as file.text or the directory such as abc/test/file.text to upload the object to the current bucket or the specified directory.
    // Specify whether to overwrite the object with the same name when you call the PutObject operation.
    // By default, if x-oss-forbid-overwrite is not specified, the object with the same name is overwritten.
    // If x-oss-forbid-overwrite is set to false, the object with the same name is overwritten.
    // If x-oss-forbid-overwrite is set to true, the object with the same name is not overwritten. If an object with the same name already exists, an error is reported.
    let result = await client.put('object-name', 'local-file',{headers: { 'x-oss-forbid-overwrite': true }});
    console.log(result);
  } catch (e) {
    console.log(e);
  }
}

put();
```

Copy objects

- Copy a small object

The following code provides an example on how to copy a small object without overwriting the object with the same object name:

```
const OSS = require("ali-oss")
const client = new OSS({
  region: '<Your region>',
  accessKeyId: '<Your AccessKeyId>',
  accessKeySecret: '<Your AccessKeySecret>',
  bucket: '<Your bucket name>',
});
// Copy objects from the same bucket.
// Specify whether to overwrite the object with the same name.
// By default, if x-oss-forbid-overwrite is not specified, the object with the same name
// is overwritten.
// If x-oss-forbid-overwrite is set to false, the object with the same name is overwritten.
// If x-oss-forbid-overwrite is set to true, the object with the same name is not overwritten.
// If an object with the same name already exists, an error is reported.
client
  .copy("test_copy", "test", { headers: { "x-oss-forbid-overwrite": true } })
  .then(res => {
    console.log(res);
  })
  .catch(e => {
    console.log(e);
  });
```

- Copy a large object

The following code provides an example on how to prevent an existing object from being overwritten by a large object with the same name when you copy the large object by using multipart copy:

```
const client = new OSS({
  region: '<Your region>',
  accessKeyId: '<Your AccessKeyId>',
  accessKeySecret: '<Your AccessKeySecret>',
  bucket: '<Your bucket name>',
});
async function put () {
  try {
    let result = await client.multipartUploadCopy(
      copyName,
      {
        sourceKey: objectkey, // Specify the name of the source object to copy.
        sourceBucketName: bucket // Specify the name of the source bucket.
      },
      {
        // Specify whether to overwrite the object with the same name.
        // By default, if x-oss-forbid-overwrite is not specified, the object with the same name is overwritten.
        // If x-oss-forbid-overwrite is set to false, the object with the same name is overwritten.
        // If x-oss-forbid-overwrite is set to true, the object with the same name is not overwritten. If an object with the same name already exists, an error is reported.
        headers: { "x-oss-forbid-overwrite": true }
      }
    );
    console.log(result);
  } catch (e) {
    console.log(e);
  }
}
```

Multipart upload

The following code provides an example on how to prevent objects from being overwritten by newly uploaded objects that have the same names in multipart upload:

```
const OSS = require("ali-oss")
const client = new OSS({
  region: '<Your region>',
  accessKeyId: '<Your AccessKeyId>',
  accessKeySecret: '<Your AccessKeySecret>',
  bucket: '<Your bucket name>',
});
// Specify whether to overwrite the object with the same name.
// By default, if x-oss-forbid-overwrite is not specified, the object with the same name is
// overwritten.
// If x-oss-forbid-overwrite is set to false, the object with the same name is overwritten.
// If x-oss-forbid-overwrite is set to true, the object with the same name is not overwritt
// en. If an object with the same name already exists, an error is reported.
client
  .multipartUpload('object-name', 'local-file', { headers: { "x-oss-forbid-overwrite": true
} })
  .then(res => {
    console.log(res);
  })
  .catch(e => {
    console.log(e);
  });
```

4.8.8. Delete objects

You can delete a single object, multiple specified objects, objects whose names contain a specified prefix, or a specified directory and all objects in the directory.



Warning Deleted objects cannot be recovered. Exercise caution when you delete objects.

Delete a single object

The following code provides an example on how to delete an object named *exampleobject.txt* from a bucket named *examplebucket*:

```
const OSS = require('ali-oss')
const client = new OSS({
  // Specify the region in which the bucket is located. For example, if your bucket is located in the China (Hangzhou) region, set the region to oss-cn-hangzhou.
  region: 'yourRegion',
  // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in Object Storage Service (OSS) is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
  accessKeyId: 'yourAccessKeyId',
  accessKeySecret: 'yourAccessKeySecret',
  // Specify the bucket name.
  bucket: 'examplebucket',
});
async function deleteFile () {
  try {
    // Specify the full path of the object. The full path of the object cannot contain the bucket name.
    let result = await client.delete('exampleobject.txt');
    console.log(result);
  } catch (e) {
    console.log(e);
  }
}
deleteFile();
```

Delete multiple objects

You can delete up to 1,000 objects at a time. You can delete multiple specified objects, objects whose names contain a specified prefix, or a specified directory and all objects in the directory.

When you delete multiple objects, the result can be returned in one of the following modes:

- **verbose**: The response includes a message body that contains the information about all deleted objects. By default, this mode is used.
- **quiet**: The response does not include a message body.

You can also configure lifecycle rules to automatically delete objects. For more information, see [Lifecycle rules based on the last modified time](#).

Delete multiple specified objects

The following code provides an example on how to delete multiple specified objects:


```
const OSS = require('ali-oss')
const client = new OSS({
  // Specify the region in which the bucket is located. For example, if your bucket is located in the China (Hangzhou) region, set the region to oss-cn-hangzhou.
  region: 'yourRegion',
  // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
  accessKeyId: 'yourAccessKeyId',
  accessKeySecret: 'yourAccessKeySecret',
  // Specify the bucket name.
  bucket: 'examplebucket',
});
async function deleteMulti () {
  try {
    // Specify the full paths of the objects to delete and set the return mode to verbose. The full paths of the objects cannot contain the bucket name.
    //let result = await client.deleteMulti(['exampleobject-1', 'exampleobject-2', 'testfolder/sampleobject.txt']);
    //console.log(result);
    // Specify the full paths of the objects to delete and set the return mode to quiet. The full paths of the objects cannot contain the bucket name.
    let result = await client.deleteMulti(['exampleobject-1', 'exampleobject-2', 'testfolder/sampleobject.txt'], {quiet: true});
    console.log(result);
  } catch (e) {
    console.log(e);
  }
}
deleteMulti();
```

Delete multiple objects with a specified prefix or in a specified directory

The following code provides an example on how to delete multiple objects with a specified prefix and how to delete a specified directory and all objects in the directory.



Warning If the prefix is not specified or is set to NULL in the following code, all objects in the bucket are deleted. Exercise caution when you configure the prefix.

```
const OSS = require('ali-oss');
const client = new OSS({
  // Specify the region in which the bucket is located. For example, if your bucket is located in the China (Hangzhou) region, set the region to oss-cn-hangzhou.
  region: 'yourRegion',
  // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
  accessKeyId: 'yourAccessKeyId',
  accessKeySecret: 'yourAccessKeySecret',
  // Specify the bucket name.
  bucket: 'examplebucket',
});
// Handle request failures, prevent the interruption of promise.all, and return failure causes and the names of objects that fail to be deleted.
async function handleDel(name, options) {
  try {
    await client.delete(name);
  } catch (error) {
    error.failObjectName = name;
    return error;
  }
}
// Delete multiple objects.
async function deletePrefix(prefix) {
  const list = await client.list({
    prefix: prefix,
  });
  list.objects = list.objects || [];
  const result = await Promise.all(list.objects.map((v) => handleDel(v.name)));
  console.log(result);
}
// Specify the prefix of the objects that you want to delete. If you want to delete all objects whose names contain the src prefix, set the prefix to src. After you set the prefix to src, all non-directory objects whose names contain the src prefix, the src directory, and all objects in the directory are deleted.
deletePrefix('src')
// If you want to delete only the src directory and all objects in the directory, set the prefix to src/.
// deletePrefix('src/')
```

References

- For the complete sample code that is used to delete a single object or multiple objects, visit [GitHub](#).
- For more information about the API operation that you can call to delete a single object, see [DeleteObject](#).
- For more information about the API operation that you can call to delete multiple objects, see [DeleteMultipleObjects](#).

4.8.9. Convert the storage classes of objects

OSS provides the following storage classes to cover various data storage scenarios from hot data to cold data: Standard, Infrequent Access (IA), Archive and Cold Archive. This topic describes how to convert the storage class of an object.

For more information about these storage classes, see [Overview](#) and [Convert storage classes](#) in the OSS Developer Guide.

The following section provides examples on how to convert the storage class of an object.

- The following code provides an example on how to convert the storage class of an object from Standard or Infrequent Access (IA) to Archive:

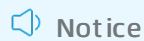
```
let OSS = require('ali-oss');
let client = new OSS({
  bucket: '<your bucket>',
  region: '<your region>',
  accessKeyId: '<your accessKeyId>',
  accessKeySecret: '<your accessKeySecret>'
})
var options = {
  headers: {'x-oss-storage-class': 'Archive'}
}
client.copy('Objectname', 'Objectname', options).then((res) => {
  console.log(res);
}).catch(err => {
  console.log(err)
})
```

- The following code provides an example on how to convert the storage class of an object from Archive to IA or Standard:

```
let OSS = require('ali-oss');
let client = new OSS({
  bucket: '<your bucket>',
  region: '<your region>',
  accessKeyId: '<your accessKeyId>',
  accessKeySecret: '<your accessKeySecret>'
})
// The following code provides an example on how to convert the storage class of an object from Archive to IA:
var options = {
  headers: {'x-oss-storage-class': 'IA'}
}
client.copy('Objectname', 'Objectname', options).then((res) => {
  console.log(res);
}).catch(err => {
  console.log(err)
})
```

For more information about the storage classes and storage fees, see the [Billing overview](#) section in Billing items.

4.8.10. Restore objects

**Notice**

Restoration

If you want to read an Archive object or a Cold Archive object, you must restore the object in advance. It takes several minutes to restore an Archive object and takes several hours to restore a Cold Archive object.

The following section describes the status of an Archive object or a Cold Archive object throughout the restoration process:

1. By default, an Archive or a Cold Archive object is in the frozen state before restoration.
2. After you submit a restore request, the object is in the restoring state.
3. After the server completes the restore task, the object enters the restored state and you can read the object.

- Archive objects

For Archive objects, the restored state lasts 24 hours by default. During the 24 hours, if you call `RestoreObject` again, the restored state is extended by 24 hours. You can extend the restored state to seven days by calling `RestoreObject` six times during the first 24 hours of restored state. You can also configure the duration of the restored state in days by calling `RestoreObject` once. You can specify a duration of at most seven days.

- Cold Archive objects

For Cold Archive objects, you can specify the duration of restored state and restoration priority. The duration of the restored state must be at least one day and at most 365 days. The time required to restore a Cold Archive object to the readable state is determined based on the restoration priority of the object:

- Expedited: The object is restored within one hour.
- Standard: The object is restored within two to five hours. If the `JobParameters` element is not passed in, the default restoration priority is Standard.
- Bulk: The object is restored within five to twelve hours.

4. After the restored state expires, the object returns to the frozen state.

Restore Archive objects

The following code provides an example on how to restore an Archive object:

```
const OSS = require('ali-oss')
const client = new OSS({
  // Set yourRegion to the endpoint of the region in which the bucket is located. For example, if your bucket is located in the China (Hangzhou) region, set yourRegion to oss-cn-hangzhou.
  region: 'yourRegion',
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
  accessKeyId: 'yourAccessKeyId',
  accessKeySecret: 'yourAccessKeySecret',
  // Specify the name of the bucket. Example: examplebucket.
  bucket: 'examplebucket',
});
// Specify the name of the object. Example: exampleobject.txt.
client.restore('exampleobject.txt').then((res) => {
  console.log(res);
}).catch(err => {
  console.log(err);
})
```

Restore a Cold Archive object

The following code provides an example on how to restore a Cold Archive object:

```
const OSS = require('ali-oss')
const client = new OSS({
  // Set yourRegion to the endpoint of the region in which the bucket is located. For example, if your bucket is located in the China (Hangzhou) region, set yourRegion to oss-cn-hangzhou.
  region: 'yourRegion',
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
  accessKeyId: 'yourAccessKeyId',
  accessKeySecret: 'yourAccessKeySecret',
  // Specify the name of the bucket. Example: examplebucket.
  bucket: 'examplebucket',
});
// Specify the name of the object. Example: exampleobject.txt.
client.restore('exampleobject.txt', {type: 'ColdArchive'})
  .then(res => console.log(res))
  .catch(e => console.log(e))
// Set type to the Cold Archive storage class.
// Set JobParameters to the restoration priority of the object.
// Expedited: The object is restored within one hour.
// Standard: The object is restored within two to five hours. If the JobParameters element is not passed in, the default restoration priority is Standard.
// Bulk: The object is restored within five to twelve hours.
// Days is used to specify the duration in which the object can remain in the restored state. Unit: days Valid values: 1 to 7.
//const result = await client.restore('exampleobject.txt', {type: 'ColdArchive', JobParameters: 'Standard', Days: 2})
//.then(r => console.log(r))
```

References

- For more information about the complete sample code that is used to restore Archive or Cold Archive objects, visit [GitHub](#).
- For more information about the API operation that is called to restore Archive or Cold Archive objects, see [RestoreObject](#).

4.8.11. Manage symbolic links

Symbolic links can be used to access objects that are frequently used in buckets. After you create a symbolic link for an object, you can use the symbolic link to access the object.

Create a symbolic link

The following code provides an example on how to create a symbolic link:

```
const OSS = require('ali-oss')
const client = new OSS({
  // Set yourRegion to the region in which the bucket is located. For example, if your bucket is located in the China (Hangzhou) region, set yourRegion to oss-cn-hangzhou.
  region: 'yourRegion',
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access Object Storage Service (OSS) because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
  accessKeyId: 'yourAccessKeyId',
  accessKeySecret: 'yourAccessKeySecret',
  // Specify the bucket name.
  bucket: 'examplebucket',
});
const headers = {
  // Specify the storage class of the object.
  'x-oss-storage-class': 'Standard',
  // Specify the access control list (ACL) of the object.
  'x-oss-object-acl': 'private',
  // Specify whether to overwrite the object with the same name when you create a symbolic link. If you set x-oss-forbid-overwrite to true, the object with the same name cannot be overwritten.
  'x-oss-forbid-overwrite': 'true '
};
async function put () {
  try {
    // Specify symlinkobject.txt as the name of the symbolic link and exampleobject.txt as the name of the object to which the symbolic link points.
    const result = await client.putSymlink('symlinkobject.txt', 'exampleobject.txt'
    // ,{ headers }
    );
    console.log(result);
  } catch (e) {
    console.log(e);
  }
}
put();
```

Query the name of the object to which the symbolic link points

To query a symbolic link, you must have read permissions on the symbolic link. The following code provides an example on how to query the name of the object to which the symbolic link points:

```
const OSS = require('ali-oss')
const client = new OSS({
  // Set yourRegion to the region in which the bucket is located. For example, if your bucket is located in the China (Hangzhou) region, set yourRegion to oss-cn-hangzhou.
  region: 'yourRegion',
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
  accessKeyId: 'yourAccessKeyId',
  accessKeySecret: 'yourAccessKeySecret',
  // Specify the bucket name.
  bucket: 'examplebucket',
});
// Specify the name of the symbolic link and query the name of the object to which the symbolic link points.
store.getSymlink('symlinkobject.txt').then(r=>console.log(r))
```

References

- For more information about the API operation that you can call to create a symbolic link, see [Put Symlink](#).
- For more information about the API operation that you can call to query a symbolic link, see [Get Symlink](#).

4.9. Configure versioning

4.9.1. Versioning

The versioning state of a bucket applies to all objects in the bucket. If you enable versioning for a bucket, you can recover any previous version of an object in the bucket when you accidentally overwrite or delete the object.

Versioning allows you to restore objects in a bucket to any previous point in time, and protects your data from being accidentally overwritten or deleted. A bucket can be in any one of the following versioning states: unversioned (default), versioning-enabled, or versioning-suspended. For more information about versioning, see [Overview](#) in OSS Developer Guide.

 **Note** Versioning is supported in OSS SDK for Node.js **6.8.0** or later.

Configure versioning for a bucket

The following code provides an example on how to set the versioning state of a bucket to Enabled or Suspended:


```
const OSS = require('ali-oss');
const client = new OSS({
  bucket: '<Your BucketName>',
  // The endpoint of the China (Hangzhou) region is used in this example. Specify the actual endpoint.
  region: '<Your Region>',
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log on to OSS because the account has permissions on all API operations. We recommend that you use your RAM user's credentials to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
  accessKeyId: '<Your AccessKeyId>',
  accessKeySecret: '<Your AccessKeySecret>'
});
async function putBucketVersioning() {
  // Set the versioning state of the bucket to Enabled or Suspended.
  const status = 'Enabled'; // `Enabled` or `Suspended`
  const result = await client.putBucketVersioning('<Bucket Name>', status);
  console.log(result);
}
putBucketVersioning();
```

For more information about how to configure versioning for a bucket, see [PutBucketVersioning](#).

Obtain the versioning state of a bucket

The following code provides an example on how to query information about the versioning state of a bucket:

```
const OSS = require('ali-oss');
const client = new OSS({
  bucket: '<Your BucketName>',
  // The endpoint of the China (Hangzhou) region is used in this example. Specify the actual endpoint.
  region: '<Your Region>',
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log on to OSS because the account has permissions on all API operations. We recommend that you use your RAM user's credentials to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
  accessKeyId: '<Your AccessKeyId>',
  accessKeySecret: '<Your AccessKeySecret>'
});
async function getBucketVersioning() {
  // Obtain the versioning state of the bucket.
  const result = await client.getBucketVersioning('Bucket Name');
  console.log(result.versionStatus);
}
getBucketVersioning();
```

For more information about how to obtain the versioning state of a bucket, see [GetBucketVersioning](#).

List the versions of all objects in a bucket

The following code provides an example on how to list the version information of all objects in a specified bucket, including delete markers:

```
const OSS = require('ali-oss');
const client = new OSS({
  bucket: '<Your BucketName>',
  // The endpoint of the China (Hangzhou) region is used in this example. Specify the actual endpoint.
  region: '<Your Region>',
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log on to OSS because the account has permissions on all API operations. We recommend that you use your RAM user's credentials to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
  accessKeyId: '<Your AccessKeyId>',
  accessKeySecret: '<Your AccessKeySecret>'
});
async function getBucketVersions() {
  // List the versions of all objects including delete markers.
  const result = await client.getBucketVersions();
  console.log(result.objects);
  console.log(result.deleteMarker);
}
getBucketVersions();
```

For more information about how to list the versions of all objects in a bucket, including delete markers, see [GetBucketVersions\(ListObjectVersions\)](#).

4.9.2. Upload objects

Simple upload

When you upload an object to a versioning-enabled bucket, OSS automatically adds a unique version ID for the uploaded object, and returns the version ID as the `x-oss-version-id` field in the response header. In a versioning-suspended bucket, the version ID added to an upload object is null. If you upload an object with the same name as an existing object to a versioning-suspended bucket, the existing object is overwritten. In this case, each object only has one version of which the ID is null.

The following code provides an example for simple upload:

```
const OSS = require('ali-oss');
const client = new OSS({
  bucket: '<Your BucketName>',
  // The endpoint of the China (Hangzhou) region is used in this example. Specify the actual endpoint.
  region: '<Your Region>',
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log on to OSS because the account has permissions on all API operations. We recommend that you use your RAM user's credentials to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
  accessKeyId: '<Your AccessKeyId>',
  accessKeySecret: '<Your AccessKeySecret>'
});
async function put() {
  const result = await client.put('fileName', 'file');
  console.log(result.res.headers['x-oss-version-id']); // View the version ID of the uploaded object.
}
put();
```

For more information about simple upload, see [PutObject](#).

Append upload

In a versioning-enabled bucket, the `AppendObject` operation can be performed only on objects of which the current version is an appendable object.

Note

- When you perform the `AppendObject` operation on an object of which the current version is an appendable object, OSS does not generate a previous version for the appendable object.
- When you perform the `PutObject` or `DeleteObject` operation on an object of which the current version is an appendable object, OSS stores the appendable object as a previous version and prevents the object from being further appended.
- You cannot perform the `AppendObject` operation on objects of which the current version is not an appendable object such as a normal object or a delete marker.

The following code provides an example for append upload:

```
const OSS = require('ali-oss');
const client = new OSS({
  bucket: '<Your BucketName>',
  // The endpoint of the China (Hangzhou) region is used in this example. Specify the actual endpoint.
  region: '<Your Region>',
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log on to OSS because the account has permissions on all API operations. We recommend that you use your RAM user's credentials to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
  accessKeyId: '<Your AccessKeyId>',
  accessKeySecret: '<Your AccessKeySecret>'
});
async function append() {
  await client.append('<file name>', '<upload file>', {
    partSize: 100 * 1024
  });
}
append();
```

For more information about append upload, see [AppendObject](#).

Multipart upload

If you call the MultipartUpload method to complete the MultipartUpload task for an object in a versioning-enabled bucket, OSS generates a unique version ID for the object and returns it as the x-oss-version-id field in the response header.

The following code provides an example for multipart upload:

```
const OSS = require('ali-oss');
const client = new OSS({
  bucket: '<Your BucketName>',
  // The endpoint of the China (Hangzhou) region is used in this example. Specify the actual endpoint.
  region: '<Your Region>',
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log on to OSS because the account has permissions on all API operations. We recommend that you use your RAM user's credentials to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
  accessKeyId: '<Your AccessKeyId>',
  accessKeySecret: '<Your AccessKeySecret>'
});
async function multipartUpload() {
  const result = await client.multipartUpload('<file name>', '<upload file>', {
    partSize: 100 * 1024
  });
  // View the version ID of the uploaded object.
  console.log(result.res.headers['x-oss-version-id']);
}
multipartUpload();
```

For more information about multipart upload, see [CompleteMultipartUpload](#).

4.9.3. Download objects

If you call `GetObject` to download an object in a bucket with versioning enabled or suspended, only the current version of the object is returned by default.

When you perform the `GetObject` operation on an object in a bucket:

- If the current version of the object is a delete marker, the 404 Not Found error is returned.
- If you specify the `versionId` of the target object in the request, the specified version of the object is returned. For example, if the `versionId` is specified as null in the request, the version of which the `versionId` is null is returned.
- If the version specified in the request is a delete marker, the 405 Method Not Allowed error is returned.

You can run the following code to download an object:

```
const OSS = require('ali-oss');
const client = new OSS({
  bucket: '<Your BucketName>',
  // The endpoint of the China (Hangzhou) region is used in this example. Specify the actual endpoint.
  region: '<Your Region>',
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log on to OSS because the account has permissions on all API operations. We recommend that you use your RAM user's credentials to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
  accessKeyId: '<Your AccessKeyId>',
  accessKeySecret: '<Your AccessKeySecret>'
});
async function get() {
  const get = await client.get(name, {
    // View the version ID of the downloaded object.
    versionId: 'versionId',
  });
  console.log(result.content);
}
get();
```

4.9.4. Copy objects

This topic describes how to copy objects in a versioned bucket. You can use `CopyObject` to copy an object that is up to 1 GB in size and use `UploadPartCopy` to copy an object larger than 1 GB in size.

Copy a small object

You can use `CopyObject` to copy an object that is up to 1 GB in size from a source bucket to a destination bucket within the same region.

- By default, the `x-oss-copy-source` header in a `CopyObject` request specifies the current version of the object to copy. If the current version of the object is a delete marker, Object Storage Service (OSS) returns 404 Not Found, which indicates that the object does not exist. You can add a version ID to `x-oss-copy-source` to copy the specified object version. Delete markers cannot be copied.

- You can copy a previous version of an object to the same bucket. The copied previous version becomes the new current version of the object. This way, the previous version of the object is recovered.
- If versioning is enabled for the destination bucket, OSS generates a unique version ID for the destination object. The version ID is returned in the response as the x-oss-version-id header value. If versioning is disabled or suspended for the destination bucket, OSS generates a version whose version ID is null for the destination object and overwrites the original version whose version ID is null.
- Appendable objects in a bucket for which versioning is enabled or suspended cannot be copied.

The following code provides an example on how to copy a small object:

```
const OSS = require('ali-oss');
const client = new OSS({
  bucket: '<Your BucketName>',
  // The endpoint of the China (Hangzhou) region is used in this example. Specify the actual endpoint.
  region: '<Your Region>',
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log on to OSS because the account has permissions on all API operations. We recommend that you use your RAM user's credentials to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
  accessKeyId: '<Your AccessKeyId>',
  accessKeySecret: '<Your AccessKeySecret>',
});
// Specify the version ID of the object to copy.
const versionId = 'your versionId';
async function copy() {
  const result = await client.copy(target, origin, { versionId });
  console.log(result.res.headers);
}
copy();
```

For more information about how to copy small objects, see [CopyObject](#).

Copy a large object

To copy an object larger than 1 GB, you must split the object into parts and copy them sequentially by using multipartUploadCopy.

By default, multipartUploadCopy uploads a part by copying data from the current version of an existing object. You can add versionId in the request header x-oss-copy-source as the condition to upload a part by copying data from a specified version of an existing object. Example: x-oss-copy-source: /SourceBucketName/SourceObjectName?versionId=111111.

 **Note** The value of SourceObjectName must be URL-encoded. The version ID of the copied object version is returned as the x-oss-copy-source-version-id header value in the response.

If versionId is not specified in the request and the current version of the source object is a delete marker, OSS returns 404 Not Found. If versionId is specified in the request and the specified version of the source object is a delete marker, OSS returns 400 Bad Request.

The following code provides an example on how to use multipart copy to copy large objects:

```
const OSS = require('ali-oss');
const client = new OSS({
  bucket: '<Your BucketName>',
  // The endpoint of the China (Hangzhou) region is used in this example. Specify the actual endpoint.
  region: '<Your Region>',
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log on to OSS because the account has permissions on all API operations. We recommend that you use your RAM user's credentials to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
  accessKeyId: '<Your AccessKeyId>',
  accessKeySecret: '<Your AccessKeySecret>'
});
// Specify the version ID of the object to copy.
versionId = 'your versionId';
async function multipartUploadCopy() {
  const result = await client.multipartUploadCopy(
    copyName,
    {
      sourceKey: name,
      sourceBucketName: bucket,
    },
    {
      versionId
    }
  );
  console.log(result);
}
multipartUploadCopy();
```

For more information about multipart copy, see [UploadPartCopy](#).

4.9.5. List objects

This topic describes how to list objects in a versioned bucket, including all objects, a specified number of objects, and objects whose names contain a specified prefix.

List the versions of all objects in a bucket

The following code provides an example on how to list the versions of all objects including delete markers within a specified bucket:

```
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log on to OSS because the account has permissions on all API operations. We recommend that you use your RAM user's credentials to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
const accessKeyId = "<yourAccessKeyId>";
const accessKeySecret = "<yourAccessKeySecret>";
// The endpoint of the China (Hangzhou) region is used in this example. Specify the actual endpoint.
const endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
const bucket = "<yourBucketName>";
// Create an OSSClient instance.
const ossClient = new OSS({
  accessKeyId,
  accessKeySecret,
  endpoint,
  bucket
});
// List the versions of all objects including delete markers within the bucket.
async function main () {
  let nextKeyMarker = null;
  let nextVersionMarker = null;
  let versionListing = null;
  do {
    versionListing = await client.getBucketVersions({
      keyMarker: nextKeyMarker,
      versionIdMarker: nextVersionMarker
    })
    versionListing.objects.forEach(o => {
      console.log(`${o.name}, ${o.versionId}`)
    })
    versionListing.deleteMarker.forEach(o => {
      console.log(`${o.name}, ${o.versionId}`)
    })
    nextKeyMarker = versionListing.NextKeyMarker;
    nextVersionMarker = versionListing.NextVersionIdMarker;
  } while (versionListing.isTruncated);
}
main()
```

List the versions of objects whose names contain a specified prefix

The following code provides an example on how to list all versions of objects whose names contain a specified prefix:


```
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log on to OSS because the account has permissions on all API operations. We recommend that you use your RAM user's credentials to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
const accessKeyId = "<yourAccessKeyId>";
const accessKeySecret = "<yourAccessKeySecret>";
// The endpoint of the China (Hangzhou) region is used in this example. Specify the actual endpoint.
const endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
const bucket = "<yourBucketName>";
// Create an OSSClient instance.
const ossClient = new OSS({
  accessKeyId,
  accessKeySecret,
  endpoint,
  bucket
});
// Specify to return all versions of objects whose names contain the "test-" prefix.
async function main () {
  let nextKeyMarker = null;
  let nextVersionMarker = null;
  let versionListing = null;
  const prefix = 'test-'
  do {
    versionListing = await client.getBucketVersions({
      keyMarker: nextKeyMarker,
      versionIdMarker: nextVersionMarker,
      prefix
    })
    versionListing.objects.forEach(o => {
      console.log(`${o.name}, ${o.versionId}`)
    })
    versionListing.deleteMarker.forEach(o => {
      console.log(`${o.name}, ${o.versionId}`)
    })
    nextKeyMarker = versionListing.NextKeyMarker;
    nextVersionMarker = versionListing.NextVersionIdMarker;
  } while (versionListing.isTruncated);
}
main()
```

List a specified number of object versions

The following code provides an example on how to list a specified number of object versions:

```
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log on to OSS because the account has permissions on all API operations. We recommend that you use your RAM user's credentials to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
const accessKeyId = "<yourAccessKeyId>";
const accessKeySecret = "<yourAccessKeySecret>";
// The endpoint of the China (Hangzhou) region is used in this example. Specify the actual endpoint.
const endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
const bucket = "<yourBucketName>";
// Create an OSSClient instance.
const ossClient = new OSS({
  accessKeyId,
  accessKeySecret,
  endpoint,
  bucket
});
// Specify to return up to 100 object versions.
const versionListing = await client.getBucketVersions({
  'max-keys': 100
})
// Display the version ID of the returned object versions. If versioning is not enabled for the bucket, the version IDs of the returned objects are none.
versionListing.objects.forEach(o => {
  console.log(`${o.name}, ${o.versionId}`)
})
versionListing.deleteMarker.forEach(o => {
  console.log(`${o.name}, ${o.versionId}`)
})
```

List objects by folder

OSS does not use a hierarchical structure for objects, but instead uses a flat structure. All elements are stored as objects in buckets. A folder is an object whose size is 0 and whose name ends with a forward slash (/). You can upload and download this object. By default, objects whose names end with a forward slash (/) is displayed as a folder in the OSS console.

You can specify the delimiter and prefix parameters to list objects by folder.

- If you set prefix to a folder name in the request, objects and subfolders whose names contain the prefix are listed.
- If you also set delimiter to a forward slash (/) in the request, the objects and subfolders whose names start with the specified prefix in the folder are listed. Each subfolder is listed as a single result element in commonPrefixes. The objects and folders in these subfolders are not listed.

Example: The following four objects are stored in the bucket: *oss.jpg*, *fun/test.jpg*, *fun/movie/001.avi*, and *fun/movie/007.avi*. The forward slash (/) is specified as the folder delimiter. The following examples describe how to list objects in simulated folders.

- List the versions of objects within the root folder of a bucket

The following code provides an example on how to list the versions of objects within the root folder of a bucket:

```
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to
log on to OSS because the account has permissions on all API operations. We recommend tha
t you use your RAM user's credentials to call API operations or perform routine operation
s and maintenance. To create a RAM user, log on to the RAM console.
const accessKeyId = "<yourAccessKeyId>";
const accessKeySecret = "<yourAccessKeySecret>";
// The endpoint of the China (Hangzhou) region is used in this example. Specify the actua
l endpoint.
const endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
const bucket = "<yourBucketName>";
// Create an OSSClient instance.
const ossClient = new OSS({
  accessKeyId,
  accessKeySecret,
  endpoint,
  bucket
});
// Set the delimiter parameter to a forward slash (/) to list the versions of objects and
the names of subfolders within the root folder.
async function main () {
  let nextKeyMarker = null;
  let nextVersionMarker = null;
  let versionListing = null;
  do {
    versionListing = await client.getBucketVersions({
      keyMarker: nextKeyMarker,
      versionIdMarker: nextVersionMarker,
      delimiter: '/'
    })
    nextKeyMarker = versionListing.NextKeyMarker;
    nextVersionMarker = versionListing.NextVersionIdMarker;
  } while (versionListing.isTruncated);
}
main()
```

Returned results

```
{
  res: {},
  objects: [
    {
      name: 'oss.jpg',
      versionId: 'xxx',
    }
  ],
  prefixes: ["fun/"],
  NextKeyMarker: null,
  NextVersionIdMarker: null,
  isTruncated: false
}
```

- List objects and subfolders within a folder

The following code provides an example on how to list the objects and subfolders within a specified folder:

```
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to
log on to OSS because the account has permissions on all API operations. We recommend tha
t you use your RAM user's credentials to call API operations or perform routine operation
s and maintenance. To create a RAM user, log on to the RAM console.
const accessKeyId = "<yourAccessKeyId>";
const accessKeySecret = "<yourAccessKeySecret>";
// The endpoint of the China (Hangzhou) region is used in this example. Specify the actua
l endpoint.
const endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
const bucket = "<yourBucketName>";
// Create an OSSClient instance.
const ossClient = new OSS({
  accessKeyId,
  accessKeySecret,
  endpoint,
  bucket
});
// Configure the prefix parameter to list all objects and subfolders within the fun folde
r. Set the delimiter parameter to a forward slash (/).
async function main () {
  let nextKeyMarker = null;
  let nextVersionMarker = null;
  let versionListing = null;
  let prefix = 'fun/'
  do {
    versionListing = await client.getBucketVersions({
      keyMarker: nextKeyMarker,
      versionIdMarker: nextVersionMarker,
      prefix,
      delimiter: '/'
    })
    nextKeyMarker = versionListing.NextKeyMarker;
    nextVersionMarker = versionListing.NextVersionIdMarker;
  } while (versionListing.isTruncated);
}
main()
```

Returned results

```
{
  res: {},
  objects: [
    {
      name: 'test.jpg',
      versionId: 'xxx',
    }
  ],
  prefixes: ["fun/movie/"],
  NextKeyMarker: null,
  NextVersionIdMarker: null,
  isTruncated: false
}
```

4.9.6. Delete objects

This topic describes how to delete a single object, multiple objects, or objects whose names contain a specified prefix from a versioned bucket.

Delete operations in a versioned bucket

When you delete an object from a versioned bucket, you must determine whether to specify a version ID in the request.

- Delete an object without specifying a version ID (temporary deletion)

By default, if you do not specify the version ID of the object that you want to delete in the request, Object Storage Service (OSS) does not delete the current version of the object but adds a delete marker to the object as the new current version. If you perform the `GetObject` operation on the object, OSS identifies the current version of the object as a delete marker and returns `404 Not Found`. Additionally, the `x-oss-delete-marker` header and the `x-oss-version-id` header that indicates the version ID of the created delete marker are included in the response to the `DeleteObject` request.

The value of the `x-oss-delete-marker` header in the response is true, which indicates that the `x-oss-version-id` value is the version ID of a delete marker.

- Delete an object by specifying a version ID (permanent deletion)

If you specify the version ID of the object that you want to delete in the request, OSS permanently deletes the version specified by the `versionId` field in the `params` parameter. To delete a version whose ID is null, add `params['versionId'] = "null"` to `params` in the request. OSS identifies the string "null" as the ID of the version to delete and deletes the version whose ID is null.

Delete a single object

The following examples demonstrate how to permanently or temporarily delete a single object from a versioned bucket.

- Permanently delete an object from a versioned bucket

The following code provides an example on how to permanently delete an object with a specified version ID from a versioned bucket:

```
const OSS = require('ali-oss');
const client = new OSS({
  bucket: '<Your BucketName>',
  // The endpoint of the China (Hangzhou) region is used in this example. Specify the actual endpoint.
  region: '<Your Region>',
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log on to OSS because the account has permissions on all API operations. We recommend that you use your RAM user's credentials to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
  accessKeyId: '<Your AccessKeyId>',
  accessKeySecret: '<Your AccessKeySecret>'
});
// Delete the object with the specified version ID.
const versionId = 'your versionId';
async function deleteVersionObject() {
  const result = await client.delete(obj.name, {
    versionId
  });
}
deleteVersionObject();
```

- Temporarily delete an object from a versioned bucket

The following code provides an example on how to temporarily delete an object from a versioned bucket:

```
const OSS = require('ali-oss');
const client = new OSS({
  bucket: '<Your BucketName>',
  // The endpoint of the China (Hangzhou) region is used in this example. Specify the actual endpoint.
  region: '<Your Region>',
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log on to OSS because the account has permissions on all API operations. We recommend that you use your RAM user's credentials to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
  accessKeyId: '<Your AccessKeyId>',
  accessKeySecret: '<Your AccessKeySecret>'
});
// Temporarily delete an object without specifying its version ID. A delete marker is added to the object.
async function deleteObject() {
  const result = await client.delete(obj.name);
}
deleteObject();
```

For more information about how to delete a single object, see [DeleteObject](#).

Delete multiple objects

The following examples describe how to permanently or temporarily delete multiple objects from a versioned bucket.

- Permanently delete multiple objects from a versioned bucket

The following code provides an example on how to permanently delete multiple objects or delete markers with specified version IDs from a versioned bucket:

```
const OSS = require('ali-oss');
const client = new OSS({
  bucket: '<Your BucketName>',
  // The endpoint of the China (Hangzhou) region is used in this example. Specify the actual endpoint.
  region: '<Your Region>',
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log on to OSS because the account has permissions on all API operations. We recommend that you use your RAM user's credentials to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
  accessKeyId: '<Your AccessKeyId>',
  accessKeySecret: '<Your AccessKeySecret>'
});
// Delete the objects with specified version IDs or the objects whose current versions are delete markers with specified version IDs.
const names = [
  { key: 'key1.js', versionId: 'versionId1' },
  { key: 'key2.js', versionId: 'versionId2' }
];
async function deleteMulti() {
  const result = await client.deleteMulti(names);
  console.log(result);
}
deleteMulti();
```

For more information about how to delete multiple objects, see [DeleteMultipleObjects](#).

- Temporarily delete multiple objects from a versioned bucket

The following code provides an example on how to temporarily delete multiple objects from a versioned bucket:

```
const OSS = require('ali-oss');
const client = new OSS({
  bucket: '<Your BucketName>',
  // The endpoint of the China (Hangzhou) region is used in this example. Specify the actual endpoint.
  region: '<Your Region>',
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log on to OSS because the account has permissions on all API operations. We recommend that you use your RAM user's credentials to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
  accessKeyId: '<Your AccessKeyId>',
  accessKeySecret: '<Your AccessKeySecret>'
});
const names = ['key1.js', 'key2.js'];
async function deleteMulti() {
  // Delete multiple objects without specifying their version IDs.
  const result = await client.deleteMulti(names);
  console.log(result);
}
deleteMulti();
```

Delete objects whose names contain a specified prefix

The following code provides an example on how to delete objects whose names contain a specified prefix:

```
const OSS = require('ali-oss');
const client = new OSS({
  bucket: '<Your BucketName>',
  // The endpoint of the China (Hangzhou) region is used in this example. Specify the actual endpoint.
  region: '<Your Region>',
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log on to OSS because the account has permissions on all API operations. We recommend that you use your RAM user's credentials to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
  accessKeyId: '<Your AccessKeyId>',
  accessKeySecret: '<Your AccessKeySecret>'
});

async function deleteMultiPrefix(prefix) {
  // List the versions of the object whose names contain the specified prefix.
  const list = client.getBucketVersions({
    prefix: prefix,
  });
  list.objects = list.objects || [];
  for (let i = 0; i < result.objects.length; i++) {
    const obj = result.objects[i];
    // Delete the objects whose names contain the specified prefix.
    const versionId = obj.versionId;
    await client.delete(obj.name, {
      versionId
    });
  }
}
```

4.9.7. Restore objects

In a versioning-enabled bucket, different versions of the objects may have different storage classes. By default, the `RestoreObject` operation restores the current version of an object. To restore a specific version of the object, specify the version ID.

The following code provides an example on how to restore an object:


```
const OSS = require('ali-oss');
const client = new OSS({
  bucket: '<Your BucketName>',
  // The endpoint of the China (Hangzhou) region is used in this example. Specify the actual endpoint.
  region: '<Your Region>',
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log on to OSS because the account has permissions on all API operations. We recommend that you use your RAM user's credentials to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
  accessKeyId: '<Your AccessKeyId>',
  accessKeySecret: '<Your AccessKeySecret>'
});
const name = '<your objectName>'
// Restore the object with the specified version ID.
const versionId = 'your versionId'
async function restore() {
  const result = await client.restore(name, {
    versionId
  });
  console.log(result);
}
restore();
```

4.9.8. Obtain object metadata

By default, when you call the `HeadObject` operation on an object in a versioned bucket, only the metadata of the object of the current version is returned. You can obtain the metadata of an object of a specified version by specifying the version ID of the object.

 **Note** If the current version of the object is a delete marker, Object Storage Service (OSS) returns 404 Not Found. If you specify a version ID in the request, OSS returns the metadata of the object of the specified version. For more information about how to obtain the version IDs of an object, see [List objects](#).

The following code provides an example on how to obtain the metadata of the `exampleobject.txt` object of the specified version in the `exampledir` directory of the `examplebucket` bucket:

```
const OSS = require('ali-oss');
const client = new OSS({
  // Set yourRegion to the endpoint of the region in which the bucket is located. For example, if your bucket is located in the China (Hangzhou) region, set yourRegion to oss-cn-hangzhou.
  region: 'yourRegion',
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a Resource Access Management (RAM) user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
  accessKeyId: 'yourAccessKeyId',
  accessKeySecret: 'yourAccessKeySecret',
  // Specify the name of the bucket. Example: examplebucket.
  bucket: 'examplebucket',
});
async function headInfo() {
  // Specify the full path of the object. Example: exampledir/exampleobject.txt. The full path of the object cannot contain bucket names.
  const name = 'exampledir/exampleobject.txt'
  // Specify the version ID of the object to obtain the metadata of the object of the specified version.
  const versionId = 'CAEQMxiBgICAof2D0BYiIDJhMGE3N2M1YTI1NDQzOGY5NTkyNTI3MGYyMzJm****'
  const options = {
    versionId
  };
  const result = await client.head(name, options);
  console.log(result);
}
headInfo();
```

For more information about how to obtain object metadata , see [HeadObject](#).

4.9.9. Manage object ACL

This topic describes how to manage ACLs of objects in a versioning-enabled bucket.

Configure the ACL of an object

The following code provides an example on how to configure the ACL of an object:

```
const OSS = require('ali-oss');
const client = new OSS({
  bucket: '<Your BucketName>',
  // The endpoint of the China (Hangzhou) region is used in this example. Specify the actual endpoint.
  region: '<Your Region>',
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log on to OSS because the account has permissions on all API operations. We recommend that you use your RAM user's credentials to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
  accessKeyId: '<Your AccessKeyId>',
  accessKeySecret: '<Your AccessKeySecret>'
});

async function putACL() {
  const name = '<your objectName>'
  const acl = '<your acl>'
  const versionId = 'your versionId' // Configure the ACL of an object with the specified version ID.
  const options = {
    versionId
  };
  const result = client.putACL(name, acl, options);
  console.log(result);
}
putACL();
```

Obtain the ACL of an object

The following code provides an example on how to obtain the ACL of an object:

```
const OSS = require('ali-oss');
const client = new OSS({
  bucket: '<Your BucketName>',
  // The endpoint of the China (Hangzhou) region is used in this example. Specify the actual endpoint.
  region: '<Your Region>',
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log on to OSS because the account has permissions on all API operations. We recommend that you use your RAM user's credentials to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
  accessKeyId: '<Your AccessKeyId>',
  accessKeySecret: '<Your AccessKeySecret>'
});
async function getACL() {
  const name = '<your objectName>'
  const versionId = 'your versionId' // View the version ID of the object of which the ACL is obtained.
  const options = {
    versionId
  };
  const result = client.getACL(name, options);
  console.log(result);
}
getACL();
```

4.9.10. Manage symbolic links

This topic describes how to manage symbolic links in a versioning-enabled bucket.

Create a symbolic link

You can create a symbol link that directs to the current version of the target object.

 **Note** In a bucket with the versioning function enabled, you cannot create a symbol link for a delete marker.

A symbol link can have multiple versions that direct to different target objects. The version ID of the symbol link is automatically generated and is returned as the x-oss-version-id field in the response header.

The following code provides an example on how to create a symbolic link:

```
const OSS = require('ali-oss');
const client = new OSS({
  bucket: '<Your BucketName>',
  // The endpoint of the China (Hangzhou) region is used in this example. Specify the actual endpoint.
  region: '<Your Region>',
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log on to OSS because the account has permissions on all API operations. We recommend that you use your RAM user's credentials to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
  accessKeyId: '<Your AccessKeyId>',
  accessKeySecret: '<Your AccessKeySecret>'
});

async function putSymlink() {
  // Create a symbolic link.
  const name = '<your objectName>'
  const targetName = '<your target objectName>'
  const result = await client.putSymlink(name, targetName);
  console.log(result.res.headers['x-oss-version-id']);
}

putSymlink();
```

For more information about creating a symbol link, see [PutSymlink](#).

Obtain a symbolic link

GetSymlink obtains the current version of the target symbol link by default. You can specify the versionId in the request to obtain the specified version of a symbol link. If the current version of the target symbol link is a delete marker, OSS returns the 404 Not Found error and includes `x-oss-delete-marker = true` and `x-oss-version-id` in the response header.

 **Note** You must have the read permission on a symbol link to obtain it.

The following code provides an example on how to obtain a symbolic link:

```
const OSS = require('ali-oss');
const client = new OSS({
  bucket: '<Your BucketName>',
  // The endpoint of the China (Hangzhou) region is used in this example. Specify the actual endpoint.
  region: '<Your Region>',
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log on to OSS because the account has permissions on all API operations. We recommend that you use your RAM user's credentials to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
  accessKeyId: '<Your AccessKeyId>',
  accessKeySecret: '<Your AccessKeySecret>'
});

async function getSymlink() {
  const name = '<your objectName>'
  const versionId = '<your versionId>'
  // Obtain the content of the symbolic link with the specified version ID.
  const result = await client.getSymlink(name, {
    versionId
  });
  console.log(result);
}
getSymlink();
```

For more information about obtaining a symbol link, see [GetSymlink](#).

4.10. Object tagging

4.10.1. Configure object tagging

Object Storage Service (OSS) allows you to configure object tags to classify objects. Tags allow you to configure lifecycle rules and access control lists (ACLs) for objects that have the same tag.

Background information

When you configure object tagging, take note of the following items:

- You can add tags to an object when and after you upload the object. If the object already has tags, the existing tags are overwritten. For more information about object tags, see [PutObjectTagging](#).
- To add tags to an object, you must have permissions to call the PutObjectTagging operation.

You can create a custom policy by using scripts to grant the permissions to RAM users. For more information, see [Attach a custom policy to a RAM user](#).

- The Last-Modified value of an object is not updated when the tags are changed.
- You can add up to 10 tags to an object. The tags added to an object must have unique keys.
- A tag key can be up to 128 characters in length. A tag value can be up to 256 characters in length.
- Tag keys and tag values are case-sensitive.
- The key and the value of a tag can contain letters, digits, spaces, and the following special characters:

+ - = . _ : /

 **Note** If you configure the tags in the HTTP header and the tags contain special characters, you can encode the keys and the values of the tags by using URL encoding.

Object tagging uses a key-value pair to identify objects. For more information about object tags, see [Object tagging](#).

Add tags to an object when you upload it

- Add tags to an object when you upload it in simple upload

The following code provides an example on how to add tags to an object when you upload it by calling Put Object:

```
const OSS = require('ali-oss')
const client = new OSS({
  // Set yourRegion to the endpoint of the region in which the bucket is located. For example, if your bucket is located in the China (Hangzhou) region, set yourRegion to oss-cn-hangzhou.
  region: 'yourRegion',
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access Object Storage Service (OSS) because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
  accessKeyId: 'yourAccessKeyId',
  accessKeySecret: 'yourAccessKeySecret',
  // Specify the name of the bucket in which the object you want to access is stored. Example: examplebucket.
  bucket: 'examplebucket',
});
// Specify the full path of the object. Example: exampledir/exampleobject.txt. The full path of the object cannot contain bucket names.
const objectName = 'exampledir/exampleobject.txt'
// Specify the full path of the local file to upload. Example: D:\\localpath\\examplefile.txt.
// By default, if you set this parameter to the name of a local file such as examplefile.txt without specifying the local path, the local file is uploaded from the local path of the project to which the sample program belongs.
const localFilepath = 'D:\\localpath\\examplefile.txt'
// Configure request headers.
const headers = {
  // Specify the key and the value of the object tag. Example: the key is owner, and the value is John.
  'x-oss-tagging': 'owner=John&type=document',
}
client.put(objectName, localFilepath, {
  headers
})
```

- Add tags to an object when you upload it by using multipart upload

The following code provides an example on how to add tags to an object when you upload it by calling Multipart Upload:

```

const OSS = require('ali-oss')
const client = new OSS({
  // Set yourRegion to the endpoint of the region in which the bucket is located. For example, if your bucket is located in the China (Hangzhou) region, set yourRegion to oss-cn-hangzhou.
  region: 'yourRegion',
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
  accessKeyId: 'yourAccessKeyId',
  accessKeySecret: 'yourAccessKeySecret',
  // Specify the name of the bucket in which the object you want to access is stored. Example: examplebucket.
  bucket: 'examplebucket',
});
// Specify the full path of the object. Example: exampledir/exampleobject.txt. The full path of the object cannot contain bucket names.
const objectName = 'exampledir/exampleobject.txt'
// Specify the full path of the local file to upload. Example: D:\\localpath\\examplefile.txt.
// By default, if you set this parameter to the name of a local file such as examplefile.txt without specifying the local path, the local file is uploaded from the local path of the project to which the sample program belongs.
const localFilepath = 'D:\\localpath\\examplefile.txt'
// Configure request headers.
const headers = {
  // Specify the key and the value of the object tag. Example: the key is owner, and the value is John.
  'x-oss-tagging': 'owner=John&type=document',
}
async function setTag() {
  await client.multipartUpload(objectName, localFilepath, {
    // Specify the part size. Unit: byte. Each part except the last part must be larger than 100 KB in size.
    partSize: 100 * 1024,
    headers
  });
  const tag = await client.getObjectTagging(objectName);
  console.log(tag);
}
setTag()

```

- Add tags to an object when you upload it by using append upload

The following code provides an example on how to add tags to an object when you upload it by calling `AppendObject`:


```

const OSS = require('ali-oss')
const client = new OSS({
  // Set yourRegion to the endpoint of the region in which the bucket is located. For example, if your bucket is located in the China (Hangzhou) region, set yourRegion to oss-cn-hangzhou.
  region: 'yourRegion',
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
  accessKeyId: 'yourAccessKeyId',
  accessKeySecret: 'yourAccessKeySecret',
  // Specify the name of the bucket in which the object you want to access is stored. Example: examplebucket.
  bucket: 'examplebucket',
});
// Specify the full path of the object. Example: exampledir/exampleobject.txt. The full path of the object cannot contain bucket names.
const objectName = 'exampledir/exampleobject.txt'
// Specify the full path of the local file to upload. Example: D:\\localpath\\examplefile.txt.
// By default, if you set this parameter to the name of a local file such as examplefile.txt without specifying the local path, the local file is uploaded from the local path of the project to which the sample program belongs.
const localFilepath = 'D:\\localpath\\examplefile.txt'
// Configure request headers.
const headers = {
  // Specify the key and the value of the object tag. Example: the key is owner, and the value is John.
  'x-oss-tagging': 'owner=John&type=document',
}
// Upload the object. Tags are added to the object when you call AppendObject and specify the x-oss-tagging header.
// Only the tags set when the object is appended for the first time are added to the object.
async function setTag() {
  await client.append(objectName, localFilepath, {
    // Specify the part size. Unit: byte. Each part except the last part must be larger than 100 KB in size.
    partSize: 100 * 1024,
    headers
  });
  const tag = await client.getObjectTagging(objectName);
  console.log(tag);
}
setTag()

```

- Add tags to an object when you upload it by using resumable upload

The following code provides an example on how to add tags to an object when you upload it by calling `MultipartUpload`:

```
const OSS = require('ali-oss')
const client = new OSS({
  // Set yourRegion to the endpoint of the region in which the bucket is located. For example, if your bucket is located in the China (Hangzhou) region, set yourRegion to oss-cn-hangzhou.
  region: 'yourRegion',
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
  accessKeyId: 'yourAccessKeyId',
  accessKeySecret: 'yourAccessKeySecret',
  // Specify the name of the bucket in which the object you want to access is stored. Example: examplebucket.
  bucket: 'examplebucket',
});
// Specify the full path of the object. Example: exampledir/exampleobject.txt. The full path of the object cannot contain bucket names.
const objectName = 'exampledir/exampleobject.txt'
// Specify the full path of the local file to upload. Example: D:\\localpath\\examplefile.txt.
// By default, if you set this parameter to the name of a local file such as examplefile.txt without specifying the local path, the local file is uploaded from the local path of the project to which the sample program belongs.
const localFilepath = 'D:\\localpath\\examplefile.txt'
// Configure the checkpoint information.
let checkpoint;
// Configure request headers.
const headers = {
  // Specify the key and the value of the object tag. Example: the key is owner, and the value is John.
  'x-oss-tagging': 'owner=John&type=document',
}
async function setTag() {
  await client.multipartUpload(objectName, localFilepath, {
    checkpoint,
    async progress(percentage, cpt) {
      checkpoint = cpt;
    },
    headers
  });
  const tag = await client.getObjectTagging(objectName);
  console.log(tag);
}
setTag()
```

Add tags to an uploaded object or modify the tags of the object

If an existing object has no tags or the added tags of the object do not meet your requirements, you can add tags to or modify the tags of the existing object.

The following code provides an example on how to add tags to or modify the tags of an existing object:

```
const OSS = require('ali-oss')
const client = new OSS({
  // Set yourRegion to the endpoint of the region in which the bucket is located. For example, if your bucket is located in the China (Hangzhou) region, set yourRegion to oss-cn-hangzhou.
  region: 'yourRegion',
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
  accessKeyId: 'yourAccessKeyId',
  accessKeySecret: 'yourAccessKeySecret',
  // Specify the name of the bucket in which the object you want to access is stored. Example: examplebucket.
  bucket: 'examplebucket',
});
// Specify the full path of the object. Example: exampledir/exampleobject.txt. The full path of the object cannot contain bucket names.
const objectName = 'exampledir/exampleobject.txt'
// Specify the key and the value of the object tag. Example: the key is owner, and the value is John.
const tag = { owner: 'John', type: 'document' };
async function putObjectTagging(objectName, tag) {
  try {
    const result = await client.putObjectTagging(objectName, tag);
    console.log(result);
  } catch (e) {
    console.log(e);
  }
}
putObjectTagging(objectName, tag)
```

Add tags to a specified version of an object or modify the tags of the object

If versioning is enabled for the bucket, you can add tags to or modify the tags of a specified version of an object by specifying the version ID of the object.

The following code provides an example on how to add tags to a specified version of an object or modify the tags of the object.

 **Note** For more information about how to obtain the version ID of the object, see [List objects](#).

```
const OSS = require('ali-oss')
const client = new OSS({
  // Set yourRegion to the endpoint of the region in which the bucket is located. For example, if your bucket is located in the China (Hangzhou) region, set yourRegion to oss-cn-hangzhou.
  region: 'yourRegion',
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
  accessKeyId: 'yourAccessKeyId',
  accessKeySecret: 'yourAccessKeySecret',
  // Specify the name of the bucket in which the object you want to access is stored. Example: examplebucket.
  bucket: 'examplebucket',
});
// Specify the full path of the object. Example: exampledir/exampleobject.txt. The full path of the object cannot contain bucket names.
const objectName = 'exampledir/exampleobject.txt'
// Specify the key and the value of the object tag. Example: the key is owner, and the value is John.
const tag = { owner: 'John', type: 'document' };
// Specify the version ID of the source object.
const versionId = 'CAEQIRiBgMDqvPqA3BciIDJhMjE4MWZkN2ViYTRmYzJhZjZkxMzk2YWw2MjJk****'
async function putObjectTagging(objectName, tag) {
  try {
    const options = {
      versionId
    };
    const result = await client.putObjectTagging(objectName, tag, options);
    console.log(result);
  } catch (e) {
    console.log(e);
  }
}
putObjectTagging(objectName, tag)
```

Add tags to an object when you copy the object

- Add tags to an object when you copy the object in simple copy mode

The following code provides an example on how to add tags to an object smaller than 1 GB when you copy it by calling `CopyObject`:

```

const OSS = require('ali-oss')
const client = new OSS({
  // Set yourRegion to the endpoint of the region in which the bucket is located. For example, if your bucket is located in the China (Hangzhou) region, set yourRegion to oss-cn-hangzhou.
  region: 'yourRegion',
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
  accessKeyId: 'yourAccessKeyId',
  accessKeySecret: 'yourAccessKeySecret',
  // Specify the name of the bucket in which the object you want to access is stored. Example: examplebucket.
  bucket: 'examplebucket',
});
// Specify the full path of the source object. Example: srcexampledir/exampleobject.txt.
const sourceObjectName = 'srcexampledir/exampleobject.txt';
// Specify the full path of the destination object. Example: destexampledir/exampleobject.txt.
const targetObjectName = 'destexampledir/exampleobject.txt';
// Configure request headers.
const headers = {
  // Specify the key and the value of the object tag. Example: the key is owner, and the value is John.
  'x-oss-tagging': 'owner=John&type=document',
  // Specifies how to set tags for the destination object. Valid values: Copy and Replace. Copy is the default value and indicates that the tags of the source object are copied to the destination object. Replace indicates that the tags specified in the request are added to the destination object.
  'x-oss-tagging-directive': 'Replace'
}
async function setTag() {
  const result = await client.copy(targetObjectName, sourceObjectName, {
    headers
  });
  const tag = await client.getObjectTagging(targetObjectName)
  console.log(tag)
}
setTag()

```

- Add tags to an object when you copy the object in multipart copy mode

The following code provides an example on how to add tags to an object larger than 1 GB when you copy it by calling `MultipartUpload`:

```
const OSS = require('ali-oss')
const client = new OSS({
  // Set yourRegion to the endpoint of the region in which the bucket is located. For example, if your bucket is located in the China (Hangzhou) region, set yourRegion to oss-cn-hangzhou.
  region: 'yourRegion',
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
  accessKeyId: 'yourAccessKeyId',
  accessKeySecret: 'yourAccessKeySecret',
  // Specify the name of the bucket in which the object you want to access is stored. Example: examplebucket.
  bucket: 'examplebucket'
});
// Specify the full path of the source object. Example: srcexampledir/exampleobject.txt.
const sourceObjectName = 'srcexampledir/exampleobject.txt'
// Specify the full path of the destination object. Example: destexampledir/exampleobject.txt.
const targetObjectName = 'destexampledir/exampleobject.txt'
// Configure request headers.
const headers = {
  // Specify the key and the value of the object tag. Example: the key is owner, and the value is John.
  'x-oss-tagging': 'owner=John&type=document',
}
async function setTag() {
  await client.multipartUploadCopy(targetObjectName, {
    sourceKey: sourceObjectName,
    sourceBucketName: 'examplebucket'
  }, {
    // Specify the part size. Unit: byte. Each part except the last part must be larger than 100 KB in size.
    partSize: 256 * 1024,
    headers
  });
  const tag = await client.getObjectTagging(targetObjectName)
  console.log(tag)
}
setTag()
```

Add tags to a symbolic link object

The following code provides an example on how to add tags to a symbolic link object:

```
const OSS = require('ali-oss')
const client = new OSS({
  // Set yourRegion to the endpoint of the region in which the bucket is located. For example, if your bucket is located in the China (Hangzhou) region, set yourRegion to oss-cn-hangzhou.
  region: 'yourRegion',
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
  accessKeyId: 'yourAccessKeyId',
  accessKeySecret: 'yourAccessKeySecret',
  // Specify the name of the bucket in which the object you want to access is stored. Example: examplebucket.
  bucket: 'examplebucket',
});
// Specify the full path of the symbolic link object. Example: shortcut/myobject.txt.
const symLink = "shortcut/myobject.txt";
// Specify the full path of the object. Example: exampledir/exampleobject.txt. The full path of the object cannot contain bucket names.
const targetObjectName = 'exampledir/exampleobject.txt'
// Configure request headers.
const headers = {
  // Specify the key and the value of the object tag. Example: the key is owner, and the value is John.
  'x-oss-tagging': 'owner=John&type=document',
}
async function setTag() {
  await client.putSymlink(symLink, targetObjectName, {
    storageClass: 'IA',
    meta: {
      uid: '1',
      slus: 'test.html'
    },
    headers
  });
  const tag = await client.getObjectTagging(targetObjectName)
  console.log(tag)
}
setTag()
```

4.10.2. Query the tags of an object

After you configure tags for an object, you can query the tags of the object. If versioning is enabled for a bucket that stores the object whose tags you want to query, Object Storage Service (OSS) returns the tags of the current version of the object by default. To query the tags of a specified version of the object, you must specify the version ID of the object.



Query the tags of an object

If versioning is not enabled for the bucket that stores the object whose tags you want to query, you can query the tags of the object based on your requirements. If versioning is enabled for the bucket that stores the object whose tags you want to query, OSS returns the tags of the current version of the object by default.

The following code provides an example on how to query the tags of the `exampleobject.txt` object in the `exampledir` directory of the `examplebucket` bucket:

```
const OSS = require('ali-oss')
const client = new OSS({
  // Set yourRegion to the endpoint of the region in which the bucket is located. For example, if your bucket is located in the China (Hangzhou) region, set yourRegion to oss-cn-hangzhou.
  region: 'yourRegion',
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access Object Storage Service (OSS) because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
  accessKeyId: 'yourAccessKeyId',
  accessKeySecret: 'yourAccessKeySecret',
  // Specify the name of the bucket in which the object you want to access is stored. Example: examplebucket.
  bucket: 'examplebucket',
});
// Specify the full path of the object. Example: exampledir/exampleobject.txt. The full path of the object cannot contain bucket names.
const objectName = 'exampledir/exampleobject.txt'
// Obtain the tags added to the object.
async function getObjectTagging(objectName) {
  try {
    const result = await client.getObjectTagging(objectName);
    console.log(result);
  } catch (e) {
    console.log(e);
  }
}
getObjectTagging(objectName)
```

Query the tags of a specified version of an object

If versioning is enabled for the bucket that stores the object whose tags you want to query, you can query the tags of a specified version of the object by specifying the version ID of the object.

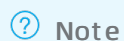
The following code provides an example on how to query the tags of a specified version of the `exampleobject.txt` object in the `exampledir` directory of the `examplebucket` bucket:

 **Note** For more information about how to obtain the version ID of the object, see [List objects](#).


```
const OSS = require('ali-oss')
const client = new OSS({
  // Set yourRegion to the endpoint of the region in which the bucket is located. For example, if your bucket is located in the China (Hangzhou) region, set yourRegion to oss-cn-hangzhou.
  region: 'yourRegion',
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
  accessKeyId: 'yourAccessKeyId',
  accessKeySecret: 'yourAccessKeySecret',
  // Specify the name of the bucket in which the object you want to access is stored. Example: examplebucket.
  bucket: 'examplebucket',
});
// Specify the full path of the object. Example: exampledir/exampleobject.txt. The full path of the object cannot contain bucket names.
const objectName = 'exampledir/exampleobject.txt'
// Specify the version ID of the source object.
const versionId = 'CAEQIRiBgMDqvPqA3BciIDJhMjE4MWZkN2ViYTRmYzJhZjZkxMzk2YWM2NjJk****'
// Obtain the tags added to the object.
async function getObjectTagging(objectName) {
  try {
    const options = {
      versionId
    };
    const result = await client.getObjectTagging(objectName, options);
    console.log(result);
  } catch (e) {
    console.log(e);
  }
}
getObjectTagging(objectName)
```

4.10.3. Delete the tags of an object

You can delete unnecessary tags of an object based on your requirements. If versioning is enabled for a bucket that stores the object whose tags you want to delete, Object Storage Service (OSS) deletes the tags of the current version of the object by default. To delete the tags of a specified version of the object, you must specify the version ID of the object.



Delete the tags of an object

If versioning is not enabled for the bucket that stores the object whose tags you want to delete, you can delete the tags of the object based on your requirements. If versioning is enabled for the bucket that stores the object whose tags you want to delete, OSS deletes the tags of the current version of the object by default.

The following code provides an example on how to delete the tags of the `exampleobject.txt` object in the `exampledir` directory of the `examplebucket` bucket:

```
const OSS = require('ali-oss')
const client = new OSS({
  // Set yourRegion to the endpoint of the region in which the bucket is located. For example, if your bucket is located in the China (Hangzhou) region, set yourRegion to oss-cn-hangzhou.
  region: 'yourRegion',
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access Object Storage Service (OSS) because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
  accessKeyId: 'yourAccessKeyId',
  accessKeySecret: 'yourAccessKeySecret',
  // Specify the name of the bucket in which the object you want to access is stored. Example: examplebucket.
  bucket: 'examplebucket',
});
// Specify the full path of the object. Example: exampledir/exampleobject.txt. The full path of the object cannot contain bucket names.
const objectName = 'exampledir/exampleobject.txt'
// Delete the tags added to the object.
async function deleteObjectTagging(objectName) {
  try {
    const result = await client.deleteObjectTagging(objectName);
    console.log(result);
  } catch (e) {
    console.log(e);
  }
}
deleteObjectTagging(objectName)
```

Delete the tags of a specified version of an object

If versioning is enabled for the bucket that stores the object whose tags you want to delete, you can delete the tags of a specified version of the object by specifying the version ID of the object.

The following code provides an example on how to delete the tags of a specified version of the `exampleobject.txt` object in the `exampledir` directory of the `examplebucket` bucket:

 **Note** For more information about how to obtain the version ID of the object, see [List objects](#).


```
const OSS = require('ali-oss')
const client = new OSS({
  bucket: '<Your BucketName>',
  // This example uses the China (Hangzhou) region. Specify the actual region based on your
  requirements.
  region: '<Your Region>',
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to
  log on to OSS, because the account has permissions on all API operations. We recommend that
  you use a RAM user to call API operations or perform routine operations and maintenance. To
  create a RAM user, log on to the RAM console.
  accessKeyId: '<Your AccessKeyId>',
  accessKeySecret: '<Your AccessKeySecret>',
});
// Set tags as the matching condition.
const tag = [{
  key: 'key1',
  value: 'value1'
},{
  key: 'key2',
  value: 'value2'
}]
client.putBucketLifecycle('<yourBucketName>', [{
  // Set the lifecycle rule ID to rule1.
  id: 'rule1',
  // Specify to apply the lifecycle rule to objects whose names are prefixed with one.
  prefix: 'one',
  status: 'Enabled',
  expiration: {
    // Specify a lifecycle rule for an object so that the object expires 60 days after it was
    last modified.
    days: 60
  },
  // Specify a lifecycle rule for an object so that the storage class of the object is conv
  erted to IA 10 days after it was last modified.
  transition: {
    days: 10,
    StorageClass: 'IA'
  },
  tag
}]);
```

View the tags set as the matching condition of a lifecycle rule

The following code provides an example on how to view the tags set as the matching condition of a lifecycle rule:

```
const OSS = require('ali-oss')
const client = new OSS({
  bucket: '<Your BucketName>',
  // This example uses the endpoint of the China (Hangzhou) region. Specify the actual endpoint based on your requirements.
  region: '<Your Region>',
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log on to OSS, because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
  accessKeyId: '<Your AccessKeyId>',
  accessKeySecret: '<Your AccessKeySecret>',
});
try {
  const result = client.getBucketLifecycle('<yourBucketName>')
  // View the tags that are set as the matching condition for a lifecycle rule.
  result.rules.map(rule => {
    if (rule.tag) {
      rule.tag.map(_ => {
        console.log(`key: ${_.key}, value: ${_.value}`)
      })
    }
  })
} catch (error) {
  console.log(error)
}
```

4.11. Common operations on LiveChannels

This topic describes common operations on LiveChannels, such as create, list, and delete LiveChannels.

Create a LiveChannel

Before you upload audio and video data by using the Real-Time Messaging Protocol (RTMP), you must call the PutLiveChannel operation to create a LiveChannel. The response to the PutLiveChannel request includes the URL that is used to ingest streams to the LiveChannel and the URL that is used to play the streams ingested to the LiveChannel.

The following code provides an example on how to create a LiveChannel:

```
const OSS = require('ali-oss');
const store = new OSS({
  // Set yourRegion to the region in which the bucket is located. For example, if your bucket is located in the China (Hangzhou) region, set yourRegion to oss-cn-hangzhou.
  region: 'yourRegion',
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access Object Storage Service (OSS) because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
  accessKeyId: 'yourAccessKeyId',
  accessKeySecret: 'yourAccessKeySecret',
  // Specify the bucket name.
  bucket: 'examplebucket',
})
// Specify the name of the LiveChannel that you want to create. Example: mychannel. The name cannot contain forward slashes (/).
const cid = 'mychannel';
const conf = {
  // Specify the information about the LiveChannel. The size of the information cannot exceed 128 bytes.
  Description: 'this is my channel',
  // Specify the status of the LiveChannel. In this example, the Status parameter is set to enabled, which indicates that the LiveChannel is enabled. If you want to disable the LiveChannel, set the Status parameter to disabled.
  Status: 'enabled',
  Target: {
    // Specify the format in which the LiveChannel stores uploaded data. Only HLS is supported.
    Type: 'HLS',
    // Specify the duration of each ts file. Unit: seconds.
    FragDuration: '10',
    // If you set Type to HLS, you must specify the number of ts files that are included in the m3u8 file.
    FragCount: '5',
    // If you set Type to HLS, you must specify the name of the generated m3u8 file. The name must be 6 to 128 bytes in length and must end with .m3u8.
    PlaylistName: 'playlist.m3u8'
  }
};
// Create the LiveChannel.
store.putChannel(cid, conf).then(r=>console.log(r))
```

Query the configurations of a LiveChannel

The following code provides an example on how to query the configurations of a LiveChannel:

```
const OSS = require('ali-oss');
const store = new OSS({
  // Set yourRegion to the region in which the bucket is located. For example, if your bucket is located in the China (Hangzhou) region, set yourRegion to oss-cn-hangzhou.
  region: 'yourRegion',
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
  accessKeyId: 'yourAccessKeyId',
  accessKeySecret: 'yourAccessKeySecret',
  // Specify the bucket name.
  bucket: 'examplebucket',
})
// Specify the name of the LiveChannel.
const cid = 'mychannel';
// Query the configurations of the LiveChannel.
store.getChannel(cid).then(r=>console.log(r));
```

Configure the LiveChannel status

The following code provides an example on how to configure the status of a LiveChannel:

```
const OSS = require('ali-oss');
const store = new OSS({
  // Set yourRegion to the region in which the bucket is located. For example, if your bucket is located in the China (Hangzhou) region, set yourRegion to oss-cn-hangzhou.
  region: 'yourRegion',
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
  accessKeyId: 'yourAccessKeyId',
  accessKeySecret: 'yourAccessKeySecret',
  // Specify the bucket name.
  bucket: 'examplebucket',
})
// Specify the name of the LiveChannel.
const cid = 'mychannel';
// A LiveChannel can be in one of the following states: enabled or disabled.
// If a LiveChannel is in the disabled state, OSS prohibits you from ingesting streams to the LiveChannel. If you are ingesting a stream to a LiveChannel whose state becomes disabled, your client is disconnected from the LiveChannel after approximately 10 seconds.
store.putChannelStatus(cid, 'disabled').then(r=>console.log(r));
```

Query the stream ingestion status of a LiveChannel

The following code provides an example on how to query the stream ingestion status of a LiveChannel:

```
const OSS = require('ali-oss');
const store = new OSS({
  // Set yourRegion to the region in which the bucket is located. For example, if your bucket is located in the China (Hangzhou) region, set yourRegion to oss-cn-hangzhou.
  region: 'yourRegion',
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
  accessKeyId: 'yourAccessKeyId',
  accessKeySecret: 'yourAccessKeySecret',
  // Specify the bucket name.
  bucket: 'examplebucket',
})
// Specify the name of the LiveChannel.
const cid = 'mychannel';
// Query the stream ingestion status of the LiveChannel.
store.getChannelStatus(cid).then(r=>console.log(r))
```

Generate a playlist for a LiveChannel

You can call the PostVodPlaylist operation to generate a VOD playlist for the specified LiveChannel. OSS queries the ts files that are generated by the streams that are ingested to the specified LiveChannel within the specified time range and converges these files into an m3u8 playlist.

The following code provides an example on how to generate a playlist for the specified LiveChannel:

```
const OSS = require('ali-oss');
const store = new OSS({
  // Set yourRegion to the region in which the bucket is located. For example, if your bucket is located in the China (Hangzhou) region, set yourRegion to oss-cn-hangzhou.
  region: 'yourRegion',
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
  accessKeyId: 'yourAccessKeyId',
  accessKeySecret: 'yourAccessKeySecret',
  // Specify the bucket name.
  bucket: 'examplebucket',
})
// Specify the name of the LiveChannel.
const cid = 'mychannel';
const r = await this.store.createVod(cid, 're-play', {
  // Specify the start time of the time range during which the ts files that you want to query are generated in the Unix timestamp format.
  startTime: 1460464870,
  // Specify the end time of the time range during which the ts files that you want to query are generated in the Unix timestamp format.
  endTime: 1460465877
  // The value of EndTime must be greater than the value of StartTime. The duration between EndTime and StartTime must be less than one day.
}).then(r=>console.log(r))
```


List specified LiveChannels

The following code provides an example on how to list specified LiveChannels:

```
const OSS = require('ali-oss');
const store = new OSS({
  // Set yourRegion to the region in which the bucket is located. For example, if your bucket is located in the China (Hangzhou) region, set yourRegion to oss-cn-hangzhou.
  region: 'yourRegion',
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
  accessKeyId: 'yourAccessKeyId',
  accessKeySecret: 'yourAccessKeySecret',
  // Specify the bucket name.
  bucket: 'examplebucket',
})
const r = await this.store.listChannels({
  // List LiveChannels whose names contain the 'my' prefix.
  prefix: 'my',
  // Specify that up to three LiveChannels can be returned.
  'max-keys': 3
}).then(r=>console.log(r))
```

Query the stream ingestion records of a LiveChannel

You can call the GetLiveChannelHistory operation to query the stream ingestion records of the specified LiveChannel. The response to a GetLiveChannelHistory request includes up to the 10 most recent stream ingestion records of the specified LiveChannel.

The following code provides an example on how to query the stream ingestion records of a LiveChannel:

```
const OSS = require('ali-oss');
const store = new OSS({
  // Set yourRegion to the region in which the bucket is located. For example, if your bucket is located in the China (Hangzhou) region, set yourRegion to oss-cn-hangzhou.
  region: 'yourRegion',
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
  accessKeyId: 'yourAccessKeyId',
  accessKeySecret: 'yourAccessKeySecret',
  // Specify the bucket name.
  bucket: 'examplebucket',
})
// Specify the name of the LiveChannel.
const cid = 'mychannel';
// Query the stream ingestion records of the LiveChannel.
store.getChannelHistory(cid).then(r=>console.log(r))
```

Delete a LiveChannel

Notice

- If you send a DeleteLiveChannel request to delete a LiveChannel to which a client is ingesting streams, the request fails.
- The DeleteLiveChannel operation deletes only the LiveChannel and does not delete the files that are generated by the streams ingested to the LiveChannel.

The following code provides an example on how to delete a LiveChannel:

```
const OSS = require('ali-oss');
const store = new OSS({
  // Set yourRegion to the region in which the bucket is located. For example, if your bucket is located in the China (Hangzhou) region, set yourRegion to oss-cn-hangzhou.
  region: 'yourRegion',
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
  accessKeyId: 'yourAccessKeyId',
  accessKeySecret: 'yourAccessKeySecret',
  // Specify the bucket name.
  bucket: 'examplebucket',
})
// Specify the name of the LiveChannel.
const cid = 'mychannel';
// Delete the LiveChannel.
store.deleteChannel(cid).then(r=>console.log(r))
```

References

- For more information about the API operation that you can call to create a LiveChannel, see [PutLiveChannel](#).
- For more information about the API operation that you can call to query the configurations of a LiveChannel, see [GetLiveChannelInfo](#).
- For more information about the API operation that you can call to configure the LiveChannel status, see [PutLiveChannelStatus](#).
- For more information about the API operation that you can call to query the stream ingestion status of a LiveChannel, see [GetLiveChannelStat](#).
- For more information about the API operation that you can call to generate a playlist for a LiveChannel, see [PostVodPlaylist](#).
- For more information about the API operation that you can call to query the playlist of a LiveChannel, see [GetVodPlaylist](#).
- For more information about the API operation that you can call to query the stream ingestion records of a LiveChannel, see [GetLiveChannelHistory](#).
- For more information about the API operation that you can call to delete a LiveChannel, see [DeleteLiveChannel](#).

4.12. Single-connection bandwidth throttling

This topic describes how to add parameters in an object upload or download request to set the limit of upload or download bandwidth. This ensures sufficient bandwidth for other applications.

Configure bandwidth throttling for simple uploads and downloads

The following code provides an example on how to configure bandwidth throttling for simple upload and download of objects:

```
const OSS = require('ali-oss')
const client = new OSS({
  bucket: '<Your BucketName>',
  // The endpoint of the China (Hangzhou) region is used in this example. Specify the actual endpoint.
  region: '<Your Region>',
  // Security risks may arise if you log on with the AccessKey pair of an Alibaba Cloud account because the account has permissions on all API operations. We recommend that you use your RAM user's credentials to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
  accessKeyId: '<Your AccessKeyId>',
  accessKeySecret: '<Your AccessKeySecret>',
});
// Configure the speed limit by using the request header.
const headers = {
  'x-oss-traffic-limit': 8 * 1024 * 100 // Set the minimum bandwidth to 100 KB/s.
}
// Configure bandwidth throttling for the object to be uploaded.
async function put() {
  const result = await client.putStream('<file name>', '<file stream>', {
    headers,
    timeout: 60000 // Set the default timeout duration to 60000. Unit:ms. When the speed of uploading objects exceeds 60000 ms, the error message is reported. We recommend that you modify the timeout duration.
  })
  console.log(result)
}
put()
// Configure bandwidth throttling for the object to be downloaded.
async function get() {
  const result = await client.get('<file name>', {
    headers,
    timeout: 60000 // Set the default timeout duration to 60000. Unit:ms. When the speed of downloading objects exceeds 60000 ms, the error message is reported. We recommend that you modify the timeout duration.
  })
  console.log(result)
}
get()
```

Configure bandwidth throttling when you use signed URLs to upload or download objects

The following code provides an example on how to configure single-connection bandwidth throttling when you use signed URLs to upload or download objects:

```
const OSS = require('ali-oss')
const client = new OSS({
  bucket: '<Your BucketName>',
  // The endpoint of the China (Hangzhou) region is used in this example. Specify the actual endpoint.
  region: '<Your Region>',
  // Security risks may arise if you log on with the AccessKey pair of an Alibaba Cloud account because the account has permissions on all API operations. We recommend that you use your RAM user's credentials to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
  accessKeyId: '<Your AccessKeyId>',
  accessKeySecret: '<Your AccessKeySecret>',
});
// Configure bandwidth throttling for the object to be uploaded by using URL Query.
async function putByQuery() {
  const url = client.signatureUrl('<file name>', {
    trafficLimit: 8 * 1024 * 100, // Set the minimum bandwidth to 100 KB/s.
    method: 'PUT' // Configure the PUT request method.
  })
  const result = await client.urlLib.request(url, {
    method: 'PUT',
    stream: '<file stream>'
    timeout: 60000 // Set the default timeout duration to 60000. Unit:ms. We recommend that you modify the timeout duration after you configure bandwidth throttling. Otherwise, the request fails.
  });
  console.log(result)
}
putByQuery()
// Configure bandwidth throttling for the object to be uploaded by using URL Query.
async function getByQuery() {
  const url = client.signatureUrl('<file name>', {
    trafficLimit: 8 * 1024 * 100, // Set the minimum bandwidth to 100 KB/s.
  })
  const result = await client.urlLib.request(url, {
    timeout: 60000 // Set the default timeout duration to 60000. Unit:ms. We recommend that you modify the timeout duration after you configure bandwidth throttling. Otherwise, the request fails.
  });
  console.log(result)
}
getByQuery()
```

4.13. Server-side encryption

Configure server-side encryption for a bucket

```
const OSS = require("ali-oss");
const store = new OSS({
  region: "<yourRegion>",
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to
  // log on to OSS because the account has permissions on all API operations. We recommend that
  // you use your RAM user's credentials to call API operations or perform routine operations and
  // maintenance. To create a RAM user, log on to the RAM console.
  accessKeyId: "<yourAccessKeyId>",
  accessKeySecret: "<yourAccessKeySecret>",
  bucket: "<yourBucketName>"
});
async function putBucketEncryption() {
  try {
    // Configure an encryption method for the bucket.
    let result = await store.putBucketEncryption("bucket-name", {
      SSEAlgorithm: "AES256", // The AES-256 encryption method is configured here as an example.
      // To use KMS for encryption, you must configure KMSMasterKeyId.
      // KMSMasterKeyId: "<yourKMSMasterKeyId>". Configure the CMK ID when SSEAlgorithm is
      // set to KMS and a specified CMK is used for encryption. In other cases, this element must be
      // set to null.
    });
    console.log(result);
  } catch (e) {
    console.log(e);
  }
}
putBucketEncryption();
```

For more information about how to configure server-side encryption for a bucket, see [PutBucketEncryption](#).

Query the server-side encryption configurations of a bucket

```
const OSS = require("ali-oss");
const store = new OSS({
  region: "<yourRegion>",
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to
  // log on to OSS because the account has permissions on all API operations. We recommend that
  // you use your RAM user's credentials to call API operations or perform routine operations and
  // maintenance. To create a RAM user, log on to the RAM console.
  accessKeyId: "<yourAccessKeyId>",
  accessKeySecret: "<yourAccessKeySecret>",
  bucket: "<yourBucketName>"
});

async function getBucketEncryption() {
  try {
    let result = await store.getBucketEncryption("bucket-name");
    console.log(result);
  } catch (e) {
    console.log(e);
  }
}

getBucketEncryption();
```

For more information about how to query the server-side encryption configurations of a bucket, see [GetBucketEncryption](#).

Delete the server-side encryption configurations of a bucket

```
const OSS = require("ali-oss");
const store = new OSS({
  region: "<yourRegion>",
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to
  // log on to OSS because the account has permissions on all API operations. We recommend that
  // you use your RAM user's credentials to call API operations or perform routine operations and
  // maintenance. To create a RAM user, log on to the RAM console.
  accessKeyId: "<yourAccessKeyId>",
  accessKeySecret: "<yourAccessKeySecret>",
  bucket: "<yourBucketName>"
});

async function deleteBucketEncryption() {
  try {
    // Delete the server-side encryption configurations of the bucket.
    let result = await store.deleteBucketEncryption("bucket-name");
    console.log(result);
  } catch (e) {
    console.log(e);
  }
}

deleteBucketEncryption();
```

For more information about how to delete the server-side encryption configurations of a bucket, see [DeleteBucketEncryption](#).

4.14. Authorized access

This topic describes how to authorize temporary access to Object Storage Service (OSS) by using Security Token Service (STS) or a signed URL.

 **Notice** A validity period must be set for an STS temporary access credential and a signed URL. When you use an STS temporary account credential to generate a signed URL that is used to perform operations such as object upload and download, the minimum validity period takes precedence. For example, you can set the validity period of your STS temporary access credential to 1,200 seconds and the validity period of your signed URL generated by using this credential to 3,600 seconds. In this case, the signed URL cannot be used to upload objects after the STS temporary access credential expires, even if the signed URL is within its validity period.

Use STS for temporary access authorization

You can use Alibaba Cloud STS to authorize temporary access to OSS. STS is a web service that provides temporary access tokens for users. You can use STS to grant an access credential that has a custom validity period and custom permissions for a third-party application or a Resource Access Management (RAM) user managed by you. For more information about STS, see [What is STS?](#).

STS has the following benefits:

- You need only to generate an access token and send the access token to a third-party application, instead of exposing your AccessKey pair to the third-party application. You can customize the access permissions and validity period of this token.
- The access token automatically expires after the validity period. Therefore, you do not need to manually revoke the permissions of an access token.

 **Note** For more information about how to configure STS, see [Use a temporary access credential provided by STS to access OSS](#). You can call the [AssumeRole](#) operation or use STS SDKs for various programming languages to obtain a temporary access credential. For more information, see [STS SDK overview](#). The temporary access credential contains a security token and a temporary AccessKey pair. The AccessKey pair consists of an AccessKey ID and an AccessKey secret. The minimum validity period of a temporary access credential is 900 seconds. The maximum validity period of a temporary access credential is the maximum session duration specified for the current role. For more information, see [Specify the maximum session duration for a RAM role](#).

The following code provides an example on how to use STS to generate a temporary access credential.

 **Notice** To use the express module, make sure that the module is installed. To install the express module, run `npm install express --save`.

```
// Use STS to generate a temporary access credential. The temporary access credential contains a security token and a temporary AccessKey pair. The AccessKey pair consists of an AccessKey ID and an AccessKey secret.
const app = express();
const { STS } = require('ali-oss');
app.get('/sts', (req, res) => {
  let sts = new STS({
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
    accessKeyId: 'yourAccessKeyId',
    accessKeySecret: 'yourAccessKeySecret'
  });
  sts.assumeRole('roleArn', 'policy', 'expiration', 'sessionName').then((result) => {
    console.log(result);
    res.set('Access-Control-Allow-Origin', '*');
    res.set('Access-Control-Allow-METHOD', 'GET');
    res.json({
      AccessKeyId: result.credentials.AccessKeyId,
      AccessKeySecret: result.credentials.AccessKeySecret,
      SecurityToken: result.credentials.SecurityToken,
      Expiration: result.credentials.Expiration
    });
  }).catch((err) => {
    console.log(err);
    res.status(400).json(err.message);
  });
});
```

The following code provides an example on how to use the temporary access credential to initialize an `OSSClient` instance on the client and then use the instance to temporarily access OSS resources:


```

const axios = require("axios");
const OSS = require("ali-oss");
// Use the temporary access credential to initialize the OSSClient instance on the client and then use the instance to temporarily access OSS resources.
const getToken = async () => {
  // Specify the address that the client uses to request the credential.
  await axios.get("https://xxx/sts").then((token) => {
    const client = new OSS({
      // Set yourRegion to the endpoint of the region in which the bucket is located. For example, if your bucket is located in the China (Hangzhou) region, set yourRegion to oss-cn-hangzhou.
      region: 'yourRegion',
      accessKeyId: token.AccessKeyId,
      accessKeySecret: token.AccessKeySecret,
      stsToken: token.SecurityToken,
      // Specify the name of the bucket. Example: examplebucket.
      bucket: "examplebucket",
      // Refresh the temporary access credential.
      refreshSTSToken: async () => {
        const refreshToken = await axios.get("https://127.0.0.1/sts");
        return {
          accessKeyId: refreshToken.AccessKeyId,
          accessKeySecret: refreshToken.AccessKeySecret,
          stsToken: refreshToken.SecurityToken,
        };
      },
    });
  });
};

```

The following table describes the STS-related parameters. For more information, see [AssumeRole](#).

Parameter	Type	Required	Example	Description
roleArn	String	Yes	acs:ram::17464958*****:role/osssts-test	The Alibaba Cloud Resource Name (ARN) of the RAM role. Format: <pre>acs:ram::\$accountID:role/\$roleName</pre> In this format, \$accountID specifies the ID of the Alibaba Cloud account and \$roleName specifies the name of the RAM role. To obtain the ARN information of the RAM role, log on to the RAM console. In the left-side navigation pane, click Roles. On the Roles page, search for and click the created RAM role. On the RAM role details page, you can view and copy the ARN information.

Parameter	Type	Required	Example	Description
policy	String	No	null	The policy that specifies the permissions of the returned STS credential. You can specify an additional policy when you specify parameters to generate an STS temporary access credential. This way, you can further limit the permissions of the credential. If you do not specify policies, the returned temporary access credential has full permissions of the specified RAM role. The value of the parameter must be 1 to 1,024 characters in length.
expiration	Number	No	3600	The validity period of the temporary access credential. Unit: seconds. Default value: 3600. Minimum value: 900. Maximum value: the value of the MaxSessionDuration parameter.  Note You can call the <code>CreateRole</code> or <code>UpdateRole</code> operation to set the <code>MaxSessionDuration</code> parameter. For more information, see CreateRole or UpdateRole.
sessionName	String	Yes	Alice	The role session name that you can customize to identify the RAM user for which an access credential is generated. You can audit the access of users that use different security tokens based on the value of this parameter. The role session name must be 2 to 32 characters in length and can contain letters, digits, periods (.), at signs (@), hyphens (-), and underscores (_).

Use a signed URL for temporary access authorization

The following section provides examples on how to use a signed URL to authorize temporary access:

- Generate a signed URL

You can generate a signed URL and provide the URL to a visitor for temporary access. When you generate a signed URL, you can specify the validity period of the URL to limit the period of access from visitors. By default, the validity period of a signed URL is 3,600 seconds. The maximum validity period of a signed URL is 32,400 seconds.

- Use the signed URL to upload and download an object

 **Note** `name {String}` specifies the name of the object stored in OSS. `[expires] {Number}` specifies the validity period of the URL. By default, the validity period is 1,800 seconds. For more information about other parameters, visit [Git Hub](#).

The following code provides an example on how to generate a signed URL for object uploads and downloads:

```
const OSS = require('ali-oss');
const client = new OSS({
  // Set yourRegion to the endpoint of the region in which the bucket is located. For example, if your bucket is located in the China (Hangzhou) region, set yourRegion to oss-cn-hangzhou.
  region: 'yourRegion',
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
  accessKeyId: 'yourAccessKeyId',
  accessKeySecret: 'yourAccessKeySecret',
  // Specify the bucket name.
  bucket: 'examplebucket'
});
// Obtain the signed URL used to download the exampleobject.txt object. By default, if you use the signed URL in a browser to access the object, the object is previewed but not downloaded.
// Specify the full path of the object. The path cannot contain the bucket name.
const url = client.signatureUrl('exampleobject.txt');
console.log(url);
// Obtain the signed URL used to download the exampleobject.txt object, and set the Content-Disposition header to attachment. This way, if you use the signed URL to access the object in a browser, the object is automatically downloaded, and you can specify the name of the downloaded object.
// To preview the object when you use the signed URL to access the object in a browser, set the Content-Disposition header to inline and use the custom domain name that is mapped to the bucket to access the object.
const filename = 'ossdemo.txt' // Customize the name of the downloaded object.
const response = {
  'content-disposition': `attachment; filename=${encodeURIComponent(filename)}`
}
// Specify the full path of the object. The path cannot contain the bucket name.
const url = client.signatureUrl('exampleobject.txt', { response });
console.log(url);
// Obtain the signed URL used to upload the exampleobject.txt object and set the validity period of the URL.
// Specify the full path of the object. The path cannot contain the bucket name.
const url = client.signatureUrl('exampleobject.txt', {
  // Set the validity period of the signed URL. By default, the validity period is 1,800
```

```
seconds.
    expires: 3600,
    // Set the request method to PUT. By default, the request method is GET.
    method: 'PUT'
  });
console.log(url);
// Obtain the signed URL used to upload the exampleobject.txt object and set Content-Type
.
// Specify the full path of the object. The path cannot contain the bucket name.
const url = client.signatureUrl('exampleobject.txt', {
  expires: 3600,
  method: 'PUT',
  'Content-Type': 'text/plain; charset=UTF-8',
});
console.log(url);
```

- Generate a signed URL that includes Image Processing (IMG) parameters

The following code provides an example on how to generate a signed URL that includes IMG parameters:

```
let OSS = require('ali-oss');
let store = new OSS({
  // Set yourRegion to the endpoint of the region in which the bucket is located. For example, if your bucket is located in the China (Hangzhou) region, set yourRegion to oss-cn-hangzhou.
  region: 'yourRegion',
  // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
  accessKeyId: 'yourAccessKeyId',
  accessKeySecret: 'yourAccessKeySecret',
  // Specify the bucket name.
  bucket: 'examplebucket'
});
// Obtain the signed URL used to process an image object named exampleobject.png.
// Specify the full path of the object. The path cannot contain the bucket name.
const url = store.signatureUrl('exampleobject.png', {
  process: 'image/resize,w_200' // Set IMG parameters used to process the image.
});
console.log(url);
// Obtain the signed URL used to process the exampleobject.png image and set the validity period of the URL.
// Specify the full path of the object. The path cannot contain the bucket name.
const url = store.signatureUrl('exampleobject.png', {
  // Set the validity period of the signed URL. By default, the validity period is 1,800 seconds.
  expires: 3600,
  // Set IMG parameters used to process the image.
  process: 'image/resize,w_200'
});
console.log(url);
```

4.15. IMG

Image Processing (IMG) is an image processing service that is provided by Object Storage Service (OSS). The service is secure, cost-effective, and highly reliable and can be used to process large numbers of images. After you upload source images to OSS, you can call RESTful API operations to process the images on a device that is connected over the Internet anytime, anywhere.

For more information about IMG parameters, see [Parameters](#).

Usage of image processing

You can use standard HTTP GET requests to access IMG. You can configure IMG parameters in QueryString of a URL.

If the access control list (ACL) of an image object is private, only authorized users can access the image.

- Anonymous access

If the ACL of an image object is `public-read`, as described in the following table, anonymous users can access the image object.

Bucket ACL	Object ACL
public-read or public-read-write	Default
Any ACL	public-read or public-read-write

You can anonymously access a processed image by using a third-level domain in the following format:

```
http://<yourBucketName>.<yourEndpoint>/<yourObjectName>?x-oss-process=image/<yourAction>,<yourParamValue>
```

Parameter	Description
bucket	The name of the bucket.
endpoint	The endpoint that is used to access the region in which the bucket is located.
object	The name of the image object.
image	The identifier reserved by IMG.
action	The operations that are performed on the image such as resizing, cropping, and rotating.
param	The parameters that correspond to the operations that are performed on the image.

- Basic operations

For example, you can resize an image to a width of 100 pixels and adjust the height based on the ratio by using the following URL:

```
http://image-demo.oss-cn-hangzhou.aliyuncs.com/example.jpg?x-oss-process=image/resize,w_100
```

- Custom image styles

You can anonymously access IMG by using a third-level domain in the following format:

```
http://<yourBucketName>.<yourEndpoint>/<yourObjectName>?x-oss-process=style/<yourStyleName>
```

- style: the identifier reserved by IMG for the custom style.
- yourStyleName: the name of the custom style. The name is specified by the Rule Name parameter when you create the custom style in the OSS console.

Example:

```
http://image-demo.oss-cn-hangzhou.aliyuncs.com/example.jpg?x-oss-process=style/oss-pic-style-w-100
```

- Cascade processing

You can use cascade processing to perform multiple operations in sequence on an image by using a URL in the following format:

```
http://<yourBucketName>.<yourEndpoint>/<yourObjectName>?x-oss-process=image/<yourAction1>,<yourParamValue1>/<yourAction2>,<yourParamValue2>/...
```

Example:

```
http://image-demo.oss-cn-hangzhou.aliyuncs.com/example.jpg?x-oss-process=image/resize,w_100/rotate,90
```

- Access by using HTTPS

IMG supports access by using HTTPS. Example:

```
https://image-demo.oss-cn-hangzhou.aliyuncs.com/example.jpg?x-oss-process=image/resize,w_100
```

- Authorized access

If the ACL of an image object is private, you must have permissions on the image object before you can perform operations on the image object.

Bucket ACL	Object ACL
Private	Default
Any ACL	Private

The following code provides an example on how to generate a signed URL for IMG:

```
const OSS = require('ali-oss');
const client = new OSS({
  bucket: '<Your BucketName>',
  // Specify the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the region to oss-cn-hangzhou.
  region: '<Your Region>',
  // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
  accessKeyId: '<Your AccessKeyId>',
  accessKeySecret: '<Your AccessKeySecret>',
});
// Set the validity period to 10 minutes. The image processing style is "image/resize,w_300".
let signUrl = client.signatureUrl('example.jpg', {expires: 600, 'process': 'image/resize,w_300'});
console.log("signUrl="+signUrl);
```

Note

- Authorized access supports custom styles, HTTPS, and cascade processing.
- The validity period (expires) is measured in seconds.

• Access by using OSS SDKs

You can use an OSS SDK to access and process an image object.

The SDK allows you to specify custom image styles, access images over HTTPS, and perform cascade processing.

◦ Basic operations

The following code provides an example on how to perform basic operations on an image:

```
const OSS = require('ali-oss');
const client = new OSS({
  bucket: '<Your BucketName>',
  // Specify the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the region to oss-cn-hangzhou.
  region: '<Your Region>',
  // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
  accessKeyId: '<Your AccessKeyId>',
  accessKeySecret: '<Your AccessKeySecret>',
});
// Resize the image.
async function scale () {
  try {
    let result = await client.get('example.jpg', './example-resize.jpg', { process: 'image/resize,m_fixed,w_100,h_100'});
  } catch (e) {
    console.log(e);
  }
}
```

```
    }
  }
  // Crop the image.
  async function cut () {
    try {
      let result = await client.get('example.jpg', './example-crop.jpg', { process: 'image/crop,w_100,h_100,x_100,y_100,r_1' });
    } catch (e) {
      console.log(e)
    }
  }
  // Rotate the image.
  async function rotate () {
    try {
      let result = await client.get('example.jpg', './example-rotate.jpg', { process: 'image/rotate,90' });
    } catch (e) {
      console.log(e);
    }
  }
  // Sharpen the image.
  async function sharpen () {
    try {
      let result = yield client.get('example.jpg', './example-sharpen.jpg', { process: 'image/sharpen,100' });
    } catch (e) {
      console.log(e);
    }
  }
  // Add watermarks to the image.
  async function watermark () {
    try {
      let result = yield client.get('example.jpg', './example-watermark.jpg', { process: 'image/watermark,text_SGVsbG8g5Zu-54mH5pyN5YqhIQ' });
    } catch (e) {
      console.log(e);
    }
  }
  // Convert the format of the image.
  async function format () {
    try {
      let result = await client.get('example.jpg', './example-format.jpg', { process: 'image/format,png' });
    } catch (e) {
      console.log(e);
    }
  }
  // Obtain information about the image.
  async function info () {
    try {
      let result = await client.get('example.jpg', './example-info.txt', {process: 'image/info'});
    } catch (e) {
      console.log(e);
    }
  }
}
```



```
}  
}
```

- Custom image styles

 **Note** You must first log on to the OSS console to create the custom style "example-style".

The following code provides an example on how to specify a custom image style:

```
const OSS = require('ali-oss');  
const client = new OSS({  
  bucket: '<Your BucketName>',  
  // Specify the region in which the bucket is located. For example, if the bucket is l  
  ocated in the China (Hangzhou) region, set the region to oss-cn-hangzhou.  
  region: '<Your Region>',  
  // The AccessKey pair of an Alibaba Cloud account has permissions on all API operatio  
  ns. Using these credentials to perform operations in OSS is a high-risk operation. We r  
  ecommend that you use a RAM user to call API operations or perform routine O&M. To crea  
  te a RAM user, log on to the RAM console.  
  accessKeyId: '<Your AccessKeyId>',  
  accessKeySecret: '<Your AccessKeySecret>',  
});  
// Specify the custom image style.  
async function style () {  
  try {  
    let result = await client.get('example.jpg', './example-custom-style.jpg', {process  
: 'style/example-style'});  
  } catch (e) {  
    console.log(e);  
  }  
}  
style();
```

- Cascade processing

The following code provides an example on how to perform cascade processing on an image:

```
const OSS = require('ali-oss');
const client = new OSS({
  bucket: '<Your BucketName>',
  // Specify the region in which the bucket is located. For example, if the bucket is l
  ocated in the China (Hangzhou) region, set the region to oss-cn-hangzhou.
  region: '<Your Region>',
  // The AccessKey pair of an Alibaba Cloud account has permissions on all API operatio
  ns. Using these credentials to perform operations in OSS is a high-risk operation. We r
  ecommend that you use a RAM user to call API operations or perform routine O&M. To crea
  te a RAM user, log on to the RAM console.
  accessKeyId: '<Your AccessKeyId>',
  accessKeySecret: '<Your AccessKeySecret>',
});
// Perform cascade processing.
async function cascade () {
  try {
    let result = await client.get('example.jpg', './example-cascade.jpg', {process: 'im
age/resize,m_fixed,w_100,h_100/rotate,90'});
  } catch (e) {
    console.log(e);
  }
}
cascade();
```

Save processed images

The following code provides an example on how to save processed images:

```
const OSS = require('ali-oss');
const client = new OSS({
  bucket: '<Your BucketName>',
  // Specify the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the region to oss-cn-hangzhou.
  region: '<Your Region>',
  // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
  accessKeyId: '<Your AccessKeyId>',
  accessKeySecret: '<Your AccessKeySecret>',
});
const sourceImage = 'sourceObject.png';
const targetImage = 'targetObject.jpg';
async function processImage(processStr, targetBucket) {
  const result = await client.processObjectSave(
    sourceImage,
    targetImage,
    processStr,
    targetBucket
  );
  console.log(result.res.status);
}
// Resize the image.
processImage("image/resize,m_fixed,w_100,h_100")
// Crop the image.
processImage("image/crop,w_100,h_100,x_100,y_100,r_1")
// Rotate the image.
processImage("image/rotate,90")
// Sharpen the image.
processImage("image/sharpen,100")
// Add watermarks to the image.
processImage("image/watermark,text_SGVsbG8g5Zu-54mH5pyN5YqhIQ")
// Convert the format of the image.
processImage("image/format,jpg")
// Convert the format of the image and configure the destination bucket that is used to save the processed image.
processImage("image/format,jpg", "<target bucket>")
```

4.16. Error handling

If a request error occurs when you use Object Storage Service (OSS) SDK for Node.js, an exception is thrown. Detailed information about the error is recorded in a log file. By default, the detailed information about the error is recorded in the `oss_sdk.log` file.

ClientError

ClientErrors indicate exceptions occurred inside the SDK. For example, parameter setting errors and errors occurred when objects are modified in the resumable upload or download process.

RequestError

RequestErrors are thrown by SDK for Node.js when an interruption or exception occurs on the network. Check the network connectivity and make sure that the network is resumed before you use OSS SDK for Node.js again.

ServerError

ServerErrors are the parsing results of the error messages generated by the server and indicate errors that occurred on the server. ServerErrors include the following attributes:

- Status: the HTTP status code of the error request.
- Code: the OSS error code.
- Message: the OSS error message.
- RequestId: the UUID of the request. If the error persists, you can provide the request ID to seek help from technical support.

For more information about common error messages in OSS, see [Error responses](#).

Debugging

If ClientErrors or ServerErrors occur when you use OSS SDK for Node.js, enable debugging by setting the `DEBUG` environment variable.

```
DEBUG=ali-oss node app.js
```

In a browser environment, set the `localStorage.debug` variable in the console to enable debugging.

```
localStorage.debug = 'ali-oss'
```

4.17. FAQ

This topic describes common problems that may occur in the use of OSS Node.js SDK and the solutions for the problems.

How to enable HTTPS access

To enable HTTPS access, specify the value of the `secure` parameter to `true` when initializing the SDK.

How to obtain the upload progress

You can specify the `progress` parameter to obtain the upload progress when performing [multipart upload](#) operations.

How to obtain the download progress

The download progress is calculated based on the size of the download traffic in OSS Node.js SDK.

How to upload Base64-encoded images

Convert the images to File objects and upload them to the OSS server by calling APIs.

```
function dataURLtoFile(dataurl, filename) {
    let arr = dataurl.split(','), mime = arr[0].match(/:(.*?);/)[1],
        bstr = atob(arr[1]), n = bstr.length, u8arr = new Uint8Array(n);
    while(n--){
        u8arr[n] = bstr.charCodeAt(n);
    }
    return new File([u8arr], filename, {type:mime});
}
let file = dataURLtoFile('<base64 content>', '');
client.multipartUpload('<oss file name>', file).then( (res)=> {
    console.log(res)
}).catch((err) => {
    console.log(err)
});
```

How to upload a file to a specified directory

You only need to add the prefix of the directory to the name of the object that you want to upload. For more information, see: [Comparison of OSS and file system](#).

```
let OSS = require('ali-oss')
let client = new OSS({
    region: '<Your region>',
    accessKeyId: '<Your AccessKeyId>',
    accessKeySecret: '<Your AccessKeySecret>',
    bucket: 'Your bucket name'
});
client.multipartUpload('base-dir/' + 'object-name', 'local-file', {
    progress: async function (p) {
        console.log('Progress: ' + p);
    }
});
console.log(result);
}).then((res) => {
    console.log(res)
}).catch((err) => {
    console.log(err);
});
```

How to obtain the signed URL of an object

You can call the `signatureUrl` method to obtain the signed URL used to download an object. For more information, see the [signatureUrl](#) part in the OSS SDK document.

Common errors

- [Error handling](#)
- [Error responses](#)

5. Python

5.1. Preface

This topic provides the sample code of Object Storage Service (OSS) SDK for Python in various use cases.

Source code

For more information about the source code of OSS SDK for Python, visit [GitHub](#).

Sample code

OSS SDK for Python provides various sample code files for your reference or use. The following table describes the sample code files provided by OSS SDK for Python.

Sample code file	Content
bucket.py	- Basic operations on a bucket, including creating a bucket, deleting buckets, listing buckets, and obtaining bucket information Static website hosting and Manage lifecycle rules
bucket_inventory.py	Bucket inventory
bucket_policy.py	Bucket policy
object_request_payment.py	Enable pay-by-requester for buckets
object_basic.py	Quick start , including creating a bucket and uploading, downloading, listing, and deleting objects
object_extra.py	Upload and manage objects, including managing object metadata , copying objects , and append upload .
upload.py	Upload objects, including resumable upload and multipart upload .
download.py	Download objects, including streaming download , downloading an object to a local file , range download , and resumable download .
object_progress.py	Upload progress bars and Download progress bars
object_callback.py	Upload callback after an object is uploaded
object_post.py	Operations related to form upload
object_basic.py	List objects and Delete objects

Sample code file	Content
select_csv.py	Query objects
traffic_limit.py	Single-connection bandwidth throttling for object upload and download
object_crypto.py	Client-side encryption
object_server_crypto.py	Server-side encryption
sts.py	Authorized access and use of Security Token Service (STS), including obtaining a temporary credential by assuming a RAM role and using the obtained temporary credential to access OSS
object_check.py	Data verification methods (including MD5 and CRC) during object upload and download and data security
image.py	Operations related to Image Processing (IMG)
live_channel.py	Operations related to LiveChannel

5.2. Installation

This topic describes how to install Object Storage Service (OSS) SDK for Python.

Environment preparation

The version of OSS SDK for Python in this topic is compatible with Python 2.6, 2.7, 3.3, 3.4, 3.5, 3.6, 3.7, and 3.8.

- Environment requirements

For more information about how to install the appropriate version of Python, visit [python](#).

 **Notice** When you install OSS SDK for Python on a computer that runs the Windows operating system, make sure that the version of Visual C++ is 15.0 or later.

- Check the Python version

Run the `python -V` command to check the Python version.

In Windows, if a message appears, which indicates that it is not an internal or external command, edit the `Path` environment variable. Add the installation path of Python and pip. The installation path of pip is the Scripts directory in the installation path of Python.

 **Notice** You may need to restart your computer to ensure that the environment variables take effect.

Download OSS SDK for Python

- [Download from GitHub](#)
- [Download previous versions](#)

For more information, see [API Reference](#).

Install python-devel

OSS SDK for Python uses the `crcmod` library to calculate the value of cyclic redundancy check (CRC). The `crcmod` library depends on the `Python.h` file. If the `Python.h` file is missing, SDK installation is not affected but `crcmod` extensions in C fail to be installed. Therefore, the efficiency of operations such as object upload and download is low. Install `python-devel` if it does not exist.

- When you install Python in Windows or Mac OS X systems, the `Python.h` file is installed together with Python. Therefore, you do not need to install `python-devel`.
- When you install Python in CentOS, RHEL, or Fedora systems, run the following command to install `python-devel`:

```
yum install python-devel
```

- When you install Python in Debian or Ubuntu systems, run the following command to install `python-devel`:

```
apt-get install python-dev
```

Install OSS SDK for Python

- Use pip

Run the following command:

```
pip install oss2
```

 **Note** For Python 2.7.9 and later and Python 3.4 and later, pip is installed by default. If you need to install pip, visit [Installation](#).

- Use the source code

Visit [GitHub](#) and download the OSS SDK for Python package of an appropriate version. Decompress the package. Check whether the `setup.py` file exists in the directory and then run the following command:

```
python setup.py install
```

Verify the SDK version

- Verify the version of OSS SDK for Python
 - i. Enter `python` in the command line and press the Enter key to go to the Python environment.
 - ii. Run the following command to verify the version of OSS SDK for Python:

```
>>> import oss2
>>> oss2.__version__
'2.0.0'
```


If an output similar to the preceding example appears, OSS SDK for Python V2.0.0 is installed.

- Verify `crcmod`

OSS SDK for Python uses the following methods to calculate the value of CRC: the C extension mode, which uses the `crcmod` dependent library to call the `_crcfunext.so` extension library to calculate the CRC value, and the Python-only mode. The performance of the C extension mode is superior to the Python-only mode. The `crcmod` library is automatically installed together with `oss2`. For more information about `crcmod`, visit [crcmod introduction](#).

When you use OSS SDK for Python, if the object upload and download speed remains much lower than that of other tools such as `ossutil` or other SDKs, `crcmod` extensions in C may fail to install. In this case, the Python-only mode is used to calculate the value of CRC for object upload and download.

You can use the following methods to determine whether the C extension mode of the `crcmod` library is installed:

- Enter `python` in the command line and press the Enter key to go to the Python environment.
- Enter `import crcmod._crcfunext` and press the Enter key.

- If no error message appears, the `crcmod` extensions in C are installed.
- If the following error message appears, `crcmod` extensions in C failed to install:

```
>>> import crcmod._crcfunext
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
ImportError: No module named _crcfunext
```

When you compile the `crcmod` library, the `_crcfunext.so` file depends on the `Python.h` file. The error occurs because the `Python.h` file does not exist in the system, which causes the `_crcfunext.so` file to fail to be generated. Therefore, the Python-only mode is used to calculate the value of CRC. OSS SDK for Python is installed, but the speed of operations such as object upload and download is low.

For Windows, to resolve this issue, download [crcmod-1.7.win32-py2.7.msi](#) or other versions of the .msi file and install the SDK. During the installation, specify the installation path of `crcmod` as the `Lib\site-packages` directory in the local installation path of the SDK. Example: `D:\python\Lib\site-packages\`. Verify `crcmod` after OSS SDK for Python is installed.

For Linux, perform the following steps to resolve the issue:

- Run the following command to uninstall the `crcmod` library:

```
pip uninstall crcmod
```

- Install `python-devel`. For more information, see [Install python-devel](#).
- Run the following command to re-install the `crcmod` library:

```
pip install crcmod
```

If the `crcmod` library fails to install after you perform the preceding steps, uninstall the `crcmod` library. Run the following command to view the detailed causes of the installation failures:

```
pip install crcmod -v
```

Uninstall OSS SDK for Python

If OSS SDK for Python fails to install, uninstall OSS SDK for Python by using pip and re-install OSS SDK for Python. Run the following code to uninstall OSS SDK for Python:

```
pip uninstall oss2
```

5.3. Initialization

This topic describes how to initialize Object Storage Service (OSS) SDK for Python.

OSS SDK for Python allows you to use `oss2.Service` and `oss2.Bucket` to perform most operations.

- `oss2.oss2.Service` is used to list buckets.
- `oss2.oss2.Bucket` is used to upload, download, and delete objects as well as configure buckets.

To initialize the `oss2.Service` and `oss2.Bucket`, you must specify the endpoint of the region in which the bucket you want to access is located. The `oss2.Service` class cannot be initialized by using custom domain names. For more information about endpoints, see [Regions and endpoints](#) and [Map custom domain names](#).

Initialize the `oss2.Service` class

For more information, see [List buckets](#).

Initialize the `oss2.Bucket` class

- Use the default domain name of a bucket to initialize the class

The following code provides an example on how to use the default domain name of a bucket to initialize the `oss2.Bucket` class:

```
# -*- coding: utf-8 -*-
import oss2

# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a Resource Access Management (RAM) user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if your bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com.
endpoint = 'yourEndpoint'
# Specify the bucket name.
bucket = oss2.Bucket(auth, endpoint, 'examplebucket')
```

- Use a custom domain name that is mapped to a bucket to initialize the class

The following code provides an example on how to use a custom domain name that is mapped to a bucket to initialize the `oss2.Bucket` class:

```
# -*- coding: utf-8 -*-
import oss2
# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# Specify the custom domain name that is mapped to the bucket. Example: example.com.
cname = 'http://example.com'
# Specify the bucket name, and set is_cname to true to enable CNAME. If CNAME is enabled, custom domain names can be mapped to the bucket.
bucket = oss2.Bucket(auth, cname, 'examplebucket', is_cname=True)
```

- Use Security Token Service (STS) to initialize the class

The following code provides an example on how to use STS to initialize the `oss2.Bucket` class.

 **Note** For more information about how to set up STS, see [Use a temporary credential provided by STS to access OSS](#) in OSS Developer Guide. You can call the [AssumeRole](#) operation or use [STS SDKs for various programming languages](#) to obtain a temporary access credential. The temporary access credential contains a security token and a temporary AccessKey pair that consists of an AccessKey ID and an AccessKey secret. The minimum validity period of a temporary access credential is 900 seconds. The maximum validity period of a temporary access credential is the maximum session duration specified for the current role. For more information, see [Specify the maximum session duration for a RAM role](#).

```
# -*- coding: utf-8 -*-
import oss2
# Initialize the StsAuth instance based on the authentication information in the temporary access credential.
auth = oss2.StsAuth('yourAccessKeyId', 'yourAccessKeySecret', 'yourSecretToken')
# Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if your bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com.
endpoint = 'yourEndpoint'
# Specify the name of the bucket. Example: examplebucket.
bucket_name = 'examplebucket'
# Initialize a bucket based on the StsAuth instance.
bucket = oss2.Bucket(auth, endpoint, bucket_name)
```

- Use anonymous access to initialize the class

 **Notice** Anonymous users can read only buckets whose access control list (ACL) is public read and read and write only buckets whose ACL is public read/write. Anonymous users cannot perform `oss2.Service-` and `oss2.Bucket-` related operations.

The following code provides an example on how to initialize the class for the `examplebucket` bucket whose ACL is public read/write:

```
# -*- coding: utf-8 -*-
import oss2
# Set yourEndpoint to the endpoint of the region in which the bucket is located. For exam
ple, if your bucket is located in the China (Hangzhou) region, set yourEndpoint to https:
//oss-cn-hangzhou.aliyuncs.com.
endpoint = 'yourEndpoint'
# Specify the name of the bucket. Example: examplebucket.
bucket_name = 'examplebucket'
# Use anonymous access to initialize the class.
bucket = oss2.Bucket(oss2.AnonymousAuth(), endpoint, bucket_name)
```

Configure parameters when you initialize the oss2.Bucket class

The following table describes the parameters that you can configure when you initialize the oss2.Bucket class.

Parameter	Example	Description	Method
is_cname	True	Specify whether the endpoint is a custom domain name. Default value: False. Valid values: - True: The endpoint that you specify is a custom domain name. - False: The endpoint that you specify is not a custom domain name.	oss2.Bucket(auth, cname, 'examplebucket', is_cname=True)
session	mytestsession	The name of the session, which indicates that a new session is started. If you set this parameter to the name of an existing session, the session is reused.	oss2.Bucket(auth, endpoint, 'examplebucket', session=oss2.Session())
connect_timeout	30	Specify the connection timeout period. Default value: None, which indicates that the connection timeout period is unlimited. Unit: seconds.	oss2.Bucket(auth, endpoint, 'examplebucket', connect_timeout=30)

Parameter	Example	Description	Method
app_name	mytool	The name of the app that uses the SDK. By default, the name of the app is empty. If the parameter is not empty, specify the corresponding value in the User-Agent field.  Notice The string must comply with HTTP standards because the string is transmitted as a value of HTTP headers.	oss2.Bucket(auth, endpoint, 'examplebucket', app_name='mytool')
enable_crc	False	Specify whether to enable cyclic redundancy check (CRC). Default value: True. - True: indicates that CRC is enabled. - False: indicates that CRC is disabled.	oss2.Bucket(auth, endpoint, 'examplebucket', enable_crc=False)

Common examples for the configurations of these parameters:

- Configure the connection timeout period

The following code provides an example on how to configure the connection timeout period:

```
# -*- coding: utf-8 -*-
import oss2
# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if your bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com.
endpoint = 'yourEndpoint'
# Specify the bucket name and set the connection timeout period to 30 seconds.
bucket = oss2.Bucket(auth, endpoint, 'examplebucket', connect_timeout=30)
```

- Disable CRC

By default, CRC is enabled to ensure data integrity when you upload or download objects.

 **Warning** We recommend that you do not disable CRC. If you disable CRC, data integrity may be affected during object upload and download.

The following code provides an example on how to disable CRC:

```
# -*- coding: utf-8 -*-
import oss2
# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if your bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com.
endpoint = 'yourEndpoint'
# Specify the bucket name, and set enable_crc to False to disable CRC.
bucket = oss2.Bucket(auth, endpoint, 'examplebucket', enable_crc=False)
```

5.4. Quick start

This topic describes how to use Object Storage Service (OSS) SDK for Python to perform routine operations such as create buckets, upload objects, and download objects.

Create buckets

A bucket is a global namespace in OSS. A bucket is a container for objects stored in OSS.

Note

The following code provides an example on how to create a bucket:

```
# -*- coding: utf-8 -*-
import oss2
# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('<yourAccessKeyId>', '<yourAccessKeySecret>')
# The endpoint of the China (Hangzhou) region is used in this region. Specify the actual endpoint based on your requirements.
bucket = oss2.Bucket(auth, 'http://oss-cn-hangzhou.aliyuncs.com', '<yourBucketName>')
# Set the access control list (ACL) of the bucket to private.
bucket.create_bucket(oss2.models.BUCKET_ACL_PRIVATE)
```

Upload an object

The following code provides an example on how to upload objects to OSS by using stream upload:

```
# -*- coding: utf-8 -*-
import oss2
# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to acc
ess OSS because the account has permissions on all API operations. We recommend that you us
e a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to
the RAM console.
auth = oss2.Auth('<yourAccessKeyId>', '<yourAccessKeySecret>')
# The endpoint of the China (Hangzhou) region is used in this region. Specify the actual en
dpoint based on your requirements.
bucket = oss2.Bucket(auth, 'http://oss-cn-hangzhou.aliyuncs.com', '<yourBucketName>')
# Upload the object to OSS.
# <yourObjectName> indicates the full path of the object you want to upload to OSS. The ful
l path of the object must include the extension of the object name and exclude the bucket n
ame. Example: abc/efg/123.jpg.
# <yourLocalFile> consists of a local file path and an object name with extension. Example:
/users/local/myfile.txt.
bucket.put_object_from_file('<yourObjectName>', '<yourLocalFile>')
```

Download an object

The following code provides an example on how to download a specified object to a local file:

```
# -*- coding: utf-8 -*-
import oss2
# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to acc
ess OSS because the account has permissions on all API operations. We recommend that you us
e a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to
the RAM console.
auth = oss2.Auth('<yourAccessKeyId>', '<yourAccessKeySecret>')
# The endpoint of the China (Hangzhou) region is used in this region. Specify the actual en
dpoint based on your requirements.
bucket = oss2.Bucket(auth, 'http://oss-cn-hangzhou.aliyuncs.com', '<yourBucketName>')
# Download the object to a local file.
# <yourObjectName> indicates the full path of the object you want to download from OSS. The
full path of the object must include the extension of the object name and exclude the bucke
t name. Example: abc/efg/123.jpg.
# <yourLocalFile> consists of a local file path and an object name with extension. Example:
/users/local/myfile.txt.
bucket.get_object_to_file('<yourObjectName>', '<yourLocalFile>')
```

List objects

The following code provides an example on how to list 10 objects in a specified bucket:

```
# -*- coding: utf-8 -*-
import oss2
from itertools import islice
# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('<yourAccessKeyId>', '<yourAccessKeySecret>')
# The endpoint of the China (Hangzhou) region is used in this region. Specify the actual endpoint based on your requirements.
bucket = oss2.Bucket(auth, 'http://oss-cn-hangzhou.aliyuncs.com', '<yourBucketName>')
# oss2.ObjectIterator is used to traverse objects.
for b in islice(oss2.ObjectIterator(bucket), 10):
    print(b.key)
```

Delete an object

The following code provides an example on how to delete an object:

```
# -*- coding: utf-8 -*-
import oss2
# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('<yourAccessKeyId>', '<yourAccessKeySecret>')
# The endpoint of the China (Hangzhou) region is used in this region. Specify the actual endpoint based on your requirements.
bucket = oss2.Bucket(auth, 'http://oss-cn-hangzhou.aliyuncs.com', '<yourBucketName>')
# <yourObjectName> indicates the full path of the object you want to delete from OSS. The full path of the object must include the extension of the object name and exclude the bucket name. Example: abc/efg/123.jpg.
bucket.delete_object('<yourObjectName>')
```

For more information about how to delete objects, see [Delete objects](#) in [Manage objects](#).

5.5. Buckets

5.5.1. Create buckets

A bucket is a container used to store objects in Object Storage Service (OSS). Each object is contained in a bucket. This topic describes how to create a bucket.

Sample code

The following code provides an example on how to create a bucket named examplebucket:


```
# -*- coding: utf-8 -*-
import oss2

# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com.
# Specify the name of the bucket. Example: examplebucket.
bucket = oss2.Bucket(auth, 'https://oss-cn-hangzhou.aliyuncs.com', 'examplebucket')
# Create the bucket.
# The following code provides an example on how to specify the storage class, access control list (ACL), and redundancy type when you create a bucket.
# In this example, the storage class is Standard, the ACL is private, and the redundancy type is zone-redundant storage (ZRS).
# bucketConfig = oss2.models.BucketCreateConfig(oss2.BUCKET_STORAGE_CLASS_STANDARD, oss2.BUCKET_DATA_REDUNDANCY_TYPE_ZRS)
# bucket.create_bucket(oss2.BUCKET_ACL_PRIVATE, bucketConfig)
bucket.create_bucket()
```

References

- For more information about the complete code that is used to create a bucket, visit [GitHub](#).
- For more information about the API operation that you can call to create a bucket, see [PutBucket](#).

5.5.2. List buckets

A bucket is a container for objects stored in Object Storage Service (OSS). Every object is contained in a bucket. This topic describes how to list all buckets or buckets whose names contain a specified prefix.

List all buckets

The following code provides an example on how to list all buckets that belong to an Alibaba Cloud account:

```
# -*- coding: utf-8 -*-
import oss2

# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# Specify the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com.
service = oss2.Service(auth, 'https://oss-cn-hangzhou.aliyuncs.com')
# List all buckets that belong to the current Alibaba Cloud account.
for b in oss2.BucketIterator(service):
    print(b.name)
```

List buckets whose names contain a specified prefix

The following code provides an example on how to list buckets whose names contain a specified prefix:

```
# -*- coding: utf-8 -*-
import oss2

# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# Specify the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com.
service = oss2.Service(auth, 'https://oss-cn-hangzhou.aliyuncs.com')
# List buckets whose names contain the test- prefix.
for b in oss2.BucketIterator(service, prefix='test-'):
    print(b.name)
```

List buckets whose names are alphabetically after the bucket specified by the marker parameter

The following code provides an example on how to list buckets whose names are alphabetically after examplebucket:

```
# -*- coding: utf-8 -*-
import oss2

# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# Specify the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com.
service = oss2.Service(auth, 'https://oss-cn-hangzhou.aliyuncs.com')
# List the buckets whose names are alphabetically after test-bucket1. The test-bucket1 bucket is not listed.
for b in oss2.BucketIterator(service, marker='test-bucket1'):
    print(b.name)
```

References

- For more information about the complete sample code that is used to list buckets, visit [GitHub](#).
-

5.5.3. Query the regions of buckets

A bucket is a container for objects stored in Object Storage Service (OSS). Every object is contained in a bucket. This topic describes how to query the region of a bucket.

Sample code

The following sample code shows how to query the region or location of a bucket:

```
# -*- coding: utf-8 -*-
import oss2

# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS, because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# Specify the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com.
# Set yourBucketName to the name of the bucket.
bucket = oss2.Bucket(auth, 'https://oss-cn-hangzhou.aliyuncs.com', 'yourBucketName')
# Obtain the region of the bucket.
result = bucket.get_bucket_location()
print('location: ' + result.location)
```

References

- For more information about the complete sample code that is used to query the region of a bucket, visit [Git Hub](#).
- For more information about the API operation that you can call to query the region of a bucket, see [Get Bucket Location](#).

5.5.4. Determine whether a bucket exists

A bucket is a container used to store objects in Object Storage Service (OSS). Every object is contained in a bucket. This topic describes how to determine whether a bucket exists.

```
# -*- coding: utf-8 -*-
import oss2

# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log
on to OSS, because the account has permissions on all API operations. We recommend that you
use a RAM user to call API operations or perform routine operations and maintenance. To cre
ate your RAM user, log on to https://ram.console.aliyun.com.
auth = oss2.Auth('<yourAccessKeyId>', '<yourAccessKeySecret>')
# This example uses the endpoint of the China (Hangzhou) region. Specify the actual endpoin
t based on your requirements.
bucket = oss2.Bucket(auth, 'http://oss-cn-hangzhou.aliyuncs.com', '<yourBucketName>')
def does_bucket_exist(bucket):
    try:
        bucket.get_bucket_info()
    except oss2.exceptions.NoSuchBucket:
        return False
    except:
        raise
    return True
exist = bucket.does_bucket_exist(bucket)
# If the returned value is true, a bucket with the specified name exists. If the returned v
alue is false, a bucket with the specified name does not exist.
if exist:
    print('bucket exist')
else:
    print('bucket not exist')
```

5.5.5. Obtain bucket information

A bucket is a container for objects stored in Object Storage Service (OSS). Every object is contained in a bucket. This topic describes how to query the information about a bucket.

The following code provides an example on how to query the information about a bucket, including the region and creation date of the bucket:

```
# -*- coding: utf-8 -*-
import oss2
# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log
on to OSS, because the account has permissions on all API operations. We recommend that you
use a RAM user to call API operations or perform routine operations and maintenance. To cre
ate your RAM user, log on to https://ram.console.aliyun.com.
auth = oss2.Auth('<yourAccessKeyId>', '<yourAccessKeySecret>')
# This example uses the endpoint of the China (Hangzhou) region. Specify the actual endpoin
t based on your requirements.
bucket = oss2.Bucket(auth, 'http://oss-cn-hangzhou.aliyuncs.com', '<yourBucketName>')
# Obtain bucket information, including the storage class, creation date, ACL, and the type
of disaster recovery.
bucket_info = bucket.get_bucket_info()
print('name: ' + bucket_info.name)
print('storage class: ' + bucket_info.storage_class)
print('creation date: ' + bucket_info.creation_date)
print('intranet_endpoint: ' + bucket_info.intranet_endpoint)
print('extranet_endpoint ' + bucket_info.extranet_endpoint)
print('owner: ' + bucket_info.owner.id)
print('grant: ' + bucket_info.acl.grant)
print('data_redundancy_type:' + bucket_info.data_redundancy_type)
```

For more information about how to query bucket information, see [GetBucketInfo](#).

5.5.6. Manage bucket ACLs

A bucket is a container for objects stored in Object Storage Service (OSS). Every object is contained in a bucket. This topic describes how to configure and query the access control list (ACL) of a bucket.

Configure the ACL of a bucket

The following table describes the bucket ACLs.

ACL	Description	Method
private	Only the bucket owner and authorized users can perform read and write operations on objects in the bucket. Other users cannot access the objects in the bucket.	oss2.BUCKET_ACL_PRIVATE
public read	Only the bucket owner and authorized users of this bucket can perform read and write operations on objects in the bucket. Other users, including anonymous users, can only read objects in the bucket. Exercise caution when you set the bucket ACL to this value.	oss2.BUCKET_ACL_PUBLIC_READ

ACL	Description	Method
public read/write	All users, including anonymous users, can perform read and write operations on objects in the bucket. Exercise caution when you set the bucket ACL to this value.	oss2.BUCKET_ACL_PUBLIC_READ_WRITE

The following code provides an example on how to configure the ACL of a bucket:

```
# -*- coding: utf-8 -*-
import oss2
# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# In this example, the endpoint of the China (Hangzhou) region is used. Specify the endpoint based on your business requirements.
# Specify yourBucketName as the name of the bucket.
bucket = oss2.Bucket(auth, 'http://oss-cn-hangzhou.aliyuncs.com', 'yourBucketName')
# Set the ACL of the bucket to private.
bucket.put_bucket_acl(oss2.BUCKET_ACL_PRIVATE)
```

For more information about how to configure the ACL of a bucket, see [PutBucketAcl](#).

Query the ACL of a bucket

The following code provides an example on how to query the ACL of a bucket:

```
# -*- coding: utf-8 -*-
import oss2
# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# In this example, the endpoint of the China (Hangzhou) region is used. Specify the endpoint based on your business requirements.
# Specify yourBucketName as the name of the bucket.
bucket = oss2.Bucket(auth, 'http://oss-cn-hangzhou.aliyuncs.com', 'yourBucketName')
# Query the ACL of the bucket.
print(bucket.get_bucket_acl().acl)
```

For more information about how to query the ACL of a bucket, see [GetBucketAcl](#).

5.5.7. Delete a bucket

If you no longer use a bucket, delete the bucket to stop unnecessary charges.

 **Warning** Deleted buckets cannot be recovered. Exercise caution when you perform this operation.

Prerequisites

- All objects in the bucket are deleted.
 - For more information about how to manually delete a small number of objects, see [删除文件](#).
 - If you want to delete a large number of objects, we recommend that you configure lifecycle rules to delete the objects in batches. For more information about how to delete a large number of objects, see [Configure lifecycle rules](#).

Notice

If versioning is enabled for the bucket that you want to delete, make sure that all versions of objects in the bucket are deleted. For more information about how to delete all versions of objects in a bucket, see [Configure versioning](#).

- Parts generated by multipart upload or resumable upload tasks in the bucket are deleted. For more information about how to delete parts in a bucket, see [Manage parts](#).
- All LiveChannels in the bucket are deleted. For more information about how to delete LiveChannels, see [DeleteLiveChannel](#).

Sample code

The following sample code shows how to delete a bucket named examplebucket:

```
# -*- coding: utf-8 -*-
import oss2

# The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to access OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine Q&M. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com.
# Specify the name of the bucket. Example: examplebucket.
bucket = oss2.Bucket(auth, 'https://oss-cn-hangzhou.aliyuncs.com', 'examplebucket')
try:
    # Delete the bucket.
    bucket.delete_bucket()
except oss2.exceptions.BucketNotEmpty:
    print('bucket is not empty.')
except oss2.exceptions.NoSuchBucket:
    print('bucket does not exist')
```

References

- For more information about the complete sample code that is used to delete a bucket, visit [GitHub](#).
- For more information about the API operation that you can call to delete a bucket, see [DeleteBucket](#).

5.5.8. Bucket tagging

You can use tags to identify buckets used for different purposes and manage these buckets.

Usage notes

The bucket tagging feature uses a key-value pair to identify a bucket. You can add tags to buckets that are used for different purposes and manage the buckets by tags.

- Only the bucket owner and users who are granted the `oss:PutBucketTagging` permission can add tags to the bucket or modify the tags of the bucket. If other users try to add tags to the bucket or modify the tags of the bucket, 403 Forbidden is returned with the `AccessDenied` error code.
- You can configure up to 20 tags for a bucket.
- The key and value of a tag must be UTF-8-encoded.
- The key of a tag can be up to 64 characters in length and cannot start with `http://`, `https://`, or `Aliyun`. In addition, the key of a tag cannot be empty.
- The value of a tag can be up to 128 characters in length and can be empty.

Configure tags for a bucket

The following code provides an example on how to configure tags for a bucket:

```
# -*- coding: utf-8 -*-
import oss2
from oss2.models import Tagging, TaggingRule
# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access Object Storage Service (OSS) because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# Specify the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com.
# Specify the bucket name. Example: examplebucket.
bucket = oss2.Bucket(auth, 'https://oss-cn-hangzhou.aliyuncs.com', 'examplebucket')
# Create tagging rules for the bucket.
rule = TaggingRule()
rule.add('key1', 'value1')
rule.add('key2', 'value2')
# Create tags for the bucket.
tagging = Tagging(rule)
# Configure tags for the bucket.
result = bucket.put_bucket_tagging(tagging)
# Query the returned HTTP status code.
print('http status:', result.status)
```

Query tags of a bucket

The following code provides an example on how to query tags of a bucket:


```
# -*- coding: utf-8 -*-
import oss2
# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# Specify the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com.
# Specify the bucket name. Example: examplebucket.
bucket = oss2.Bucket(auth, 'https://oss-cn-hangzhou.aliyuncs.com', 'examplebucket')
# Query tags of the bucket.
result = bucket.get_bucket_tagging()
# Query the obtained tagging rule.
tag_rule = result.tag_set.tagging_rule
print('tag rule:', tag_rule)
```

List buckets that have specific tags

The following code provides an example on how to list buckets that have specific tags:

```
# -*- coding: utf-8 -*-
import oss2
# Create a server object.
# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# Specify the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com.
service = oss2.Service(auth, 'https://oss-cn-hangzhou.aliyuncs.com')
# Add the tag-key and tag-value fields to the params parameter of the ListBuckets API operation.
params = {}
params['tag-key'] = '<yourTagging_key>'
params['tag-value'] = '<yourTagging_value>'
# List buckets that have specific tags
result = service.list_buckets(params=params)
# Query the results of the list operation.
for bucket in result.buckets:
    print('result bucket_name:', bucket.name)
```

Delete tags of a bucket

The following code provides an example on how to delete tags of a bucket:

```
# -*- coding: utf-8 -*-
import oss2

# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# Specify the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com.
# Specify the bucket name. Example: examplebucket.
bucket = oss2.Bucket(auth, 'https://oss-cn-hangzhou.aliyuncs.com', 'examplebucket')
# Delete tags of the bucket.
result = bucket.delete_bucket_tagging()
# Query the returned HTTP status code.
print('http status:', result.status)
```

References

- For the complete sample code that is used to configure tags for a bucket, visit [GitHub](#).
- For more information about the API operation that you can call to configure tags for a bucket, see [PutBucketTags](#).
- For more information about the API operation that you can call to query tags of a bucket, see [GetBucketTags](#).
- For more information about the API operation that you can call to delete tags of a bucket, see [DeleteBucketTags](#).

5.5.9. Bucket inventory

This topic describes how to add an inventory for a bucket and how to query, list, and delete inventories configured for a bucket.

Create an inventory for a bucket

The following code provides an example on how to create an inventory for a bucket:

```
# -*- coding: utf-8 -*-
import oss2
from oss2.models import (InventoryConfiguration,
                          InventoryFilter,
                          InventorySchedule,
                          InventoryDestination,
                          InventoryBucketDestination,
                          InventoryServerSideEncryptionKMS,
                          InventoryServerSideEncryptionOSS,
                          INVENTORY_INCLUDED_OBJECT_VERSIONS_CURRENT,
                          INVENTORY_INCLUDED_OBJECT_VERSIONS_ALL,
                          INVENTORY_FREQUENCY_DAILY,
                          INVENTORY_FREQUENCY_WEEKLY,
                          INVENTORY_FORMAT_CSV,
                          FIELD_SIZE,
                          FIELD_LAST_MODIFIED_DATE,
                          -----)
```

```

FIELD_STORAG_CLASS,
FIELD_ETAG,
FIELD_IS_MULTIPART_UPLOADED,
FIELD_ENCRYPTION_STATUS)

# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to acc
ess OSS because the account has permissions on all API operations. We recommend that you us
e a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to
the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# Set yourEndpoint to the endpoint of the region in which the bucket is located. For exampl
e, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss
-cn-hangzhou.aliyuncs.com.
# Specify the name of the bucket. Example: examplebucket.
bucket = oss2.Bucket(auth, 'https://oss-cn-hangzhou.aliyuncs.com', 'examplebucket')
# Specify the account ID granted by the bucket owner to perform the PutBucketInventory oper
ation.
account_id = 'yourBucketDestinationAccountId'
# Specify the RAM role for which you want to configure inventories. The RAM role must have
permissions to read the source bucket for which you want to configure the inventory and per
missions to write data to the destination bucket in which you want to store the generated i
nventory lists.
role_arn = 'yourBucketDestinationRoleArn'
# Specify the name of the bucket in which you want to store the generated inventory lists.
dest_bucket_name = 'yourDestinationBucketName'
# Specify the inventory name.
inventory_id = "inventory1"
# Specify the object attributes to include in the inventory lists.
optional_fields = [FIELD_SIZE, FIELD_LAST_MODIFIED_DATE, FIELD_STORAG_CLASS,
FIELD_ETAG, FIELD_IS_MULTIPART_UPLOADED, FIELD_ENCRYPTION_STATUS]
# Create an configuration for the destination bucket.
bucket_destination = InventoryBucketDestination(
    # Specify the ID of the Alibaba Cloud account that owns the destination bucket.
    account_id=account_id,
    # Specify the role ARN of the destination bucket.
    role_arn=role_arn,
    # Specify the name of the destination bucket.
    bucket=dest_bucket_name,
    # Specify the format of the inventory lists.
    inventory_format=INVENTORY_FORMAT_CSV,
    # Specify the prefix of the path in which you want to store the generated inventory
lists.
    prefix='destination-prefix',
    # The following code provides an example on how to encrypt the inventory list objec
t by using CMKS hosted in KMS.
    # sse_kms_encryption=InventoryServerSideEncryptionKMS("test-kms-id"),
    # The following code provides an example on how to encrypt the inventory list objec
t on the OSS server.
    # sse_oss_encryption=InventoryServerSideEncryptionOSS()
)
# Create an inventory configuration.
inventory_configuration = InventoryConfiguration(
    # Specify the inventory ID.
    inventory_id=inventory_id,
    # Specify whether the inventory is enabled. True indicates that the inventory list
is generated, while false indicates that no inventory list is generated.

```

```
is generated, while false indicates that no inventory list is generated.
    is_enabled=True,
    # Specify whether to generate the inventory on a daily or weekly basis. The followi
ng code provides an example to generate an inventory on a daily basis.
    inventory_schedule=InventorySchedule(frequency=INVENTORY_FREQUENCY_DAILY),
    # Set the inventory list to include only the current version. If you set this param
eter to INVENTORY_INCLUDED_OBJECT_VERSIONS_ALL, all versions are inventoried. Versioning mu
st be enabled for this configuration to take effect.
    included_object_versions=INVENTORY_INCLUDED_OBJECT_VERSIONS_CURRENT,
    # Specify the prefix of object names to filter objects to include in the inventory
list.
    inventory_filter=InventoryFilter(prefix="obj-prefix"),
    # Specify the object attributes to include in the inventory list.
    optional_fields=optional_fields,
    # Specify the destination for the inventory lists.
    inventory_destination=InventoryDestination(bucket_destination=bucket_destination))
# Apply the inventory configuration to the bucket.
bucket.put_bucket_inventory_configuration(inventory_configuration)
```

Query an inventory configured for a bucket

The following code provides an example on how to query inventories configured for a bucket:

```
# -*- coding: utf-8 -*-
import oss2
import os

# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com.
# Specify the name of the bucket. Example: examplebucket.
bucket = oss2.Bucket(auth, 'https://oss-cn-hangzhou.aliyuncs.com', 'examplebucket')
# Specify the inventory name.
inventory_id = "inventory1"
# Obtain the inventory configuration.
result = bucket.get_bucket_inventory_configuration(inventory_id=inventory_id);
# Display the information about the inventory configuration.
print('====inventory configuration====')
print('inventory_id', result.inventory_id)
print('is_enabled', result.is_enabled)
print('frequency', result.inventory_schedule.frequency)
print('included_object_versions', result.included_object_versions)
print('inventory_filter prefix', result.inventory_filter.prefix)
print('fields', result.optional_fields)
bucket_destin = result.inventory_destination.bucket_destination
print('===bucket destination===')
print('account_id', bucket_destin.account_id)
print('role_arn', bucket_destin.role_arn)
print('bucket', bucket_destin.bucket)
print('format', bucket_destin.inventory_format)
print('prefix', bucket_destin.prefix)
if bucket_destin.sse_kms_encryption is not None:
    print('server side encryption by kms, key id:', bucket_destin.sse_kms_encryption.key_id)
elif bucket_destin.sse_oss_encryption is not None:
    print('server side encryption by oss.')
```

List multiple inventories configured for a bucket

 **Note** You can query up to 100 inventories by sending a request. To query more than 100 inventory rules, you must send multiple requests and use the token returned for each request as the parameter for the next request.

The following code provides an example on how to list multiple inventories configured for a bucket at a time:

```
# -*- coding: utf-8 -*-
import oss2
import os

# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
```

```

e a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to
the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# Set yourEndpoint to the endpoint of the region in which the bucket is located. For exampl
e, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss
-cn-hangzhou.aliyuncs.com.
# Specify the name of the bucket. Example: examplebucket.
bucket = oss2.Bucket(auth, 'https://oss-cn-hangzhou.aliyuncs.com', 'examplebucket')
# Display the information about the inventory configuration.
def print_inventory_configuration(configuration):
    print('====inventory configuration====')
    print('inventory_id', configuration.inventory_id)
    print('is_enabled', configuration.is_enabled)
    print('frequency', configuration.inventory_schedule.frequency)
    print('included_object_versions', configuration.included_object_versions)
    print('inventory_filter prefix', configuration.inventory_filter.prefix)
    print('fields', configuration.optional_fields)
    bucket_destin = configuration.inventory_destination.bucket_destination
    print('===bucket destination===')
    print('account_id', bucket_destin.account_id)
    print('role_arn', bucket_destin.role_arn)
    print('bucket', bucket_destin.bucket)
    print('format', bucket_destin.inventory_format)
    print('prefix', bucket_destin.prefix)
    if bucket_destin.sse_kms_encryption is not None:
        print('server side encryption by kms, key id:', bucket_destin.sse_kms_encryption.ke
y_id)
    elif bucket_destin.sse_oss_encryption is not None:
        print('server side encryption by oss.')
# List all inventory configurations.
# If the number inventories exceeds 100, the results are returned in pages. The pagination
information is stored in the <oss2.models.ListInventoryConfigurationResult> class.
continuation_token = None
while 1:
    result = bucket.list_bucket_inventory_configurations(continuation_token=continuation_to
ken)
    # Obtain whether the results are listed by page.
    print('is truncated', result.is_truncated)
    # Obtain the token to carry in this list.
    print('continuation_token', result.continuation_token)
    # Obtain the token to carry in the subsequent list.
    print('next_continuation_token', result.next_continuation_token)
    # Display the information about the inventory configuration.
    for inventory_config in result.inventory_configurations:
        print_inventory_configuration(inventory_config)
    # If the results are listed by page, continue the listing. The subsequent list must car
ry the continuation token.
    if result.is_truncated:
        continuation_token = result.next_continuation_token
    else:
        break

```

Delete an inventory configured for a bucket

The following code provides an example on how to delete an inventory configured for a bucket:

```
# -*- coding: utf-8 -*-
import oss2
import os

# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com.
# Specify the name of the bucket. Example: examplebucket.
bucket = oss2.Bucket(auth, 'https://oss-cn-hangzhou.aliyuncs.com', 'examplebucket')
# Specify the inventory name.
inventory_id = "inventory1"
# Delete the inventory.
bucket.delete_bucket_inventory_configuration(inventory_id)
```

References

- For more information about the complete sample code for operations related to bucket inventories, visit [GitHub](#).
- For more information about the API operation that is called to create an inventory, see [PutBucketInventory](#).
- For more information about the API operation that is called to query inventories configured for a bucket, see [GetBucketInventory](#).
- For more information about the API operation that is called to list multiple inventories configured for a bucket, see [ListBucketInventory](#).
- For more information about the API operation that is called to delete inventories configured for a bucket, see [DeleteBucketInventory](#).

5.5.10. Bucket policy

Bucket policies are resource-based authorization policies. You can use bucket policies to grant other users the permission to access specific resources in Object Storage Service (OSS).

Configure bucket policies

The following code provides an example on how to configure a bucket policy:

```
# -*- coding: utf-8 -*-
import oss2
import json
# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# In this example, the endpoint of the China (Hangzhou) region is used. Specify your actual endpoint.
# Specify yourBucketName as the name of the bucket.
bucket = oss2.Bucket(auth, 'http://oss-cn-hangzhou.aliyuncs.com', 'yourBucketName')
# Configure the value of policy_text, which is the content of the bucket policy.
policy_text = '{"Statement": [{"Effect": "Allow", "Action": ["oss:GetObject", "oss:ListObjects"], "Resource": ["acs:oss:*:*/*user1/*"]}], "Version": "1"}'
# Configure the bucket policy.
bucket.put_bucket_policy(policy_text)
```

Query the bucket policies configured for a bucket

The following code provides an example on how to query the policies configured for a bucket:

```
# -*- coding: utf-8 -*-
import oss2
import json
# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# In this example, the endpoint of the China (Hangzhou) region is used. Specify your actual endpoint.
# Specify yourBucketName as the name of the bucket.
bucket = oss2.Bucket(auth, 'http://oss-cn-hangzhou.aliyuncs.com', 'yourBucketName')
# Query the bucket policies configured for the bucket.
result = bucket.get_bucket_policy()
policy_json = json.loads(result.policy)
print("Get policy text: ", policy_json)
```

Delete the bucket policies configured for a bucket

The following code provides an example on how to delete the policies configured for a bucket:


```
# -*- coding: utf-8 -*-
import oss2

# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# In this example, the endpoint of the China (Hangzhou) region is used. Specify your actual endpoint.
# Specify yourBucketName as the name of the bucket.
bucket = oss2.Bucket(auth, 'http://oss-cn-hangzhou.aliyuncs.com', 'yourBucketName')
# Delete the bucket policies configured for the bucket.
result = bucket.delete_bucket_policy()
assert int(result.status)//100 == 2
```

References

- For more information about the complete sample code for bucket policies, visit [GitHub](#).
- For more information about the API operation that you can call to configure bucket policies for a bucket, see [PutBucketPolicy](#).
- For more information about the API operation that you can call to query the bucket policies configured for a bucket, see [GetBucketPolicy](#).
- For more information about the API operation that you can call to delete bucket policies configured for a bucket, see [DeleteBucketPolicy](#).

5.5.11. Retention policy

You can configure a time-based retention policy for a bucket. A retention policy has a retention period that ranges from one day to 70 years. This topic describes how to create, query, and lock a retention policy.

Background information

OSS supports the Write Once Read Many (WORM) strategy that prevents an object from being deleted or overwritten for a specified period of time.

If a retention policy is not locked within 24 hours after it is created, the retention policy becomes invalid. After the retention policy configured for a bucket is locked, you can upload objects to or read objects from the bucket. However, you cannot delete objects in the bucket or the retention policy within the retention period of the policy. The retention period of the policy cannot be shortened but only be extended. For more information about retention policies, see [Retention policies](#).

Create a retention policy

The following code provides an example on how to create a retention policy:

```
# -*- coding: utf-8 -*-
import oss2
# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log
on to OSS because the account has permissions on all API operations. We recommend that you
use your RAM user's credentials to call API operations or perform routine operations and ma
intenance. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('<yourAccessKeyId>', '<yourAccessKeySecret>')
# This example uses the endpoint of the China (Hangzhou) region. Specify the actual endpoin
t based on your requirements.
bucket = oss2.Bucket(auth, 'http://oss-cn-hangzhou.aliyuncs.com', '<yourBucketName>')
# Create the retention policy and set the retention period to 1 days.
result = bucket.init_bucket_worm(1)
# Query the ID of the retention policy.
print(result.worm_id)
```

For more information about how to create a retention policy, see [InitiateBucketWorm](#).

Cancel an unlocked retention policy

The following code provides an example on how to cancel an unlocked retention policy:

```
# -*- coding: utf-8 -*-
import oss2
# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log
on to OSS because the account has permissions on all API operations. We recommend that you
use your RAM user's credentials to call API operations or perform routine operations and ma
intenance. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('<yourAccessKeyId>', '<yourAccessKeySecret>')
# This example uses the endpoint of the China (Hangzhou) region. Specify the actual endpoin
t based on your requirements.
bucket = oss2.Bucket(auth, 'http://oss-cn-hangzhou.aliyuncs.com', '<yourBucketName>')
# Cancel the unlocked retention policy.
bucket.abort_bucket_worm()
```

For more information about how to cancel an unlocked retention policy, see [AbortBucketWorm](#).

Lock a retention policy

The following code provides an example on how to lock a retention policy:

```
# -*- coding: utf-8 -*-
import oss2
# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log
on to OSS because the account has permissions on all API operations. We recommend that you
use your RAM user's credentials to call API operations or perform routine operations and ma
intenance. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('<yourAccessKeyId>', '<yourAccessKeySecret>')
# This example uses the endpoint of the China (Hangzhou) region. Specify the actual endpoin
t based on your requirements.
bucket = oss2.Bucket(auth, 'http://oss-cn-hangzhou.aliyuncs.com', '<yourBucketName>')
# Lock the retention policy.
bucket.complete_bucket_worm('<yourWromId>')
```

For more information about how to lock a retention policy, see [CompleteBucketWorm](#).

Query a retention policy

The following code provides an example on how to query a retention policy:

```
# -*- coding: utf-8 -*-
import oss2

# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log
on to OSS because the account has permissions on all API operations. We recommend that you
use your RAM user's credentials to call API operations or perform routine operations and ma
intenance. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('<yourAccessKeyId>', '<yourAccessKeySecret>')
# This example uses the endpoint of the China (Hangzhou) region. Specify the actual endpoin
t based on your requirements.
bucket = oss2.Bucket(auth, 'http://oss-cn-hangzhou.aliyuncs.com', '<yourBucketName>')
# Query the retention policy.
result = self.bucket.get_bucket_worm()
# Query the ID of the retention policy.
print(result.worm_id)
# Query the status of the retention policy. InProgress indicates that the retention policy
is not locked. Locked indicates that the retention policy is locked.
print(result.state)
# Query the retention period of the retention policy.
print(result.retention_period_days)
# Query the created time of the retention policy.
print(result.creation_date)
```

For more information about how to query a retention policy, see [GetBucketWorm](#).

Extend the retention period of a retention policy

The following code provides an example on how to extend the retention period of a locked retention policy:

```
# -*- coding: utf-8 -*-
import oss2

# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log
on to OSS because the account has permissions on all API operations. We recommend that you
use your RAM user's credentials to call API operations or perform routine operations and ma
intenance. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('<yourAccessKeyId>', '<yourAccessKeySecret>')
# This example uses the endpoint of the China (Hangzhou) region. Specify the actual endpoin
t based on your requirements.
bucket = oss2.Bucket(auth, 'http://oss-cn-hangzhou.aliyuncs.com', '<yourBucketName>')
# Extend the retention period of the locked retention policy.
bucket.extend_bucket_worm('<yourWormId>', 2)
```

For more information about how to extend the retention period of a locked retention policy, see [ExtendBucketWorm](#).

5.5.12. Enable pay-by-requester

After pay-by-requester mode is enabled for a bucket in Alibaba Cloud OSS, the requester instead of the bucket owner pays the cost of requests and traffic. The bucket owner always pays the cost of storing data. You can enable this feature to share data without having to pay for all requests and traffic by yourself.

Enable pay-by-requester

The following code provides an example of how to enable pay-by-requester mode:

```
# -*- coding: utf-8 -*-
import oss2
from oss2.models import PAYER_BUCKETOWNER, PAYER_REQUESTER
# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com.
# Specify the name of the bucket. Example: examplebucket.
bucket = oss2.Bucket(auth, 'https://oss-cn-hangzhou.aliyuncs.com', 'examplebucket')
# Enable the pay-by-requester mode. The default request payment mode is PAYER_BUCKETOWNER.
result = bucket.put_bucket_request_payment(PAYER_REQUESTER)
print("http respon status: ", result.status)
```

Query the pay-by-requester configurations of a bucket

The following code provides an example of how to query pay-by-requester configurations:

```
# -*- coding: utf-8 -*-
import oss2
# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com.
# Specify the name of the bucket. Example: examplebucket.
bucket = oss2.Bucket(auth, 'https://oss-cn-hangzhou.aliyuncs.com', 'examplebucket')
# Query the pay-by-requester configurations of the bucket.
result = bucket.get_bucket_request_payment()
print('payer:', result.payer)
```

Specify that third parties are charged for access to objects

If you specify that third parties are charged for access to objects, requesters must include the x-oss-request-payer:requester header in the HTTP requests to perform operations on your objects. If this header is not included, an error is returned.

The following code provides an example on how to specify that third parties are charged when they access objects by calling the PutObject, GetObject, and DeleteObject operations. You can use the same method to specify that third parties are charged when they call other operations to access your objects.

```
# -*- coding: utf-8 -*-
import oss2
from oss2.headers import OSS_REQUEST_PAYER
# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com.
# Specify the name of the bucket. Example: examplebucket.
bucket = oss2.Bucket(auth, 'https://oss-cn-hangzhou.aliyuncs.com', 'examplebucket')
# Specify the full path of the object. The full path cannot contain the bucket name. Example: exampledir/exampleobject.txt.
object_name = 'exampledir/exampleobject.txt'
headers = dict()
headers[OSS_REQUEST_PAYER] = "requester"
# Specify the x-oss-request-payer header in the request to upload the object.
result = bucket.put_object(object_name, 'test-content', headers=headers)
# Specify the x-oss-request-payer header in the request to download the object.
result = bucket.get_object(object_name, headers=headers)
# Specify the x-oss-request-payer header in the request to delete the object.
result = bucket.delete_object(object_name, headers=headers);
```

References

- For more information about the complete sample code that is used to enable pay-by-requester for a bucket, visit [GitHub](#).
- For the API operation that you can call to enable pay-by-requester mode, see [Put Bucket Request Payment](#).
- For the API operation that you can call to query pay-by-requester configurations, see [Get Bucket Request Payment](#).

5.5.13. Lifecycle

Object Storage Service (OSS) allows you to configure lifecycle rules to delete expired objects and parts or convert the storage class of expired objects to Infrequent Access (IA) or Archive to reduce your storage costs. This topic describes how to manage lifecycle rules for buckets.

Configure lifecycle rules

The following code provides an example on how to configure lifecycle rules for a bucket named examplebucket:

```
# -*- coding: utf-8 -*-
import oss2
import datetime
```

```

from oss2.models import (LifecycleExpiration, LifecycleRule,
                        BucketLifecycle, AbortMultipartUpload,
                        TaggingRule, Tagging, StorageTransition,
                        NoncurrentVersionStorageTransition,
                        NoncurrentVersionExpiration)

# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to acc
ess OSS because the account has permissions on all API operations. We recommend that you us
e a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to
the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# Specify the endpoint of the region in which the bucket is located. For example, if the bu
cket is located in the China (Hangzhou) region, set Endpoint to https://oss-cn-hangzhou.ali
yuncs.com.
# Specify yourBucketName as the name of the bucket. Example: examplebucket.
bucket = oss2.Bucket(auth, 'http://oss-cn-hangzhou.aliyuncs.com', 'examplebucket')
# Specify that objects expire three days after they are last modified.
rule1 = LifecycleRule('rule1', 'tests/',
                    status=LifecycleRule.ENABLED,
                    expiration=LifecycleExpiration(days=3))
# Specify that the objects last modified before the specified date expire.
rule2 = LifecycleRule('rule2', 'tests2/',
                    status=LifecycleRule.ENABLED,
                    expiration=LifecycleExpiration(created_before_date=datetime.date(2022, 12, 12)))
# Specify that parts expire three days after they are last modified.
rule3 = LifecycleRule('rule3', 'tests3/',
                    status=LifecycleRule.ENABLED,
                    abort_multipart_upload=AbortMultipartUpload(days=3))
# Specify that the parts last modified before the specified date expire.
rule4 = LifecycleRule('rule4', 'tests4/',
                    status=LifecycleRule.ENABLED,
                    abort_multipart_upload = AbortMultipartUpload(created_before_date=dat
etime.date(2022, 12, 12)))
# Specify that the storage class of objects is converted to IA 20 days after they are last
modified, and to Archive 30 days after they are last modified.
rule5 = LifecycleRule('rule5', 'tests5/',
                    status=LifecycleRule.ENABLED,
                    storage_transitions=[StorageTransition(days=20,storage_class=oss2.BUC
KET_STORAGE_CLASS_IA),
                                        StorageTransition(days=30,storage_class=oss2.BUCKET_STORAGE_CLA
SS_ARCHIVE)])
# Specify the tags to match the rules.
tagging_rule = TaggingRule()
tagging_rule.add('key1', 'value1')
tagging_rule.add('key2', 'value2')
tagging = Tagging(tagging_rule)
# Specify that the storage class of objects is converted to Archive 365 days after they are
last modified.
# Compared with the preceding rules, rule6 specifies tags to match. The rule applies to obj
ects whose tagging configurations are key1=value1 and key2=value2.
rule6 = LifecycleRule('rule6', 'tests6/',
                    status=LifecycleRule.ENABLED,
                    storage_transitions=[StorageTransition(created_before_date=datetime.d
ate(2022, 12, 12),storage_class=oss2.BUCKET_STORAGE_CLASS_IA)],
                    tagging = tagging)

```

```
# Create rule7 for a versioned bucket.
# Specify that the storage class of objects is converted to Archive 365 days after they are
last modified.
# Specify that delete markers are automatically removed when they expire.
# Specify that the storage class of objects is converted to Infrequent Access (IA) 12 days
after they become previous versions.
# Specify that the storage class of objects is converted to Archive 20 days after they beco
me previous versions.
# Specify that objects are deleted 30 days after they become previous versions.
rule7 = LifecycleRule('rule7', 'tests7/',
                      status=LifecycleRule.ENABLED,
                      storage_transitions=[StorageTransition(days=365, storage_class=oss2.BUCKET_ST
ORAGE_CLASS_ARCHIVE)],
                      expiration=LifecycleExpiration(expired_delete_marker=True),
                      noncurrent_version_storage_transitions =
                          [NoncurrentVersionStorageTransition(12, oss2.BUCKET_STORAGE_CLASS_IA),
                           NoncurrentVersionStorageTransition(20, oss2.BUCKET_STORAGE_CLASS_ARCHI
VE)],
                      noncurrent_version_expiration = NoncurrentVersionExpiration(30))
lifecycle = BucketLifecycle([rule1, rule2, rule3, rule4, rule5, rule6, rule7])
bucket.put_bucket_lifecycle(lifecycle)
```

Query lifecycle rules

The following code provides an example on how to query the lifecycle rules configured for the bucket named `examplebucket`:

```
# -*- coding: utf-8 -*-
import oss2

# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to acc
ess OSS because the account has permissions on all API operations. We recommend that you us
e a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to
the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# Specify the endpoint of the region in which the bucket is located. For example, if the bu
cket is located in the China (Hangzhou) region, set Endpoint to https://oss-cn-hangzhou.ali
yuncs.com.
# Specify yourBucketName as the name of the bucket. Example: examplebucket.
bucket = oss2.Bucket(auth, 'http://oss-cn-hangzhou.aliyuncs.com', 'examplebucket')
# Query the lifecycle rules.
lifecycle = bucket.get_bucket_lifecycle()
for rule in lifecycle.rules:
    print('====')
    print('id:', rule.id)
    print('prefix:', rule.prefix)
    print('status', rule.status)
    if rule.tagging is not None:
        print('tagging:', rule.tagging)
    if rule.abort_multipart_upload is not None:
        if rule.abort_multipart_upload.days is not None:
            print('abort_multipart_upload days:', rule.abort_multipart_upload.days)
        else:
            print('abort_multipart_upload created_before_date:', rule.abort_multipart_uploa
d.created_before_date)
```

```

if rule.expiration is not None:
    if rule.expiration.days is not None:
        print('expiration days:', rule.expiration.days)
    elif rule.expiration.expired_detete_marker is not None:
        print('expiration expired_detete_marker:', rule.expiration.expired_detete_marke
r)

    elif rule.expiration.created_before_date is not None:
        print('expiration created_before_date:', rule.expiration.created_before_date)
if len(rule.storage_transitions) > 0:
    storage_info = ''
    for storage_rule in rule.storage_transitions:
        if storage_rule.days is not None:
            storage_info += 'days={0}, storage_class={1} *** '.format(
                storage_rule.days, storage_rule.storage_class)
        else:
            storage_info += 'created_before_date={0}, storage_class={1} *** '.format(
                storage_rule.created_before_date, storage_rule.storage_class)
    print('storage_transitions:', storage_info)
if len(rule.noncurrent_version_sotrage_transitions) > 0:
    noncurrent_storage_info = ''
    for storage_rule in rule.noncurrent_version_sotrage_transitions:
        noncurrent_storage_info += 'days={0}, storage_class={1} *** '.format(
            storage_rule.noncurrent_days, storage_rule.storage_class)
    print('noncurrent_version_sotrage_transitions:', noncurrent_storage_info)
if rule.noncurrent_version_expiration is not None:
    print('noncurrent_version_expiration days:', rule.noncurrent_version_expiration.non
current_days)

```

Delete lifecycle rules

The following code provides an example on how to delete lifecycle rules configured for the bucket named examplebucket:

```

# -*- coding: utf-8 -*-
import oss2

# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to acc
ess OSS because the account has permissions on all API operations. We recommend that you us
e a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to
the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# Specify the endpoint of the region in which the bucket is located. For example, if the bu
cket is located in the China (Hangzhou) region, set Endpoint to https://oss-cn-hangzhou.ali
yuncs.com.
# Specify yourBucketName as the name of the bucket. Example: examplebucket.
bucket = oss2.Bucket(auth, 'http://oss-cn-hangzhou.aliyuncs.com', 'examplebucket')
# Delete lifecycle rules.
bucket.delete_bucket_lifecycle()
# An exception occurs when you query lifecycle rules again.
try:
    lifecycle = bucket.get_bucket_lifecycle()
except oss2.exceptions.NoSuchLifecycle:
    print('lifecycle is not configured')

```


5.5.14. Hotlink protection

This topic describes how to use hotlink protection.

To prevent your data on OSS from being leached, OSS supports hotlink protection through the referer field settings in the HTTP header, including the following parameters:

- Referrer whitelist: Used to allow access only for specified domains to OSS data.
- Empty referer: Determines whether the referer can be empty. If it is not allowed, only requests with the referer filed in their HTTP or HTTPS headers can access OSS data.

For more information about hotlink protection, see [Hotlink protection](#) in OSS Developer Guide.

Configure a Referrer whitelist for a bucket

The following code provides an example on how to configure a Referrer whitelist for a bucket:

```
# -*- coding: utf-8 -*-
import oss2
from oss2.models import BucketReferer
# The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('<yourAccessKeyId>', '<yourAccessKeySecret>')
# In this example, the endpoint of the China (Hangzhou) region is used. Specify your actual endpoint.
bucket = oss2.Bucket(auth, 'http://oss-cn-hangzhou.aliyuncs.com', '<yourBucketName>')
# Set BucketReferer to True. The value of True indicates that empty Referers are allowed and the value of False indicates that empty Referers are not allowed. Then, configure the Referrer whitelist.
bucket.put_bucket_referer(BucketReferer(True, ['http://aliyun.com', 'http://*.aliyuncs.com']))
```

Query the Referrer whitelist of a bucket

The following code provides an example on how to query a Referrer whitelist of a bucket:

```
# -*- coding: utf-8 -*-
import oss2
# The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('<yourAccessKeyId>', '<yourAccessKeySecret>')
# In this example, the endpoint of the China (Hangzhou) region is used. Specify your actual endpoint.
bucket = oss2.Bucket(auth, 'http://oss-cn-hangzhou.aliyuncs.com', '<yourBucketName>')
config = bucket.get_bucket_referer()
print('allow empty referer={0}, referers={1}'.format(config.allow_empty_referer, config.referers))
```

Clear a Referer whitelist of a bucket

The following code provides an example on how to clear a Referer whitelist of a bucket:

```
# -*- coding: utf-8 -*-
import oss2

# The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('<yourAccessKeyId>', '<yourAccessKeySecret>')
# In this example, the endpoint of the China (Hangzhou) region is used. Specify your actual endpoint.
bucket = oss2.Bucket(auth, 'http://oss-cn-hangzhou.aliyuncs.com', '<yourBucketName>')
# The Referer whitelist of a bucket cannot be directly cleared. You must create a Referer whitelist that allows empty Referer fields to overwrite the existing Referer whitelist.
bucket.put_bucket_referer(BucketReferer(True, []))
```

5.5.15. Logging

A large number of logs are generated when Object Storage Service (OSS) resources are accessed. After you enable and configure logging for a bucket, OSS generates logs every hour in accordance with predefined naming conventions and then stores the logs as objects in a specified bucket.

Enable logging for a bucket

The following code provides an example on how to enable logging for a bucket:

```
# -*- coding: utf-8 -*-
import oss2
from oss2.models import BucketLogging
# The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in Object Storage Service (OSS) is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com.
# Specify the bucket name. Example: examplebucket.
bucket = oss2.Bucket(auth, 'yourEndpoint', 'examplebucket')
# Specify that the generated log objects are stored in the current bucket.
# Specify that the log objects are stored in the log/ directory. If you specify this parameter, log objects are stored in a specified directory of the destination bucket. If you do not specify this parameter, log objects are stored in the root directory of the destination bucket.
# Enable logging for the bucket.
logging = bucket.put_bucket_logging(BucketLogging(bucket.bucket_name, 'log/'))
if logging.status == 200:
    print("Enable access logging")
else:
    print("request_id:", logging.request_id)
    print("resp:", logging.resp.response)
```

View the logging configurations of a bucket

The following code provides an example on how to view the logging configurations of a bucket:

```
# -*- coding: utf-8 -*-
import oss2
# The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com.
# Specify the bucket name. Example: examplebucket.
bucket = oss2.Bucket(auth, 'yourEndpoint', 'examplebucket')
# View the logging configurations of the bucket.
logging = bucket.get_bucket_logging()
print('TargetBucket={0}, TargetPrefix={1}'.format(logging.target_bucket, logging.target_prefix))
```

Disable logging for a bucket

The following code provides an example on how to disable logging for a bucket:

```
# -*- coding: utf-8 -*-
import oss2

# The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com.
# Specify the bucket name. Example: examplebucket.
bucket = oss2.Bucket(auth, 'yourEndpoint', 'examplebucket')
# Disable logging for the bucket.
logging = bucket.delete_bucket_logging()
if logging.status == 204:
    print("Disable access logging")
else:
    print("request_id:", logging.request_id)
    print("resp:", logging.resp.response)
```

References

- For the complete sample code that is used to configure logging for a bucket, visit [GitHub](#).
- For more information about the API operation that you can call to enable logging for a bucket, see [PutBucketLogging](#).
- For more information about the API operation that you can call to view the logging configurations of a bucket, see [GetBucketLogging](#).
- For more information about the API operation that you can call to disable logging for a bucket, see [DeleteBucketLogging](#).

5.5.16. Static website hosting (mirroring-based back-to-origin)

You can enable static website hosting for buckets and configure mirroring-based back-to-origin rules. After you host a static website on a bucket, you can access the bucket to visit the website. You are automatically redirected to the specified index page or error page. After mirroring-based back-to-origin rules are configured and take effect, you can use mirroring-based back-to-origin to seamlessly migrate data to Object Storage Service (OSS).

Static website hosting

- The following code provides an example on how to configure static website hosting:

```
# -*- coding: utf-8 -*-
import oss2
from oss2.models import BucketWebsite
# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to O
SS because the account has permissions on all API operations. We recommend that you use a
RAM user to call API operations or perform routine operations and maintenance. To create
a RAM user, log on to the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# In this example, the endpoint of the China (Hangzhou) region is used. Specify the endpo
int based on your business requirements. For more information about the endpoints of othe
r regions, see Regions and endpoints.
# Specify the name of the bucket. Example: examplebucket. For more information about the
naming conventions for buckets, see Bucket naming conventions.
bucket = oss2.Bucket(auth, 'https://oss-cn-hangzhou.aliyuncs.com', 'examplebucket')
# Enable static website hosting and set the default homepage to index.html and the defaul
t 404 page to error.html.
bucket.put_bucket_website(BucketWebsite('index.html', 'error.html'))
```

Configure static website hosting

- The following code provides an example on how to query static website hosting configurations:

```
# -*- coding: utf-8 -*-
import oss2
# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to a
ccess OSS because the account has permissions on all API operations. We recommend that yo
u use a RAM user to call API operations or perform routine operations and maintenance. To
create a RAM user, log on to the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# In this example, the endpoint of the China (Hangzhou) region is used. Specify the endpo
int based on your business requirements.
# Specify the name of the bucket. Example: examplebucket.
bucket = oss2.Bucket(auth, 'https://oss-cn-hangzhou.aliyuncs.com', 'examplebucket')
try:
    # If static website hosting is not enabled for the specified bucket, a NoSuchWebsite
exception is thrown when you call get_bucket_website.
    website = bucket.get_bucket_website()
    print('Index file is {0}, error file is {1}'.format(website.index_file, website.error
_file))
except oss2.exceptions.NoSuchWebsite as e:
    print('Website is not configured, request_id={0}'.format(e.request_id))
```

Query static website hosting configurations

- The following code provides an example on how to delete static website hosting configurations:

```
# -*- coding: utf-8 -*-
import oss2

# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# In this example, the endpoint of the China (Hangzhou) region is used. Specify the endpoint based on your business requirements.
# Specify the name of the bucket. Example: examplebucket.
bucket = oss2.Bucket(auth, 'https://oss-cn-hangzhou.aliyuncs.com', 'examplebucket')
bucket.delete_bucket_website()
```

Delete static website hosting configurations

Mirroring-based back-to-origin

Mirroring-based back-to-origin is used to seamlessly migrate data to OSS. For example, you can migrate services from a self-managed origin or from another cloud service to OSS without service interruption. You can use mirroring-based back-to-origin rules during migration to obtain the data that is not migrated to OSS. This ensures service continuity.

- If a requester attempts to access an object in the specified bucket and the object does not exist, you can specify the URL of the object in the origin and back-to-origin conditions to allow the requester to obtain the object from the origin. For example, a bucket named examplebucket resides in the China (Hangzhou) region. When a requester attempts to access an object in the examplefolder directory of the root directory of the bucket but the object does not exist, the requester is redirected to *https://www.example.com/* to access the required object that is stored in the examplefolder directory of the origin.

The following code provides an example on how to configure mirroring-based back-to-origin rules in the preceding scenario:

```

# -*- coding: utf-8 -*-
import oss2
from oss2.models import BucketWebsite
# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# In this example, the endpoint of the China (Hangzhou) region is used. Specify the endpoint based on your business requirements.
# Specify the name of the bucket. Example: examplebucket. For more information about the naming conventions for buckets, see Bucket naming conventions.
bucket = oss2.Bucket(auth, 'https://oss-cn-hangzhou.aliyuncs.com', 'examplebucket')
# Enable static website hosting and set the default homepage to index.html and the default 404 page to error.html.
index_file = 'index.html'
error_file = 'error.html'
# Specify the matching conditions.
condition1 = Condition(key_prefix_equals='examplefolder',
                        http_err_code_return_equals=404)
# Specify the headers included in the request when you use mirroring-based back-to-origin.
mirror_headers_set_1 = MirrorHeadersSet("myheader-key5", "myheader-value5")
mirror_headers_set_2 = MirrorHeadersSet("myheader-key6", "myheader-value6")
set_list = [mirror_headers_set_1, mirror_headers_set_2]
pass_list = ['myheader-key1', 'myheader-key2']
remove_list = ['myheader-key3', 'myheader-key4']
mirror_header = RedirectMirrorHeaders(pass_all=True, pass_list=pass_list, remove_list=remove_list, set_list=set_list)
# Specify the operation to perform after the rule is matched.
redirect1 = Redirect(redirect_type=REDIRECT_TYPE_MIRROR, mirror_url='https://www.example.com/',
                    mirror_pass_query_string=True, mirror_follow_redirect=True, mirror_check_md5=True, mirror_headers=mirror_header)
rule1 = RoutingRule(rule_num=1, condition=condition1, redirect=redirect1)
# Configure mirroring-based back-to-origin.
website_set = BucketWebsite(index_file, error_file, [rule1])

```

Configure mirroring-based back-to-origin

- The following code provides an example on how to query the mirroring-based back-to-origin configurations:

```
# -*- coding: utf-8 -*-
import oss2

# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# In this example, the endpoint of the China (Hangzhou) region is used. Specify the endpoint based on your business requirements.
# Specify the name of the bucket. Example: examplebucket.
bucket = oss2.Bucket(auth, 'https://oss-cn-hangzhou.aliyuncs.com', 'examplebucket')
try:
    # If static website hosting is not enabled for the specified bucket, a NoSuchWebsite exception is thrown when you call get_bucket_website.
    website_get = bucket.get_bucket_website()
    # Query the default homepage.
    print(website_get.index_file)
    # Query the default 404 page.
    print(website_get.error_file)
    for rule in website_get.rules:
        print(rule.rule_num)
        # Query the prefix for rule matching.
        print(rule.condition.key_prefix_equals)
        # Query the HTTP status code.
        print(rule.condition.http_err_code_return_equals)
        # Query the redirect type.
        print(rule.redirect.redirect_type)
        # Query the origin URL for mirroring-based back-to-origin.
        print(rule.redirect.mirror_url)
        # Query the parameters in the request.
        print(rule.redirect.pass_query_string)
        # Specify the string that is used to replace the object name when the request is redirected. This parameter can be set to a variable.
        # print(rule.redirect.replace_key_with)
        # Specify the string that is used to replace the prefix of the object name when the request is redirected. If the prefix of an object is empty, this string precedes the object name.
        # print(rule.redirect.replace_key_prefix_with)
        # Query the protocol that is used for redirection.
        # print(rule.redirect.proto)
        # Query the domain name to which the request is redirected.
        # print(rule.redirect.host_name)
        # Query the HTTP status code in the response.
        # print(rule.redirect.http_redirect_code)
except oss2.exceptions.NoSuchWebsite as e:
    print('Website is not configured, request_id={0}'.format(e.request_id))
```

Query mirroring-based back-to-origin configurations

- The following code provides an example on how to delete mirroring-based back-to-origin configurations:


```
# -*- coding: utf-8 -*-
import oss2
# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# In this example, the endpoint of the China (Hangzhou) region is used. Specify the endpoint based on your business requirements.
# Specify the name of the bucket. Example: examplebucket.
bucket = oss2.Bucket(auth, 'https://oss-cn-hangzhou.aliyuncs.com', 'examplebucket')
bucket.delete_bucket_website()
```

Delete mirroring-based back-to-origin configurations

References

- For more information about the API operation that you can call to configure static website hosting or mirroring-based back-to-origin, see [PutBucketWebsite](#).
- For more information about the API operation that you can call to query static website hosting or mirroring-based back-to-origin configurations, see [GetBucketWebsite](#).
- For more information about the API operation that you can call to delete static website hosting or mirroring-based back-to-origin configurations, see [DeleteBucketWebsite](#).

5.5.17. CORS

Cross-origin resource sharing (CORS) is a standard cross-origin solution provided by HTML5 to allow web application servers to control cross-origin access and secure cross-origin data transmission.

Configure CORS rules

The following code provides an example on how to configure CORS rules for a specific bucket:

```
# -*- coding: utf-8 -*-
import oss2
from oss2.models import BucketCors, CorsRule
# The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to access OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine Q&M. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com.
# Specify the name of the bucket. Example: examplebucket.
bucket = oss2.Bucket(auth, 'https://oss-cn-hangzhou.aliyuncs.com', 'examplebucket')
rule = CorsRule(allowed_origins=['*'],
                allowed_methods=['GET', 'HEAD'],
                allowed_headers=['*'],
                max_age_seconds=1000)
# The existing rules are overwritten.
bucket.put_bucket_cors(BucketCors([rule]))
```

Query CORS rules

The following code provides an example on how to query CORS rules that are configured for a specific bucket:

```
# -*- coding: utf-8 -*-
import oss2

# The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to access OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com.
# Specify the name of the bucket. Example: examplebucket.
bucket = oss2.Bucket(auth, 'https://oss-cn-hangzhou.aliyuncs.com', 'examplebucket')
try:
    cors = bucket.get_bucket_cors()
except oss2.exceptions.NoSuchCORS:
    print('cors is not set')
else:
    for rule in cors.rules:
        print('AllowedOrigins={0}'.format(rule.allowed_origins))
        print('AllowedMethods={0}'.format(rule.allowed_methods))
        print('AllowedHeaders={0}'.format(rule.allowed_headers))
        print('ExposeHeaders={0}'.format(rule.expose_headers))
        print('MaxAgeSeconds={0}'.format(rule.max_age_seconds))
```

Delete CORS rules

The following code provides an example on how to delete the CORS rules that are configured for a specific bucket:

```
# -*- coding: utf-8 -*-
import oss2

# The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to access OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com.
# Specify the name of the bucket. Example: examplebucket.
bucket = oss2.Bucket(auth, 'https://oss-cn-hangzhou.aliyuncs.com', 'examplebucket')
bucket.delete_bucket_cors()
```

References

- For more information about the complete sample code of CORS, visit [GitHub](#).
- For more information about the API operation that you can call to configure CORS rules, see [PutBucketCORS](#).

- For more information about the API operation that you can call to query CORS rules, see [GetBucketCors](#).
- For more information about the API operation that you can call to delete CORS rules, see [DeleteBucketCors](#).

5.5.18. Transfer acceleration

The transfer acceleration feature allows users worldwide to access objects stored in Object Storage Service (OSS) more quickly. This feature is applicable to scenarios where data must be transferred over long geographical distances. This feature can also be used to download or upload large objects that are gigabytes or terabytes in size.

Enable transfer acceleration

The following code provides an example on how to enable transfer acceleration for a bucket named `examplebucket`:

```
import oss2

# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set Endpoint to https://oss-cn-hangzhou.aliyuncs.com.
# Specify the name of the bucket. Example: examplebucket.
bucket = oss2.Bucket(auth, 'https://oss-cn-hangzhou.aliyuncs.com', 'examplebucket')
# Configure transfer acceleration for the bucket.
# If enabled is set to true, transfer acceleration is enabled. If enabled is set to false, transfer acceleration is disabled.
enabled = 'true'
bucket.put_bucket_transfer_acceleration(enabled)
```

Query the transfer acceleration state of a bucket

The following code provides an example on how to query the transfer acceleration state of a bucket named `examplebucket`:

```
import oss2
# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set Endpoint to https://oss-cn-hangzhou.aliyuncs.com.
# Specify the name of the bucket. Example: examplebucket.
bucket = oss2.Bucket(auth, 'https://oss-cn-hangzhou.aliyuncs.com', 'examplebucket')
# Query the transfer acceleration state of the bucket.
# If the returned value is true, the transfer acceleration feature is enabled for the bucket. If the returned value is false, the transfer acceleration feature is disabled for the bucket.
result = bucket.get_bucket_transfer_acceleration()
enabled_text = result.enabled
print("Returns whether to enable transfer acceleration: ", enabled_text)
```

References

- For more information about the complete sample code that is used to enable transfer acceleration for a bucket, visit [GitHub](#).
- For more information about the API operation that is called to enable transfer acceleration for a bucket, see [PutBucketTransferAcceleration](#).
- For more information about the API operation that is called to query the transfer acceleration state of a bucket, see [GetBucketTransferAcceleration](#).

5.5.19. Cross-region replication

Cross-region replication (CRR) can automatically replicate objects across buckets in different Object Storage Service (OSS) regions in asynchronous mode. CRR can also synchronize operations, for example, create, overwrite, and delete operations on objects, from a source bucket to a destination bucket. This feature can meet your requirements for geo-disaster recovery and data replication.

Enable CRR

The following code provides an example on how to enable CRR to replicate data from the srcexamplebucket bucket in the China (Hangzhou) region to the dstexamplebucket bucket in the China (Beijing) region.

 **Notice** Before you enable CRR, make sure that both the source bucket and the destination bucket are in the unversioned or versioned state.

```
# -*- coding: utf-8 -*-
import oss2
from oss2.models import ReplicationRule
# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
```

```

auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# Specify the endpoint of the region in which the source bucket is located. For example, if
the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-ha
ngzhou.aliyuncs.com. For more information about the endpoints of other regions, see Regions
and endpoints.
# Specify the name of the source bucket. Example: srcexamplebucket. For more information ab
out the naming conventions for buckets, see Bucket naming conventions.
bucket = oss2.Bucket(auth, 'https://oss-cn-hangzhou.aliyuncs.com', 'srcexamplebucket')
replica_config = ReplicationRule(
    # Specify the destination bucket to which the data is replicated.
    target_bucket_name='destexamplebucket',
    # Specify the region in which the destination bucket is located.
    target_bucket_location='oss-cn-beijing'
)
# Specify the prefix that is used to determine the object that you want to replicate. After
you specify the prefix, only objects whose names contain the prefix are replicated to the d
estination bucket.
# prefix_list = ['prefix1', 'prefix2']
# Specify CRR rules.
# replica_config = ReplicationRule(
    # prefix_list=prefix_list,
    # Specify the operations that can be synchronized to the destination bucket. By defaul
t, the value is ALL, which indicates that all operations on the source bucket are synchroni
zed to the destination bucket.
    # action_list=[ReplicationRule.ALL],
    # Specify the destination bucket to which the data is replicated.
    # target_bucket_name='destexamplebucket1',
    # Specify the region in which the destination bucket is located.
    # target_bucket_location='oss-cn-shanghai',
    # Specify whether to synchronize historical data. By default, the historical data is s
ynchronized. In this example, this parameter is set to False, which indicates that the hist
orical data is not synchronized.
    # is_enable_historical_object_replication=False,
    # Specify the link that is used to transfer data during data replication.
    # target_transfer_type='oss_acc',
    # Specify the role that you authorize OSS to use to replicate data. If you want to use
SSE-KMS to encrypt the objects that are replicated to the destination bucket, you must conf
igure this parameter.
    # sync_role_name='roleNameTest',
    # Replicate the objects that are encrypted by using SSE-KMS.
    # sse_kms_encrypted_objects_status=ReplicationRule.ENABLED
    # Specify the customer master key (CMK) ID used in SSE-KMS. If you want to use SSE-KMS
to encrypt the objects that are replicated to the destination bucket, you must configure th
is parameter.
    # replica_kms_keyid='9468da86-3509-4f8d-a61e-6eableac****',
    #)
# Enable CRR.
bucket.put_bucket_replication(replica_config)

```

Query CRR configurations

The following code provides an example on how to query the CRR configurations of the examplebucket bucket:

```
# -*- coding: utf-8 -*-
import oss2
from oss2.models import ReplicationRule
# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com.
# Specify the name of the bucket. Example: examplebucket.
bucket = oss2.Bucket(auth, 'yourEndpoint', 'examplebucket')
# Query the CRR configurations.
result = bucket.get_bucket_replication()
# Display the returned information.
for rule in result.rule_list:
    print(rule.rule_id)
    print(rule.target_bucket_name)
    print(rule.target_bucket_location)
```

Query the progress of data synchronization

You can query the progress of historical data synchronization and the progress of real-time data synchronization.

- Percent signs (%) indicate the progress of historical data synchronization. This function only works for buckets for which historical data synchronization has been enabled.
- Time points (the point of latest time the data is written) indicate the progress of real-time data synchronization. If a time point is displayed, the data created before this time point has been synchronized.

The following code provides an example on how to query the progress of data synchronization that is performed on the examplebucket bucket configured with the test_replication_1 replication rule:

```
# -*- coding: utf-8 -*-
import oss2
from oss2.models import ReplicationRule
# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to acc
ess OSS because the account has permissions on all API operations. We recommend that you us
e a RAM user to call API operations or perform routine operations and maintenance. To creat
e a RAM user, log on to the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# Set yourEndpoint to the endpoint of the region in which the bucket is located. For exampl
e, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss
-cn-hangzhou.aliyuncs.com.
# Specify the name of the bucket. Example: examplebucket.
bucket = oss2.Bucket(auth, 'yourEndpoint', 'examplebucket')
# Query the progress of the CRR task that is performed on the bucket.
# Specify the replication rule ID. Example: test_replication_1.
result = bucket.get_bucket_replication_progress('test_replication_1')
print(result.progress.rule_id)
# Check whether CRR is enabled for historical data in the bucket.
print(result.progress.is_enable_historical_object_replication)
# Display the progress of historical data synchronization.
print(result.progress.historical_object_progress)
# Display the progress of real-time data synchronization.
print(result.progress.new_object_progress)
```

Query the regions in which CRR can be performed

The following code provides an example on how to query the regions to which data in the examplebucket bucket can be synchronized:

```
# -*- coding: utf-8 -*-
import oss2
from oss2.models import ReplicationRule
# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to acc
ess OSS because the account has permissions on all API operations. We recommend that you us
e a RAM user to call API operations or perform routine operations and maintenance. To creat
e a RAM user, log on to the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# Set yourEndpoint to the endpoint of the region in which the bucket is located. For exampl
e, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss
-cn-hangzhou.aliyuncs.com.
# Specify the name of the bucket. Example: examplebucket.
bucket = oss2.Bucket(auth, 'yourEndpoint', 'examplebucket')
# Query the regions to which data in the source bucket can be synchronized.
result = bucket.get_bucket_replication_location()
for location in result.location_list:
    print(location)
```

Disable CRR

You can disable CRR for the specified source bucket by deleting the replication rule that is configured for the bucket.

The following code provides an example on how to delete the `test_replication_1` replication rule that is configured for the `examplebucket` bucket:

```
# -*- coding: utf-8 -*-
import oss2
from oss2.models import ReplicationRule
# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com.
# Specify the name of the bucket. Example: examplebucket.
bucket = oss2.Bucket(auth, 'yourEndpoint', 'examplebucket')
# Disable CRR for the bucket.
# Specify the replication rule ID. Example: test_replication_1.
result = bucket.delete_bucket_replication('test_replication_1')
```

References

- For more information about the API operation that you can call to enable CRR, see [PutBucketReplication](#).
- For more information about the API operation that you can call to query CRR configurations, see [GetBucketReplication](#).
- For more information about the API operation that you can call to query the regions in which CRR can be performed, see [GetBucketReplicationLocation](#).
- For more information about the API operation that you can call to query the progress of a CRR task, see [GetBucketReplicationProgress](#).
- For more information about the API operation that you can call to disable CRR, see [DeleteBucketReplication](#).

5.6. Upload objects

5.6.1. Overview

This topic is the overview of object upload in OSS Python SDK.

In OSS, an object is the basic unit for data operations. OSS Python SDK provides the following file upload methods:

- **Simple upload**: supports the upload of a file up to 5 GB in size.
- **Append upload**: supports the upload of a file up to 5 GB in size.
- **Resumable upload**: supports the upload of a file up to 48.8 TB in size. It applies to the upload of large files. It supports concurrent upload and you can define the size of each part.
- **Multipart upload**: supports the upload of a file up to 48.8 TB in size. It applies to the upload of large files.

 **Note** For more information about the usage scenarios of each upload method, see “Upload files” in OSS Developer Guide.

During the file upload, you can set [Object Meta](#), and view the upload progress in the [upload progress bar](#). After you complete the file upload, you can perform [upload callback](#).

5.6.2. Simple upload

You can use the `put_object` method to upload a single object. This method is called simple upload. You can use simple upload to upload the following types of data: strings, bytes, Unicode characters, network streams, and local files.

 **Note** For more information about how to perform simple upload, see [Put Object](#).

Usage notes

If you upload an object whose name is the same as that of an accessible existing object, the existing object is overwritten by the uploaded object.

The following table lists the common parameters that you must configure when you upload an object.

Parameter	Description
<code>bucket_name</code>	The name of the Object Storage Service (OSS) bucket. Bucket names must comply with the following conventions: • The name can contain only lowercase letters, digits, and hyphens (-). • The name must start and end with a lowercase letter or a digit. • The name must be 3 to 63 characters in length.
<code>object_name</code>	The full path of the object that you want to upload. The path cannot contain the bucket name. Object names must comply with the following conventions: • The name must be encoded in UTF-8. • The name must be 1 to 1,023 characters in length. • The name cannot start with a forward slash (/) or a backslash (\).

Upload a string

The following code provides an example on how to upload a string to an object named `exampleobject.txt` in a bucket named `examplebucket`:

```
# -*- coding: utf-8 -*-
import oss2

# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS, because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com.
# Specify the name of the bucket.
bucket = oss2.Bucket(auth, 'yourEndpoint', 'examplebucket')
# Upload an object.
# You can configure the x-oss-storage-class and x-oss-object-acl headers for put_object to specify the storage class and access control list (ACL) of the object that you want to upload.
# headers = dict()
# headers["x-oss-storage-class"] = "Standard"
# headers["x-oss-object-acl"] = oss2.OBJECT_ACL_PRIVATE
# Specify the full path of the object and the string that you want to upload. The full path of the object cannot contain the bucket name.
# result = bucket.put_object('exampleobject.txt', 'Hello OSS', headers=headers)
result = bucket.put_object('exampleobject.txt', 'Hello OSS')
# Obtain the returned HTTP status code.
print('http status: {0}'.format(result.status))
# Obtain the request ID. A request ID uniquely identifies a request. We recommend that you add this parameter to the logs.
print('request_id: {0}'.format(result.request_id))
# Obtain the ETag value returned by the put_object method. The ETag value of an object is used to identify the object content.
print('ETag: {0}'.format(result.etag))
# Obtain the HTTP response headers.
print('date: {0}'.format(result.headers['date']))
```

Upload bytes

The following code provides an example on how to upload bytes to an object named exampleobject.txt in a bucket named examplebucket:

```
# -*- coding: utf-8 -*-
import oss2
# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to acc
ess OSS, because the account has permissions on all API operations. We recommend that you u
se a RAM user to call API operations or perform routine O&M. To create a RAM user, log on t
o the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# Set yourEndpoint to the endpoint of the region in which the bucket is located. For exampl
e, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss
-cn-hangzhou.aliyuncs.com.
# Specify the name of the bucket.
bucket = oss2.Bucket(auth, 'yourEndpoint', 'examplebucket')
# Specify the full path of the object and the bytes that you want to upload. The full path
of the object cannot contain the bucket name.
bucket.put_object('exampleobject.txt', b'Hello OSS')
```

Upload Unicode characters

The following code provides an example on how to upload Unicode characters to an object named `exampleobject.txt` in a bucket named `examplebucket`. When you upload Unicode characters, OSS converts the characters into UTF-8-encoded bytes.

```
# -*- coding: utf-8 -*-
import oss2
# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to acc
ess OSS, because the account has permissions on all API operations. We recommend that you u
se a RAM user to call API operations or perform routine O&M. To create a RAM user, log on t
o the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# Set yourEndpoint to the endpoint of the region in which the bucket is located. For exampl
e, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss
-cn-hangzhou.aliyuncs.com.
# Specify the name of the bucket.
bucket = oss2.Bucket(auth, 'yourEndpoint', 'examplebucket')
# Specify the full path of the object and the Unicode characters that you want to upload. T
he full path of the object cannot contain the bucket name.
bucket.put_object('exampleobject.txt', u'Hello OSS')
```

Upload a network stream

The following code provides an example on how to upload a network stream to an object named `exampleobject.txt` in a bucket named `examplebucket`. OSS uploads network streams as iterable objects by using chunked encoding.

```
# -*- coding: utf-8 -*-
import oss2
import requests

# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS, because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com.
# Specify the name of the bucket.
bucket = oss2.Bucket(auth, 'yourEndpoint', 'examplebucket')
# The requests.get method returns an iterable object. OSS SDK for Python uploads the object by using chunked encoding.
# Specify the address of the network stream.
input = requests.get('http://www.aliyun.com')
# Specify the full path of the object. The full path of the object cannot contain the bucket name.
bucket.put_object('exampleobject.txt', input)
```

Upload a local file

The following code provides an example on how to upload a local file named `examplefile.txt` to a bucket named `examplebucket`. The uploaded file is stored as an object named `exampleobject.txt` in OSS. OSS uploads local files as file objects and opens the files in binary mode, such as the `rb` mode, during upload.

```
# -*- coding: utf-8 -*-
import oss2
import os
# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS, because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com.
# Specify the name of the bucket.
bucket = oss2.Bucket(auth, 'yourEndpoint', 'examplebucket')
# The file must be opened in binary mode.
# Specify the full path of the local file. By default, if you do not specify the full path of a local file, the local file is uploaded from the path of the project to which the sample program belongs.
with open('D:\\localpath\\examplefile.txt', 'rb') as fileobj:
    # Use the seek method to read data from byte 1000 of the file. The data is uploaded from byte 1000 to the last byte of the local file.
    fileobj.seek(1000, os.SEEK_SET)
    # Use the tell method to obtain the current position.
    current = fileobj.tell()
    # Specify the full path of the object. The full path of the object cannot contain the bucket name.
    bucket.put_object('exampleobject.txt', fileobj)
```

OSS SDK for Python also provides a more convenient method to upload local files. The following code provides an example of this method:

```
# Specify the full paths of the local file and object. The full path of the object cannot contain the bucket name.
# By default, if you do not specify the full path of a local file, the local file is uploaded from the path of the project to which the sample program belongs.
bucket.put_object_from_file('exampleobject.txt', 'D:\\localpath\\examplefile.txt')
```

5.6.3. Append upload

You can call the AppendObject operation to append content to existing appendable objects.

Usage notes

- If the object to which you want to append content does not exist, an appendable object is created when you call the AppendObject operation.
- When you call the AppendObject operation to append content to an object, you may encounter a few situations: If the object is an appendable object, and the specified position from which the append operation starts is equal to the current object length, the content is appended to the end of the object. If the specified position from which the append operation starts is not equal to the current object length, a `PositionNotEqualToLength` exception is thrown. If the object is not an appendable object, an `ObjectNotAppendable` exception is thrown.
- The CopyObject operation cannot be performed on appendable objects.

Sample code

The following code provides an example on how to perform append upload:

```
# -*- coding: utf-8 -*-
import oss2

# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to acc
ess Object Storage Service (OSS) because the account has permissions on all API operations.
We recommend that you use a RAM user to call API operations or perform routine operations a
nd maintenance. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('<yourAccessKeyId>', '<yourAccessKeySecret>')
# In this example, the endpoint of the China (Hangzhou) region is used. Specify the endpoin
t based on your business requirements.
# Specify the name of the bucket. Example: examplebucket.
bucket = oss2.Bucket(auth, 'https://oss-cn-hangzhou.aliyuncs.com', 'examplebucket')
# The following code provides an example on how to specify headers for the append upload:
# headers = dict()
# Specify the caching behavior of the web page for the object.
# headers['Cache-Control'] = 'no-cache'
# Specify the name of the object when the object is downloaded.
# headers['Content-Disposition'] = 'oss_MultipartUpload.txt'
# Specify the encoding format for the content of the object.
# headers['Content-Encoding'] = 'utf-8'
# Specify the request header that is used to check whether the message content is the same
as the content of the message when it was sent.
# headers['Content-MD5'] = 'ohhngLBJFiKkPSBOleNaUA=='
# Specify the expiration time.
# headers['Expires'] = 'Wed, 08 Jul 2022 16:57:01 GMT'
# Specify the access control list (ACL) of the object. In this example, this parameter is s
et to OBJECT_ACL_PRIVATE, which indicates private.
# headers['x-oss-object-acl'] = oss2.OBJECT_ACL_PRIVATE
# Specify whether to overwrite the object with the same name in the append upload.
# headers['x-oss-forbid-overwrite'] = 'true'
# Specify the method that is used to encrypt objects on the OSS server. In this example, th
e method is set to server-side encryption by using OSS-managed keys (SSE-OSS)
# headers[OSS_SERVER_SIDE_ENCRYPTION] = SERVER_SIDE_ENCRYPTION_AES256
# Specify the storage class of the object.
# headers['x-oss-storage-class'] = oss2.BUCKET_STORAGE_CLASS_STANDARD
# You can add parameters whose names are prefixed with x-oss-meta- when you call the Append
Object operation to create an append object. These parameters cannot be included in the req
uests when you append content to the existing append objects. Parameters whose names are pr
efixed with x-oss-meta- are considered the metadata of the object.
# headers['x-oss-meta-author'] = 'Alice'
# result = bucket.append_object(exemplerdir/exampleobject.txt, 0, 'content of first append',
headers=headers)
# Set the position from which the first append operation starts to 0.
# Specify the full path of the object. The full path of the object cannot contain the bucke
t name. Example: exemplerdir/exampleobject.txt.
result = bucket.append_object('exemplerdir/exampleobject.txt', 0, 'content of first append')
# If you have appended content to the object, you can obtain the position from which the cu
rrent append operation starts from the next_position field in the response returned by the
last operation or by using the bucket.head_object method.
bucket.append_object('<yourObjectName>', result.next_position, 'content of second append')
```

References

- For more information about the complete sample code of append upload, visit [GitHub](#).
- For more information about the API operation that you can call to perform append upload, see [AppendObject](#).

5.6.4. Resumable upload

Before you upload an object to Object Storage Service (OSS) by using resumable upload, you can specify a directory for the checkpoint file that stores resumable upload records. If an object fails to upload due to a network exception or program error, the upload task is resumed from the position recorded in the checkpoint file to upload the remaining data.

Notice

- Multiple threads are used in resumable upload. Therefore, you do not need to encapsulate multiple upload threads when you call the resumable upload method. Otherwise, data may be transmitted repeatedly.
- We recommend that you increase the size of each part when the network connection is stable and decrease the size of each part when the network connection is unstable.

Sample code

The following code provides an example on how to perform resumable upload by using multipart upload:

```
# -*- coding: utf-8 -*-
import oss2

# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# Specify the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com.
# Specify the bucket name. Example: examplebucket.
bucket = oss2.Bucket(auth, 'https://oss-cn-hangzhou.aliyuncs.com', 'examplebucket')
# Set yourObjectName to the full path of the object. The full path cannot contain the bucket name. Example: exampledir/exampleobject.txt.
# Set yourLocalFile to the full path of the local file. Example: D:\\localpath\\examplefile.txt. By default, if you do not specify the full path of the local file, the local file is uploaded from the path of the project to which the sample program belongs.
oss2.resumable_upload(bucket, 'exampledir/exampleobject.txt', 'D:\\localpath\\examplefile.txt')
# If you do not specify a directory by using the store parameter, the py-oss-upload directory is created in the HOME directory to store the checkpoint information.
# OSS SDK for Python V2.1.0 and later support the configuration of optional parameters in resumable upload. The following code provides an example on how to configure optional parameters in resumable upload:
# import sys
# # If the size of the data to upload cannot be determined, the total_bytes parameter is set to None.
```

```

# def percentage(consumed_bytes, total_bytes):
#     if total_bytes:
#         rate = int(100 * (float(consumed_bytes) / float(total_bytes)))
#         print('\r{0}% '.format(rate), end='')
#         sys.stdout.flush()
# If you use the store parameter to specify a directory, the checkpoint information is stor
ed in the specified directory. If you use the num_threads parameter to specify the number o
f concurrent upload threads, make sure that the oss2.defaults.connection_pool_size value is
greater than or equal to the number of concurrent upload threads. The default number of con
current upload threads is 1.
# oss2.resumable_upload(bucket, '<yourObjectName>', '<yourLocalFile>',
#                         store=oss2.ResumableStore(root='/tmp'),
#                         # Specify that multipart upload is used when the object size is gre
ater than or equal to the value of the multipart_threshold parameter. The default value of
the parameter is 10 MB.
#                         multipart_threshold=100*1024,
#                         # Specify the size of each part. Unit: bytes. Valid values: 100 KB
to 5 GB. Default value: 100 KB.
#                         part_size=100*1024,
#                         # Configure the function used to indicate the progress of the resum
able upload task by implementing upload callback.
#                         progress_callback=percentage,
#                         # If you use the num_threads parameter to specify the number of con
current upload threads, make sure that the oss2.defaults.connection_pool_size value is grea
ter than or equal to the number of concurrent upload threads. The default number of concurr
ent upload threads is 1.
#                         num_threads=4)

```

References

For the complete sample code that is used to perform resumable upload, visit [GitHub](#).

5.6.5. Multipart upload

By using the multipart upload feature provided by OSS, you can split a large object into multiple parts and upload them separately. After all parts are uploaded, call the CompleteMultipartUpload operation to combine these parts into a single object to implement resumable upload.

Process

To upload an object by using multipart upload, perform the following steps:

1. Initiate a multipart upload task.
Call the bucket.init_multipart_upload method to obtain an unique upload ID in Object Storage Service (OSS).
2. Upload parts.
Call the bucket.upload_part method to upload the parts.

 Note

- Part numbers identify the relative positions of parts in an object that share the same upload ID. If you upload a part by using an existing part number, the existing part is overwritten.
- OSS includes the MD5 hash of part data in the ETag header and returns the MD5 hash to the user.
- OSS calculates the MD5 hash of uploaded data and compares the MD5 hash with the MD5 hash calculated by the SDK. If the two hashes are different, the InvalidDigest error code is returned.

3. Complete the multipart upload task.

After you upload all the parts, call the `bucket.complete_multipart_upload` method to combine the parts into a complete object.

Complete sample code

The following code provides a complete example that describes the process of multipart upload:

```
# -*- coding: utf-8 -*-
import os
from oss2 import SizedFileAdapter, determine_part_size
from oss2.models import PartInfo
import oss2

# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# In this example, the endpoint of the China (Hangzhou) region is used. Specify the endpoint based on your business requirements.
# Specify the name of the bucket. Example: examplebucket.
bucket = oss2.Bucket(auth, 'https://oss-cn-hangzhou.aliyuncs.com', 'examplebucket')
# Specify the full path of the object. The full path cannot contain the bucket name. Example: exampledir/exampleobject.txt.
key = 'exampledir/exampleobject.txt'
# Specify the full path of the local file that you want to upload. Example: D:\\localpath\\examplefile.txt.
filename = 'D:\\localpath\\examplefile.txt'
total_size = os.path.getsize(filename)
# Use the determine_part_size method to determine the size of each part.
part_size = determine_part_size(total_size, preferred_size=100 * 1024)
# Initiate a multipart upload task.
# If you want to specify the storage class of the object when you initiate the multipart upload task, configure the related headers when you call the init_multipart_upload method.
# headers = dict()
# Specify the caching behavior of the web page for the object.
# headers['Cache-Control'] = 'no-cache'
# Specify the name of the object when it is downloaded.
# headers['Content-Disposition'] = 'oss_MultipartUpload.txt'
# Specify the encoding format for the content of the object.
# headers['Content-Encoding'] = 'utf-8'
```

```

# Specify the validity period. Unit: milliseconds.
# headers['Expires'] = '1000'
# Specify whether to overwrite the existing object with the same name as the uploaded object
# when you initiate the multipart upload task. In this example, this parameter is set to true,
# which indicates that the existing object with the same name cannot be overwritten by the
# uploaded object.
# headers['x-oss-forbid-overwrite'] = 'true'
# Specify the server-side encryption method that is used to encrypt each part of the object
# that you want to upload.
# headers[OSS_SERVER_SIDE_ENCRYPTION] = SERVER_SIDE_ENCRYPTION_KMS
# Specify the algorithm that is used to encrypt the object. If you do not configure this parameter,
# objects are encrypted by using AES-256.
# headers[OSS_SERVER_SIDE_DATA_ENCRYPTION] = SERVER_SIDE_ENCRYPTION_KMS
# Specify the ID of the Customer Master Key (CMK) that is managed by Key Management Service
# (KMS).
# headers[OSS_SERVER_SIDE_ENCRYPTION_KEY_ID] = '9468da86-3509-4f8d-a61e-6eableac****'
# Specify the storage class of the object.
# headers['x-oss-storage-class'] = oss2.BUCKET_STORAGE_CLASS_STANDARD
# Specify tags for the object. You can specify multiple tags for the object at the same time.
# headers[OSS_OBJECT_TAGGING] = 'k1=v1&k2=v2&k3=v3'
# upload_id = bucket.init_multipart_upload(key, headers=headers).upload_id
upload_id = bucket.init_multipart_upload(key).upload_id
parts = []
# Upload the parts one by one.
with open(filename, 'rb') as fileobj:
    part_number = 1
    offset = 0
    while offset < total_size:
        num_to_upload = min(part_size, total_size - offset)
        # Call the SizedFileAdapter(fileobj, size) method to generate a new object and recalculate
        # the length of the append object.
        result = bucket.upload_part(key, upload_id, part_number,
                                    SizedFileAdapter(fileobj, num_to_upload))
        parts.append(PartInfo(part_number, result.etag))
        offset += num_to_upload
        part_number += 1
# Complete the multipart upload task.
# The following code provides an example on how to configure headers when you complete the
# multipart upload task:
headers = dict()
# Specify the access control list (ACL) of the object. In this example, this parameter is set
# to OBJECT_ACL_PRIVATE, which indicates private.
# headers["x-oss-object-acl"] = oss2.OBJECT_ACL_PRIVATE
# If you configure x-oss-complete-all:yes in the request, OSS lists all parts that are uploaded
# by using the current upload ID, sorts the parts by part number, and then performs the
# CompleteMultipartUpload operation.
# If you configure x-oss-complete-all:yes in the request, the request body cannot be specified.
# Otherwise, an error occurs.
headers["x-oss-complete-all"] = 'yes'
bucket.complete_multipart_upload(key, upload_id, parts, headers=headers)
# bucket.complete_multipart_upload(key, upload_id, parts)
# Verify the result of the multipart upload task.
with open(filename, 'rb') as fileobj:
    result = bucket.get_object(key).read()
    # Check the result.

```

```
assert bucket.get_object(key).read() == fileobj.read()
```

 **Note** We recommend that you increase the size of each part when network conditions are stable. We recommend that you decrease the size of each part when network conditions are unstable.

Cancel a multipart upload task

You can call the `bucket.abort_multipart_upload` method to cancel a multipart upload task. If a multipart upload task is canceled, the upload ID cannot be used to upload parts. In addition, the uploaded parts are deleted.

The following code provides an example on how to cancel a multipart upload task:

```
# -*- coding: utf-8 -*-
import os
import oss2

# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# In this example, the endpoint of the China (Hangzhou) region is used. Specify the endpoint based on your business requirements.
# Specify the name of the bucket. Example: examplebucket.
bucket = oss2.Bucket(auth, 'https://oss-cn-hangzhou.aliyuncs.com', 'examplebucket')
# Specify the full path of the object. The full path cannot contain the bucket name. Example: exampledir/exampleobject.txt.
key = 'exampledir/exampleobject.txt'
# Obtain the upload ID returned by the init_multipart_upload method.
upload_id = 'yourUploadId'
# Cancel the multipart upload task that uses the specified upload ID. The uploaded parts are deleted.
bucket.abort_multipart_upload(key, upload_id)
```

List the uploaded parts

The following code provides an example on how to list the uploaded parts:

```
# -*- coding: utf-8 -*-
import os
import oss2
# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# In this example, the endpoint of the China (Hangzhou) region is used. Specify the endpoint based on your business requirements.
bucket = oss2.Bucket(auth, 'https://oss-cn-hangzhou.aliyuncs.com', 'yourBucketName')
# Specify the full path of the object. The full path cannot contain the bucket name. Example: exampledir/exampleobject.txt.
key = 'exampledir/exampleobject.txt'
# Obtain the upload ID returned by the init_multipart_upload method.
upload_id = 'yourUploadId'
# List the uploaded parts that use the specified upload ID.
for part_info in oss2.PartIterator(bucket, key, upload_id):
    print('part_number:', part_info.part_number)
    print('etag:', part_info.etag)
    print('size:', part_info.size)
```

List multipart upload tasks

- List the multipart upload tasks of the specified object

The following code provides an example on how to list the multipart upload tasks of the specified object:

```
# -*- coding: utf-8 -*-
import os
import oss2
# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# In this example, the endpoint of the China (Hangzhou) region is used. Specify the endpoint based on your business requirements.
# Specify the name of the bucket. Example: examplebucket.
bucket = oss2.Bucket(auth, 'https://oss-cn-hangzhou.aliyuncs.com', 'examplebucket')
# Specify the full path of the object. The full path cannot contain the bucket name. Example: exampledir/exampleobject.txt.
key = 'exampledir/exampleobject.txt'
# List all multipart upload tasks of the object. Each time the init_multipart_upload method is called for the same object, a different upload ID is returned.
# An upload ID uniquely identifies a multipart upload task.
for upload_info in oss2.ObjectUploadIterator(bucket, key):
    print('key:', upload_info.key)
    print('upload_id:', upload_info.upload_id)
```

- List all multipart upload tasks initiated for a bucket

The following code provides an example on how to list all multipart upload tasks initiated for a bucket:

```
# -*- coding: utf-8 -*-
import os
import oss2

# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# In this example, the endpoint of the China (Hangzhou) region is used. Specify the endpoint based on your business requirements.
# Specify the name of the bucket. Example: examplebucket.
bucket = oss2.Bucket(auth, 'https://oss-cn-hangzhou.aliyuncs.com', 'examplebucket')
# List all multipart upload tasks initiated for the bucket.
for upload_info in oss2.MultipartUploadIterator(bucket):
    print('key:', upload_info.key)
    print('upload_id:', upload_info.upload_id)
```

- List the multipart upload tasks of objects whose names contain the specified prefix in a bucket

The following code provides an example on how to list the multipart upload tasks of objects whose names contain the specified prefix in a bucket:

```
# -*- coding: utf-8 -*-
import os
import oss2

# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# In this example, the endpoint of the China (Hangzhou) region is used. Specify the endpoint based on your business requirements.
# Specify the name of the bucket. Example: examplebucket.
bucket = oss2.Bucket(auth, 'https://oss-cn-hangzhou.aliyuncs.com', 'examplebucket')
# List the multipart upload tasks of objects whose names contain the test prefix in the bucket.
for upload_info in oss2.MultipartUploadIterator(bucket, prefix='test'):
    print('key:', upload_info.key)
    print('upload_id:', upload_info.upload_id)
```

References

- The following API operations are required to perform multipart upload:
 - The API operation that you can call to initiate a multipart upload task. For more information, see [InitiateMultipartUpload](#).
 - The API operation that you can call to upload data by part. For more information, see [UploadPart](#).
 - The API operation that you can call to complete a multipart upload task. For more information, see [CompleteMultipartUpload](#).
- For more information about the API operation that you can call to cancel a multipart upload task, see [AbortMultipartUpload](#).

- For more information about the API operation that you can call to list uploaded parts, see [ListParts](#).
- For more information about the API operation that you can call to list ongoing multipart upload tasks, see [ListMultipartUploads](#). Ongoing multipart upload tasks include tasks that have been initiated but are not completed and tasks that have been canceled.

5.6.6. Upload progress bars

You can use progress bars to indicate the upload or download progress.

For the complete code of progress bars, see [GitHub](#).

The following code is used as an example to describe how to view progress information with `bucket.put_object`:

```
# -*- coding: utf-8 -*-
from __future__ import print_function
import os, sys
import oss2

# It is highly risky to log on with AccessKey of an Alibaba Cloud account because the account has permissions on all the APIs in OSS. We recommend that you log on as a RAM user to access APIs or perform routine operations and maintenance. To create a RAM account, log on to https://ram.console.aliyun.com.
auth = oss2.Auth('<yourAccessKeyId>', '<yourAccessKeySecret>')
# This example uses endpoint China (Hangzhou). Specify the actual endpoint based on your requirements.
bucket = oss2.Bucket(auth, 'http://oss-cn-hangzhou.aliyuncs.com', '<yourBucketName>')
# If the length of the data to be uploaded cannot be determined, the value of total_bytes is None.
def percentage(consumed_bytes, total_bytes):
    if total_bytes:
        rate = int(100 * (float(consumed_bytes) / float(total_bytes)))
        print('\r{0}% '.format(rate), end='')
        sys.stdout.flush()
# progress_callback is an optional parameter used to implement progress bars.
bucket.put_object('<yourObjectName>', 'a'*1024*1024, progress_callback=percentage)
```

5.6.7. Upload callback

After Object Storage Service (OSS) completes simple upload by using `put_object` and `put_object_from_file` and multipart upload by using `complete_multipart_upload`, a callback request can be sent to the application server. To implement callback, you need only to send a request that contains relevant callback parameters to OSS.

Sample code

The following code provides an example on how to implement upload callback when you upload a string to the `exampleobject.txt` object in the `examplefiles` directory of the `examplebucket` bucket:

```
# -*- coding: utf-8 -*-
import json
import base64
import oss2

# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com.
# Specify the name of the bucket.
bucket = oss2.Bucket(auth, 'yourEndpoint', 'examplebucket')
# Define the function used to encode the callback parameter in Base64.
def encode_callback(callback_params):
    cb_str = json.dumps(callback_params).strip()
    return oss2.compat.to_string(base64.b64encode(oss2.compat.to_bytes(cb_str)))
# Set the upload callback parameter.
callback_params = {}
# Set the address of the server to which the callback request is sent. Example: http://oss-demo.aliyuncs.com:23450.
callback_params['callbackUrl'] = 'http://oss-demo.aliyuncs.com:23450'
# Optional. Set the value of the Host field included in the callback request header.
#callback_params['callbackHost'] = 'yourCallbackHost'
# Set the value of the body field included in the callback request.
callback_params['callbackBody'] = 'bucket=${bucket}&object=${object}'
# Set Content-Type for the callback request.
callback_params['callbackBodyType'] = 'application/x-www-form-urlencoded'
encoded_callback = encode_callback(callback_params)
# Configure custom parameters for the callback request. Each custom parameter consists of a key and a value. The key must start with x:.
callback_var_params = {'x:my_var1': 'my_val1', 'x:my_var2': 'my_val2'}
encoded_callback_var = encode_callback(callback_var_params)
# Implement upload callback.
params = {'x-oss-callback': encoded_callback, 'x-oss-callback-var': encoded_callback_var}
# Specify the string and the full path of the object. The full path of the object cannot contain the bucket name.
result = bucket.put_object('examplefiles/exampleobject.txt', 'a'*1024*1024, params)
```

References

- For more information about the complete sample code for upload callback, visit [GitHub](#).
- For more information about the API operation that is called to configure upload callback, see [Callback](#).

5.7. Download objects

5.7.1. Overview

This topic is the overview of object download in OSS Python SDK.

OSS Python SDK provides the following download methods:

- [Streaming download](#)
- [Download to your local file](#)
- [Range download](#)
- [Resumable download](#)

You can view the download progress shown in the [download progress bar](#).

5.7.2. Streaming download

If you need to download a large object or it takes a long time to download an object at a time, you can use streaming download to download the object in increments.

The following code provides an example on how to use streaming download to download an object named `exampleobject.txt` from a bucket named `examplebucket`:

```
# -*- coding: utf-8 -*-
import oss2

# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log
# on to OSS because the account has permissions on all API operations. We recommend that you
# use your RAM user's credentials to call API operations or perform routine operations and ma
# intenance. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# Set yourEndpoint to the endpoint of the region in which the bucket is located. For exampl
# e, if your bucket is located in the China (Hangzhou) region, set yourEndpoint to https://os
# s-cn-hangzhou.aliyuncs.com.
# Specify the bucket name.
bucket = oss2.Bucket(auth, 'yourEndpoint', 'examplebucket')
# The value returned by bucket.get_object is a file-like and iterable object.
# Specify the full path of the object. The full path of the object cannot contain bucket na
# mes.
object_stream = bucket.get_object('exampleobject.txt')
print(object_stream.read())
# You must call read() to read the object from the stream returned by get_object before you
# can calculate the CRC-64 value of the object.
if object_stream.client_crc != object_stream.server_crc:
    print("The CRC checksum between client and server is inconsistent!")
```

The following code provides an example on how to download the data of an object named `exampleobject.txt` from a stream to a file named `examplefile.txt` in the local directory `D:\localpath`:


```
import shutil
import oss2
# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log
on to OSS because the account has permissions on all API operations. We recommend that you
use your RAM user's credentials to call API operations or perform routine operations and ma
intenance. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# Set yourEndpoint to the endpoint of the region in which the bucket is located. For exampl
e, if your bucket is located in the China (Hangzhou) region, set yourEndpoint to https://os
s-cn-hangzhou.aliyuncs.com.
# Specify the bucket name.
bucket = oss2.Bucket(auth, 'yourEndpoint', 'examplebucket')
# object_stream is a file-like object. You can use the shutil.copyfileobj method to downloa
d the data from a stream to the local file.
# Specify the full path of the object. The full path of the object cannot contain bucket na
mes.
object_stream = bucket.get_object('exampleobject.txt')
# Download the object to a local file in the specified path. If the specified local file ex
ists, the downloaded object overwrites the local file. Otherwise, the local file is created
.
# If the path for the object is not specified, the downloaded object is saved to the path o
f the project to which the sample program belongs.
with open('D:\\localpath\\examplefile.txt', 'wb') as local_fileobj:
    shutil.copyfileobj(object_stream, local_fileobj)
```

The following code provides an example on how to copy the data of an object named `exampleobject.txt` from a stream to an object named `exampleobjectnew.txt`:

```
import oss2
# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log
on to OSS because the account has permissions on all API operations. We recommend that you
use your RAM user's credentials to call API operations or perform routine operations and ma
intenance. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# Set yourEndpoint to the endpoint of the region in which the bucket is located. For exampl
e, if your bucket is located in the China (Hangzhou) region, set yourEndpoint to https://os
s-cn-hangzhou.aliyuncs.com.
# Specify the bucket name.
bucket = oss2.Bucket(auth, 'yourEndpoint', 'examplebucket')
# object_stream is an iterable object. You can copy the data of the object from a stream to
another object in the same bucket.
# Specify the full path of the source object. The full path of the object cannot contain bu
cket names.
object_stream = bucket.get_object('exampleobject.txt')
# Specify the full path of the destination object. The full path of the object cannot conta
in bucket names.
bucket.put_object('exampleobjectnew.txt', object_stream)
```

5.7.3. Download objects to local files

This topic describes how to download an object from a bucket to your computer.

Sample code

```
# -*- coding: utf-8 -*-
import oss2

# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# In this example, the endpoint of the China (Hangzhou) region is used. Specify the endpoint based on your business requirements.
# Specify the name of the bucket. Example: examplebucket.
bucket = oss2.Bucket(auth, 'http://oss-cn-hangzhou.aliyuncs.com', 'examplebucket')
# Specify the full path of the object. The path cannot contain the bucket name. Example: testfolder/exampleobject.txt.
# Download the object to your local computer and save the object to the specified local path D:\\localpath\\examplefile.txt. If the specified local file exists, the object to download overwrites the file. Otherwise, a file is created.
bucket.get_object_to_file('testfolder/exampleobject.txt', 'D:\\localpath\\examplefile.txt')
```

References

- For more information about the complete sample code that is used to download objects to local files, visit [GitHub](#).
- For more information about the API operation that you can call to download objects to local files, see [Get Object](#).

5.7.4. Range download

If you need only part of the data in an object, you can use range download to download data within a specified range.

Specify a valid range

The following code provides an example on how to specify a valid range to download data:

```
# -*- coding: utf-8 -*-
import oss2

# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log on to OSS because the account has permissions on all API operations. We recommend that you use your RAM user's credentials to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('<yourAccessKeyId>', '<yourAccessKeySecret>')
# The endpoint of the China (Hangzhou) region is used in this example. Specify the actual endpoint.
bucket = oss2.Bucket(auth, 'http://oss-cn-hangzhou.aliyuncs.com', '<yourBucketName>')
# For an object whose size is 1,000 bytes, the valid range is from byte 0 to byte 999.
# Query the data that is within the range from byte 0 to byte 999, which includes a total of 1,000 bytes. If the specified value range is invalid, for example, the specified range includes a negative number, or the specified value is larger than the object size, the entire object is downloaded.
object_stream = bucket.get_object('<yourObjectName>', byte_range=(0, 999))
```

An invalid range is specified

For an object whose size is 1,000 bytes, the valid range is from byte 0 to byte 999. If the specified range is not within the range from byte 0 to byte 999, the range does not take effect. Then, OSS returns HTTP status code 200 and the data of the entire object. The following list provides examples of invalid requests and the returned results:

- If you set Range: bytes to 500-2000, the value at the end of the range is invalid. Then, OSS returns HTTP status code 200 and the data of the entire object.
- If you set Range: bytes to 1000-2000, the value at the start of the range is invalid. Then, OSS returns HTTP status code 200 and the data of the entire object.

Compatible download

If you add x-oss-range-behavior:standard to the request header, the download behavior is modified when the specified range is not within the valid range. Example: Use range download to download an object whose size is 1,000 bytes.

- If you set Range: bytes to 500-2000, the value at the end of the range is invalid. Then, OSS returns HTTP status code 206 and the data that is within the range from byte 500 to byte 999.
- If you set Range: bytes to 1000-2000, the value at the start of the range is invalid. Then, OSS returns HTTP status code 416 and error code InvalidRange.

The following code provides an example on how to specify compatible behaviors to download data by range:

```
# -*- coding: utf-8 -*-
import oss2

# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log
on to OSS because the account has permissions on all API operations. We recommend that you
use your RAM user's credentials to call API operations or perform routine operations and ma
intenance. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('<yourAccessKeyId>', '<yourAccessKeySecret>')
# The endpoint of the China (Hangzhou) region is used in this example. Specify the actual e
ndpoint.
bucket = oss2.Bucket(auth, 'http://oss-cn-hangzhou.aliyuncs.com', '<yourBucketName>')
# Create an object whose size is 1,000 bytes.
object_name = 'rangeTest.txt'
content = 'a' * 1000
bucket.put_object(object_name, content)
headers = {'x-oss-range-behavior': 'standard'}
# Specify the compatible behavior.
# When the value at the end of the range is invalid, OSS returns the data that is within th
e range from byte 500 to byte 999. The HTTP status code 206 is also returned.
object_stream = bucket.get_object(object_name, byte_range=(500, 2000), headers=headers)
print('standard get 500~2000 http status code:', object_stream.status)
print('standard get 500~2000 content length:', object_stream.content_length)
try:
    # When the value at the start of the range is invalid, exceptions are thrown. The retur
ned HTTP status code is 416 and the returned error code is InvalidRange.
    object_stream = bucket.get_object(object_name, byte_range=(1000, 2000), headers=headers
)
except oss2.exceptions.ServerError as e:
    print('standard get 1000~2000 http status code:', e.status)
    print('standard get 1000~2000 error code:', e.code)
```

5.7.5. Resumable download

You may fail to download a large object if the network is unstable or other exceptions occur. In some cases, you may still fail to download the object even after multiple attempts. To resolve this issue, Object Storage Service (OSS) SDK for Python provides the resumable download feature. In resumable download, the object that you want to download is split into multiple parts and each part is separately downloaded. After all parts are downloaded, these parts are combined into a complete object.

Procedure

To use resumable download, perform the following steps:

1. Create a temporary local file with a name that consists of the original object name and a random suffix.
2. Specify the Range header in the HTTP request so that the object is read based on the range. Then, the read content is written to the corresponding position of the temporary local file.
3. After the download is completed, rename the temporary file as the destination file. If the destination file already exists, the downloaded data overwrites the data in the existing file. Otherwise, a new file is created.



Warning Do not use multiple programs or threads to call the method simultaneously to download the same object to a same destination file. This is because one piece of checkpoint information overwrites another on the local disk, or one temporary file name conflicts with another.

Sample code

The following code provides an example on how to perform resumable download:

```

# -*- coding: utf-8 -*-
import oss2
# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com.
# Specify the name of the bucket. Example: examplebucket.
bucket = oss2.Bucket(auth, 'https://oss-cn-hangzhou.aliyuncs.com', 'examplebucket')
# Set yourObjectName to the full path of the object. The full path cannot contain the bucket name. Example: exampledir/exampleobject.txt.
# Set yourLocalFile to the full path of the local file. Example: D:\\localpath\\examplefile.txt.
oss2.resumable_download(bucket, 'exampledir/exampleobject.txt', 'D:\\localpath\\examplefile.txt')
# If you do not specify a directory by using the store parameter, the .py-oss-download directory is created in the HOME directory to store the checkpoint information.
# The following optional parameters are supported by OSS SDK for Python that is later than version 2.1.0.
# import sys
# # If the length of the data to download cannot be determined, the value of total_bytes is None.
# def percentage(consumed_bytes, total_bytes):
#     if total_bytes:
#         rate = int(100 * (float(consumed_bytes) / float(total_bytes)))
#         print('\r{0}% '.format(rate), end='')
#         sys.stdout.flush()
# # If you use the store parameter to specify a directory, the checkpoint information is stored in the specified directory. If you use the num_threads parameter to specify the number of concurrent download threads, make sure that the value of oss2.defaults.connection_pool_size is greater than or equal to the number of concurrent download threads. The default number of concurrent threads is 1.
# oss2.resumable_download(bucket, 'exampledir/exampleobject.txt', 'D:\\localpath\\examplefile.txt',
#                         store=oss2.ResumableDownloadStore(root='/tmp'),
#                         # Specify that resumable download is used when the object size is greater than or equal to the optional multipart_threshold parameter. The default value of multipart_threshold is 10 MB.
#                         multiget_threshold=100*1024,
#                         # Set the size of each part. Unit: bytes. Valid values: 100 KB to 5 GB. Default value: 100 KB.
#                         part_size=100*1024,
#                         # Configure the download progress callback function.
#                         progress_callback=percentage,
#                         # If you use num_threads to set the number of concurrent download threads, set oss2.defaults.connection_pool_size to a value that is greater than or equal to the number of concurrent download threads. The default number of concurrent download threads is 1.
#                         num_threads=4)

```

References

- For more information about the complete sample code that is used to perform resumable download, visit [Git Hub](#).
- For more information about the API operation that you can call to perform resumable download, see [Get Object](#).

5.7.6. Download progress bars

You can use progress bars to indicate the upload or download progress.

For the complete code of progress bars, see [Git Hub](#).

The following code is used as an example to describe how to view progress information with `bucket.get_object_to_file`:

```
# -*- coding: utf-8 -*-
from __future__ import print_function
import os, sys
import oss2

# It is highly risky to log on with AccessKey of an Alibaba Cloud account because the account has permissions on all the APIs in OSS. We recommend that you log on as a RAM user to access APIs or perform routine operations and maintenance. To create a RAM account, log on to https://ram.console.aliyun.com.
auth = oss2.Auth('<yourAccessKeyId>', '<yourAccessKeySecret>')
# This example uses endpoint China (Hangzhou). Specify the actual endpoint based on your requirements.
bucket = oss2.Bucket(auth, 'http://oss-cn-hangzhou.aliyuncs.com', '<yourBucketName>')
# If the HTTP response header does not carry Content-Length, the value of total_bytes is None.
def percentage(consumed_bytes, total_bytes):
    if total_bytes:
        rate = int(100 * (float(consumed_bytes) / float(total_bytes)))
        print('\r{0}% '.format(rate), end='')
        sys.stdout.flush()
# progress_callback is an optional parameter used to implement progress bar.
bucket.get_object_to_file('<yourObjectName>', '<yourLocalFile>', progress_callback=percentage)
```

5.7.7. Conditional download

You can specify one or more conditions to download objects. If the specified conditions are met, the object is downloaded. If the specified conditions are not met, an error is returned and the object is not downloaded.

Sample code

The following code provides an example on how to perform conditional download:

```
import oss2
# Set endpoint to the endpoint of the region in which the bucket is located. For example, if
# the bucket is located in the China (Hangzhou) region, set endpoint to https://oss-cn-hang
# zhou.aliyuncs.com.
endpoint = 'https://oss-cn-hangzhou.aliyuncs.com'
# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to acc
# ess Object Storage Service (OSS) because the account has permissions on all API operations.
# We recommend that you use a RAM user to call API operations or perform routine operations a
# nd maintenance. To create a RAM user, log on to the RAM console.
access_key_id = 'yourAccessKeyId'
access_key_secret = 'yourAccessKeySecret'
# Specify the name of the bucket. Example: examplebucket. For more information about the na
# ming conventions for buckets, see Bucket naming conventions.
bucket_name = 'examplebucket'
# Specify the full path of the object. The full path of the object cannot contain the bucke
# t name. Example: exampledir/exampleobject.txt. For more information about the naming conven
# tions for objects, see Object naming conventions.
object_name = 'exampledir/exampleobject.txt'
auth = oss2.Auth(access_key_id, access_key_secret)
bucket = oss2.Bucket(auth, endpoint, bucket_name)
headers = dict()
# If the time that is specified in the request is earlier than the actual modified time of
# the object, the object is downloaded. Otherwise, the error code 304 Not modified is returne
# d.
headers['If-Modified-Since'] = 'Mon, 13 Dec 2021 14:47:53 GMT'
# If the time that is specified in the request is the same as or later than the actual modi
# fied time of the object, the object is downloaded. Otherwise, the error code 412 Preconditi
# on failed is returned.
# headers['If-Unmodified-Since'] = 'Mon, 13 Dec 2021 14:47:53 GMT'
# If the ETag value that is specified in the request matches the ETag value of the object,
# the object is downloaded. Otherwise, the error code 412 Precondition failed is returned.
# headers['If-Match'] = 'DC21493F505BA3739562D8CC452C****'
# If the ETag value that is specified in the request does not match the ETag value of the o
# bject, the object is downloaded. Otherwise, the error code 304 Not modified is returned.
# headers['If-None-Match'] = 'DC21493F505BA3739562D8CC452C****'
bucket.get_object(object_name, headers=headers)
```

References

For more information about the API operation that you can call to perform conditional download, see [GetObject](#).

5.8. Manage objects

5.8.1. Overview

This topic is the overview of object management in OSS Python SDK.

You can perform the following operations on objects in a bucket with APIs:

- [Determine whether an object exists](#)
- [Manage ACL for an object](#)

- [Manage Object Meta](#)
- [List objects](#)
- [Delete an object](#)
- [Copy an object](#)
- [Restore an archive object](#)
- [Manage a symbolic link](#)

5.8.2. Determine whether an object exists

This topic describes how to determine whether an object exists.

```
# -*- coding: utf-8 -*-
import oss2

# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log
on to Object Storage Service (OSS) because the account has permissions on all API operation
s. We recommend that you use a Resource Access Management (RAM) user to call API operations
or perform routine operations and maintenance. To create a RAM user, log on to the RAM cons
ole.
auth = oss2.Auth('<yourAccessKeyId>', '<yourAccessKeySecret>')
# The endpoint of the China (Hangzhou) region is used in this region. Specify the actual en
dpoint.
bucket = oss2.Bucket(auth, 'http://oss-cn-hangzhou.aliyuncs.com', 'examplebucket')
# Specify the full path of the object. The full path of the object cannot contain bucket na
mes.
exist = bucket.object_exists('exampleobject.txt')
# If the returned value is true, the specified object exists. If the returned value is fals
e, the specified object does not exist.
if exist:
    print('object exist')
else:
    print('object not exist')
```

5.8.3. Manage the ACL of an object

This topic describes how to manage the access control list (ACL) of an object.

The following table describes the ACLs that you can configure for an object.

ACL	Description	Value
Inherited from the bucket	The ACL of the object is the same as the ACL of the bucket in which the object is stored.	oss2.OBJECT_ACL_DEFAULT
Private	Only the object owner and authorized users are granted the read and write permissions on the object.	oss2.OBJECT_ACL_PRIVATE

ACL	Description	Value
Public read	Only the object owner and authorized users are granted the read and write permissions on the object. Other users are granted only the read permissions on the object. Exercise caution when you set the ACL of the object to this value.	oss2.OBJECT_ACL_PUBLIC_READ
Public read/write	All users are granted the read and write permissions on the object. Exercise caution when you set the ACL of the object to this value.	oss2.OBJECT_ACL_PUBLIC_READ_WRITE

The ACL of the object takes precedence over the ACL of the bucket. For example, if the ACL of a bucket is private and the ACL of an object that is stored in the bucket is public, all users are granted the read and write permissions on the object. If the ACL of an object is not configured, the ACL of the object is the same as the ACL of the bucket in which the object is stored.

Configure the ACL of an object

The following code provides an example on how to configure the ACL of a specified object:

```
# -*- coding: utf-8 -*-
import oss2

# The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('<yourAccessKeyId>', '<yourAccessKeySecret>')
# In this example, the endpoint of the China (Hangzhou) region is used. Specify your actual endpoint.
bucket = oss2.Bucket(auth, 'http://oss-cn-hangzhou.aliyuncs.com', '<yourBucketName>')
# Configure the ACL of the object.
bucket.put_object_acl('<yourObjectName>', oss2.OBJECT_ACL_PUBLIC_READ)
```

Obtain the ACL of an object

The following code provides an example on how to query the ACL of a specified object:

```
# -*- coding: utf-8 -*-
import oss2
# The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('<yourAccessKeyId>', '<yourAccessKeySecret>')
# In this example, the endpoint of the China (Hangzhou) region is used. Specify your actual endpoint.
bucket = oss2.Bucket(auth, 'http://oss-cn-hangzhou.aliyuncs.com', '<yourBucketName>')
print(bucket.get_object_acl('<yourObjectName>').acl)
```

5.8.4. Manage object metadata

Object metadata includes HTTP headers and user metadata.

 **Note** For more information about object metadata, see [Manage object metadata](#) in OSS Developer Guide.

Configure HTTP headers

The following code provides an example on how to configure HTTP headers for the `exampleobject.txt` object in the `exampledir` directory stored in the `examplebucket` bucket.

 **Note** For more information about HTTP headers, see [RFC 2616](#).

```
# -*- coding: utf-8 -*-
import oss2
# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access Object Storage Service (OSS) because the account has permissions on all API operations. We recommend that you use a Resource Access Management (RAM) user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com.
# Specify the name of the bucket. Example: examplebucket.
bucket = oss2.Bucket(auth, 'yourEndpoint', 'examplebucket')
# Specify the full path of the object. Example: exampledir/exampleobject.txt. The full path of the object cannot contain the bucket name.
object_name = 'exampledir/exampleobject.txt'
# Specify the string to upload.
content = '{"age": 1}'
# Configure HTTP headers. For example, set the value of the header Content-Type to 'application/json; charset=utf-8'.
bucket.put_object(object_name, content, headers={'Content-Type': 'application/json; charset=utf-8'})
```

Configure user metadata

You can configure the user metadata of an object to describe the object.

 **Note** The user metadata is prefixed with `x-oss-meta-` in OSS. For more information, see [PutObject](#).

The following code provides an example on how to configure the user metadata of the `exampleobject.txt` object in the `exampledir` directory stored in the `examplebucket` bucket:

```
# -*- coding: utf-8 -*-
import oss2

# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to acc
ess OSS because the account has permissions on all API operations. We recommend that you us
e a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to
the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# Set yourEndpoint to the endpoint of the region in which the bucket is located. For exampl
e, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss
-cn-hangzhou.aliyuncs.com.
# Specify the name of the bucket. Example: examplebucket.
bucket = oss2.Bucket(auth, 'yourEndpoint', 'examplebucket')
# Specify the full path of the object. Example: exampledir/exampleobject.txt. The full path
of the object cannot contain the bucket name.
object_name = 'exampledir/exampleobject.txt'
# Specify the string to upload.
content = 'a novel'
# Configure the user metadata that must be prefixed with x-oss-meta-. For example, set the
value of the x-oss-meta-author metadata to 'O.Henry'.
bucket.put_object(object_name, content, headers={'x-oss-meta-author': 'O. Henry', 'Content-
Type': 'application/json; charset=utf-8'})
```

Modify object metadata

The following code provides an example on how to modify the metadata of the `exampleobject.txt` object in the `exampledir` directory stored in the `examplebucket` bucket:

```
# -*- coding: utf-8 -*-
import oss2
# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com.
# Specify the name of the bucket. Example: examplebucket.
bucket = oss2.Bucket(auth, 'yourEndpoint', 'examplebucket')
# Specify the full path of the object. Example: exampledir/exampleobject.txt. The full path of the object cannot contain THE bucket name.
object_name = 'exampledir/exampleobject.txt'
bucket.update_object_meta(object_name, {'x-oss-meta-author': 'O. Henry'})
# Each time you call bucket.update_object_meta, the existing user metadata is updated.
bucket.update_object_meta(object_name, {'x-oss-meta-price': '100 dollar'})
```

The following code provides an example on how to modify object metadata such as Content-Type:

```
# -*- coding: utf-8 -*-
import oss2
# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com.
# Specify the name of the bucket. Example: examplebucket.
bucket = oss2.Bucket(auth, 'yourEndpoint', 'examplebucket')
# Specify the full path of the object. Example: exampledir/exampleobject.txt. The full path of the object cannot contain the bucket name.
object_name = 'exampledir/exampleobject.txt'
bucket.update_object_meta(object_name, {'x-oss-meta-author': 'O. Henry'})
# Each time you call bucket.update_object_meta, the existing user metadata is updated.
bucket.update_object_meta(object_name, {'Content-Type': 'text/plain'})
```

Obtain object metadata

You can use the methods provided by OSS SDK for Python to query object metadata.

Method	Description	Advantage
get_object_meta	Queries part of object metadata such as the ETag, Size, and LastModified (the last modified time) values of the object. For more information, see GetObjectMeta .	More lightweight and faster

Method	Description	Advantage
head_object	Queries all object metadata. For more information, see HeadObject .	N/A

The following code provides an example on how to query the metadata of the `exampleobject.txt` object in the `exampledir` directory stored in the `examplebucket` bucket:

```
# -*- coding: utf-8 -*-
import oss2

# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com.
# Specify the name of the bucket. Example: examplebucket.
bucket = oss2.Bucket(auth, 'yourEndpoint', 'examplebucket')
# Specify the full path of the object. Example: exampledir/exampleobject.txt. The full path of the object cannot contain the bucket name.
object_name = 'exampledir/exampleobject.txt'
# Query part of the object metadata by using the get_object_meta method.
simplifiedmeta = bucket.get_object_meta(object_name)
print(simplifiedmeta.headers['Last-Modified'])
print(simplifiedmeta.headers['Content-Length'])
print(simplifiedmeta.headers['ETag'])
# Query all of the object metadata by using the head_object method.
objectmeta = bucket.head_object(object_name)
# In this example, the returned part of the object metadata is displayed. You can add the code to display all of the object metadata.
print(objectmeta.headers['Content-Type'])
print(objectmeta.headers['Last-Modified'])
print(objectmeta.headers['x-oss-object-type'])
```

5.8.5. Convert the storage classes of objects

OSS provides the following storage classes to cover various data storage scenarios from hot data to cold data: Standard, Infrequent Access (IA), Archive and Cold Archive. This topic describes how to convert the storage class of an object.

Examples

Convert the storage class of an object from Standard or Infrequent Access (IA) to Archive or Cold Archive

 Notice

- If an object is overwritten or deleted within less than 60 days after its storage class is converted to Archive, you are charged for the storage usage of the Archive object that is stored for less than the minimum storage duration.
- If an object is overwritten or deleted within less than 180 days after its storage class is converted to Cold Archive, you are charged for the storage usage of the Cold Archive object that is stored for less than the minimum storage duration.

For more information, see [Storage fees](#).

The following code provides an example on how to convert the storage class of an object from Standard or IA to Archive or Cold Archive:

```
# -*- coding: utf-8 -*-
import oss2
import os

# The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in Object Storage Service (OSS) is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('<yourAccessKeyId>', '<yourAccessKeySecret>')
# In this example, the endpoint of the China (Hangzhou) region is used. Specify your actual endpoint.
# Specify the bucket name. Example: examplebucket.
bucket = oss2.Bucket(auth, 'https://oss-cn-hangzhou.aliyuncs.com', 'examplebucket')
# Specify the full path of the object. The full path of the object cannot contain the bucket name. Example: exampledir/exampleobject.txt.
object_name = 'exampledir/exampleobject.txt'
# Make sure that the storage class of the object is Standard or IA.
# Convert the storage class of the object to Archive by adding a header that specifies the Archive storage class in the request.
headers = {'x-oss-storage-class': oss2.BUCKET_STORAGE_CLASS_ARCHIVE}
# Convert the storage class of the object to Cold Archive by adding a header that specifies the Cold Archive storage class in the request.
# headers = {'x-oss-storage-class': oss2.BUCKET_STORAGE_CLASS_COLDARCHIVE}
# Modify the storage class of the object.
bucket.copy_object(bucket.bucket_name, object_name, object_name, headers)
```

Convert the storage class of an object from Archive to Standard or IA

 Notice If an object is overwritten or deleted within less than 30 days after its storage class is converted to IA, you are charged for the storage usage of the IA object that is stored for less than the minimum storage duration. For more information, see [Storage fees](#).

The following code provides an example of how to convert the storage class of an object from Archive to Standard or IA:

```
# -*- coding: utf-8 -*-
import oss2
import os
import time

# The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('<yourAccessKeyId>', '<yourAccessKeySecret>')
# In this example, the endpoint of the China (Hangzhou) region is used. Specify your actual endpoint.
# Specify the bucket name. Example: examplebucket.
bucket = oss2.Bucket(auth, 'https://oss-cn-hangzhou.aliyuncs.com', 'examplebucket')
# Specify the full path of the object. The full path of the object cannot contain the bucket name. Example: exampledir/exampleobject.txt.
# Make sure that the storage class of the object is Archive.
object_name = 'exampledir/exampleobject.txt'
# Obtain the metadata of the object.
meta = bucket.head_object(object_name)
# Restore the object. The time required to restore an object varies with the size of the object.
if meta.resp.headers['x-oss-storage-class'] == oss2.BUCKET_STORAGE_CLASS_ARCHIVE:
    bucket.restore_object(object_name)
    while True:
        meta = bucket.head_object(object_name)
        if meta.resp.headers['x-oss-restore'] == 'ongoing-request="true"':
            time.sleep(5)
        else:
            break
# Convert the storage class of the object to Standard by adding a header that specifies the Standard storage class in the request.
headers = {'x-oss-storage-class': oss2.BUCKET_STORAGE_CLASS_STANDARD}
# Convert the storage class of the object to IA by adding a header that specifies the IA storage class in the request.
# headers = {'x-oss-storage-class': oss2.BUCKET_STORAGE_CLASS_IA}
# Modify the storage class of the object.
bucket.copy_object(bucket.bucket_name, object_name, object_name, headers)
```

Convert the storage class of an object from Cold Archive to Standard or IA

The following code provides an example on how to convert the storage class of an object from Cold Archive to Standard or IA:


```

# -*- coding: utf-8 -*-
import oss2
import os
import time

# The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('<yourAccessKeyId>', '<yourAccessKeySecret>')
# In this example, the endpoint of the China (Hangzhou) region is used. Specify your actual endpoint.
# Specify the bucket name. Example: examplebucket.
bucket = oss2.Bucket(auth, 'https://oss-cn-hangzhou.aliyuncs.com', 'examplebucket')
# Specify the full path of the object. The full path of the object cannot contain the bucket name. Example: exampledir/exampleobject.txt.
# Make sure that the storage class of the object is Cold Archive.
object_name = 'exampledir/exampleobject.txt'
# Obtain the metadata of the object.
meta = bucket.head_object(object_name)
# Restore the object.
if meta.resp.headers['x-oss-storage-class'] == oss2.BUCKET_STORAGE_CLASS_COLD_ARCHIVE:
    # Specify the restoration priority. RESTORE_TIER_EXPEDITED indicates that the object is restored within 1 hour.
    job_parameters = RestoreJobParameters(RESTORE_TIER_EXPEDITED)
    # Specify days. This parameter is used to specify the duration within which the restored object remains in the restored state.
    restore_config = oss2.models.RestoreConfiguration(days=5, job_parameters=job_parameters)
    # bucket.restore_object(object_name, input=restore_config)
    bucket.restore_object(object_name)
    while True:
        meta = bucket.head_object(object_name)
        if meta.resp.headers['x-oss-restore'] == 'ongoing-request="true"':
            time.sleep(5)
        else:
            break
# Convert the storage class of the object to Standard by adding a header that specifies the Standard storage class in the request.
headers = {'x-oss-storage-class': oss2.BUCKET_STORAGE_CLASS_STANDARD}
# Convert the storage class of the object to IA by adding a header that specifies the IA storage class in the request.
# headers = {'x-oss-storage-class': oss2.BUCKET_STORAGE_CLASS_IA}
# Modify the storage class of the object.
bucket.copy_object(bucket.bucket_name, object_name, object_name, headers)

```

References

For more information about the API operation that you can call to convert the storage class of an object, see [CopyObject](#).

5.8.6. List objects

This topic describes how to list all objects, objects whose names contain a specified prefix, and objects and subdirectories within a specified directory of a bucket.

Background information

You can call the GetBucket (ListObjects) or GetBucketV2 (ListObjectsV2) operation to list up to 1,000 objects in a bucket at a time. This operation provides several parameters to customize the output. For example, you can configure parameters to list objects from a specified start position, list objects and subdirectories in a specified directory, and list objects by page. The GetBucket (ListObjects) and GetBucketV2 (ListObjectsV2) operations are different in the following aspects:

- By default, when you call the GetBucket (ListObjects) operation to list objects, the information about the bucket owner is included in the response.
- When you call the GetBucketV2 (ListObjectsV2) operation to list objects, you can set the fetchOwner parameter to specify whether to include the information about the bucket owner in the response.

 **Note** If you want to list objects in a versioned bucket, we recommend that you call the GetBucketV2 (ListObjectsV2) operation.

The following tables describe the parameters that you can configure when you call the GetBucket (ListObjects) or GetBucketV2 (ListObjectsV2) operation to list objects.

- Call GetBucket (ListObjects) to list objects

The following table describes the parameters that you can configure when you call the GetBucket (ListObjects) operation to list objects.

Parameter	Description
prefix	The prefix that must be included in the names of objects to list.
delimiter	The character used to group objects by name.
marker	The position from which the list operation starts.

- Call GetBucketV2 (ListObjectsV2) to list objects

The following table describes the parameters that you can configure when you call the GetBucketV2 (ListObjectsV2) operation to list objects.

Parameter	Description
prefix	The prefix that must be included in the names of objects to list.
delimiter	The character used to group objects by name.
startAfter	The position from which the list operation starts.
fetchOwner	Specifies whether to include the information about the bucket owner in the response. ◦ true: The response includes the owner information. ◦ false: The response does not include the owner information.

Simple list

- Call GetBucket (ListObjects) to list objects

The following code provides an example on how to use GetBucket (ListObjects) to list 10 objects in a specified bucket:

```
# -*- coding: utf-8 -*-
import oss2
from itertools import islice
# The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to access OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# Specify the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com.
# Specify the name of the bucket. Example: examplebucket.
bucket = oss2.Bucket(auth, 'https://oss-cn-hangzhou.aliyuncs.com', 'examplebucket')
# List 10 objects in the bucket.
for b in islice(oss2.ObjectIterator(bucket), 10):
    print(b.key)
```

- Call GetBucketV2 (ListObjectsV2) to list objects

The following code provides an example on how to call GetBucketV2 (ListObjectsV2) to list 10 objects in a specified bucket:

```
# -*- coding: utf-8 -*-
import oss2
from itertools import islice
# The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to access OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# Specify the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com.
# Specify the name of the bucket. Example: examplebucket.
bucket = oss2.Bucket(auth, 'https://oss-cn-hangzhou.aliyuncs.com', 'examplebucket')
# List 10 objects in the bucket.
for obj in islice(oss2.ObjectIteratorV2(bucket), 10):
    print(obj.key)
```

List all objects in a bucket

- Call GetBucket (ListObjects) to list objects

The following code provides an example on how to call GetBucket (ListObjects) to list all objects in a specified bucket:

```
# -*- coding: utf-8 -*-
import oss2
# The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. U
sing these credentials to access OSS is a high-risk operation. We recommend that you use
a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to
the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# Specify the endpoint of the region in which the bucket is located. For example, if the
bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hang
zhou.aliyuncs.com.
# Specify the name of the bucket. Example: examplebucket.
bucket = oss2.Bucket(auth, 'https://oss-cn-hangzhou.aliyuncs.com', 'examplebucket')
# List all objects in the bucket.
for obj in oss2.ObjectIterator(bucket):
    print(obj.key)
```

- Call `GetBucketV2 (ListObjectsV2)` to list objects

The following code provides an example on how to call `GetBucketV2 (ListObjectsV2)` to list all objects in a specified bucket:

```
# -*- coding: utf-8 -*-
import oss2
# The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. U
sing these credentials to access OSS is a high-risk operation. We recommend that you use
a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to
the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# Specify the endpoint of the region in which the bucket is located. For example, if the
bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hang
zhou.aliyuncs.com.
# Specify the name of the bucket. Example: examplebucket.
bucket = oss2.Bucket(auth, 'https://oss-cn-hangzhou.aliyuncs.com', 'examplebucket')
# List all objects in the bucket.
for obj in oss2.ObjectIteratorV2(bucket):
    print(obj.key)
```

- Call `GetBucketV2 (ListObjectsV2)` to list objects in a bucket and include the information about the bucket owner in the response

The following code provides an example on how to call `GetBucketV2 (ListObjectsV2)` to list all objects in a specified bucket and how to specify the `fetchOwner` parameter to include the information about the bucket owner in the response:

```
# -*- coding: utf-8 -*-
import oss2
# The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. U
sing these credentials to access OSS is a high-risk operation. We recommend that you use
a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to
the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# Specify the endpoint of the region in which the bucket is located. For example, if the
bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hang
zhou.aliyuncs.com.
# Specify the name of the bucket. Example: examplebucket.
bucket = oss2.Bucket(auth, 'https://oss-cn-hangzhou.aliyuncs.com', 'examplebucket')
# List all objects in the specified bucket and specify the fetchOwner parameter to includ
e the information about the bucket owner in the response.
for obj in oss2.ObjectIteratorV2(bucket, fetch_owner=True):
    print(obj.key)
    print('file owner display name: ' + obj.owner.display_name)
    print('file owner id: ' + obj.owner.id)
```

List objects whose names contain a specified prefix

In the following examples, four objects are stored in the bucket: oss.jpg, fun/test.jpg, fun/movie/001.avi, and fun/movie/007.avi. A forward slash (/) is used as the directory delimiter.

- Call GetBucket (List Objects) to list objects

The following code provides an example on how to call GetBucket (List Objects) to list objects whose names contain a specified prefix:

```
# -*- coding: utf-8 -*-
import oss2
# The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. U
sing these credentials to access OSS is a high-risk operation. We recommend that you use
a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to
the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# Specify the endpoint of the region in which the bucket is located. For example, if the
bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hang
zhou.aliyuncs.com.
# Specify the name of the bucket. Example: examplebucket.
bucket = oss2.Bucket(auth, 'https://oss-cn-hangzhou.aliyuncs.com', 'examplebucket')
# List all objects in the fun directory and its subdirectories.
for obj in oss2.ObjectIterator(bucket, prefix='fun/'):
    print(obj.key)
```

- Call GetBucketV2 (ListObjectsV2) to list objects

The following code provides an example on how to call GetBucketV2 (ListObjectsV2) to list objects whose names contain a specified prefix:

```
# -*- coding: utf-8 -*-
import oss2
# The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. U
sing these credentials to access OSS is a high-risk operation. We recommend that you use
a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to
the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# Specify the endpoint of the region in which the bucket is located. For example, if the
bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hang
zhou.aliyuncs.com.
# Specify the name of the bucket. Example: examplebucket.
bucket = oss2.Bucket(auth, 'https://oss-cn-hangzhou.aliyuncs.com', 'examplebucket')
# List all objects in the fun directory and its subdirectories.
for obj in oss2.ObjectIteratorV2(bucket, prefix='fun/'):
    print(obj.key)
```

- Response

```
fun/movie/001.avi
fun/movie/007.avi
fun/test.jpg
```

List all objects from a specified start position

- Call GetBucket (ListObjects) to list objects

You can configure the marker parameter to specify the position from which the list operation starts. All objects whose names are alphabetically after the value of marker are returned. In the following example, the bucket stores four objects: x1.txt, x2.txt, z1.txt, and z2.txt.

The following code provides an example on how to call GetBucket (ListObjects) to list all objects from a specified start position:

```
# -*- coding: utf-8 -*-
import oss2
# The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. U
sing these credentials to access OSS is a high-risk operation. We recommend that you use
a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to
the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# Specify the endpoint of the region in which the bucket is located. For example, if the
bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hang
zhou.aliyuncs.com.
# Specify the name of the bucket. Example: examplebucket.
bucket = oss2.Bucket(auth, 'https://oss-cn-hangzhou.aliyuncs.com', 'examplebucket')
# List all objects whose names are alphabetically after the value of marker. The object w
hose name is the same as the value of marker is not returned.
for obj in oss2.ObjectIterator(bucket, marker="x2.txt"):
    print(obj.key)
```

- Call GetBucketV2 (ListObjectsV2) to list objects

You can configure the start_after parameter to specify the position from which the list operation starts. All objects whose names are alphabetically after the value of start_after are returned. In the following example, the bucket stores four objects: x1.txt, x2.txt, z1.txt, and z2.txt.

The following code provides an example on how to call `GetBucketV2` (`ListObjectsV2`) to list all objects from a specified start position:

```
# -*- coding: utf-8 -*-
import oss2
# The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to access OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# Specify the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com.
# Specify the name of the bucket. Example: examplebucket.
bucket = oss2.Bucket(auth, 'https://oss-cn-hangzhou.aliyuncs.com', 'examplebucket')
# List all objects whose names are alphabetically after the value of start_after. The object whose name is the same as the value of start_after is not returned.
for obj in oss2.ObjectIteratorV2(bucket, start_after="x2.txt"):
    print(obj.key)
```

List objects and subdirectories in a specified directory

OSS uses a flat structure for objects instead of a hierarchical structure to store objects. A directory is an object whose size is 0 and whose name ends with a forward slash (/). You can upload and download this object. By default, objects whose names end with a forward slash (/) are displayed as directories in the OSS console. For the complete sample code that is used to create directories, visit [GitHub](#).

You can specify the delimiter and prefix parameters to list objects by directory.

- If you set `prefix` to the name of a directory, objects and subdirectories whose names contain the prefix are listed.
- If you set `delimiter` to a forward slash (/), only the objects and subdirectories in the directory are listed. The objects and directories in subdirectories are not listed.

The following examples show how to call `GetBucket` (`ListObjects`) and `GetBucketV2` (`ListObjectsV2`) to list objects and subdirectories in a specified directory.

- Call `GetBucket` (`ListObjects`) to list objects

The following code provides an example on how to call `GetBucket` (`ListObjects`) to list objects and subdirectories in a specified directory:

```
# -*- coding: utf-8 -*-
import oss2
# The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. U
sing these credentials to access OSS is a high-risk operation. We recommend that you use
a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to
the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# Specify the endpoint of the region in which the bucket is located. For example, if the
bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hang
zhou.aliyuncs.com.
# Specify the name of the bucket. Example: examplebucket.
bucket = oss2.Bucket(auth, 'https://oss-cn-hangzhou.aliyuncs.com', 'examplebucket')
# List objects and subdirectories in the fun directory. Objects in the subdirectories are
not listed.
for obj in oss2.ObjectIterator(bucket, prefix = 'fun/', delimiter = '/', start_after='fun
/'):
    # Use is_prefix to determine whether obj is a directory.
    if obj.is_prefix(): # Specify the operation to perform if obj is determined as a dir
ectory.
        print('directory: ' + obj.key)
    else: # Specify the operation to perform if obj is determined as an ob
ject.
        print('file: ' + obj.key)
```

- Call GetBucketV2 (ListObjectsV2) to list objects

The following code provides an example on how to call GetBucketV2 (ListObjectsV2) to list objects and subdirectories in a specified directory:


```
# -*- coding: utf-8 -*-
import oss2
# The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. U
sing these credentials to access OSS is a high-risk operation. We recommend that you use
a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to
the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# Specify the endpoint of the region in which the bucket is located. For example, if the
bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hang
zhou.aliyuncs.com.
# Specify the name of the bucket. Example: examplebucket.
bucket = oss2.Bucket(auth, 'https://oss-cn-hangzhou.aliyuncs.com', 'examplebucket')
# List objects and subdirectories in the fun directory. Objects in the subdirectories are
not listed. If you do not need the bucket owner information, you can leave the fetchOwner
parameter unspecified.
for obj in oss2.ObjectIteratorV2(bucket, prefix = 'fun/', delimiter = '/', start_after='f
un/', fetch_owner=True):
    # Use is_prefix to determine whether obj is a directory.
    if obj.is_prefix(): # Specify the operation to perform if obj is determined as a dir
ectory.
        print('directory: ' + obj.key)
    else: # Specify the operation to perform if obj is determined as an ob
ject.
        print('file: ' + obj.key)
        print('file owner display name: ' + obj.owner.display_name)
        print('file owner id: ' + obj.owner.id)
```

- Response

```
directory: fun/movie/
file: fun/test.jpg
```

List the sizes of objects in a specified directory

- Call GetBucket (ListObjects) to list the sizes of objects

The following code provides an example on how to call GetBucket (ListObjects) to list the sizes of objects in a specified directory:

```
# -*- coding: utf-8 -*-
import oss2
def CalculateFolderLength(bucket, folder):
    length = 0
    for obj in oss2.ObjectIterator(bucket, prefix=folder):
        length += obj.size
    return length
# The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to access OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# Specify the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com.
# Specify the name of the bucket. Example: examplebucket.
bucket = oss2.Bucket(auth, 'https://oss-cn-hangzhou.aliyuncs.com', 'examplebucket')
for obj in oss2.ObjectIterator(bucket, delimiter='/'):
    if obj.is_prefix(): # Specify the operation to perform if obj is determined as a directory.
        length = CalculateFolderLength(bucket, obj.key)
        print('directory: ' + obj.key + ' length:' + str(length) + "Byte.")
    else: # Specify the operation to perform if obj is determined as an object.
        print('file:' + obj.key + ' length:' + str(obj.size) + "Byte.")
```

- Call `GetBucketV2 (ListObjectsv2)` to list the sizes of objects

The following code provides an example on how to call `GetBucketV2 (ListObjectsv2)` to list the sizes of objects within a specified directory:

```
# -*- coding: utf-8 -*-
import oss2
def CalculateFolderLength(bucket, folder):
    length = 0
    for obj in oss2.ObjectIteratorV2(bucket, prefix=folder):
        length += obj.size
    return length
# The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to access OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# Specify the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com.
# Specify the name of the bucket. Example: examplebucket.
bucket = oss2.Bucket(auth, 'https://oss-cn-hangzhou.aliyuncs.com', 'examplebucket')
for obj in oss2.ObjectIteratorV2(bucket, delimiter='/'):
    if obj.is_prefix(): # Specify the operation to perform if obj is determined as a directory.
        length = CalculateFolderLength(bucket, obj.key)
        print('directory: ' + obj.key + ' length:' + str(length) + "Byte.")
    else: # Specify the operation to perform if obj is determined as an object.
        print('file:' + obj.key + ' length:' + str(obj.size) + "Byte.")
```

5.8.7. Query objects

This topic describes how to use SelectObject in Object Storage Service (OSS) SDK for Python to query CSV files and JSON objects.

Example

The following code provides an example on how to query CSV files and JSON objects:

```
import oss2

def select_call_back(consumed_bytes, total_bytes = None):
    print('Consumed Bytes:' + str(consumed_bytes) + '\n')

# The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
access_key_id = 'yourAccessKeyId'
access_key_secret = 'yourAccessKeySecret'
# Specify the bucket name. Example: examplebucket.
bucket_name = 'yourBucketName'
# Specify the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com.
endpoint = 'https://oss-cn-hangzhou.aliyuncs.com'
# Create the bucket. All object-related methods are called based on the bucket.
bucket = oss2.Bucket(oss2.Auth(access_key_id, access_key_secret), endpoint, bucket_name)
key = 'python_select.csv'
content = 'Tom Hanks,USA,45\r\n'*1024
filename = 'python_select.csv'
# Upload a CSV file.
bucket.put_object(key, content)
# Configure the parameters of the SelectObject API operation.
csv_meta_params = {'RecordDelimiter': '\r\n'}
select_csv_params = {'CsvHeaderInfo': 'None',
                    'RecordDelimiter': '\r\n',
                    'LineRange': (500, 1000)}
csv_header = bucket.create_select_object_meta(key, csv_meta_params)
print(csv_header.rows)
print(csv_header.splits)
result = bucket.select_object(key, "select * from ossobject where _3 > 44", select_call_back, select_csv_params)
select_content = result.read()
print(select_content)
result = bucket.select_object_to_file(key, filename,
                                     "select * from ossobject where _3 > 44", select_call_back, select_csv_params)
bucket.delete_object(key)
###JSON DOCUMENT
key = 'python_select.json'
content = '{"contacts":[{"key1":1,"key2":"hello world1"}, {"key1":2,"key2":"hello world2"}]}'
filename = 'python_select.json'
# Upload a JSON DOCUMENT object.
bucket.put_object(key, content)
select_json_params = {'Json Type': 'DOCUMENT'}
```

```

result = bucket.select_object(key, "select s.key2 from ossobject.contacts[*] s where s.key1
= 1", None, select_json_params)
select_content = result.read()
print(select_content)
result = bucket.select_object_to_file(key, filename,
    "select s.key2 from ossobject.contacts[*] s where s.key1 = 1", None, select_json_params)
bucket.delete_object(key)
###JSON LINES
key = 'python_select_lines.json'
content = '{"key1":1,"key2":"hello world1"}\n{"key1":2,"key2":"hello world2"}'
filename = 'python_select.json'
# Upload a JSON LINES object.
bucket.put_object(key, content)
select_json_params = {'Json_Type': 'LINES'}
json_header = bucket.create_select_object_meta(key, select_json_params)
print(json_header.rows)
print(json_header.splits)
result = bucket.select_object(key, "select s.key2 from ossobject s where s.key1 = 1", None,
select_json_params)
select_content = result.read()
print(select_content)
result = bucket.select_object_to_file(key, filename,
    "select s.key2 from ossobject s where s.key1 = 1", None, select_json_params)
bucket.delete_object(key)

```

Python SelectObject

The following section describes the elements used for the SelectObject API operation in Python in detail. These elements include `select_object`, `select_object_to_file`, and `create_select_object_meta`.

- `select_object`
 - The following code provides an example on how to specify parameters for `select_object`:

```

def select_object(self, key, sql,
                  progress_callback=None,
                  select_params=None
                  byte_range=None
                  headers=None
                  ):

```

The preceding code example is used to execute an SQL statement on the object that has the specified key and return the query results.

- `sql`: the SQL statement that can be directly used as the value of the `sql` parameter and does not need to be Base64-encoded.
- `Progress_callback`: optional. This parameter specifies a callback function used to report the query progress.
- `select_params`: the parameters and actions of the SelectObject operation.
- `headers`: the header information included in the request. The header information functions the same as that for the GetObject operation. For example, you can configure the `bytes` field in the request header to specify the range of a CSV file to query.

- The following table describes the parameters supported by the `select_params` parameter.

Parameter	Description
Json_Type	■ By default, the object is a CSV file if this parameter is not specified. ■ The object is a JSON Document object if this parameter is set to <code>DOCUMENT</code>. ■ The object is a JSON LINES object if this parameter is set to <code>LINES</code>.
CsvHeaderInfo	The header information of the CSV file. Valid values: <i>None</i>, <i>Ignore</i>, and <i>Use</i>. ■ <i>None</i>: This object has no header information configured. ■ <i>Ignore</i>: This object has header information configured, which is not used when the SQL statement is executed. ■ <i>Use</i>: This object has header information configured. The column names in the header information are used when the SQL statement is executed.
CommentCharacter	The comment character in the CSV file. This parameter value can be only one character in length. The default value is <code>None</code> , which indicates that no comment characters are allowed.
RecordDelimiter	The row delimiter in the CSV file. This parameter value can be only one or two characters in length. Default value: <code>\n</code>
OutputRecordDelimiter	The row delimiter in the output result of the <code>SELECT</code> statement. Default value: <code>\n</code>
FieldDelimiter	The column delimiter in the CSV file. This parameter value can be only one character in length. Default value: comma (`,`).
OutputFieldDelimiter	The column delimiter in the output result of the <code>SELECT</code> statement. Default value: comma (`,`).
QuoteCharacter	The quote character for the columns in the CSV file. This parameter value can be only one character in length. Default value: double quotation marks (`"`). Row and column delimiters enclosed in quote characters are processed as normal characters.
SplitRange	The split range in multipart query. The parameter value is a closed interval in the (start, end) format, which indicates the range of splits to query.
LineRange	The row range in multipart query. The parameter value is a closed interval in the (start, end) format, which indicates the range of rows to query.
CompressionType	The format in which the object is compressed. Valid value: <code>GZIP</code> . Default value: <code>None</code> .

Parameter	Description
KeepAllColumns	If this parameter is set to true, columns that are excluded by the SELECT statement in the CSV file are left empty in the output result. However, the column positions are kept. Default value: False. Example: The columns in the CSV file are firstname, lastname, and age. The SQL statement is <code>select firstname, age from ossobject</code>. ■ If KeepAllColumns is set to <i>true</i>, the output result is <code>firstname,,age</code>, in which a comma (,) is added to indicate the position of the excluded lastname column. ■ If KeepAllColumns is set to <i>false</i>, the output result is <code>firstname,age</code>.  Note By using this parameter, you can directly use the code that is used to process GetObject to process SelectObject without modifications.
OutputRawData	■ If this parameter is set to <i>True</i>, SelectObject directly returns the original data. If data is not returned for a long time, a timeout error may occur. ■ If this parameter is set to <i>False</i>, the output data is encapsulated in frames. Default value: False.
EnablePayloadCrc	Specifies whether a cyclic redundancy check (CRC) value is calculated for each frame. Default value: False.
OutputHeader	The header information in the first line of the output result. This parameter applies only to CSV files.
SkipPartialDataRecord	If this parameter is set to True for a CSV file, the current record is skipped when a column of this record has no data. If this parameter is set to True for a JSON object, the current record is skipped when a key of this record does not exist. If this parameter is set to False, a column without data is left empty in the output result. For example, a row includes the firstname, lastname, and age columns. The SQL statement is <code>select _1, _4 from ossobject</code>. ■ If this parameter is set to True, this row is skipped. ■ If this parameter is set to False, <code>firstname,\n</code> is returned.
MaxSkippedRecordsAllowed	The maximum number of skipped rows. The default value is 0, which indicates that an error is returned if a row is skipped.

Parameter	Description
ParseJsonNumberAsString	■ If this parameter is set to True, all numbers in the JSON object are parsed as strings. ■ If this parameter is set to False, all numbers in the JSON object are parsed as integers or floating-point numbers. Default value: False. High-accuracy floating-point numbers in a JSON object suffer loss of accuracy when they are parsed as floating-point numbers. To keep the accuracy, you can set this parameter to True and use the CAST function to convert the parsed data to the decimal type.

- Returned result of `select_object`: A `SelectObjectResult` object is returned. You can use the `read()` function or the `__iter__` method to obtain all results.

 **Note** If you call the `read()` function to read all results when multiple results are returned, excess memory resources are used and you have to wait for a long time. We recommend that you use the `__iter__` method (foreach chunk in result) to obtain results and process each chunk in the results. The `__iter__` method reduces memory usage and enables the client to process each chunk request processed by the OSS server in a timely manner. This way, the client does not need to wait until all results are returned.

- `select_object_to_file`

```
def select_object_to_file(self, key, filename, sql,
                          progress_callback=None,
                          select_params=None,
                          headers=None
                          ):
```

The preceding code example is used to execute an SQL statement on the object that has the specified key and write the query results to another specified object.

Other parameters are the same as those of `select_object`.

- `create_select_object_meta`

- Syntax of `create_select_object_meta`

```
def create_select_object_meta(self, key, select_meta_params=None, header=None):
```

The preceding code example is used to create Select Meta for the object that has the specified key or obtain Select Meta from such an object. Select Meta includes the total number of rows, total number of columns (for CSV files), and total number of splits in an object.

If Select Meta has been created for an object, this function does not re-create Select Meta unless the `OverwriteIfExists` parameter is set to true.

To create Select Meta for an object, the object must be completely scanned.

- The following table describes the parameters supported by the `select_meta_params` parameter.

Parameter	Description
Json_Type	■ By default, the object is a CSV file if this parameter is not specified. ■ If this parameter is specified, the parameter value must be <code>LINES</code>, which indicates that the object is a JSON LINES object.  Note This operation does not apply to JSON Document objects.
RecordDelimiter	The row delimiter within the CSV file.
FieldDelimiter	The column delimiter within the CSV file.
QuoteCharacter	The quote character within the CSV file. Row and column delimiters enclosed in quote characters are processed as normal characters.
CompressionType	The format in which the object is compressed. Currently, no compression formats are supported. The default value is <code>None</code> .
OverwriteIfExists	Specifies whether the created Select Meta overwrites the original Select Meta. You do not need to set this parameter in most cases.

- Returned results of `create_select_object_meta`: A `GetSelectObjectMetaResult` object is returned, which includes the `rows` and `splits` attributes. For a CSV file, the `select_resp` object in the result includes the `columns` attribute, which indicates the number of columns in the CSV file.

5.8.8. Delete objects

You can delete a single object, multiple specified objects, objects whose names contain a specified prefix, or a specified directory and all objects in the directory.

 **Warning** Deleted objects cannot be recovered. Exercise caution when you delete objects.

Delete a single object

The following code provides an example on how to delete an object named `exampleobject.txt` from a bucket named `examplebucket`:


```
# -*- coding: utf-8 -*-
import oss2
# The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in Object Storage Service (OSS) is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# Specify the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com.
# Specify the bucket name. Example: examplebucket.
bucket = oss2.Bucket(auth, 'https://oss-cn-hangzhou.aliyuncs.com', 'examplebucket')
# Delete the object.
# Set yourObjectName to the full path of the object that you want to delete. The full path cannot contain the bucket name. Example: exampledir/exampleobject.txt.
# To delete a directory, set yourObjectName to the directory name. If the directory contains objects, you must delete all objects from the directory before you can delete the directory.
bucket.delete_object('exampledir/exampleobject.txt')
```

Delete multiple objects

You can delete up to 1,000 objects at a time. You can delete multiple specified objects, objects whose names contain a specified prefix, or a specified directory and all objects in the directory.

You can also configure lifecycle rules to automatically delete objects. For more information, see [Lifecycle rules based on the last modified time](#).

Delete multiple specified objects

The following code provides an example on how to delete multiple specified objects:

```
# -*- coding: utf-8 -*-
import oss2
# The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# Specify the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com.
# Specify the bucket name. Example: examplebucket.
bucket = oss2.Bucket(auth, 'https://oss-cn-hangzhou.aliyuncs.com', 'examplebucket')
# Delete three objects at a time. You can delete up to 1,000 objects at a time.
# Specify the full paths of the three objects that you want to delete. The full paths cannot contain the bucket name.
result = bucket.batch_delete_objects(['exampleobject1.jpg', 'testobject2.png', 'destobject3.txt'])
# Display the names of the deleted objects.
print('\n'.join(result.deleted_keys))
```

Delete multiple objects with a specified prefix or in a specified directory

The following code provides an example on how to delete multiple objects with a specified prefix and how to delete a specified directory and all objects in the directory.

 **Warning** If the prefix is not specified or is set to NULL in the following code, all objects in the bucket are deleted. Exercise caution when you configure the prefix.

```
# -*- coding: utf-8 -*-
import oss2

# The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# Specify the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com.
# Specify the bucket name. Example: examplebucket.
bucket = oss2.Bucket(auth, 'https://oss-cn-hangzhou.aliyuncs.com', 'examplebucket')
# Specify the prefix of the objects that you want to delete. If you want to delete all objects whose names contain the src prefix, set the prefix to src. After you set the prefix to src, all non-directory objects whose names contain the src prefix, the src directory, and all objects in the directory are deleted.
prefix = 'src'
# If you want to delete only the src directory and all objects in the directory, set the prefix to src/.
# prefix = 'src/'
# Delete multiple objects.
for obj in oss2.ObjectIterator(bucket, prefix=prefix):
    bucket.delete_object(obj.key)
```

References

- For the complete sample code that is used to delete a single object or multiple objects, visit [GitHub](#).
- For more information about the API operation that you can call to delete a single object, see [DeleteObject](#).
- For more information about the API operation that you can call to delete multiple objects, see [DeleteMultipleObjects](#).

5.8.9. Copy objects

Copy a small object

You can use simple copy to copy an object whose size is smaller than 1 GB. The following code provides an example on how to use simple copy to copy an object named `srcexampleobject.txt` from the `srcexamplebucket` bucket to an object named `destexampleobject.txt` in the `destexamplebucket` bucket:

```
# -*- coding: utf-8 -*-
```

```

import oss2

# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to acc
ess Object Storage Service (OSS) because the account has permissions on all API operations.
We recommend that you use a RAM user to call API operations or perform routine operations a
nd maintenance. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# Specify the name of the source bucket. Example: srcexamplebucket.
src_bucket_name = 'srcexamplebucket'
# Specify the name of the destination bucket, which is located in the same region as the so
urce bucket. Example: destexamplebucket.
# If you copy an object within a bucket, make sure that you specify the same bucket name fo
r the source and destination buckets.
dest_bucket_name = 'destexamplebucket'
# Set yourEndpoint to the endpoint of the region in which the bucket is located. For exampl
e, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss
-cn-hangzhou.aliyuncs.com.
bucket = oss2.Bucket(auth, 'yourEndpoint', dest_bucket_name)
# Specify the full path of the source object. The full path of the object cannot contain th
e bucket name. Example: srcexampleobject.txt.
src_object_name = 'srcexampleobject.txt'
# Specify the full path of the destination object. The full path of the object cannot conta
in the bucket name. Example: destexampleobject.txt.
dest_object_name = 'destexampleobject.txt'
# headers = dict()
# Specify whether the CopyObject operation overwrites the object with the same name. In thi
s example, this parameter is set to true, which indicates that the object with the same nam
e cannot be overwritten.
# headers['x-oss-forbid-overwrite'] = 'true'
# Specify the path of the source object.
# headers[OSS_COPY_OBJECT_SOURCE] = '/example-bucket-by-util/recode-test.txt'
# If the ETag value of the source object is the same as the ETag value that is specified in
the request, OSS copies the object and returns 200 OK.
# headers['x-oss-copy-source-if-match'] = '5B3C1A2E053D763E1B002CC607C5****'
# If the ETag value of the source object is different from the ETag value that is specified
in the request, OSS copies the object and returns 200 OK.
# headers['x-oss-copy-source-if-none-match'] = '5B3C1A2E053D763E1B002CC607C5****'
# If the time that is specified in the request is equal to or later than the actual modifie
d time of the object, OSS copies the object and returns 200 OK.
# headers['x-oss-copy-source-if-unmodified-since'] = '2021-12-09T07:01:56.000Z'
# If the source object is modified after the time that is specified in the request, OSS cop
ies the object.
# headers['x-oss-copy-source-if-modified-since'] = '2021-12-09T07:01:56.000Z'
# Specify the method that is used to configure the metadata of the destination object. If y
ou set the method to COPY, the metadata of the source object is copied to the destination o
bject.
# headers[OSS_METADATA_DIRECTIVE] = 'COPY'
# Specify the server-side encryption algorithm that is used to encrypt the destination obje
ct when OSS creates the destination object.
# headers[OSS_SERVER_SIDE_ENCRYPTION] = 'KMS'
# Specify the customer master key (CMK) that is managed by Key Management Service (KMS). Th
is parameter takes effect only when x-oss-server-side-encryption is set to KMS.
# headers['x-oss-server-side-encryption-key-id'] = '9468da86-3509-4f8d-a61e-6eableac****'
# Specify the access control list (ACL) of the destination object. In this example, this pa
rameter is set to OBJECT_ACL_PRIVATE, which indicates that only the object owner and author
ized users have read and write permissions on the object.

```

```

ized users have read and write permissions on the object.
# headers[OSS_OBJECT_ACL] = oss2.OBJECT_ACL_PRIVATE
# Specify the storage class of the destination object. In this example, this parameter is s
et to BUCKET_STORAGE_CLASS_STANDARD, which indicates Standard.
# headers['x-oss-storage-class'] = oss2.BUCKET_STORAGE_CLASS_STANDARD
# Specify the tag for the destination object. You can specify multiple tags for the destina
tion object at the same time.
# headers[OSS_OBJECT_TAGGING] = 'k1=v1&k2=v2&k3=v3'
# Specify the method that is used to configure the tag of the destination object. If you se
t the method to COPY, the tags of the source object are copied to the destination object.
# headers[OSS_OBJECT_TAGGING_COPY_DIRECTIVE] = 'COPY'
# result = bucket.copy_object(src_bucket_name, src_object_name, dest_object_name, headers=h
eaders)
# Copy the object from the source bucket to the destination bucket.
result = bucket.copy_object(src_bucket_name, src_object_name, dest_object_name)
# View the HTTP status code in the returned results. If the HTTP status code 200 is returne
d, the execution succeeds.
print('result.status:', result.status)

```

Copy a large object

1. Use `bucket.init_multipart_upload` to initiate a multipart copy task.
2. Use `bucket.upload_part_copy` to copy the parts. All parts must be larger than 100 KB, except for the last part.
3. Use `bucket.complete_multipart_upload` to complete the multipart copy task.

The following code provides an example on how to use multipart copy to copy an object named `srcexampleobject.txt` from the `srcexamplebucket` bucket to an object named `destexampleobject.txt` in the `destexamplebucket` bucket:

```

# -*- coding: utf-8 -*-
import oss2
from oss2.models import PartInfo
from oss2 import determine_part_size
# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to acc
ess OSS because the account has permissions on all API operations. We recommend that you us
e a RAM user to call API operations or perform routine operations and maintenance. To creat
e a RAM user, log on to the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# Specify the name of the source bucket. Example: srcexamplebucket.
src_bucket_name = 'srcexamplebucket'
# Specify the name of the destination bucket, which is located in the same region as the so
urce bucket. Example: destexamplebucket.
# If you copy an object within a bucket, make sure that you specify the same bucket name fo
r the source and destination buckets.
dest_bucket_name = 'destexamplebucket'
# Set yourEndpoint to the endpoint of the region in which the bucket is located. For exampl
e, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss
-cn-hangzhou.aliyuncs.com.
bucket = oss2.Bucket(auth, 'yourEndpoint', dest_bucket_name)
# If you copy an object within a bucket, comment out this line of code and change src_bucke
t in src_bucket_name to bucket.
src_bucket = oss2.Bucket(auth, 'https://oss-cn-hangzhou.aliyuncs.com', src_bucket_name)
# Specify the full path of the source object. The full path of the object cannot contain th

```

```

e bucket name. Example: srcexampleobject.txt.
src_object_name = 'srcexampleobject.txt'
# Specify the full path of the destination object. The full path of the object cannot contain
# the bucket name. Example: destexampleobject.txt.
dest_object_name = 'destexampleobject.txt'
# Obtain the size of the source object. If you copy an object within a bucket, change src_bucket
# to bucket.
head_info = src_bucket.head_object(src_object_name)
total_size = head_info.content_length
print('src object size:', total_size)
# Specify the determine_part_size method to determine the part size.
part_size = determine_part_size(total_size, preferred_size=100 * 1024)
print('part_size:', part_size)
# Initiate a multipart copy task.
upload_id = bucket.init_multipart_upload(dest_object_name).upload_id
parts = []
# Copy the parts one by one.
part_number = 1
offset = 0
while offset < total_size:
    num_to_upload = min(part_size, total_size - offset)
    end = offset + num_to_upload - 1
    # headers = dict()
    # Specify the path of the source object.
    # headers[OSS_COPY_OBJECT_SOURCE] = '/example-bucket-by-util/recode-test.txt'
    # Specify the range of data that you want to copy. For example, if you set bytes to 0~1023,
    # the first 1024 bytes of the source object are copied.
    # headers[OSS_COPY_OBJECT_SOURCE_RANGE] = 'bytes=0~1023'
    # If the ETag value of the source object is the same as the ETag value that is specified
    # in the request, OSS copies the object and returns 200 OK.
    # headers['x-oss-copy-source-if-match'] = '5B3C1A2E053D763E1B002CC6****'
    # If the ETag value of the source object is different from the ETag value that is specified
    # in the request, OSS copies the object and returns 200 OK.
    # headers['x-oss-copy-source-if-none-match'] = '5B3C1A2E053D763E1B002CC6****'
    # If the time that is specified in the request is equal to or later than the actual modified
    # time of the object, OSS copies the object and returns 200 OK.
    # headers['x-oss-copy-source-if-unmodified-since'] = '2021-12-09T07:01:56.000Z'
    # If the time that is specified in the request is earlier than the actual modified time
    # of the object, OSS copies the object and returns 200 OK.
    # headers['x-oss-copy-source-if-modified-since'] = '2021-12-09T07:01:56.000Z'
    # result = bucket.upload_part_copy(src_bucket_name, src_object_name, (offset, end), dest_
    # object_name, upload_id, part_number, headers=headers)
    result = bucket.upload_part_copy(src_bucket_name, src_object_name, (offset, end), dest_
    object_name, upload_id, part_number)
    # Save the part information.
    parts.append(PartInfo(part_number, result.etag))
    offset += num_to_upload
    part_number += 1
# Complete the multipart copy task.
result = bucket.complete_multipart_upload(dest_object_name, upload_id, parts)
# View the HTTP status code in the returned response.
print('result :', result.status)
# Query the metadata of the destination object.
head_info = bucket.head_object(dest_object_name)

```

```
# Query the size of the destination object.
dest_object_size = head_info.content_length
print('dest object size:', dest_object_size)
# Compare the sizes between the source and destination objects.
assert dest_object_size == total_size
```

References

- Copy a small object
 - For more information about the complete sample code that is used to copy a small object, visit [GitHub](#).
 - For more information about the API operation that you can call to copy a small object, see [CopyObject](#).
- Copy a large object

For more information about the API operation that you can call to copy a large object, see [UploadPartCopy](#).

5.8.10. Disable overwriting object with the same name

By default, if you upload an object that has the same name as an existing object, the existing object is overwritten by the uploaded object. This topic describes how to configure the `x-oss-forbid-overwrite` request header to prevent objects from being overwritten by objects with the same names when you copy objects or perform simple upload or multipart upload.

Simple upload

The following code provides an example on how to prevent objects from being overwritten by objects that have the same names in simple upload:

```
# -*- coding: utf-8 -*-
import oss2

# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log
on to OSS, because the account has permissions on all API operations. We recommend that you
use a RAM user to call API operations or perform routine operations and maintenance. To cre
ate your RAM user, log on to https://ram.console.aliyun.com.
auth = oss2.Auth('<yourAccessKeyId>', '<yourAccessKeySecret>')
# This example uses the endpoint of the China (Hangzhou) region. Specify the actual endpoin
t based on your requirements.
bucket = oss2.Bucket(auth, 'http://oss-cn-hangzhou.aliyuncs.com', '<yourBucketName>')
# Upload the object.
# Specify whether the PutObject operation overwrites the object with the same name.
# By default, if x-oss-forbid-overwrite is not specified, the object with the same name is
overwritten.
# If x-oss-forbid-overwrite is set to false, the object with the same name is overwritten.
# If x-oss-forbid-overwrite is set to true, the object with the same name is not overwritte
n. If the object with the same name already exists, an error is returned.
headers["x-oss-forbid-overwrite"] = "true"
result = bucket.put_object('<yourObjectName>', 'content of object', headers=headers)
# Obtain the HTTP status code.
print('http status: {0}'.format(result.status))
# Obtain the unique request ID. We recommend that you add this parameter in the program log
s.
print('request_id: {0}'.format(result.request_id))
# Obtain the ETag value returned by the put_object method.
print('ETag: {0}'.format(result.etag))
# Obtain the HTTP response headers.
print('date: {0}'.format(result.headers['date']))
```

Copy objects

- Copy a small object

The following code provides an example on how to copy a small object without overwriting the object with the same object name:

```
# -*- coding: utf-8 -*-
import oss2
# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log on to OSS, because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine operations and maintenance. To create your RAM user, log on to https://ram.console.aliyun.com.
auth = oss2.Auth('<yourAccessKeyId>', '<yourAccessKeySecret>')
# This example uses the endpoint of the China (Hangzhou) region. Specify the actual endpoint based on your requirements.
bucket = oss2.Bucket(auth, 'http://oss-cn-hangzhou.aliyuncs.com', '<yourRequestBucketName>')
# Specify whether the copy_object operation overwrites the object with the same object name.
# By default, if x-oss-forbid-overwrite is not specified, the object with the same name is overwritten.
# If x-oss-forbid-overwrite is set to false, the object with the same name is overwritten.
# If x-oss-forbid-overwrite is set to true, the object with the same name is not overwritten. If the object with the same name already exists, an error is returned.
headers = dict()
headers["x-oss-forbid-overwrite"] = "true"
bucket.copy_object('<yourSourceBucketName>', '<yourSourceObjectName>', '<yourDestinationObjectName>', headers=headers)
```

- Copy a large object

The following code provides an example on how to prevent an existing object from being overwritten by a large object with the same name when you copy the large object by using multipart copy:


```

# -*- coding: utf-8 -*-
import oss2
from oss2.models import PartInfo
from oss2 import determine_part_size
# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to l
og on to OSS, because the account has permissions on all API operations. We recommend tha
t you use a RAM user to call API operations or perform routine operations and maintenance
. To create your RAM user, log on to https://ram.console.aliyun.com.
auth = oss2.Auth('<yourAccessKeyId>', '<yourAccessKeySecret>')
# This example uses the endpoint of the China (Hangzhou) region. Specify the actual endpo
int based on your requirements.
bucket = oss2.Bucket(auth, 'http://oss-cn-hangzhou.aliyuncs.com', '<yourBucketName>')
src_object = '<yourSourceObjectName>'
dst_object = '<yourDestinationObjectName>'
total_size = bucket.head_object(src_key).content_length
part_size = determine_part_size(total_size, preferred_size=100 * 1024)
# Initiate a multipart copy task.
# Specify whether to overwrite an object with the same name when copying an object.
# By default, if x-oss-forbid-overwrite is not specified, the object with the same name i
s overwritten.
# If x-oss-forbid-overwrite is set to false, the object with the same name is overwritten
.
# If x-oss-forbid-overwrite is set to true, the object with the same name is not overwrit
ten. If the object with the same name already exists, an error is returned.
headers = dict()
headers["x-oss-forbid-overwrite"] = "true"
upload_id = bucket.init_multipart_upload(dst_object, headers=headers).upload_id
parts = []
# Copy each part sequentially.
part_number = 1
offset = 0
while offset < total_size:
    num_to_upload = min(part_size, total_size - offset)
    byte_range = (offset, offset + num_to_upload - 1)
    result = bucket.upload_part_copy(bucket.bucket_name, src_object, byte_range,dst_objec
t, upload_id, part_number)
    parts.append(PartInfo(part_number, result.etag))
    offset += num_to_upload
    part_number += 1
# Complete multipart copy.
# Specify whether to overwrite an object with the same name when copying an object.
# By default, if x-oss-forbid-overwrite is not specified, the object with the same name i
s overwritten.
# If x-oss-forbid-overwrite is set to false, the object with the same name is overwritten
.
# If x-oss-forbid-overwrite is set to true, the object with the same name is not overwrit
ten. If the object with the same name already exists, an error is returned.
headers = dict()
headers["x-oss-forbid-overwrite"] = "true"
bucket.complete_multipart_upload(dst_object, upload_id, parts, headers=headers)

```

Multipart upload

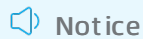
The following code provides an example on how to prevent objects from being overwritten by newly uploaded objects that have the same names in multipart upload:

```
# -*- coding: utf-8 -*-
import os
from oss2 import SizedFileAdapter, determine_part_size
from oss2.models import PartInfo
import oss2

# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log
on to OSS, because the account has permissions on all API operations. We recommend that you
use a RAM user to call API operations or perform routine operations and maintenance. To cre
ate your RAM user, log on to https://ram.console.aliyun.com.
auth = oss2.Auth('<yourAccessKeyId>', '<yourAccessKeySecret>')
# This example uses the endpoint of the China (Hangzhou) region. Specify the actual endpoin
t based on your requirements.
bucket = oss2.Bucket(auth, 'http://oss-cn-hangzhou.aliyuncs.com', '<yourBucketName>')
key = '<yourObjectName>'
filename = '<yourLocalFile>'
total_size = os.path.getsize(filename)
# Use the determine_part_size method to determine the size of each part.
part_size = determine_part_size(total_size, preferred_size=100 * 1024)
# Initiate a multipart upload task.
# Specify whether to overwrite the object with the same name when performing multipart uplo
ad.
# By default, if x-oss-forbid-overwrite is not specified, the object with the same name is
overwritten.
# If x-oss-forbid-overwrite is set to false, the object with the same name is overwritten.
# If x-oss-forbid-overwrite is set to true, the object with the same name is not overwritte
n. If the object with the same name already exists, an error is returned.
headers["x-oss-forbid-overwrite"] = "true"
upload_id = bucket.init_multipart_upload(key, headers=headers).upload_id
parts = []
# Upload each part sequentially.
with open(filename, 'rb') as fileobj:
    part_number = 1
    offset = 0
    while offset < total_size:
        num_to_upload = min(part_size, total_size - offset)
        # The SizedFileAdapter(fileobj, size) method generates a new object and recalculate
s the length of the append object.
        result = bucket.upload_part(key, upload_id, part_number,
                                   SizedFileAdapter(fileobj, num_to_upload))
        parts.append(PartInfo(part_number, result.etag))
        offset += num_to_upload
        part_number += 1
# Complete multipart upload.
# Specify whether to overwrite the object with the same name when performing multipart uplo
ad.
# By default, if x-oss-forbid-overwrite is not specified, the object with the same name is
overwritten.
# If x-oss-forbid-overwrite is set to false, the object with the same name is overwritten.
# If x-oss-forbid-overwrite is set to true, the object with the same name is not overwritte
n. If the object with the same name already exists, an error is returned.
headers["x-oss-forbid-overwrite"] = "true"
```

```
bucket.complete_multipart_upload(key, upload_id, parts, headers=headers)
# Verify multipart upload.
with open(filename, 'rb') as fileobj:
    assert bucket.get_object(key).read() == fileobj.read()
```

5.8.11. Restore objects



Restoration

If you want to read an Archive object or a Cold Archive object, you must restore the object in advance. It takes several minutes to restore an Archive object and takes several hours to restore a Cold Archive object.

The following section describes the status of an Archive object or a Cold Archive object throughout the restoration process:

1. By default, an Archive or a Cold Archive object is in the frozen state before restoration.
2. After you submit a restore request, the object is in the restoring state.
3. After the server completes the restore task, the object enters the restored state and you can read the object.
 - Archive objects

For Archive objects, the restored state lasts 24 hours by default. During the 24 hours, if you call `RestoreObject` again, the restored state is extended by 24 hours. You can extend the restored state to seven days by calling `RestoreObject` six times during the first 24 hours of restored state. You can also configure the duration of the restored state in days by calling `RestoreObject` once. You can specify a duration of at most seven days.
 - Cold Archive objects

For Cold Archive objects, you can specify the duration of restored state and restoration priority. The duration of the restored state must be at least one day and at most 365 days. The time required to restore a Cold Archive object to the readable state is determined based on the restoration priority of the object:

 - Expedited: The object is restored within one hour.
 - Standard: The object is restored within two to five hours. If the `JobParameters` element is not passed in, the default restoration priority is Standard.
 - Bulk: The object is restored within five to twelve hours.
4. After the restored state expires, the object returns to the frozen state.

Restore Archive objects

The following code provides an example on how to restore an Archive object:

```
# -*- coding: utf-8 -*-
import oss2
# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to acc
ess OSS because the account has permissions on all API operations. We recommend that you us
e a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to
the RAM console.
auth = oss2.Auth('<yourAccessKeyId>', '<yourAccessKeySecret>')
# Set yourEndpoint to the endpoint of the region in which the bucket is located. For exampl
e, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss
-cn-hangzhou.aliyuncs.com.
# Specify the name of the bucket. Example: examplebucket.
bucket = oss2.Bucket(auth, 'https://oss-cn-hangzhou.aliyuncs.com', 'examplebucket')
# Specify the full path of the object. The path cannot contain the bucket name.
objectName = 'yourObjectName'
bucket.restore_object(objectName)
```

Restore a Cold Archive object

The following code provides an example on how to restore a Cold Archive object:

```
# -*- coding: utf-8 -*-
import oss2
from oss2.models import (RestoreJobParameters,
                        RestoreConfiguration,
                        RESTORE_TIER_EXPEDITED,
                        RESTORE_TIER_STANDARD,
                        RESTORE_TIER_BULK)

# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('<yourAccessKeyId>', '<yourAccessKeySecret>')
# Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com.
# Specify the name of the bucket. Example: examplebucket.
bucket = oss2.Bucket(auth, 'https://oss-cn-hangzhou.aliyuncs.com', 'examplebucket')
# Specify the full path of the object. The path cannot contain the bucket name.
object_name = "yourColdArchiveObjectName"
# Refer to the following code if you want to set the storage class of the object to Cold Archive when you upload the object.
# bucket.put_object(object_name, 'yourContent', headers={"x-oss-storage-class": oss2.BUCKET_STORAGE_CLASS_COLD_ARCHIVE})
# Configure the restoration priority of the Cold Archive object.
# RESTORE_TIER_EXPEDITED: The object is restored within one hour.
# RESTORE_TIER_STANDARD: The object is restored within two to five hours.
# RESTORE_TIER_BULK: The object is restored within five to twelve hours.
job_parameters = RestoreJobParameters(RESTORE_TIER_STANDARD)
# Configure parameters. For example, set the restoration priority of the object to Standard and set the duration for which the object can remain in the restored state to two days.
# The days parameter specifies the duration for which the object can remain in the restored state. The default value is one day. This parameter applies to Archive and Cold Archive objects.
# The job_parameters parameter specifies the restoration priority of the object. This parameter applies only to Cold Archive objects.
restore_config= RestoreConfiguration(days=2, job_parameters=job_parameters)
# Initiate a restore request.
bucket.restore_object(object_name, input=restore_config)
```

References

- For more information about the complete sample code that is used to restore Archive or Cold Archive objects, visit [GitHub](#).
-

5.8.12. Manage symbolic links

Symbolic links can be used to access objects that are frequently used in buckets. After you create a symbolic link for an object, you can use the symbolic link to access the object.

Create a symbolic link

The following code provides an example on how to create a symbolic link:

```
# -*- coding: utf-8 -*-
import oss2

# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access Object Storage Service (OSS) because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# In this example, the endpoint of the China (Hangzhou) region is used. Specify the endpoint based on your business requirements. For more information about the endpoints of other regions, see Regions and endpoints.
# Specify the name of the bucket. Example: examplebucket. For more information about the naming conventions for buckets, see Bucket naming conventions.
bucket = oss2.Bucket(auth, 'https://oss-cn-hangzhou.aliyuncs.com', 'examplebucket')
# Specify the name of the object to which the symbolic link points. Example: exampleobject.txt. For more information about the naming conventions for objects, see Object naming conventions.
object_name = 'exampleobject.txt'
# Specify the name of the symbolic link. Example: examplesymlick.txt.
symlink = 'examplesymlick.txt';
# headers = dict()
# Specify whether to overwrite the object with the same name when the symbolic link is created. In this example, this parameter is set to true, which indicates that the object with the same name cannot be overwritten.
# headers['x-oss-forbid-overwrite'] = 'true'
# Specify the access control list (ACL) of the object. In this example, this parameter is set to OBJECT_ACL_PRIVATE, which indicates private. Only the owner of the object and authorized users have read and write permissions on the object.
# headers[OSS_OBJECT_ACL] = oss2.OBJECT_ACL_PRIVATE
# Specify the storage class of the object. In this example, this parameter is set to BUCKET_STORAGE_CLASS_STANDARD, which indicates Standard.
# headers['x-oss-storage-class'] = oss2.BUCKET_STORAGE_CLASS_STANDARD
# bucket.put_symlink(object_name, symlink, headers=headers)
# Create the symbolic link.
bucket.put_symlink(object_name, symlink)
```

Query the name of the object to which a symbolic link points

To query a symbolic link, you must have read permissions on the symbolic link. The following code provides an example on how to query a symbolic link and the name of the object to which the symbolic link points:

```
# -*- coding: utf-8 -*-
import oss2

# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# In this example, the endpoint of the China (Hangzhou) region is used. Specify the endpoint based on your business requirements.
# Specify the name of the bucket. Example: examplebucket.
bucket = oss2.Bucket(auth, 'https://oss-cn-hangzhou.aliyuncs.com', 'examplebucket')
# Specify the name of the symbolic link that you want to query.
symlink = 'examplesymlick.txt';
# Query the symbolic link and the name of the object to which the symbolic link points.
bucket.get_symlink(symlink)
result = bucket.get_symlink(symlink)
print(result.target_key)
```

References

- For more information about the API operation that you can call to create a symbolic link, see [Put Symlink](#).
- For more information about the API operation that you can call to query a symbolic link, see [Get Symlink](#).

5.9. Versioning

5.9.1. Versioning

The versioning state of a bucket applies to all objects in the bucket. Versioning allows you to restore objects in a bucket to previous version and protects your data from being accidentally overwritten or deleted.

A bucket can be in one of the following versioning states: unversioned (default), versioning-enabled, or versioning-suspended. For more information about versioning, see [Overview](#).

Configure versioning for a bucket

The following code provides an example on how to set versioning for a bucket to Enabled or Suspended:

```
# -*- coding: utf-8 -*-
import oss2
from oss2.models import BucketVersioningConfig
# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log
on to OSS because the account has permissions on all API
operations. We recommend that you use your RAM user's credentials to call API operations or
perform routine operations and maintenance. To create a RAM user, log on to the RAM console
.
auth = oss2.Auth('<yourAccessKeyId>', '<yourAccessKeySecret>')
// The endpoint of the China (Hangzhou) region is used in this region. Specify the actual e
ndpoint based on your requirements.
bucket = oss2.Bucket(auth, 'http://oss-cn-hangzhou.aliyuncs.com', '<yourBucketName>')
# Initialize the versioning configurations for the bucket.
config = BucketVersioningConfig()
# Set the versioning state to Enabled or Suspended.
config.status = oss2.BUCKET_VERSIONING_ENABLE
# Configure versioning for the bucket.
result = bucket.put_bucket_versioning(config)
# View the HTTP status code.
print('http response code:', result.status)
```

For more information about how to configure versioning for a bucket, see [PutBucketVersioning](#).

Obtain the versioning state of a bucket

The following code provides an example on how to obtain the versioning state of a bucket:

```
# -*- coding: utf-8 -*-
import oss2
# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log
on to OSS because the account has permissions on all API
operations. We recommend that you use your RAM user's credentials to call API operations or
perform routine operations and maintenance. To create a RAM user, log on to the RAM console
.
auth = oss2.Auth('<yourAccessKeyId>', '<yourAccessKeySecret>')
// The endpoint of the China (Hangzhou) region is used in this region. Specify the actual e
ndpoint based on your requirements.
bucket = oss2.Bucket(auth, 'http://oss-cn-hangzhou.aliyuncs.com', '<yourBucketName>')
# Obtain the versioning state of the bucket.
versioning_info = bucket.get_bucket_versioning()
# View the versioning state of the bucket. If versioning has been enabled, Enabled or Suspe
nded is returned. If versioning has not been enabled, None is returned.
print('bucket versioning status:', versioning_info.status)
```

For more information about how to obtain the versioning state of a bucket, see [GetBucketVersioning](#).

5.9.2. Upload objects

This topic describes how to upload objects to a versioning-enabled bucket by using simple upload, append upload, or multipart upload.

Simple upload

When you upload an object to a versioning-enabled bucket, OSS automatically adds a unique version ID for the uploaded object, and returns the version ID as the x-oss-version-id field in the response header. In a versioning-suspended bucket, the version ID added to an upload object is null. If you upload an object with the same name as an existing object to a versioning-suspended bucket, the existing object is overwritten. In this case, each object only has one version of which the ID is null.

The following code provides an example for simple upload:

```
# -*- coding: utf-8 -*-
import oss2

# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log
on to OSS because the account has permissions on all API operations. We recommend that you
use your RAM user's credentials to call API operations or perform routine operations and ma
intenance. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('<yourAccessKeyId>', '<yourAccessKeySecret>')
# The endpoint of the China (Hangzhou) region is used in this example. Specify the actual e
ndpoint.
bucket = oss2.Bucket(auth, 'http://oss-cn-hangzhou.aliyuncs.com', '<yourBucketName>')
# Upload an object.
result = bucket.put_object('<yourObjectName>', 'content of object')
# Obtain the returned HTTP status code.
print('http status: {0}'.format(result.status))
# View the version ID of the uploaded object.
print('put object version:', result.versionid)
```

For more information about simple upload, see [PutObject](#).

Append upload

In a versioning-enabled bucket, the AppendObject operation can be performed only on objects of which the current version is an appendable object.

Note

- When you perform the AppendObject operation on an object of which the current version is an appendable object, OSS does not generate a previous version for the appendable object.
- When you perform the PutObject or DeleteObject operation on an object of which the current version is an appendable object, OSS stores the appendable object as a previous version and prevents the object from being further appended.
- You cannot perform the AppendObject operation on objects of which the current version is not an appendable object such as a normal object or a delete marker.

The following code provides an example for append upload:

```
# -*- coding: utf-8 -*-
import oss2
# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log
on to OSS because the account has permissions on all API operations. We recommend that you
use your RAM user's credentials to call API operations or perform routine operations and ma
intenance. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('<yourAccessKeyId>', '<yourAccessKeySecret>')
# The endpoint of the China (Hangzhou) region is used in this example. Specify the actual e
ndpoint.
bucket = oss2.Bucket(auth, 'http://oss-cn-hangzhou.aliyuncs.com', '<yourBucketName>')
# Set the position for the first upload (the Position parameter) to 0.
result = bucket.append_object('<yourObjectName>', 0, 'content of first append')
# View the version ID of the object that is appended.
print('append object versionid:', result.versionid)
# If you have performed the AppendObject operation on the object before, you can obtain the
position from where the operation starts this time from the next_position field in the resp
onse returned in the last operation or by using the bucket.head_object method.
bucket.append_object('<yourObjectName>', result.next_position, 'content of second append')
```

For more information about append upload, see [AppendObject](#).

Multipart upload

If you call `CompleteMultipartUpload` to complete the MultipartUpload task for an object in a versioning-enabled bucket, OSS generates a unique version ID for the object and returns it as the `x-oss-version-id` field in the response header.

The following code provides an example for multipart upload:

```

# -*- coding: utf-8 -*-
import os
import oss2
from oss2 import SizedFileAdapter, determine_part_size
from oss2.models import PartInfo
# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log
on to OSS because the account has permissions on all API operations. We recommend that you
use your RAM user's credentials to call API operations or perform routine operations and ma
intenance. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('<yourAccessKeyId>', '<yourAccessKeySecret>')
# The endpoint of the China (Hangzhou) region is used in this example. Specify the actual e
ndpoint.
bucket = oss2.Bucket(auth, 'http://oss-cn-hangzhou.aliyuncs.com', '<yourBucketName>')
key = '<yourObjectName>'
filename = '<yourLocalFile>'
total_size = os.path.getsize(filename)
# The determine_part_size method is used to determine the part size.
part_size = determine_part_size(total_size, preferred_size=100 * 1024)
# Initialize a multipart upload task.
upload_id = bucket.init_multipart_upload(key).upload_id
parts = []
# Upload the parts one by one.
with open(filename, 'rb') as fileobj:
    part_number = 1
    offset = 0
    while offset < total_size:
        num_to_upload = min(part_size, total_size - offset)
        # The SizedFileAdapter(fileobj, size) method generates a new object, and resets the app
end position.
        result = bucket.upload_part(key, upload_id, part_number,
                                   SizedFileAdapter(fileobj, num_to_upload))
        parts.append(PartInfo(part_number, result.etag))
        offset += num_to_upload
        part_number += 1
# Complete the multipart upload task.
result = bucket.complete_multipart_upload(key, upload_id, part_info_list)
# View the version ID of the uploaded object, which is returned in the response.
print('result.versionid:', result.versionid)
# Verify the result of the multipart upload task.
with open(filename, 'rb') as fileobj:
    assert bucket.get_object(key).read() == fileobj.read()

```

For more information about multipart upload, see [CompleteMultipartUpload](#).

5.9.3. Download objects

If you call `GetObject` to download an object in a bucket with versioning enabled or suspended, only the current version of the object is returned by default.

When you perform the `GetObject` operation on an object in a bucket:

- If the current version of the object is a delete marker, the 4040 Not Found error is returned.
- If you specify the versionid of the target object in the request, the specified version of the object is

returned. For example, if the versionId is specified as null in the request, the version of which the versionId is null is returned.

- If the version specified in the request is a delete marker, the 405 Method Not Allowed error is returned.

You can run the following code to download an object:

```
# -*- coding: utf-8 -*-
import oss2

# It is highly risky to log on with the AccessKey of an Alibaba Cloud account because the account has permissions on all the APIs in OSS. We recommend that you log on as a RAM user to access APIs or perform routine operations and maintenance. To create a RAM user account, log on to the RAM console.
auth = oss2.Auth('<yourAccessKeyId>', '<yourAccessKeySecret>')
# This example uses the China East 1 (Hangzhou) endpoint. Specify the actual endpoint based on your requirements.
bucket = oss2.Bucket(auth, 'http://oss-cn-hangzhou.aliyuncs.com', '<yourBucketName>')
# Downloads a specified version of the object.
params = dict()
params['versionId'] = '<yourObjectVersionId>'
object_stream = bucket.get_object('<yourObjectName>', params=params)
# Reads the content of the downloaded object.
read_content = object_stream.read()
print('get object content:', read_content)
# Views the version ID of the downloaded object.
print('get object versionid:', object_stream.versionid)
# The get_object interface returns a stream. You can get the CRC checksum of the object only after you perform the read() operation. Therefore, you must perform CRC verification after the read() operation.
if object_stream.client_crc != object_stream.server_crc:
    print "The CRC checksum between client and server is inconsistent!"
```

5.9.4. Copy objects

This topic describes how to copy objects in a versioned bucket. You can use CopyObject to copy an object that is up to 1 GB in size and use UploadPartCopy to copy an object larger than 1 GB.

Copy a small object

You can use CopyObject to copy an object that is up to 1 GB in size from a source bucket to a destination bucket within the same region.

- By default, the x-oss-copy-source header in a CopyObject request specifies the current version of the object to copy. If the current version of the object is a delete marker, Object Storage Service (OSS) returns 404, which indicates that the object does not exist. You can add a version ID to x-oss-copy-source to copy the specified object version. Delete markers cannot be copied.
- You can copy a previous version of an object to the same bucket. The copied previous version becomes the new version of the object. This way, the previous version of the object is recovered.
- If versioning is enabled for the destination bucket, OSS generates a unique version ID for the destination object. The version ID is returned in the response as the x-oss-version-id header value. If versioning is not enabled or is suspended for the destination bucket, OSS generates a version whose

version ID is null for the destination object and the generated version overwrites the original version whose version ID is null.

- If versioning is not enabled for the destination bucket, you can copy appendable objects. If versioning is enabled or suspended for the destination bucket, you cannot copy appendable objects.

```
# -*- coding: utf-8 -*-
import oss2

# The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# Specify the name of the source bucket. Example: srcexamplebucket.
src_bucket_name = 'srcexamplebucket'
# Specify the name of the destination bucket. The destination bucket must be in the same region as the source bucket. Example: destexamplebucket.
# If you copy an object within a bucket, make sure that you specify the same bucket name for the source and destination buckets.
dest_bucket_name = 'destexamplebucket'
# Specify the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com.
bucket = oss2.Bucket(auth, 'yourEndpoint', dest_bucket_name)
# Specify the full path of the source object. Example: srcexampledir/exampleobject.txt. The full path cannot contain the bucket name.
src_object_name = 'srcexampledir/exampleobject.txt'
# Specify the full path of the destination object. Example: destexampledir/exampleobject.txt. The full path cannot contain the bucket name.
# If you copy an object within a bucket, make sure that you specify the same bucket name for the source and destination buckets.
dest_object_name = 'destexampledir/exampleobject.txt'
# Specify the version ID of the source object.
versionId = 'CAEQMxiBgICAof2D0BYiIDJhMGE3N2MlYTI1NDQzOGY5NTkyNTI3MGYyMzJm****'
# Copy the specified version of the object.
params = dict()
params['versionId'] = versionId
# Copy the object from the source bucket to the destination bucket.
result = bucket.copy_object(src_bucket_name, src_object_name, dest_object_name, params=params)
# View the response. If the return value is 200, the source object is copied to the destination bucket.
print('result.status:', result.status)
```

For more information about how to copy small objects, see [CopyObject](#).

Copy a large object

To copy an object that is larger than 1 GB in size, you must split the object into parts and copy the parts by using UploadPartCopy.

By default, the UploadPartCopy operation uploads a part by copying data from the current version of an existing object. You can add versionId to the x-oss-copy-source request header to copy the specified version of the object. Example: x-oss-copy-source : /SourceBucketName/SourceObjectName?versionId=CAEQMxiBgICAof2D0BYiIDJhMGE3N2M1YTl1NDQzOGY5NTkyNTl3MGYyMzM****.

 **Note** The value of SourceObjectName must be URL-encoded. The version ID of the copied object is returned in the response as the value of the x-oss-copy-source-version-id header.

If versionId is not specified in the request and the current version of the source object is a delete marker, OSS returns 404 Not Found. If versionId is specified in the request and the specified version of the source object is a delete marker, OSS returns 400 Bad Request.

The following code provides an example on how to copy an object by part:

```
# -*- coding: utf-8 -*-
import os
import oss2
from oss2 import determine_part_size
from oss2.models import PartInfo
# The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# Specify the name of the source bucket. Example: srcexamplebucket.
src_bucket_name = 'srcexamplebucket'
# Specify the name of the destination bucket. The destination bucket must be in the same region as the source bucket. Example: destexamplebucket.
# If you copy an object within a bucket, make sure that you specify the same bucket name for the source and destination buckets.
dest_bucket_name = 'destexamplebucket'
# Specify the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com.
bucket = oss2.Bucket(auth, 'yourEndpoint', dest_bucket_name)
# If you copy an object within a bucket, comment out this line of code and change src_bucket in src_bucket_name to bucket.
src_bucket = oss2.Bucket(auth, 'yourEndpoint', src_bucket_name)
# Specify the full path of the source object. Example: srcexampledir/exampleobject.txt. The full path cannot contain the bucket name.
src_object_name = 'srcexampledir/exampleobject.txt'
# Specify the full path of the destination object. Example: destexampledir/exampleobject.txt. The full path cannot contain the bucket name.
dest_object_name = 'destexampledir/exampleobject.txt'
# Specify the version ID of the source object.
versionId = 'CAEQQhiBgMC5kZDE4RciIGZmNzd1YzRjYzRkODQwYjQ5ZDE0Njg5MWM0ODQzNDg3'
params = dict()
params['versionId'] = versionId
# Query the size of the specified version of the source object. If you copy an object within a bucket, change src_bucket to bucket.
head_info = src_bucket.head_object(src_object_name, params=params)
total_size = head_info.content_length
print('src object size:', total_size)
# Use the determine_part_size method to determine the part size.
```

```
part_size = determine_part_size(total_size, preferred_size=100 * 1024)
print('part_size:', part_size)
# Initiate a multipart upload task.
upload_id = bucket.init_multipart_upload(dest_object_name).upload_id
parts = []
# Upload the parts one by one.
part_number = 1
offset = 0
while offset < total_size:
    num_to_upload = min(part_size, total_size - offset)
    end = offset + num_to_upload - 1
    result = bucket.upload_part_copy(src_bucket_name, src_object_name, (offset, end), dest_
object_name, upload_id, part_number, params=params)
    # Save the part information.
    parts.append(PartInfo(part_number, result.etag))
    offset += num_to_upload
    part_number += 1
# Complete the multipart upload task.
result = bucket.complete_multipart_upload(dest_object_name, upload_id, parts)
# Display the version ID of the destination object.
print('result version id:', result.versionid)
# Query the metadata of the destination object.
head_info = bucket.head_object(dest_object_name)
# Query the size of the destination object.
dest_object_size = head_info.content_length
print('dest object size:', dest_object_size)
# Compare the sizes between the source and destination objects.
assert dest_object_size == total_size
```

For more information about multipart copy, see [UploadPartCopy](#).

5.9.5. List objects

This topic describes how to list all objects, a specified number of objects, and objects whose names contain a specified prefix in a versioned bucket.

Lists the versions of all objects in a bucket

The following code provides an example on how to list the versions of all objects including delete markers in a bucket:

```

# -*- coding: utf-8 -*-
import oss2
# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log
on to OSS because the account has permissions on all API operations. We recommend that you
use your RAM user's credentials to call API operations or perform routine operations and ma
intenance. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('<yourAccessKeyId>', '<yourAccessKeySecret>')
# The endpoint of the China (Hangzhou) region is used in this example. Specify the actual e
ndpoint.
bucket = oss2.Bucket(auth, 'http://oss-cn-hangzhou.aliyuncs.com', '<yourBucketName>')
# Use the list_object_versions operation to list the versions of objects in a versioned buc
ket.
# List the versions of all objects including delete markers in a bucket.
result = bucket.list_object_versions()
# List the versions of all objects in the bucket.
next_key_marker = None
next_versionid_marker = None
while True:
    result = bucket.list_object_versions(key_marker=next_key_marker, versionid_marker=next_
versionid_marker)
    # Display the version information about the listed objects.
    for version_info in result.versions:
        print('version_info.versionid:', version_info.versionid)
        print('version_info.key:', version_info.key)
        print('version_info.is_latest:', version_info.is_latest)
    # Display the version information of the listed delete markers.
    for del_maker_Info in result.delete_marker:
        print('del_maker.key:', del_maker_Info.key)
        print('del_maker.versionid:', del_maker_Info.versionid)
        print('del_maker.is_latest:', version_info.is_latest)
    is_truncated = result.is_truncated
    # Check whether the versions of all objects are listed. If not the versions of all obje
cts are listed, continue the list operation. If the versions of all objects are listed, the
list operation ends.
    if is_truncated:
        next_key_marker = result.next_key_marker
        next_versionid_marker = result.next_versionid_marker
    else:
        break

```

List the versions of objects whose names contain the specified prefix

The following code provides an example on how to list the versions of objects whose names contain the specified prefix:


```
# -*- coding: utf-8 -*-
import oss2

# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log
on to OSS because the account has permissions on all API operations. We recommend that you
use your RAM user's credentials to call API operations or perform routine operations and ma
intenance. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('<yourAccessKeyId>', '<yourAccessKeySecret>')
# The endpoint of the China (Hangzhou) region is used in this example. Specify the actual e
ndpoint.
bucket = oss2.Bucket(auth, 'http://oss-cn-hangzhou.aliyuncs.com', '<yourBucketName>')
# Use the list_object_versions operation to list the versions of objects in a versioned buc
ket.
# List the versions of all objects including delete markers in a bucket.
result = bucket.list_object_versions()
# List the versions of objects whose names contain the test- prefix.
prefix = 'test-'
next_key_marker = None
next_versionid_marker = None
while True:
    result = bucket.list_object_versions(prefix=prefix, key_marker=next_key_marker, version
id_marker=next_versionid_marker)
    # Display the version information of the listed objects.
    for version_info in result.versions:
        print('version_info.versionid:', version_info.versionid)
        print('version_info.key:', version_info.key)
        print('version_info.is_latest:', version_info.is_latest)
    # Display the version information of the listed delete markers.
    for del_maker_Info in result.delete_marker:
        print('del_maker.key:', del_maker_Info.key)
        print('del_maker.versionid:', del_maker_Info.versionid)
        print('del_maker.is_latest:', version_info.is_latest)
    is_truncated = result.is_truncated
    # Check whether the versions of all objects whose names contain the test- prefix are li
sted. If not the versions of all objects whose names contain the test- prefix are listed, c
ontinue the list operation. If the versions of all objects whose names contain the test- pr
efix are listed, the list operation ends.
    if is_truncated:
        next_key_marker = result.next_key_marker
        next_versionid_marker = result.next_versionid_marker
    else:
        break
```

List a specific number of object versions

The following code provides an example on how to list a specific numbers of object versions:

```
# -*- coding: utf-8 -*-
import oss2
# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log
on to OSS because the account has permissions on all API operations. We recommend that you
use your RAM user's credentials to call API operations or perform routine operations and ma
intenance. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('<yourAccessKeyId>', '<yourAccessKeySecret>')
# The endpoint of the China (Hangzhou) region is used in this example. Specify the actual e
ndpoint.
bucket = oss2.Bucket(auth, 'http://oss-cn-hangzhou.aliyuncs.com', '<yourBucketName>')
# Use the list_object_versions operation to list the versions of objects in a versioned buc
ket.
# List the versions of all objects including delete markers in a bucket.
result = bucket.list_object_versions()
# List up to 200 object versions.
max_keys = 200
result = bucket.list_object_versions(max_keys=max_keys)
# Display the version information of the listed objects.
for version_info in result.versions:
    print('version_info.versionid:', version_info.versionid)
    print('version_info.key:', version_info.key)
    print('version_info.is_latest:', version_info.is_latest)
# Display the version information of the listed delete markers.
for del_maker_Info in result.delete_marker:
    print('del_maker.key:', del_maker_Info.key)
    print('del_maker.versionid:', del_maker_Info.versionid)
    print('del_maker.is_latest:', version_info.is_latest)
# Check whether the listed result is truncated.
# If the number of objects in the bucket is larger than 200, the value of is_truncated is T
rue, which indicates that the listed result is truncated. If the number of objects in the b
ucket is smaller than 200, the value of is_truncated is False, which indicates that the lis
ted result is not truncated.
print('is truncated', result.is_truncated)
```

List objects by folder

OSS does not use a hierarchical structure for objects, but instead uses a flat structure. All elements are stored as objects in buckets. A folder is an object 0 KB in size whose name ends with a forward slash (/). You can upload and download this object. By default, objects whose names end with a forward slash (/) are displayed as folders in the OSS console.

You can use the delimiter and prefix parameters to list objects by folder.

- If you set prefix to a folder name in the request, objects and subfolders whose names contain the prefix are listed.
- If you also set delimiter to a forward slash (/) in the request, the objects and subfolders whose names start with the specified prefix in the folder are listed. Each subfolder is listed as a single result element in commonPrefixes. The objects and folders in these subfolders are not listed.

Example: A bucket named examplebucket contains the following objects: *oss.jpg*, *fun/test.jpg*, *fun/movie/001.avi*, and *fun/movie/007.txt*. The forward slash (/) is used as the folder delimiter. The structure of the bucket named examplebucket is as following.

```
examplebucket
├── oss.jpg
├── fun
│   ├── test.jpg
│   └── movie
│       ├── 001.avi
│       └── 007.txt
```

The following examples describe how to list objects by simulating folders.

- List the versions of objects in the root folder

The following code provides an example on how to list the versions of objects in the root folder:

```
# -*- coding: utf-8 -*-
import oss2
# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log on to OSS because the account has permissions on all API operations. We recommend that you use your RAM user's credentials to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('<yourAccessKeyId>', '<yourAccessKeySecret>')
# The endpoint of the China (Hangzhou) region is used in this example. Specify the actual endpoint.
bucket = oss2.Bucket(auth, 'http://oss-cn-hangzhou.aliyuncs.com', '<yourBucketName>')
# Use the list_object_versions operation to list the versions of objects in a versioned bucket.
# List the versions of all objects including delete markers in a bucket.
result = bucket.list_object_versions()
# Specify the delimiter as a forward slash (/).
delimiter = "/"
next_key_marker = None
next_versionid_marker = None
while True:
    result = bucket.list_object_versions(delimiter=delimiter, key_marker=next_key_marker, versionid_marker=next_versionid_marker)
    # Display the version information of the listed objects.
    for version_info in result.versions:
        print('version_info.versionid:', version_info.versionid)
        print('version_info.key:', version_info.key)
        print('version_info.is_latest:', version_info.is_latest)
    # Display the version information of the listed delete markers.
    for del_maker_Info in result.delete_marker:
        print('del_maker.key:', del_maker_Info.key)
        print('del_maker.versionid:', del_maker_Info.versionid)
        print('del_maker.is_latest:', version_info.is_latest)
    # Display the folders whose names end with a forward slash (/).
    for common_prefix in result.common_prefix:
        print("common_prefix:", common_prefix)
    is_truncated = result.is_truncated
    # Check whether the versions of all objects in the root folder are listed. If not the versions of all objects in the root folder are listed, continue the list operation. If the versions of all objects in the root folder are listed, the list operation ends.
    if is_truncated:
        next_key_marker = result.next_key_marker
        next_versionid_marker = result.next_versionid_marker
    else:
        break
```

Returned results:

```
('version_info.versionid:', 'CAEQEhiBgMCw8Y7FqBciIGIzMDE3MTEzOWRiMDRmZmFhMmRlMjZlZWlOMWU4****')
('version_info.key:', 'oss.jpg')
('version_info.is_latest:', True)
('common_prefix:', 'fun/')
```

- List objects and subfolders within a folder

The following code provides an example on how to list objects and subfolders within a folder in a bucket:

```
# -*- coding: utf-8 -*-
import oss2
# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log on to OSS because the account has permissions on all API operations. We recommend that you use your RAM user's credentials to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('<yourAccessKeyId>', '<yourAccessKeySecret>')
# The endpoint of the China (Hangzhou) region is used in this example. Specify the actual endpoint.
bucket = oss2.Bucket(auth, 'http://oss-cn-hangzhou.aliyuncs.com', '<yourBucketName>')
# Use the list_object_versions operation to list the versions of objects in a versioned bucket.
# List the versions of all objects including delete markers in a bucket.
result = bucket.list_object_versions()
# Specify the delimiter as a forward slash (/) and prefix as fun/.
prefix = "fun/"
delimiter = "/"
next_key_marker = None
next_versionid_marker = None
while True:
    result = bucket.list_object_versions(prefix=prefix, delimiter=delimiter, key_marker=next_key_marker, versionid_marker=next_versionid_marker)
    # Display the version information of the listed objects.
    for version_info in result.versions:
        print('version_info.versionid:', version_info.versionid)
        print('version_info.key:', version_info.key)
        print('version_info.is_latest:', version_info.is_latest)
    # Display the version information of the listed delete markers.
    for del_maker_Info in result.delete_marker:
        print('del_maker.key:', del_maker_Info.key)
        print('del_maker.versionid:', del_maker_Info.versionid)
        print('del_maker.is_latest:', version_info.is_latest)
    # Display the folders whose names end with a forward slash (/).
    for common_prefix in result.common_prefix:
        print("common_prefix:", common_prefix)
    is_truncated = result.is_truncated
    # Check whether all objects and subfolders within the folder are listed. If not all objects and subfolders within the folder are listed, continue the list operation. If all objects and subfolders within the folder are listed, the list operation ends.
    if is_truncated:
        next_key_marker = result.next_key_marker
        next_versionid_marker = result.next_versionid_marker
    else:
        break
```

Returned results:

```
('version_info.versionid:', 'CAEQFRiBgMCh9JDkrxcIIGE3OTNkYzFhYTc2YzQzOTQ4Y2MzYjg2YjQ4ODg*
****')
('version_info.key:', 'fun/test.jpg')
('version_info.is_latest:', True)
('commonPrefix:', 'fun/movie/')
```

5.9.6. Delete objects

This topic describes how to delete a single object, multiple objects, or objects whose names contain a specified prefix from a versioned bucket.

Delete operations in versioned bucket

When you delete an object from a versioned bucket, you must determine whether to specify a version ID in the request.

- Do not specify a version ID (temporary deletion)

By default, if you do not specify the version ID of the object you want to delete in the request, OSS does not delete the current version of the object and instead add a delete marker to the object as the new current version. If you perform the `GetObject` operation on the object, OSS identifies the current version of the object as a delete marker and returns `404 Not Found`. In addition, the `x-oss-s-delete-marker` header and the `x-oss-version-id` header that indicates the version ID of the created delete marker are included in the response.

The value of the `x-oss-delete-marker` header in the response is `true`, which indicates that the `x-oss-version-id` is the version ID of a delete marker.

- Specify a version ID (permanent deletion)

If you specify the version ID of the object you want to delete in the request, OSS permanently deletes the version whose ID is specified by the `versionId` field in the `params` parameter. To delete a version whose ID is null, add `params['versionId'] = "null"` in `params` in the request. OSS identifies the string "null" as the ID of the version to delete and deletes the version whose ID is null.

Delete a single object

The following examples describe how to permanently or temporarily delete a single object from a versioned bucket.

- Permanently delete an object from a versioned bucket

The following code provides an example on how to permanently delete an object from a versioned bucket by specifying the version ID of the object in the request:

```
# -*- coding: utf-8 -*-
import os
import oss2
# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log on to OSS because the account has permissions on all API operations. We recommend that you use your RAM user's credentials to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('<yourAccessKeyId>', '<yourAccessKeySecret>')
# The endpoint of the China (Hangzhou) region is used in this example. Specify the actual endpoint.
bucket = oss2.Bucket(auth, 'http://oss-cn-hangzhou.aliyuncs.com', '<yourBucketName>')
# Specify the complete name of the object excluding the bucket name. Example: example/test.txt.
object_name = '<yourObjectName>'
# Specify the version ID of the object or delete marker you want to delete.
params = dict()
params['versionId'] = '<yourObjectVersionIdOrDeleteMarkerVersionId>'
# Delete the object with the specified version ID or the object whose current version is a delete marker with the specified ID.
result = bucket.delete_object(object_name, params=params)
print("delete object name: ", object_name)
# If the version ID of an object is specified in the request, the value of the delete_marker header in the response is None and the value of the versionId header in the response is the version ID specified in the request.
# If the version ID of a delete marker is specified in the request, the value of the delete_marker header in the response is True and the value of the versionId header in the response is the version ID specified in the request.
if result.delete_marker:
    print("delete del-marker versionid: ",result.versionid)
else:
    print("delete object versionid:", result.versionid)
```

- Temporarily delete an object from a versioned bucket

The following code provides an example on how to temporarily delete an object from a versioned:

```
# -*- coding: utf-8 -*-
import os
import oss2
# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log on to OSS because the account has permissions on all API operations. We recommend that you use your RAM user's credentials to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('<yourAccessKeyId>', '<yourAccessKeySecret>')
# The endpoint of the China (Hangzhou) region is used in this example. Specify the actual endpoint.
bucket = oss2.Bucket(auth, 'http://oss-cn-hangzhou.aliyuncs.com', '<yourBucketName>')
object_name = '<yourObjectName>'
# Temporarily delete an object without specifying a version ID. A delete marker is added to the object.
result = bucket.delete_object(object_name)
# Display the delete marker added to the object.
print("delete marker: ", result.delete_marker)
# Display the version ID of the added delete marker.
print("delete marker versionid: ", result.versionid)
```

For more information about how to delete a single object, see [DeleteObject](#).

Delete multiple objects

The following examples describe how to permanently or temporarily delete multiple objects from a versioned bucket.

- Permanently delete multiple objects from a versioned bucket

The following code provides an example on how to permanently delete multiple objects or delete markers with specified version IDs from a versioned bucket:


```

# -*- coding: utf-8 -*-
import os
import oss2
from oss2.models import BatchDeleteObjectVersion
from oss2.models import BatchDeleteObjectVersionList
# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to l
og on to OSS because the account has permissions on all API operations. We recommend that
you use your RAM user's credentials to call API operations or perform routine operations
and maintenance. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('<yourAccessKeyId>', '<yourAccessKeySecret>')
# The endpoint of the China (Hangzhou) region is used in this example. Specify the actual
endpoint.
bucket = oss2.Bucket(auth, 'http://oss-cn-hangzhou.aliyuncs.com', '<yourBucketName>')
version_list = BatchDeleteObjectVersionList()
# Specify the version IDs of the objects or delete markers you want to delete.
obj1_versionid = '<yourObject1VersionId>'
obj1_del_marker_versionid = '<yourObject1DelMarkerVersionId>'
obj2_versionid = '<yourObject2VersionId>'
obj2_del_marker_versionid = '<yourObject2DelMarkerVersionId>'
version_list.append(BatchDeleteObjectVersion(key='<yourObject1Name>', versionid=obj1_vers
ionid))
version_list.append(BatchDeleteObjectVersion(key='<yourObject1Name>', versionid=obj1_del_
marker_versionid))
version_list.append(BatchDeleteObjectVersion(key='<yourObject2Name>', versionid=obj2_vers
ionid))
version_list.append(BatchDeleteObjectVersion(key='<yourObject2Name>', versionid=obj2_del_
marker_versionid))
# Delete the objects or delete markers with the specified version IDs.
result = bucket.delete_object_versions(version_list)
# Display the version IDs of the deleted objects or delete markers.
for del_version in result.delete_versions:
    print('del object name:', del_version.key)
    # Display whether the deleted objects are delete markers.
    print('Is del marker:', del_version.delete_marker)
    # If a deleted object is a delete marker, display the version ID of the delete marker
    . If the deleted object is not a delete marker, display the version ID of the deleted obj
    ect.
    if del_version.delete_marker :
        print('del object del_marker.versionid', del_version.delete_marker_versionid)
    else:
        print('del object versionid:', del_version.versionid)

```

- Temporarily delete multiple objects from a versioned bucket

The following code provides an example on how to temporarily delete multiple objects from a versioned bucket:

```
# -*- coding: utf-8 -*-
import os
import oss2
from oss2.models import BatchDeleteObjectVersion
from oss2.models import BatchDeleteObjectVersionList
# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to l
og on to OSS because the account has permissions on all API operations. We recommend that
you use your RAM user's credentials to call API operations or perform routine operations
and maintenance. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('<yourAccessKeyId>', '<yourAccessKeySecret>')
# The endpoint of the China (Hangzhou) region is used in this example. Specify the actual
endpoint.
bucket = oss2.Bucket(auth, 'http://oss-cn-hangzhou.aliyuncs.com', '<yourBucketName>')
key_list = ['<yourObject1Name>', '<yourObject2Name>']
# If you delete an object without specifying a version ID in the request, a delete marker
is added to the object.
result = bucket.batch_delete_objects(key_list)
for del_version in result.delete_versions:
    print('key name:', del_version.key)
    # Display the returned delete markers.
    print('Is del marker:', del_version.delete_marker)
    print('key del_marker.versionid', del_version.delete_marker_versionid)
```

For more information about how to delete multiple objects, see [DeleteMultipleObjects](#).

Delete objects whose names contain a specified prefix

The following code provides an example on how to delete objects whose names contain a specified prefix:

```
# -*- coding: utf-8 -*-
import oss2
# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log
on to OSS because the account has permissions on all API operations. We recommend that you
use your RAM user's credentials to call API operations or perform routine operations and ma
intenance. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('<yourAccessKeyId>', '<yourAccessKeySecret>')
# The endpoint of the China (Hangzhou) region is used in this example. Specify the actual e
ndpoint.
bucket = oss2.Bucket(auth, 'http://oss-cn-hangzhou.aliyuncs.com', '<yourBucketName>')
prefix = "<yourKeyPrefix>"
# List the version IDs of the objects whose names contain the specified prefix and delete t
hese objects.
next_key_marker = None
next_versionid_marker = None
while True:
    result = bucket.list_object_versions(prefix=prefix, key_marker=next_key_marker, version
id_marker=next_versionid_marker)
    for version_info in result.versions:
        bucket.delete_object(version_info.key, params={'versionId': version_info.versionid}
)
    for del_marker_info in result.delete_marker:
        bucket.delete_object(del_marker_info.key, params={'versionId': del_marker_info.vers
ionid})
    is_truncated = result.is_truncated
    if is_truncated:
        next_key_marker = result.next_key_marker
        next_versionid_marker = result.next_versionid_marker
    else:
        break
```

5.9.7. Restore objects

In a bucket with versioning enabled or suspended, the storage class of each version of an object can be different. `RestoreObject` restores the current version of the target object by default. You can specify the `versionId` in the request to restore the specified version of the object.

You can run the following code to restore an object:

```
# -*- coding: utf-8 -*-
import oss2

# It is highly risky to log on with the AccessKey of an Alibaba Cloud account because the account has permissions on all the APIs in OSS. We recommend that you log on as a RAM user to access APIs or perform routine operations and maintenance. To create a RAM user account, log on to https://ram.console.aliyun.com.
auth = oss2.Auth('<yourAccessKeyId>', '<yourAccessKeySecret>')
# This example uses the China East 1 (Hangzhou) endpoint. Specify the actual endpoint based on your requirements.
bucket = oss2.Bucket(auth, 'http://oss-cn-hangzhou.aliyuncs.com', '<yourBucketName>')
# Inputs the versionId field to the param parameter.
params = dict()
params['versionId'] = '<yourObjectVersionId>'
# Inputs the name of an object of the Archive storage class. If no object of the Archive storage class exists in the bucket, call the put_object interface to create one.
object_name = 'yourArchiveObjectName'
# Obtains the metadata of a specified version of the object.
meta = bucket.head_object(object_name, params=params)
# Determines whether the storage class of the object is Archive.
if meta.resp.headers['x-oss-storage-class'] == oss2.BUCKET_STORAGE_CLASS_ARCHIVE:
    # Restores the specified version of the object.
    result = bucket.restore_object(object_name, params=params)
    # Views the version ID of the object that is restored.
    print('restore object version id:', result.versionid)
    while True:
        meta = bucket.head_object(object_name, params=params)
        print('x-oss-restore:', meta.resp.headers['x-oss-restore'])
        # Determines the status of the restored object.
        if meta.resp.headers['x-oss-restore'] == 'ongoing-request="true"':
            time.sleep(5)
        else:
            break
```

5.9.8. Obtain object metadata

In a bucket with versioning enabled or suspended, the HeadObject operation only obtains the metadata of the current version of an object.

 **Note** If the current version of the target object is a delete marker, the 404 Not Found error is returned. You can specify the versionId to obtain the metadata of a specified version of the target object.

You can run the following code to obtain the metadata of an object:

```
# -*- coding: utf-8 -*-
import oss2

# It is highly risky to log on with the AccessKey of an Alibaba Cloud account because the account has permissions on all the APIs in OSS. We recommend that you log on as a RAM user to access APIs or perform routine operations and maintenance. To create a RAM user account, log on to https://ram.console.aliyun.com.
auth = oss2.Auth('<yourAccessKeyId>', '<yourAccessKeySecret>')
# This example uses the China East 1 (Hangzhou) endpoint. Specify the actual endpoint based on your requirements.
bucket = oss2.Bucket(auth, 'http://oss-cn-hangzhou.aliyuncs.com', '<yourBucketName>')
# Inputs the versionId field to the param parameter.
params = dict()
params['versionId'] = '<yourObjectVersionId>'
# Obtains a part of metadata of the object.
simplifiedmeta = bucket.get_object_meta("<yourObjectName>", params=params)
print(simplifiedmeta.headers['Last-Modified'])
print(simplifiedmeta.headers['Content-Length'])
print(simplifiedmeta.headers['ETag'])
# Obtains all metadata of the object.
objectmeta = bucket.head_object("<yourObjectName>", params=params)
print(objectmeta.headers['Content-Type'])
print(objectmeta.headers['Last-Modified'])
print(objectmeta.headers['x-oss-object-type'])
```

For more information about obtaining object metadata, see [HeadObject](#) and [GetObjectMeta](#).

5.9.9. Manage object ACL

This topic describes how to manage the ACLs for objects in a bucket with versioning enabled or suspended.

Set the ACL for an object

PutObjectACL sets the ACL for the current version of the target object by default. If the current version of the target object is a delete marker, the 404 Not Found error is returned. You can set the versionId in the request to set the ACL for a specified version of the target object.

You can run the following code to set the ACL for an object:

```
# -*- coding: utf-8 -*-
import oss2
# It is highly risky to log on with the AccessKey of an Alibaba Cloud account because the account has permissions on all the APIs in OSS. We recommend that you log on as a RAM user to access APIs or perform routine operations and maintenance. To create a RAM user account, log on to https://ram.console.aliyun.com.
auth = oss2.Auth('<yourAccessKeyId>', '<yourAccessKeySecret>')
# This example uses the China East 1 (Hangzhou) endpoint. Specify the actual endpoint based on your requirements.
bucket = oss2.Bucket(auth, 'http://oss-cn-hangzhou.aliyuncs.com', '<yourBucketName>')
# Sets the ACL for a specified version of the object. In this example, the ACL for the object is modified to public-read.
params = dict()
params['versionId'] = '<yourObjectVersionId>'
result = bucket.put_object_acl('<yourObjectName>', oss2.OBJECT_ACL_PUBLIC_READ, params = params)
# Views the version ID of the object for which the ACL is modified.
print('set acl object versionid:', result.versionid)
```

For more information about setting the ACL for an object, see [PutObjectACL](#).

Obtain the ACL for an object

GetObjectACL obtains the ACL for the current version of the target object by default. If the current version of the object is a delete marker, the 404 Not Found error is returned. You can specify the versionId in the request to obtain the ACL for a specified version of the target object.

You can run the following code to obtain the ACL for an object:

```
# -*- coding: utf-8 -*-
import oss2
# It is highly risky to log on with the AccessKey of an Alibaba Cloud account because the account has permissions on all the APIs in OSS. We recommend that you log on as a RAM user to access APIs or perform routine operations and maintenance. To create a RAM user account, log on to https://ram.console.aliyun.com.
auth = oss2.Auth('<yourAccessKeyId>', '<yourAccessKeySecret>')
# This example uses the China East 1 (Hangzhou) endpoint. Specify the actual endpoint based on your requirements.
bucket = oss2.Bucket(auth, 'http://oss-cn-hangzhou.aliyuncs.com', '<yourBucketName>')
# Obtains the ACL for a specified version of the object.
params = dict()
params['versionId'] = '<yourObjectVersionId>'
result = bucket.get_object_acl('<yourObjectName>', params = params)
# Views the obtained ACL for the specified version of the object.
print('get object acl:', result.acl)
# Views the version ID of the object for which the ACL is obtained.
print('object version id:', result.versionid)
```

For more information about obtaining the ACL for an object, see [GetObjectACL](#).

5.9.10. Manage symbolic links

You can configure a symbolic link to point to a frequently accessed object in the target bucket for easier object retrieval. After configuring a symbolic link for an object, you can use the symbolic link to quickly access the object. A symbolic link works in a similar manner as the short cut in Windows. This topic describes how to manage symbolic links in a versioning-enabled bucket.

Create a symbolic link

A symbolic link can have multiple versions. When you create a symbolic link in a versioning-enabled bucket, the version ID of the link is automatically generated by OSS and is returned as the `x-oss-version-id` header in the response. You can create a symbolic link and point it to the current version of the target object.

 **Note** In a versioning-enabled bucket, a symbolic link cannot be created for a delete marker.

The following code provides an example on how to create a symbolic link in a versioning-enabled bucket:

```
# -*- coding: utf-8 -*-
import oss2

# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log
# on to OSS because the account has permissions on all API operations. We recommend that you
# use your RAM user's credentials to call API operations or perform routine operations and ma
# intenance. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('<yourAccessKeyId>', '<yourAccessKeySecret>')
# The endpoint of the China (Hangzhou) region is used in this example. Specify the actual e
# ndpoint.
bucket = oss2.Bucket(auth, 'http://oss-cn-hangzhou.aliyuncs.com', '<yourBucketName>')
objectName = '<yourTargetObjectName>'
symlink = '<yourSymlink>'
# Upload a symbolic link.
result = bucket.put_symlink(objectName, symlink)
# View the version ID of the uploaded symbolic link.
print('symlink versionid:', result.versionid)
```

For more information about how to create a symbolic link, see [PutSymlink](#).

Obtain a symbolic link

By default, `GetSymlink` obtains the current version of the target symbolic link. You can specify the version ID in the request to obtain the specified version of a symbolic link. If the current version of the target symbolic link is a delete marker, OSS returns the 404 Not Found error and includes `oss-delete-marker = true` and `x-oss-version-id` in the response header.

 **Note** You must have the read permission on a symbolic link to obtain it.

The following code provides an example on how to obtain a symbolic link:

```
# -*- coding: utf-8 -*-
import oss2
# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log
on to OSS because the account has permissions on all API operations. We recommend that you
use your RAM user's credentials to call API operations or perform routine operations and ma
intenance. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('<yourAccessKeyId>', '<yourAccessKeySecret>')
# The endpoint of the China (Hangzhou) region is used in this example. Specify the actual e
ndpoint.
bucket = oss2.Bucket(auth, 'http://oss-cn-hangzhou.aliyuncs.com', '<yourBucketName>')
symlink = '<yourSymlink>'
# Obtain the symbolic link with the specified version ID.
params = dict()
params['versionId'] = 'yourSymlinkVersionId'
result = bucket.get_symlink(symlink, params=params)
# View the version ID of the returned symbolic link.
print('get symlink versionid:', result.versionid)
```

For more information about how to obtain a symbolic link, see [GetSymlink](#).

5.10. Object tagging

5.10.1. Configure object tagging

Object Storage Service (OSS) allows you to configure object tags to classify objects. Tags allow you to configure lifecycle rules and access control lists (ACLs) for objects that have the same tag.

Background information

When you configure object tagging, take note of the following items:

- You can add tags to an object when and after you upload the object. If the object already has tags, the existing tags are overwritten. For more information about object tags, see [PutObjectTagging](#).
- To add tags to an object, you must have permissions to call the PutObjectTagging operation.

You can create a custom policy by using scripts to grant the permissions to RAM users. For more information, see [Attach a custom policy to a RAM user](#).

- The Last-Modified value of an object is not updated when the tags are changed.
- You can add up to 10 tags to an object. The tags added to an object must have unique keys.
- A tag key can be up to 128 characters in length. A tag value can be up to 256 characters in length.
- Tag keys and tag values are case-sensitive.
- The key and the value of a tag can contain letters, digits, spaces, and the following special characters:

+ - = . _ : /

 **Note** If you configure the tags in the HTTP header and the tags contain special characters, you can encode the keys and the values of the tags by using URL encoding.

Object tagging uses a key-value pair to identify objects. For more information about object tags, see [Object tagging](#).

Add tags to an object when you upload it

- Add tags to an object when you upload it by using simple upload

The following code provides an example on how to add tags to an object when you upload the object by using simple upload:

```
# -*- coding: utf-8 -*-
import oss2
from oss2.headers import OSS_OBJECT_TAGGING
# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access Object Storage Service (OSS) because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com.
# Specify the name of the bucket. Example: examplebucket.
bucket = oss2.Bucket(auth, 'yourEndpoint', 'examplebucket')
# Specify the full path of the object. The full path of the object cannot contain bucket names. Example: exampledir/exampleobject.txt.
object_name = 'exampledir/exampleobject.txt'
# Configure the tagging string.
tagging = "k1=v1&k2=v2&k3=v3"
# If tags contain characters, you must encode the keys and the values of the tags by using URL encoding.
k4 = "k4+=="
v4 = "+==._:/"
tagging += "&" + oss2.urlquote(k4) + "=" + oss2.urlquote(v4)
# Configure the tags in the HTTP headers.
headers = dict()
headers[OSS_OBJECT_TAGGING] = tagging
# Specify the headers when you call the put_object operation so that the tags are added to the object when it is uploaded.
result = bucket.put_object(object_name, 'content', headers=headers)
print('http response status: ', result.status)
# Query the tags added to the object.
result = bucket.get_object_tagging(object_name)
for key in result.tag_set.tagging_rule:
    print('tagging key: {}, value: {}'.format(key, result.tag_set.tagging_rule[key]))
```

- Add tags to an object when you upload it by using multipart upload

The following code provides an example on how to add tags to an object when you upload the object by using multipart upload:

```
# -*- coding: utf-8 -*-
import os
import oss2
from oss2 import SizedFileAdapter, determine_part_size
from oss2.models import PartInfo
from oss2.headers import OSS_OBJECT_TAGGING
# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log
```

```

on to the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# Set yourEndpoint to the endpoint of the region in which the bucket is located. For exam-
ple, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://
/oss-cn-hangzhou.aliyuncs.com.
# Specify the name of the bucket. Example: examplebucket.
bucket = oss2.Bucket(auth, 'yourEndpoint', 'examplebucket')
# Specify the full path of the object. The full path of the object cannot contain bucket
names. Example: exampledir/exampleobject.txt.
object_name = 'exampledir/exampleobject.txt'
# Specify the full path of the local file that you want to upload. Example: D:\\localpath
\\examplefile.txt.
# By default, if you specify only the name of the local file such as examplefile.txt with-
out specifying the local path, the local file is uploaded from the path of the project to
which the sample program belongs.
filename = 'D:\\localpath\\examplefile.txt'
total_size = os.path.getsize(filename)
# Use the determine_part_size method to determine the size of each part.
part_size = determine_part_size(total_size, preferred_size=100 * 1024)
# Configure the tagging string.
tagging = "k1=v1&k2=v2&k3=v3"
# If tags contain characters, you must encode the keys and values of the tags by using UR
L encoding.
k4 = "k4+=="
v4 = "+==._:/"
tagging += "&" + oss2.urlquote(k4) + "=" + oss2.urlquote(v4)
# Configure the tags in the HTTP headers.
headers = dict()
headers[OSS_OBJECT_TAGGING] = tagging
# Initiate a multipart upload task.
# Specify the headers when you call the init_multipart_upload operation so that the tags
are added to the object to upload.
upload_id = bucket.init_multipart_upload(object_name, headers=headers).upload_id
parts = []
# Upload the parts one by one.
with open(filename, 'rb') as fileobj:
    part_number = 1
    offset = 0
    while offset < total_size:
        num_to_upload = min(part_size, total_size - offset)
        # The SizedFileAdapter(fileobj, size) method generates a new object and recalcula-
tes the position from which the append operation starts.
        result = bucket.upload_part(object_name, upload_id, part_number,
                                   SizedFileAdapter(fileobj, num_to_upload))
        parts.append(PartInfo(part_number, result.etag))
        offset += num_to_upload
        part_number += 1
# Complete the multipart upload task.
result = bucket.complete_multipart_upload(object_name, upload_id, parts)
print('http response status: ', result.status)
# Query the tags added to the object.
result = bucket.get_object_tagging(object_name)
for key in result.tag_set.tagging_rule:
    print('tagging key: {}, value: {}'.format(key, result.tag_set.tagging_rule[key]))

```

```
# Verify the result of the multipart upload task.
with open(filename, 'rb') as fileobj:
    assert bucket.get_object(object_name).read() == fileobj.read()
```

- Add tags to an object when you upload it by using append upload

The following code provides an example on how to add tags to an object when you upload the object by using append upload:

```
# -*- coding: utf-8 -*-
import oss2
from oss2.headers import OSS_OBJECT_TAGGING
# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com.
# Specify the name of the bucket. Example: examplebucket.
bucket = oss2.Bucket(auth, 'yourEndpoint', 'examplebucket')
# Specify the full path of the object. The full path of the object cannot contain bucket names. Example: exampledir/exampleobject.txt.
object_name = 'exampledir/exampleobject.txt'
# Configure the tagging string.
tagging = "k1=v1&k2=v2&k3=v3"
# If tags contain characters, you must encode the keys and values of the tags by using URL encoding.
k4 = "k4+-="
v4 = "+-._:/"
tagging += "&" + oss2.urlquote(k4) + "=" + oss2.urlquote(v4)
# Configure the tags in the HTTP headers.
headers = dict()
headers[OSS_OBJECT_TAGGING] = tagging
# Append the object. Specify the headers when you call the append_object operation so that the tags are added to the object.
# Only the tags configured the first time the object is appended are added to the object.

result = bucket.append_object(object_name, 0, '<yourContent>', headers=headers)
# Query the tags added to the object.
result = bucket.get_object_tagging(object_name)
for key in result.tag_set.tagging_rule:
    print('tagging key: {}, value: {}'.format(key, result.tag_set.tagging_rule[key]))
```

- Add tags to an object when you upload the object by using resumable upload

The following code provides an example on how to add tags to an object when you upload the object by using resumable upload:

```
# -*- coding: utf-8 -*-
import oss2
from oss2.headers import OSS_OBJECT_TAGGING
# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com.
# Specify the name of the bucket. Example: examplebucket.
bucket = oss2.Bucket(auth, 'yourEndpoint', 'examplebucket')
# Specify the full path of the object. The full path of the object cannot contain bucket names. Example: exampledir/exampleobject.txt.
object_name = 'exampledir/exampleobject.txt'
# Specify the full path of the local file. By default, if you do not specify the local file, the local file is uploaded from the path of the project to which the sample program belongs.
local_file = 'D:\\localpath\\examplefile.txt'
# Configure the tagging string.
tagging = "k1=v1&k2=v2&k3=v3"
# If tags contain characters, you must encode the keys and values of the tags by using URL encoding.
k4 = "k4+=="
v4 = "+=._:/"
tagging += "&" + oss2.urlquote(k4) + "=" + oss2.urlquote(v4)
# Configure the tags in the HTTP headers.
headers = dict()
headers[OSS_OBJECT_TAGGING] = tagging
# When the object length is greater than or equal to the value of the multipart_threshold parameter, multipart upload is used. The multipart_threshold parameter is optional. The default value of multipart_threshold is 10 MB. If you do not specify a directory by using the store parameter, the .py-oss-upload directory is created in the HOME directory to store the checkpoint information.
# Specify the headers when you call the resumable_upload operation so that the tags are added to the object to be uploaded.
oss2.resumable_upload(bucket, object_name, local_file, headers=headers)
result = bucket.get_object_tagging(object_name)
for key in result.tag_set.tagging_rule:
    print('object tagging key: {}, value: {}'.format(key, result.tag_set.tagging_rule[key]))
```

Add tags to an uploaded object or modify the tags of the object

If an existing object has no tags or the added tags of the object do not meet your requirements, you can add tags to or modify the tags of the existing object.

The following code provides an example on how to add tags to or modify the tags of an existing object:

```
# -*- coding: utf-8 -*-
import oss2
from oss2.models import Tagging, TaggingRule
# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com.
# Specify the name of the bucket.
bucket = oss2.Bucket(auth, 'yourEndpoint', 'examplebucket')
# Specify the full path of the object. The full path of the object cannot contain bucket names. Example: exampledir/exampleobject.txt.
object_name = 'exampledir/exampleobject.txt'
# Create a tagging rule.
rule = TaggingRule()
rule.add('key1', 'value1')
rule.add('key2', 'value2')
# Create a tag.
tagging = Tagging(rule)
# Add the tag to the object.
result = bucket.put_object_tagging(object_name, tagging)
# Query the HTTP status code.
print('http response status:', result.status)
```

Add tags to a specified version of an object or modify the tags of the object

If versioning is enabled for the bucket, you can add tags to a specified version of an object or modify the tags of the object by specifying the version ID of the object.

The following code provides an example on how to add tags to a specified version of an object or modify the tags of the object.

 **Note** For more information about how to obtain the version IDs of an object, see [List objects](#).

```
# -*- coding: utf-8 -*-
import oss2
from oss2.models import Tagging
# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com.
# Specify the name of the bucket. Example: examplebucket.
bucket = oss2.Bucket(auth, 'yourEndpoint', 'examplebucket')
# Specify the full path of the object. The full path of the object cannot contain bucket names. Example: exampledir/exampleobject.txt.
object_name = 'exampledir/exampleobject.txt'
# Specify the version ID of the object. Example: CAEQMxiBgICAof2D0BYiIDJhMGE3N2M1YTI1NDQzOGY5NTkyNTI3MGYyMzJm****.
version_id = 'CAEQMxiBgICAof2D0BYiIDJhMGE3N2M1YTI1NDQzOGY5NTkyNTI3MGYyMzJm****'
tagging = Tagging()
# Specify the key and the value of the object tag. Example: The key is owner, and the value is John.
tagging.tag_set.add('owner', 'John')
tagging.tag_set.add('type', 'document')
params = dict()
params['versionId'] = version_id
bucket.put_object_tagging(object_name, tagging, params=params)
```

Add tags to an object when you copy the object

You can configure one of the following tagging parameters when you copy an object. Default value: Copy. Valid values:

- Copy: The tag of the source object is copied to the destination object.
- Replace: The tag of the destination object is set to the tag specified in the request instead of the tag of the source object.

The following examples describe how to add tags to objects smaller than 1 GB in simple copy mode and larger than 1 GB in multipart copy mode:

- Add tags to an object when you copy the object by calling CopyObject

The following code provides an example on how to add tags to an object smaller than 1 GB when you copy it by calling CopyObject:

```
# -*- coding: utf-8 -*-
import oss2
from oss2.headers import OSS_OBJECT_TAGGING, OSS_OBJECT_TAGGING_COPY_DIRECTIVE
# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://
```

```

/oss-cn-hangzhou.aliyuncs.com.
# Specify the name of the bucket. Example: examplebucket.
bucket = oss2.Bucket(auth, 'yourEndpoint', 'examplebucket')
# Specify the full path of the object. The full path of the object cannot contain bucket
names. Example: srcexampledir/exampleobject.txt.
src_object_name = 'srcexampledir/exampleobject.txt'
# Specify the full path of the object. The full path of the object cannot contain bucket
names. Example: destexampledir1/exampleobject.txt.
dest_object_name1 = 'destexampledir1/exampleobject.txt'
# Specify the full path of the object. The full path of the object cannot contain bucket
names. Example: destexampledir2/exampleobject.txt.
dest_object_name2 = 'destexampledir2/exampleobject.txt'
# Configure the tagging string.
tagging = "k1=v1&k2=v2&k3=v3"
# If tags contain characters, you must encode the keys and the values of the tags by usin
g URL encoding.
k4 = "k4+-="
v4 = "+-=._:/"
tagging += "&" + oss2.urlquote(k4) + "=" + oss2.urlquote(v4)
# Set OSS_OBJECT_TAGGING_COPY_DIRECTIVE to COPY or keep the default value in the HTTP hea
ders, so that the tags of the source object are added to the dest_object_name1 object.
headers=dict()
headers[OSS_OBJECT_TAGGING_COPY_DIRECTIVE] = 'COPY'
bucket.copy_object(bucket.bucket_name, src_object_name, dest_object_name1, headers=header
s)
# Set OSS_OBJECT_TAGGING_COPY_DIRECTIVE to REPLACE in the HTTP headers, so that the tags
specified in OSS_OBJECT_TAGGING are added to the dest_object_name2 object.
headers[OSS_OBJECT_TAGGING_COPY_DIRECTIVE] = 'REPLACE'
headers[OSS_OBJECT_TAGGING] = tagging
bucket.copy_object(bucket.bucket_name, src_object_name, dest_object_name2, headers=header
s)
# Query the tags added to the src_object_name object.
result = bucket.get_object_tagging(src_object_name)
for key in result.tag_set.tagging_rule:
    print('src tagging key: {}, value: {}'.format(key, result.tag_set.tagging_rule[key]))
# Query the tags added to the dest_object_name1 object. The tags added to the dest_object
_name1 object are the same as those of the src_object_name object.
result = bucket.get_object_tagging(dest_object_name1)
for key in result.tag_set.tagging_rule:
    print('dest1 object tagging key: {}, value: {}'.format(key, result.tag_set.tagging_ru
le[key]))
# Query the tags added to the dest_object_name2 object. The tags added to the dest_object
_name2 object are those specified in headers[OSS_OBJECT_TAGGING].
result = bucket.get_object_tagging(dest_object_name2)
for key in result.tag_set.tagging_rule:
    print('dest2 object tagging key: {}, value: {}'.format(key, result.tag_set.tagging_ru
le[key]))

```

- Add tags to an object when you copy the object in multipart copy mode

The following code provides an example on how to add tags to an object larger than 1 GB when you copy it by calling `MultipartUpload`:

```

# -*- coding: utf-8 -*-
import os

```

```

import oss2
from oss2 import determine_part_size
from oss2.models import PartInfo
from oss2.headers import OSS_OBJECT_TAGGING
# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com.
# Specify the name of the bucket. Example: examplebucket.
bucket = oss2.Bucket(auth, 'yourEndpoint', 'examplebucket')
# Specify the full path of the object. The full path of the object cannot contain bucket names. Example: srcexampledir/exampleobject.txt.
src_object_name = 'srcexampledir/exampleobject.txt'
# Specify the full path of the object. The full path of the object cannot contain bucket names. Example: destexampledir/exampleobject.txt.
dest_object_name = 'destexampledir/exampleobject.txt'
# Obtain the size of the source object.
head_info = bucket.head_object(src_object_name)
total_size = head_info.content_length
print('src object size:', total_size)
# Use the determine_part_size method to determine the size of each part.
part_size = determine_part_size(total_size, preferred_size=100 * 1024)
print('part_size:', part_size)
# Configure the tagging string.
tagging = "k1=v1&k2=v2&k3=v3"
# If tags contain characters, you must encode the keys and the values of the tags by using URL encoding.
k4 = "k4+=="
v4 = "+-._:/"
tagging += "&" + oss2.urlquote(k4) + "=" + oss2.urlquote(v4)
# Configure the tags in the HTTP headers.
headers = dict()
headers[OSS_OBJECT_TAGGING] = tagging
# Initiate a multipart upload task.
# Specify the headers when you call the init_multipart_upload operation so that the tags are added to the destination object.
upload_id = bucket.init_multipart_upload(dest_object_name, headers=headers).upload_id
parts = []
# Upload the parts one by one.
part_number = 1
offset = 0
while offset < total_size:
    num_to_upload = min(part_size, total_size - offset)
    end = offset + num_to_upload - 1;
    result = bucket.upload_part_copy(bucket.bucket_name, src_object_name, (offset, end), dest_object_name, upload_id, part_number)
    # Save the part information.
    parts.append(PartInfo(part_number, result.etag))
    offset += num_to_upload
    part_number += 1

```



```
# Complete the multipart upload task.
result = bucket.complete_multipart_upload(dest_object_name, upload_id, parts)
# Obtain the metadata of the destination object.
head_info = bucket.head_object(dest_object_name)
# Query the size of the destination object.
dest_object_size = head_info.content_length
print('dest object size:', dest_object_size)
# Compare the size of the destination object with that of the source object.
assert dest_object_size == total_size
# Query the tags of the source object.
result = bucket.get_object_tagging(src_object_name)
for key in result.tag_set.tagging_rule:
    print('src tagging key: {}, value: {}'.format(key, result.tag_set.tagging_rule[key]))
# Query the tags added to the destination object.
result = bucket.get_object_tagging(dest_object_name)
for key in result.tag_set.tagging_rule:
    print('dest tagging key: {}, value: {}'.format(key, result.tag_set.tagging_rule[key]))
)
```

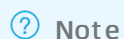
Add tags to a symbolic link

The following code provides an example on how to add tags to a symbolic link:

```
# -*- coding: utf-8 -*-
import oss2
from oss2.headers import OSS_OBJECT_TAGGING
# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com.
# Specify the name of the bucket. Example: examplebucket.
bucket = oss2.Bucket(auth, 'yourEndpoint', 'examplebucket')
# Specify the full path of the object. The full path of the object cannot contain bucket names. Example: exampledir/exampleobject.txt.
object_name = 'exampledir/exampleobject.txt'
# Specify the full path of the symbolic link object. Example: shortcut/myobject.txt.
symlink_name = 'shortcut/myobject.txt'
# Configure the tagging string.
tagging = "k1=v1&k2=v2&k3=v3"
# If tags contain characters, you must encode the keys and values of the tags by using URL encoding.
k4 = "k4+=="
v4 = "+=._:/"
tagging += "&" + oss2.urlquote(k4) + "=" + oss2.urlquote(v4)
# Configure the tags in the HTTP headers.
headers = dict()
headers[OSS_OBJECT_TAGGING] = tagging
# Add a symbolic link to the object.
# Specify the headers when you call the put_symlink operation so that the tags are added to the symbolic link.
result = bucket.put_symlink(object_name, symlink_name, headers=headers)
print('http response status: ', result.status)
# Query the tags added to the symbolic link.
result = bucket.get_object_tagging(symlink_name)
for key in result.tag_set.tagging_rule:
    print('tagging key: {}, value: {}'.format(key, result.tag_set.tagging_rule[key]))
```

5.10.2. Query the tags of an object

After you configure tags for an object, you can query the tags of the object. If versioning is enabled for a bucket that stores the object whose tags you want to query, Object Storage Service (OSS) returns the tags of the current version of the object by default. To query the tags of a specified version of the object, you must specify the version ID of the object.



Note

Query the tags of an object

If versioning is not enabled for the bucket that stores the object whose tags you want to query, you can query the tags of the object based on your requirements. If versioning is enabled for the bucket that stores the object whose tags you want to query, OSS returns the tags of the current version of the object by default.

The following code provides an example on how to query the tags of the `exampleobject.txt` object in the `exampledir` directory of the `examplebucket` bucket:

```
# -*- coding: utf-8 -*-
import oss2
from oss2.models import Tagging, TaggingRule
# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to acc
ess OSS because the account has permissions on all API operations. We recommend that you us
e a Resource Access Management (RAM) user to call API operations or perform routine O&M. To
create a RAM user, log on to the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# Set yourEndpoint to the endpoint of the region in which the bucket is located. For exampl
e, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss
-cn-hangzhou.aliyuncs.com.
# Specify the name of the bucket. Example: examplebucket.
bucket = oss2.Bucket(auth, 'yourEndpoint', 'examplebucket')
# Specify the full path of the object. Example: exampledir/exampleobject.txt. The full path
of the object cannot contain the bucket name.
object_name = 'exampledir/exampleobject.txt'
# Query the tags of the object.
result = bucket.get_object_tagging(object_name)
# View the tags of the object.
for key in result.tag_set.tagging_rule:
    print('tagging key: {}, value: {}'.format(key, result.tag_set.tagging_rule[key]))
```

Query the tags of a specified version of an object

If versioning is enabled for the bucket that stores the object whose tags you want to query, you can query the tags of a specified version of the object by specifying the version ID of the object.

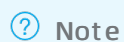
The following code provides an example on how to query the tags of a specified version of the `exampleobject.txt` object in the `exampledir` directory of the `examplebucket` bucket:

 **Note** For more information about how to obtain the version IDs of an object, see [List objects](#).

```
# -*- coding: utf-8 -*-
import oss2
from oss2.models import Tagging
# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com.
# Specify the name of the bucket. Example: examplebucket.
bucket = oss2.Bucket(auth, 'yourEndpoint', 'examplebucket')
# Specify the full path of the object. Example: exampledir/exampleobject.txt. The full path of the object cannot contain the bucket name.
object_name = 'exampledir/exampleobject.txt'
# Specify the version ID of the object. Example: CAEQMxiBgICAof2D0BYiIDJhMGE3N2M1YTI1NDQzOGY5NTkyNTI3MGYyMzJm****.
version_id = 'CAEQMxiBgICAof2D0BYiIDJhMGE3N2M1YTI1NDQzOGY5NTkyNTI3MGYyMzJm****'
params = dict()
params['versionId'] = version_id
result = bucket.get_object_tagging(object_name, params=params)
print(result)
```

5.10.3. Remove the tags of an object

You can delete unnecessary tags of an object based on your requirements. If versioning is enabled for a bucket that stores the object whose tags you want to delete, Object Storage Service (OSS) deletes the tags of the current version of the object by default. To delete the tags of a specified version of the object, you must specify the version ID of the object.



Remove the tags of an object

If versioning is not enabled for the bucket that stores the object whose tags you want to delete, you can delete the tags of the object based on your requirements. If versioning is enabled for the bucket that stores the object whose tags you want to delete, OSS deletes the tags of the current version of the object by default.

The following code provides an example on how to delete the tags of the exampleobject.txt object in the exampledir directory of the examplebucket bucket:

```
# -*- coding: utf-8 -*-
import oss2
from oss2.models import Tagging, TaggingRule
# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to acc
ess OSS because the account has permissions on all API operations. We recommend that you us
e a Resource Access Management (RAM) user to call API operations or perform routine O&M. To
create a RAM user, log on to the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# Set yourEndpoint to the endpoint of the region in which the bucket is located. For exampl
e, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss
-cn-hangzhou.aliyuncs.com.
# Specify the name of the bucket. Example: examplebucket.
bucket = oss2.Bucket(auth, 'yourEndpoint', 'examplebucket')
# Specify the full path of the object. Example: exampledir/exampleobject.txt. The full path
of the object cannot contain the bucket name.
object_name = 'exampledir/exampleobject.txt'
# Remove the tags of the object.
result = bucket.delete_object_tagging(object_name)
print('http response status: ', result.status)
```

Remove the tags of a specified version of an object

If versioning is enabled for the bucket that stores the object whose tags you want to delete, you can delete the tags of a specified version of the object by specifying the version ID of the object.

The following code provides an example on how to delete the tags of a specified version of the exampleobject.txt object in the exampledir directory of the examplebucket bucket:

Note

```
# -*- coding: utf-8 -*-
import oss2
from oss2.models import Tagging
# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to acc
ess OSS because the account has permissions on all API operations. We recommend that you us
e a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to
the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# Set yourEndpoint to the endpoint of the region in which the bucket is located. For exampl
e, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss
-cn-hangzhou.aliyuncs.com.
# Specify the name of the bucket. Example: examplebucket.
bucket = oss2.Bucket(auth, 'yourEndpoint', 'examplebucket')
# Specify the full path of the object. Example: exampledir/exampleobject.txt. The full path
of the object cannot contain the bucket name.
object_name = 'exampledir/exampleobject.txt'
# Specify the version ID of the object.
version_id = 'CAEQMxiBgICAof2D0BYiIDJhMGE3N2M1YTI1NDQzOGY5NTkyNTI3MGYyMzJm*****'
params = dict()
params['versionId'] = version_id
bucket.delete_object_tagging(object_name, params=params)
```

5.10.4. Object tagging and lifecycle rules

Lifecycle rules can take effect on objects configured with specified prefixes or tags. You can also specify a combination of prefixes and tags as conditions of lifecycle rules at the same time.

Note If you configure tag conditions, the rule applies only to objects that meet the tag key and value conditions. If a prefix and multiple object tags are configured in a rule, the rule applies only to objects that match the prefix and object tag conditions.

Add tagging configurations to a lifecycle rule

The following code provides an example on how to add tagging configurations to a lifecycle rule:

```
# -*- coding: utf-8 -*-
import oss2
import datetime
from oss2.models import (LifecycleExpiration, LifecycleRule,
                        BucketLifecycle, AbortMultipartUpload,
                        TaggingRule, Tagging, StorageTransition)

# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log
# on to Object Storage Service (OSS) because the account has permissions on all API operation
# s. We recommend that you use a Resource Access Management (RAM) user to call API operations
# or perform routine operations and maintenance. To create a RAM user, log on to the RAM cons
# ole.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# Set yourEndpoint to the endpoint of the region in which the bucket is located. For exampl
# e, if your bucket is located in the China (Hangzhou) region, set yourEndpoint to https://os
# s-cn-hangzhou.aliyuncs.com.
# Specify the bucket name.
bucket = oss2.Bucket(auth, 'yourEndpoint', 'examplebucket')
# Specify that objects expire three days after they are last modified.
# Set the name of the expiration rule and the prefix to match the objects.
rule1 = LifecycleRule('rule1', 'tests/',
                    # Enable the expiration rule.
                    status=LifecycleRule.ENABLED,
                    # Set the validity period to three days after the last modified date.

                    expiration=LifecycleExpiration(days=3))
# Specify that the objects last modified before the specified date expire.
# Set the name of the expiration rule and the prefix to match the objects.
rule2 = LifecycleRule('rule2', 'logging-',
                    # Disable the expiration rule.
                    status=LifecycleRule.DISABLED,
                    # Specify that the objects last modified before the specified date ex
# pire.

                    expiration=LifecycleExpiration(created_before_date=datetime.date(2018
, 12, 12)))
# Specify that parts expire three days after they are last modified.
rule3 = LifecycleRule('rule3', 'tests1/',
                    status=LifecycleRule.ENABLED,
                    abort_multipart_upload=AbortMultipartUpload(days=3))
# Specify that the parts last modified before the specified date expire.
rule4 = LifecvcleRule('rule4', 'logging1-',
```

```
        status=LifecycleRule.DISABLED,
        abort_multipart_upload = AbortMultipartUpload(created_before_date=datetime.date(2018, 12, 12)))
# Configure tags to match objects.
tagging_rule = TaggingRule()
tagging_rule.add('key1', 'value1')
tagging_rule.add('key2', 'value2')
tagging = Tagging(tagging_rule)
# Configure the rule to convert the storage class of objects. Specify that the storage class of objects is converted to Archive 365 days after the objects are last modified.
# Tags to match objects are specified in rule5. The rule applies only to objects that match tag conditions of key1=value1 and key2=value2.
rule5 = LifecycleRule('rule5', 'logging2-',
                      status=LifecycleRule.ENABLED,
                      storage_transitions=[StorageTransition(days=365, storage_class=oss2.BUCKET_STORAGE_CLASS_ARCHIVE)],
                      tagging = tagging)
lifecycle = BucketLifecycle([rule1, rule2, rule3, rule4, rule5])
bucket.put_bucket_lifecycle(lifecycle)
```

View the tagging configurations of a lifecycle rule

The following code provides an example on how to view the tagging configurations of a lifecycle rule:

```
# -*- coding: utf-8 -*-
import oss2

# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log
on to OSS because the account has permissions on all API operations. We recommend that you
use a RAM user to call API operations or perform routine operations and maintenance. To cre
ate a RAM user, log on to the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# Set yourEndpoint to the endpoint of the region in which the bucket is located. For exampl
e, if your bucket is located in the China (Hangzhou) region, set yourEndpoint to https://os
s-cn-hangzhou.aliyuncs.com.
# Specify the bucket name.
bucket = oss2.Bucket(auth, 'yourEndpoint', 'examplebucket')
# View the lifecycle rules.
lifecycle = bucket.get_bucket_lifecycle()
for rule in lifecycle.rules:
    # View the part expiration rules.
    if rule.abort_multipart_upload is not None:
        print('id={0}, prefix={1}, tagging={2}, status={3}, days={4}, created_before_date={
5}')
        .format(rule.id, rule.prefix, rule.tagging, rule.status,
                rule.abort_multipart_upload.days,
                rule.abort_multipart_upload.created_before_date))
    # View the object expiration rules.
    if rule.expiration is not None:
        print('id={0}, prefix={1}, tagging={2}, status={3}, days={4}, created_before_date={
5}')
        .format(rule.id, rule.prefix, rule.tagging, rule.status,
                rule.expiration.days,
                rule.expiration.created_before_date))
    # View the rules to convert the storage class.
    if len(rule.storage_transitions) > 0:
        storage_trans_info = ''
        for storage_rule in rule.storage_transitions:
            storage_trans_info += 'days={0}, created_before_date={1}, storage_class={2} ***
* '.format(
                storage_rule.days, storage_rule.created_before_date, storage_rule.storage_c
lass)
        print('id={0}, prefix={1}, tagging={2}, status={3},, StorageTransition={4}'
              .format(rule.id, rule.prefix, rule.tagging, rule.status, storage_trans_info
))
```

5.11. Single-connection bandwidth throttling

This topic describes how to add parameters in an object upload and download request to set the limit of upload or download bandwidth. This way, you can ensure other applications have sufficient bandwidth.

Configure single-connection bandwidth throttling for simple upload and download

The following code provides an example on how to configure single-connection bandwidth throttling for simple upload and download:

```
# -*- coding: utf-8 -*-
import oss2
from oss2.models import OSS_TRAFFIC_LIMIT
# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com.
# Specify the name of the bucket.
bucket = oss2.Bucket(auth, 'yourEndpoint', 'examplebucket')
# Specify the full path of the object. The path cannot contain the bucket name. Example: exampledir/exampleobject.txt.
object_name = 'exampledir/exampleobject.txt'
# Specify the full path of the local file that you want to upload. Example: D:\\localpath\\examplefile.txt. By default, if you do not specify the local file, the local file is uploaded from the path of the project to which the sample program belongs.
local_file_name = 'D:\\localpath\\examplefile.txt'
# Specify the full path of the local file to which you want the object to be downloaded. If the specified local file exists, the object to download replaces the file. Otherwise, a file is created.
# If the path for the object is not specified, the downloaded object is saved to the path of the project to which the sample program belongs.
down_file_name = 'D:\\localpath\\exampleobject.txt'
# Set the related headers to limit bandwidth throttling to 100 KB/s, which is equal to 819200 bit/s.
limit_speed = (100 * 1024 * 8)
headers = dict()
headers[OSS_TRAFFIC_LIMIT] = str(limit_speed);
# Configure single-connection bandwidth throttling for the object upload.
result = bucket.put_object_from_file(object_name, local_file_name, headers=headers)
print('http response status:', result.status)
# Configure single-connection bandwidth throttling for the object download.
result = bucket.get_object_to_file(object_name, down_file_name, headers=headers)
print('http response status:', result.status)
```

Configure single-connection bandwidth throttling for object uploads and downloads that use signed URLs

The following code provides an example on how to configure single-connection bandwidth throttling when you use signed URLs to upload or download objects:

```
# -*- coding: utf-8 -*-
import oss2
from oss2.models import OSS_TRAFFIC_LIMIT
# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com.
# Specify the name of the bucket.
bucket = oss2.Bucket(auth, 'yourEndpoint', 'examplebucket')
# Specify the full path of the object. The path cannot contain the bucket name. Example: exampledir/exampleobject.txt.
object_name = 'exampledir/exampleobject.txt'
# Specify the full path of the local file that you want to upload. Example: D:\\localpath\\examplefile.txt. By default, if you do not specify the local file, the local file is uploaded from the path of the project to which the sample program belongs.
local_file_name = 'D:\\localpath\\examplefile.txt'
# Specify the full path of the local file to which you want the object to be downloaded. If the specified local file exists, the object to download replaces the file. Otherwise, a file is created.
# If the path for the object is not specified, the downloaded object is saved to the path of the project to which the sample program belongs.
down_file_name = 'D:\\localpath\\exampleobject.txt'
# Set the related parameters to limit bandwidth throttling to 100 KB/s, which is equal to 819,200 bit/s.
limit_speed = (100 * 1024 * 8)
params = dict()
params[OSS_TRAFFIC_LIMIT] = str(limit_speed);
# Generate a signed URL used to upload an object with single-connection bandwidth throttling configured and set the validity period of the URL to 60 seconds.
url = bucket.sign_url('PUT', object_name, 60, params=params)
print('put object url:', url)
# Configure single-connection bandwidth throttling for the object upload.
result = bucket.put_object_with_url_from_file(url, local_file_name)
print('http response status:', result.status)
# Generate a signed URL used to download an object with single-connection bandwidth throttling configured and set the validity period of the URL to 60 seconds.
url = bucket.sign_url('GET', object_name, 60, params=params)
print('get object url:', url)
# Configure single-connection bandwidth throttling for the object download.
result = bucket.get_object_with_url_to_file(url, down_file_name)
print('http response status:', result.status)
```

References

For more information about the complete sample code for single-connection bandwidth throttling, visit [GitHub](#).

5.12. Data encryption

5.12.1. Client-side encryption

Client-side encryption is used to encrypt objects on the local client before they are uploaded to Object Storage Service (OSS).

Disclaimer

- When you use client-side encryption, you must ensure the integrity and validity of the customer master key (CMK). If the CMK is incorrectly used or lost due to improper maintenance, you will be held responsible for all losses and consequences caused by decryption failures.
- When you copy or migrate encrypted data, you must ensure the integrity and validity of the object metadata related to client-side encryption. If the encrypted metadata is incorrectly used or lost due to improper maintenance, you will be held responsible for all losses and consequences caused by decryption failures.

Background information

In client-side encryption, a random data key is generated for each object to perform symmetric encryption on the object. The client uses a CMK to encrypt the random data key. The encrypted data key is uploaded as a part of the object metadata and stored in the OSS server. When an encrypted object is downloaded, the client uses the CMK to decrypt the random data key and then uses the data key to decrypt the object. The CMK is used only on the client and is not transmitted over the network or stored in the server. This ensures data security.

Encryption methods

You can use CMKs managed in one of the following ways:

- Use KMS-managed CMKs

When you use a CMK stored in KMS for client-side encryption, you must send the CMK ID to OSS SDK.

- Use RSA-based CMKs managed by yourself

When you use a CMK managed by yourself for client-side encryption, you must send the public key and the private key of your CMK to OSS SDK as parameters.

You can use the preceding methods to prevent data leaks and protect your data on the client. Even if your data is leaked, it cannot be decrypted by others.

For more information about client-side encryption, see [Client-side encryption](#) in OSS Developer Guide. For more information about the complete sample code of client-side encryption, visit [GitHub](#).

Client-side encryption V2 (recommended)

Notice

- Client-side encryption supports multipart upload for objects larger than 5 GB. When you use multipart upload to upload an object, you must specify the total size of the object and the size of each part. The size of each part except for the last part must be the same and be a multiple of 16 bytes.
- After you upload objects encrypted on the client, the object metadata related to client-side encryption is protected and cannot be modified by calling operations such as [CopyObject](#).

- Object metadata related to client-side encryption

Parameter	Description	Required
x-oss-meta-client-side-encryption-key	The encrypted data key. The encrypted data key is a string encrypted by using a RSA-based CMK or a CMK managed by KMS and encoded in Base64.	Yes
x-oss-meta-client-side-encryption-start	The initialization vector generated randomly for data encryption. The encrypted data key is a string encrypted by using a RSA-based CMK or a CMK managed by KMS and encoded in Base64.	Yes
x-oss-meta-client-side-encryption-cek-alg	The algorithm used to encrypt data.	Yes
x-oss-meta-client-side-encryption-wrap-alg	The algorithm used to encrypt the data key.	Yes
x-oss-meta-client-side-encryption-matdesc	The description of the content encryption key (CEK) in the JSON format.	No
x-oss-meta-client-side-encryption-unencrypted-content-length	The length of data before encryption. This parameter is not generated if Content-Length is not specified.	No
x-oss-meta-client-side-encryption-unencrypted-content-md5	The MD5 hash of data before encryption. This parameter is not generated if Content-MD5 is not specified.	No
x-oss-meta-client-side-encryption-data-size	The total size of the object to upload by using multipart upload.	No (required for multipart upload)
x-oss-meta-client-side-encryption-part-size	The size of each part in multipart upload.	No (required for multipart upload)

- Create a bucket for client-side encryption

Before you encrypt and upload an object or download and decrypt an object on the client, you must initialize a bucket instance. You can call the operations of the bucket instance to upload or download objects. In client-side encryption, the `CryptoBucket` class inherits the operations from the `Bucket` class. You can specify corresponding parameters to initialize a `CryptoBucket` instance in the same way as a bucket instance.

- Initialize a bucket for object encryption by using a RSA-based CMK managed by yourself

 **Notice** If you use a RSA-based CMK, you must manage the CMK by yourself. The loss of the CMK or the damage to the CMK data may cause decryption failures. We recommend that you use CMKs managed by KMS. If you must use a RSA-based CMK to perform encryption, we recommend that you back up your CMK data.

The following code provides an example on how to initialize a bucket for object encryption by using a RSA-based CMK managed by yourself:

```
# -*- coding: utf-8 -*-
import os
import oss2
from oss2.crypto import RsaProvider
# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to
# access OSS because the account has permissions on all API operations. We recommend that
# you use a RAM user to call API operations or perform routine O&M. To create a RAM user,
# log on to the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# If you want only to decrypt objects, specify only the private key.
# key_pair = {'private_key': 'yourPrivateKey'}
# If you want only to encrypt objects, specify only the public key.
# key_pair = {'public_key': 'yourPublicKey'}
# If you want to encrypt and decrypt objects, you must specify both the public key and
# private key.
key_pair = {'private_key': 'yourPrivateKey', 'public_key': 'yourPublicKey'}
bucket = oss2.CryptoBucket(oss2.Auth(access_key_id, access_key_secret), endpoint, bucket_name,
                           crypto_provider=RsaProvider(key_pair))
```

- Initialize a bucket for object encryption by using a CMK managed by KMS

The following code provides an example on how to initialize a bucket for object encryption by using a CMK managed by KMS:

```
import os
import oss2
from oss2.crypto import AliKMSPProvider
kms_provider=AliKMSPProvider('yourAccessKeyId', 'yourAccessKeySecret', 'yourRegion', 'yourCMKID')
bucket = oss2.CryptoBucket(oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret'), 'yourEndpoint', 'yourBucketName', crypto_provider = kms_provider)
```

- Use and manage multiple CMKs

You may use different CMKs to encrypt objects to upload to a bucket or decrypt objects downloaded from the same bucket. You can configure different descriptions for the CMKs and add the CMKs and their descriptions to the encryption-related information about the bucket. When data is decrypted, OSS SDK automatically matches a CMK based on the description for seamless decryption. Sample code:

```
# -*- coding: utf-8 -*-
import os
import oss2
from oss2.crypto import RsaProvider
# Create a RSA-based key pair.
key_pair_1 = {'private_key': 'yourPrivateKey_1', 'public_key': 'yourPublicKey_1'}
mat_desc_1 = {'key1': 'value1'}
# Create another RSA-based key pair.
key_pair_2 = {'private_key': 'yourPrivateKey_2', 'public_key': 'yourPublicKey_2'}
mat_desc_2 = {'key2': 'value2'}
# Add the description of key_pair_1 to provider.
provider = RsaProvider(key_pair={'private_key': private_key_str_2, 'public_key': public_key_str_2}, mat_desc=mat_desc_2)
encryption_materials = oss2.EncryptionMaterials(mat_desc_1, key_pair=key_pair_1)
provider.add_encryption_materials(encryption_materials)
# Use provider to initialize a crypto_bucket instance. Then, you can use the operations of crypto_bucket to download the object data that is encrypted by a CMK whose description is mat_desc_1.
crypto_bucket = oss2.CryptoBucket(oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret'), 'yourEndpoint', 'yourBucketName', crypto_provider=provider)
```

- Perform client-side encryption in simple upload and download

The following code provides an example on how to use a CMK managed by KMS to encrypt objects to upload in simple upload or decrypt objects when you download the objects:

```
# -*- coding: utf-8 -*-
import os
import oss2
from oss2.crypto import RsaProvider
# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
kms_provider=AliKMSProvider('yourAccessKeyId', 'yourAccessKeySecret', 'yourRegion', 'yourCMKID')
bucket = oss2.CryptoBucket(oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret'), 'yourEndpoint', 'yourBucketName', crypto_provider = kms_provider)
key = 'motto.txt'
content = b'a' * 1024 * 1024
filename = 'download.txt'
# Upload an object.
bucket.put_object(key, content, headers={'content-length': str(1024 * 1024)})
# Download the object from OSS to the local memory.
result = bucket.get_object(key)
# Verify the data consistency between the content of the downloaded object and that of the object before upload.
content_got = b''
for chunk in result:
    content_got += chunk
assert content_got == content
# Download the object from OSS to the local memory.
result = bucket.get_object_to_file(key, filename)
# Verify the data consistency between the content of the downloaded object and that of the object before upload.
with open(filename, 'rb') as fileobj:
    assert fileobj.read() == content
```

- Perform client-side encryption in multipart upload

Note

- During multipart upload, if the upload task is interrupted and the process is exited, the context for multipart upload may be lost. If a multipart upload task is interrupted and you want to upload the object again, you must upload the entire object again.
- We recommend that you use the operation of resumable upload in OSS to upload large objects. This way, you can store the context of your multipart upload in the local client so that the context can be retained even if the upload task is interrupted.

The following code provides an example on how to use a CMK managed by KMS to encrypt an object to upload in multipart upload:

```

# -*- coding: utf-8 -*-
import os
import oss2
from oss2.crypto import RsaProvider
# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to a
ccess OSS because the account has permissions on all API operations. We recommend that yo
u use a RAM user to call API operations or perform routine O&M. To create a RAM user, log
on to the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
kms_provider=AliKMSProvider('yourAccessKeyId', 'yourAccessKeySecret', 'yourRegion', 'your
CMKID')
bucket = oss2.CryptoBucket(oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret'), 'yourEndp
oint', 'yourBucketName', crypto_provider = kms_provider)
"""
Multipart upload
"""
# Initialize the parts to upload.
part_a = b'a' * 1024 * 100
part_b = b'b' * 1024 * 100
part_c = b'c' * 1024 * 100
multi_content = [part_a, part_b, part_c]
parts = []
data_size = 100 * 1024 * 3
part_size = 100 * 1024
multi_key = "test_crypto_multipart"
# Initialize the context for the multipart upload when you use client-side encryption.
context = models.MultipartUploadCryptoContext(data_size, part_size)
res = bucket.init_multipart_upload(multi_key, upload_context=context)
upload_id = res.upload_id
# In this example, parts are uploaded in sequence. In practice, multipart upload supports
multi-threaded upload to speed up the upload.
for i in range(3):
    # The context value cannot be modified. If the values in context are modified, data u
pload fails.
    result = bucket.upload_part(multi_key, upload_id, i + 1, multi_content[i], upload_con
text=context)
    parts.append(oss2.models.PartInfo(i + 1, result.etag, size=part_size, part_crc=result
.crc))
# Complete the multipart upload task.
result = bucket.complete_multipart_upload(multi_key, upload_id, parts)
# Verify the data consistency between the content of the downloaded object and that of th
e object before upload.
result = bucket.get_object(multi_key)
content_got = b''
for chunk in result:
    content_got += chunk
assert content_got[0:102400] == part_a
assert content_got[102400:204800] == part_b
assert content_got[204800:307200] == part_c

```

- Perform client-side encryption in resumable upload

The following code provides an example on how to use a RSA-based CMK managed by yourself to encrypt objects to upload in resumable upload:


```

# -*- coding: utf-8 -*-
import os
import oss2
from oss2.crypto import RsaProvider
key = 'motto.txt'
content = b'a' * 1024 * 1024 * 100
file_name_put = 'upload.txt'
# Write the data included in content to the file.
with open(file_name_put, 'wb') as fileobj:
    fileobj.write(content)

# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# Create a bucket and use the RSA-based CMK to perform client-side encryption. This encryption method supports only uploads and downloads of entire objects.
# If you use OSS SDK for Python V2.9.0 or later, we recommend that you do not use LocalRsaProvider to initialize the bucket.
# bucket = oss2.CryptoBucket(auth, 'yourEndpoint', 'yourBucketName', crypto_provider=LocalRsaProvider())
# If you want only to decrypt objects, specify only the private key.
# key_pair = {'private_key': 'yourPrivateKey'}
# If you want only to encrypt objects, specify only the public key.
# key_pair = {'public_key': 'yourPublicKey'}
# If you want to encrypt and decrypt objects, you must specify both the public key and private key.
key_pair = {'private_key': 'yourPrivateKey', 'public_key': 'yourPublicKey'}
# Initializes the bucket. Set upload_contexts_flag to True, and you do not need to specify parameters for multipart upload context when you call the upload_part operation.
bucket = oss2.CryptoBucket(oss2.Auth(access_key_id, access_key_secret), endpoint, bucket_name,
                           crypto_provider=RsaProvider(key_pair))

# In this example, the value of multipart_threshold is set to 100*1024 for convenience. In practice, set the value of multipart_threshold based on your requirements. Default value: 10 MB.
# multipart_threshold indicates a threshold for object sizes. If the object size exceeds the threshold, multipart upload is used. If the object size is smaller than the threshold, we recommend that you use put_object to upload the object.
# part_size indicates the size of each part in multipart upload. Default value: 10 MB.
# num_threads indicates the number of concurrent upload threads. Default value: 1.
oss2.resumable_upload(bucket, key, file_name_put, multipart_threshold=10 * 1024 * 1024, part_size=1024 * 1024, num_threads=3)

```

- Perform client-side encryption in resumable download

The following code provides an example on how to use a CMK managed by KMS to decrypt objects to download in resumable download:

```
# -*- coding: utf-8 -*-
import os
import oss2
from oss2.crypto import RsaProvider
key = 'motto.txt'
content = b'a' * 1024 * 1024 * 100
file_name_get = 'download.txt'
kms_provider=AliKMSProvider('yourAccessKeyId', 'yourAccessKeySecret', 'yourRegion', 'yourCMKID')
bucket = oss2.CryptoBucket(oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret'), 'yourEndpoint', 'yourBucketName', crypto_provider = kms_provider)
# Download the objects by using resumable download.
oss2.resumable_download(bucket, key, file_name_get, multiget_threshold=10 * 1024 * 1024, part_size=1024 * 1024, num_threads=3)
# Verify the data consistency between the content of the downloaded object and that of the object before upload.
with open(file_name_get, 'rb') as fileobj:
    assert fileobj.read() == content
```

Client-side encryption V1 (not recommended)

Note

- Client-side encryption V1 supports only the upload of objects smaller than 5 GB by using PutObject. The multipart upload, resumable upload and resumable download operations are not supported.
- After you upload an object by using the operations of the CryptoBucket class, you cannot call operations such as CopyObject to modify the metadata related to client-side encryption. If the metadata is modified, the data may fail to be decrypted.
- Only OSS SDK for Python supports client-side encryption V1. OSS SDKs for other languages cannot decrypt data uploaded by using client-side encryption V1.
- OSS SDK for Python V2.11.0 and later supports client-side encryption V2. Client-side encryption V2 provides more features than client-side encryption V1. We recommend that you upgrade to the latest version.

- Object metadata related to client-side encryption

Parameter	Description	Required
x-oss-meta-oss-crypto-key	The encrypted data key. The encrypted data key is a string encrypted by using a RSA-based CMK and encoded in Base64.	Yes
x-oss-meta-oss-crypto-start	The initial value generated randomly for data encryption. The encrypted data key is a string encrypted by using a RSA-based CMK and encoded in Base64.	Yes
x-oss-meta-oss-cek-alg	The algorithm used to encrypt data.	Yes

Parameter	Description	Required
x-oss-meta-oss-wrap-alg	The algorithm used to encrypt the data key.	Yes
x-oss-meta-oss-matdesc	The description of the CEK in the JSON format. This parameter does not take effect currently.	No
x-oss-meta-unencrypted-content-length	The length of data before encryption. This parameter is not generated if Content-Length is not specified.	No
x-oss-meta-unencrypted-content-md5	The MD5 hash of data before encryption. This parameter is not generated if Content-MD5 is not specified.	No

- Use RSA-based CMKs managed by yourself to perform client-side encryption in object uploads and downloads

The following code provides an example on how to use a RSA-based CMK to encrypt an object to upload or decrypt an object to download:

```
# -*- coding: utf-8 -*-
import os
import oss2
from oss2.crypto import LocalRsaProvider

# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# Create a bucket and use the RSA-based CMK to perform client-side encryption. This encryption method supports only uploads and downloads of entire objects.
bucket = oss2.CryptoBucket(auth, 'yourEndpoint', 'yourBucketName', crypto_provider=LocalRsaProvider())
key = 'motto.txt'
content = b'a' * 1024 * 1024
filename = 'download.txt'
# Upload an object.
bucket.put_object(key, content, headers={'content-length': str(1024 * 1024)})
# Download the object from OSS to the local memory.
result = bucket.get_object(key)
# Verify the data consistency between the content of the downloaded object and that of the object before upload.
content_got = b''
for chunk in result:
    content_got += chunk
assert content_got == content
# Download the object from OSS to the local memory.
result = bucket.get_object_to_file(key, filename)
# Verify the data consistency between the content of the downloaded object and that of the object before upload.
with open(filename, 'rb') as fileobj:
    assert fileobj.read() == content
```

5.12.2. Server-side encryption

Configure server-side encryption for a bucket

```
# -*- coding: utf-8 -*-
import oss2
from oss2.models import ServerSideEncryptionRule
# The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in Object Storage Service (OSS) is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('<yourAccessKeyId>', '<yourAccessKeySecret>')
# In this example, the endpoint of the China (Hangzhou) region is used. Specify your actual endpoint.
bucket = oss2.Bucket(auth, 'http://oss-cn-hangzhou.aliyuncs.com', '<yourBucketName>')
# Create encryption configurations for the bucket. In this example, AES-256 encryption is used.
rule = ServerSideEncryptionRule()
rule.sse_algorithm = oss2.SERVER_SIDE_ENCRYPTION_AES256
# Specify the customer master key (CMK) ID. The CMK ID can be specified only when you use the OSS-KMS encryption method. If you want to use a specified CMK for encryption, enter the CMK ID. If you want to use the default CMK managed by Key Management Service (KMS) for encryption, leave this parameter empty. If you use AES-256 encryption, you must leave this parameter empty.
rule.kms_master_keyid = ""
# Apply the encryption configurations to the bucket.
result = bucket.put_bucket_encryption(rule)
# Display the returned HTTP status code.
print('http response code:', result.status)
```

Obtain the server-side encryption configurations of a bucket

```
# -*- coding: utf-8 -*-
import oss2
# The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('<yourAccessKeyId>', '<yourAccessKeySecret>')
# In this example, the endpoint of the China (Hangzhou) region is used. Specify your actual endpoint.
bucket = oss2.Bucket(auth, 'http://oss-cn-hangzhou.aliyuncs.com', '<yourBucketName>')
# Obtain the server-side encryption configurations of the bucket.
result = bucket.get_bucket_encryption()
# Display the obtained server-side encryption configurations.
print('sse_algorithm:', result.sse_algorithm)
print('kms_master_keyid:', result.kms_master_keyid) # If AES-256 encryption is used for the bucket, the value of kms_master_keyid is None.
```

Delete the server-side encryption configurations of a bucket

```
# -*- coding: utf-8 -*-
import oss2
# The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('<yourAccessKeyId>', '<yourAccessKeySecret>')
# In this example, the endpoint of the China (Hangzhou) region is used. Specify your actual endpoint.
bucket = oss2.Bucket(auth, 'http://oss-cn-hangzhou.aliyuncs.com', '<yourBucketName>')
# Delete the server-side encryption configurations of the bucket.
result = bucket.delete_bucket_encryption()
# Display the returned HTTP status code.
print('http status:', result.status)
```

5.13. Authorized access

This topic describes how to authorize temporary access to Object Storage Service (OSS) by using Security Token Service (STS) or a signed URL.

Use STS for temporary access authorization

You can use Alibaba Cloud STS to authorize temporary access to OSS. STS is a web service that provides temporary access tokens for users. You can use STS to grant a set of temporary access credentials that have a custom validity period and custom permissions to a third-party application or a RAM user managed by you. For more information about STS, see [What is STS?](#)

STS provides the following benefits:

- You need only to generate an access token and send the access token to a third-party application, instead of exposing your AccessKey pair to the third-party application. You can define the access permissions and validity period of this token.
- The token automatically expires after the validity period. Therefore, you do not need to manually revoke the access permissions of a token.

 **Note** For more information about how to use STS, see [Use a temporary credential provided by STS to access OSS](#) in OSS User Guide. You can call the [AssumeRole](#) operation or use STS SDKs for various programming languages to obtain temporary access credentials. For more information, see [STS SDK overview](#). The temporary access credentials consist of an AccessKey pair and a security token. The AccessKey pair contains an AccessKey ID and an AccessKey secret. The minimum validity period of temporary access credentials is 900 seconds. The maximum validity period of temporary access credentials is the maximum session duration specified for the current role. For more information, see [Specify the maximum session duration for a RAM role](#).

Run the `pip install aliyun-python-sdk-sts` command to install the official Security Token Service (STS) client for Python. For more information about the complete sample code, visit [GitHub](#).

Make sure that you use Object Storage Service (OSS) SDK for Python V2.0.6 or later. The following sample code shows how to use a temporary access credential obtained from STS to download an object:

```
# -*- coding: utf-8 -*-
from aliyunsdkcore import client
```

```

from aliyunsdksts.request.v20150401 import AssumeRoleRequest
import json
import oss2

# Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com.
endpoint = 'yourEndpoint'

# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine operations and management. To create a RAM user, log on to the RAM console.
access_key_id = 'yourAccessKeyId'
access_key_secret = 'yourAccessKeySecret'

# Specify the name of the bucket. Example: examplebucket.
bucket_name = 'examplebucket'

# Specify the full path of the object. Example: exampledir/exampleobject.txt. The full path of the object cannot contain the bucket name.
object_name = 'exampledir/exampleobject.txt'

# To obtain the Alibaba Cloud Resource Name (ARN) information of the RAM role, log on to the RAM console. In the left-side navigation pane, choose Identities > Roles. On the Roles page, search for and click the created RAM role. In the Basic Information section of the role details page, you can view and copy the ARN information.
# Specify the ARN information of the RAM role. Format: acs:ram::$accountID:role/$roleName.
# $accountID specifies the Alibaba Cloud account ID. To view the Alibaba Cloud account ID, perform the following steps: Log on to the OSS console, and move the pointer over the profile picture in the upper-right corner to view and copy the account ID, or click Basic Information to view the account ID.
# $roleName specifies the name of the RAM role. To view the RAM role name, perform the following steps: Log on to the RAM console. In the left-side navigation pane, click Roles. On the Roles page, view the name in the Role Name column.
role_arn = 'acs:ram::17464958*****:role/ossststest'

# Create a RAM policy.
# The policy specifies that GetObject operations can be performed on resources only in the bucket named examplebucket.
policy_text = '{"Version": "1", "Statement": [{"Action": ["oss:GetObject"], "Effect": "Allow", "Resource": ["acs:oss:*:*:examplebucket/*"]}]}
'
clt = client.AcsClient(access_key_id, access_key_secret, 'cn-hangzhou')
req = AssumeRoleRequest.AssumeRoleRequest()
# Set the format of the returned value to JSON.
req.set_accept_format('json')
req.set_RoleArn(role_arn)
# Specify a custom role session name to distinguish different tokens. Example: session-test.
req.set_RoleSessionName('session-test')
req.set_Policy(policy_text)
body = clt.do_action_with_exception(req)
# Use the AccessKey pair of the RAM user to apply for a temporary access credential from STS.
token = json.loads(oss2.to_unicode(body))
# Initialize the StsAuth instance based on the authentication information in the temporary access credential.
auth = oss2.StsAuth(token['Credentials']['AccessKeyId'],
                    token['Credentials']['AccessKeySecret'],
                    token['Credentials']['SecurityToken'])

```

```
# Initialize a bucket based on the StsAuth instance.
bucket = oss2.Bucket(auth, endpoint, bucket_name)
# Download the object from the bucket.
read_obj = bucket.get_object(object_name)
print(read_obj.read())
```

Use a signed URL for temporary access authorization

You can generate a signed URL and provide the URL to a visitor for temporary access. When you generate a signed URL, you can specify the validity period of the URL to limit the period of access from visitors. By default, the validity period of a signed URL is 3,600 seconds. The maximum validity period of a signed URL is 32,400 seconds.

You can add signature information to a URL and provide the URL to a third-party user for authorized access. For more information, see [Add signatures to URLs](#).

 **Notice** A validity period must be set for an STS temporary access credential and a signed URL. When you use an STS temporary account credential to generate a signed URL that is used to perform operations such as object upload and download, the minimum validity period takes precedence. For example, you can set the validity period of your STS temporary access credential to 1,200 seconds and the validity period of your signed URL generated by using this credential to 3,600 seconds. In this case, the signed URL cannot be used to upload objects after the STS temporary access credential expires, even if the signed URL is within its validity period.

- Use a signed URL to upload an object

The following sample code shows how to use a signed URL to upload an object with temporary authorization:

```
# -*- coding: utf-8 -*-
import oss2

# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine operations and management. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com.
# Specify the name of the bucket. Example: examplebucket.
bucket = oss2.Bucket(auth, 'yourEndpoint', 'examplebucket')
# Specify the full path of the object. Example: exampledir/exampleobject.txt. The full path of the object cannot contain the bucket name.
object_name = 'exampledir/exampleobject.txt'
# Generate a signed URL that is used to upload the object. The validity period of the URL is 60 seconds.
# By default, OSS identifies the forward slashes (/) in the full path of an object as escape characters when the signed URL is generated. Therefore, you cannot directly use the signed URL.
# Set the slash_safe parameter to True. This way, OSS does not identify the forward slashes (/) in the full path of the object as escape characters. Then, you can directly use the generated signed URL.
url = bucket.sign_url('PUT', object_name, 60, slash_safe=True)
print('the address of the signed URL:', url)
# Use the signed URL to upload an object.
# Specify the full path of the local file that you want to upload. Example: D:\\localpath\\examplefile.txt.
# By default, if you set this parameter to the name of a local file such as examplefile.txt without specifying the local path, the local file is uploaded from the local path of the project to which the sample program belongs.
result = bucket.put_object_with_url_from_file(url, 'D:\\localpath\\examplefile.txt')
print(result.status)
```

- Use a signed URL to download an object

The following sample code shows how to use a signed URL to download an object with temporary authorization:


```
# -*- coding: utf-8 -*-
import oss2
# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine operations and management. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com.
# Specify the name of the bucket. Example: examplebucket.
bucket = oss2.Bucket(auth, 'yourEndpoint', 'examplebucket')
# Specify the full path of the object. Example: exampledir/exampleobject.txt. The full path of the object cannot contain the bucket name.
object_name = 'exampledir/exampleobject.txt'
# Generate a signed URL that is used to download the object. The validity period of the URL is 60 seconds.
# By default, OSS identifies the forward slashes (/) in the full path of an object as escape characters when the signed URL is generated. Therefore, you cannot directly use the signed URL.
# Set the slash_safe parameter to True. This way, OSS does not identify the forward slashes (/) in the full path of the object as escape characters. Then, you can directly use the generated signed URL.
url = bucket.sign_url('GET', object_name, 60, slash_safe=True)
print('the address of the signed URL:', url)
# Use the signed URL to download the object to the local path.
# Specify the full path of the local file that is downloaded from OSS. Example: D:\\localpath\\examplefile.txt.
# By default, if you set this parameter to the name of a local file such as examplefile.txt without specifying the local path, the downloaded object is saved to the local path of the project to which the sample program belongs.
result = bucket.get_object_with_url_to_file(url, 'D:\\localpath\\examplefile.txt')
print(result.read())
```

5.14. Data security

Object Storage Service (OSS) SDK for Java uses MD5 verification and CRC-64 to ensure data integrity when you upload, download, and copy objects.

MD5 verification

If you configure Content-MD5 in an object upload request, OSS calculates the MD5 hash of the uploaded object. If the calculated MD5 hash is different from the MD5 hash configured in the upload request, `InvalidDigest` is returned. This allows OSS to ensure data integrity for object upload. If `InvalidDigest` is returned, you must upload the object again.

The following code provides an example on how to perform MD5 verification for object upload operations:

```
# -*- coding: utf-8 -*-
import oss2
# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log
on to OSS because the account has permissions on all API operations. We recommend that you
use your RAM user's credentials to call API operations or perform routine operations and ma
intenance. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('<yourAccessKeyId>', '<yourAccessKeySecret>')
# The endpoint of the China (Hangzhou) region is used in this example. Specify the actual e
ndpoint.
bucket = oss2.Bucket(auth, 'http://oss-cn-hangzhou.aliyuncs.com', '<yourBucketName>')
object_name = '<yourObjectName>'
content = '<yourContent>'
# Calculate the MD5 hash of the uploaded content.
content_md5 = oss2.utils.content_md5(content);
print('content_md5', content_md5)
# If the upload request contains the Content-MD5 header, the OSS server verifies the MD5 ha
sh of the uploaded content.
headers = dict()
headers['Content-MD5'] = content_md5
bucket.put_object(object_name, content, headers=headers)
```

 **Note** MD5 verification can be used for `put_object`, `append_object`, `post_object`, and `upload_part`.

CRC-64

 Note

- CRC-64 can be used for `put_object`, `get_object`, `append_object`, and `upload_part`. By default, CRC-64 is enabled when you upload an object. If the CRC-64 value calculated on the client is different from the CRC-64 value returned by the OSS server, `InconsistentError` is returned.
- CRC-64 is not supported in range download.
- CRC-64 consumes CPU resources and slows down uploads and downloads.

- CRC-64 in object download

The following code provides an example on how to perform CRC-64 in object download:

```
# -*- coding: utf-8 -*-
import oss2
# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log on to OSS because the account has permissions on all API operations. We recommend that you use your RAM user's credentials to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('<yourAccessKeyId>', '<yourAccessKeySecret>')
# The endpoint of the China (Hangzhou) region is used in this example. Specify the actual endpoint.
bucket = oss2.Bucket(auth, 'http://oss-cn-hangzhou.aliyuncs.com', '<yourBucketName>')
object_name = '<yourObjectName>'
# Check whether CRC-64 is enabled by default.
print('bucket.enable-crc:', bucket.enable_crc)
# The return value of bucket.get_object is a file-like and iterable object.
object_stream = bucket.get_object(object_name)
print(object_stream.read())
# You must call read() to read the object from the stream returned by get_object before you can calculate the CRC-64 value of the object.
if object_stream.client_crc != object_stream.server_crc:
    print "The CRC checksum between client and server is inconsistent!"
```

- CRC-64 in append upload

If you set `init_crc` in append upload, CRC-64 is enabled by default.

```
# -*- coding: utf-8 -*-
import oss2
# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log on to OSS because the account has permissions on all API operations. We recommend that you use your RAM user's credentials to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('<yourAccessKeyId>', '<yourAccessKeySecret>')
# The endpoint of the China (Hangzhou) region is used in this example. Specify the actual endpoint.
bucket = oss2.Bucket(auth, 'http://oss-cn-hangzhou.aliyuncs.com', '<yourBucketName>')
object_name = "<yourAppendObjectName>"
first_content = "<yourFirstContent>"
second_content = "<yourSecondContent>"
# Set init_crc in the first append.
# When init_crc is set, OSS SDK for Python performs CRC-64 for the returned result.
result = bucket.append_object(object_name, 0, first_content, init_crc=0)
# Set init_crc in the second append.
# Set init_crc to the CRC-64 value of the uploaded object.
result = bucket.append_object(object_name, result.next_position, second_content, init_crc=result.crc)
```

5.15. Enable log recording

To facilitate troubleshooting, OSS SDK for Python provides the log recording feature. By default, log recording is disabled.

 **Note** To enable log recording for OSS SDK for Python, use OSS SDK V2.6.x and later.

The log recording feature provided by OSS SDK for Python can be used to collect log information of operations in OSS. The log information is recorded in log files stored in your computer.

- Log format: `<time><name><level><threadId><message>`
- Log levels in ascending order: CRITICAL, ERROR, WARNING, INFO, DEBUG, and NOTSET

 **Note** After a log level is specified, a local log file records only logs whose levels are equal to or higher than the specified level. For example, if you set the log level to INFO, the log file records logs whose levels are CRITICAL, ERROR, WARNING, and INFO.

Enable log recording

The following code provides an example on how to enable log recording for OSS SDK for Python:

```
# -*- coding: utf-8 -*-
import os
import logging
import oss2
from itertools import islice

# Download log information to a local log file, and store the log file in the specified local path.
# By default, if you specify the name of a local file such as examplelogfile.log without specifying the local path, the local file is saved to the local path of the project to which the sample program belongs.
log_file_path = "D:\\localpath\\examplelogfile.log"
# Enable log recording.
oss2.set_file_logger(log_file_path, 'oss2', logging.INFO)
# Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
auth = oss2.Auth('yourAccessKeyId', 'yourAccessKeySecret')
# Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if your bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com.
# Specify the name of the bucket. Example: examplebucket.
bucket = oss2.Bucket(auth, 'yourEndpoint', 'examplebucket')
# Traverse objects and directories.
for b in islice(oss2.ObjectIterator(bucket), 10):
    print(b.key)
# Obtain the metadata of the object.
# Specify the full path of the object. Example: exampledir/exampleobject.txt. The full path of the object cannot contain bucket names.
object_meta = bucket.get_object_meta('exampledir/exampleobject.txt')
```

An example of log information

 **Note** To obtain more log details, you can change the log level to DEBUG.

```
2018-11-20 16:37:46,437 oss2.api [INFO] 26716 : Exception: {
  'status': 404,
  'x-oss-request-id': '5BF3C7DA236B3A201CE64679',
  'details': {
    'HostId': 'examplebucket.oss-cn-shenzhen.aliyuncs.com',
    'Message': 'The specified key does not exist.',
    'Code': 'NoSuchKey',
    'RequestId': '5BF3C7DA236B3A201CE64679',
    'Key': 'exemplerdir/exampleobject.txt'
  }
}
```

The log information shows that all objects in a bucket are traversed if the bucket is initialized. If the requested object does not exist, an error occurs. The error information is `NoSuchKey`.

5.16. IMG

Image Processing (IMG) provided by Object Storage Service (OSS) is a secure, cost-effective, and highly reliable image processing service that can process large amounts of data. After you upload source images to OSS, you can call RESTful APIs to process the images anytime, anywhere, and on all Internet devices.

For more information about IMG parameters, see [Parameters](#).

For more information about the complete sample code, visit [GitHub](#).

Use a single IMG parameter to process images

- The following code provides an example on how to use a single IMG parameter to process an image and save the image to your computer:

```
# -*- coding: utf-8 -*-
import os
import oss2
# In this example, the endpoint of the China (Hangzhou) region is used. Specify your actual endpoint.
endpoint = 'https://oss-cn-hangzhou.aliyuncs.com'
# The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to access OSS is a high-risk operation. We recommend that you use a Resource Access Management (RAM) user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
access_key_id = '<yourAccessKeyId>'
access_key_secret = '<yourAccessKeySecret>'
# The name of the bucket in which the source image is stored.
bucket_name = '<yourBucketName>'
# The name of the source image. If the image is not stored in the root directory of the bucket, you must add the access path of the image. Example: example/example.jpg.
key = '<yourObjectName>'
# Specify the name of the processed image.
new_pic = '<LocalFileName>'
# Specify the bucket instance. You must use the bucket instance to call all object-related methods.
bucket = oss2.Bucket(oss2.Auth(access_key_id, access_key_secret), endpoint, bucket_name)
# If the image does not exist in the specified bucket, you must upload the image to the bucket.
# bucket.put_object_from_file(key, '<yourLocalFile>')
# After you resize the image to a height and width of 100 pixels, save the image to your computer.
style = 'image/resize,m_fixed,w_100,h_100'
bucket.get_object_to_file(key, new_pic, process=style)
# After the IMG is complete, you can delete the source image from the bucket if you no longer use the image.
# bucket.delete_object(key)
# If you no longer use the processed image, you can delete the processed image.
# os.remove(new_pic)
```

- Use multiple IMG parameters to process an image and save the image to your computer

The following code provides an example on how to use multiple IMG parameters to process an image. Separate these parameters with forward slashes (/).

```
# -*- coding: utf-8 -*-
import os
import oss2
# Specify the endpoint of the region in which the bucket is located. For example, if the
bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hang
zhou.aliyuncs.com.
endpoint = 'https://oss-cn-hangzhou.aliyuncs.com'
# The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. U
sing these credentials to access OSS is a high-risk operation. We recommend that you use
a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to
the RAM console.
access_key_id = 'yourAccessKeyId'
access_key_secret = 'yourAccessKeySecret'
# Specify the name of the bucket in which the source image is stored. Example: examplebuc
ket.
bucket_name = 'examplebucket'
# Specify the name of the source image. If the image is not stored in the root directory
of the bucket, you must add the full path of the image. Example: exampledir/example.jpg.
key = 'exampledir/example.jpg'
# Specify the name of the processed image.
new_pic = 'exampledir/newexample.jpg'
# Specify the bucket instance. You must use the bucket instance to call all object-relate
d methods.
bucket = oss2.Bucket(oss2.Auth(access_key_id, access_key_secret), endpoint, bucket_name)
# If the image is not stored in the specified bucket, you must upload the image from the
local path to the specified bucket.
# bucket.put_object_from_file(key, 'D:\\\\localpath\\\\example.jpg')
# After you resize the image to a height and width of 100 pixels, rotate the image 90 deg
rees, and save the image to your computer.
style = 'image/resize,m_fixed,w_100,h_100/rotate,90'
bucket.get_object_to_file(key, new_pic, process=style)
# After the IMG is complete, you can delete the source image from the bucket if you no lo
nger use the image.
# bucket.delete_object(key)
# If you no longer use the processed image, you can delete it.
# os.remove(new_pic)
```

Use an image style to process images

You can encapsulate multiple IMG parameters in a style, and then use the style to concurrently process multiple images. For more information, see [Image styles](#). The following code provides an example on how to use an image style to process an image:

```
# -*- coding: utf-8 -*-
import os
import oss2

# Specify the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com.
endpoint = 'https://oss-cn-hangzhou.aliyuncs.com'

# The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to access OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
access_key_id = 'yourAccessKeyId'
access_key_secret = 'yourAccessKeySecret'

# Specify the name of the bucket in which the source image is stored. Example: examplebucket.
bucket_name = 'examplebucket'

# Specify the name of the source image. If the image is not stored in the root directory of the bucket, you must add the full path of the image. Example: exampledir/example.jpg.
key = 'exampledir/example.jpg'

# Specify the name of the processed image.
new_pic = 'exampledir/newexample.jpg'

# Specify the bucket instance. You must use the bucket instance to call all object-related methods.
bucket = oss2.Bucket(oss2.Auth(access_key_id, access_key_secret), endpoint, bucket_name)

# If the image is not stored in the specified bucket, you must upload the image from the local path to the specified bucket.
# bucket.put_object_from_file(key, 'D:\\\\localpath\\\\example.jpg')

# Use the custom style to process the image. Set yourCustomStyleName to the name of the image style that you create in the OSS console.
style = 'yourCustomStyleName'

# Save the processed image to your computer.
bucket.get_object_to_file(key, new_pic, process=style)

# After the IMG is complete, you can delete the source image from the bucket if you no longer use the image.
# bucket.delete_object(key)

# If you no longer use the processed image, you can delete it.
# os.remove(new_pic)
```

Save processed images

You can call the `ImgSaveAs` operation to save the processed images to a specified bucket. The following code provides an example on how to save a processed image:


```

# -*- coding: utf-8 -*-
import os
import base64
import oss2
# Specify the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com.
endpoint = 'https://oss-cn-hangzhou.aliyuncs.com'
# The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to access OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
access_key_id = 'yourAccessKeyId'
access_key_secret = 'yourAccessKeySecret'
# Specify the name of the bucket in which the source image is stored.
source_bucket_name = 'srcbucket'
# Specify the name of the bucket to which the processed image is saved. The bucket must be within the same region as the bucket that stores the source image.
target_bucket_name = 'destbucket'
# Specify the name of the source image. If the image is not stored in the root directory of the bucket, you must add the full path of the image. Example: example/example.jpg.
source_image_name = 'example/example.jpg'
# Specify the bucket instance. You must use the bucket instance to call all object-related methods.
bucket = oss2.Bucket(oss2.Auth(access_key_id, access_key_secret), endpoint, source_bucket_name)
# Resize the image to a height and width of 100 pixels.
style = 'image/resize,m_fixed,w_100,h_100'
# Specify the name of the processed image. If the image is not stored in the root directory of the bucket, you must add the full path of the image. Example: exampledir/example.jpg.
target_image_name = 'exampledir/example.jpg'
process = "{0}|sys/saveas,o_{1},b_{2}".format(style,
    oss2.compat.to_string(base64.urlsafe_b64encode(oss2.compat.to_bytes(target_image_name))),
    oss2.compat.to_string(base64.urlsafe_b64encode(oss2.compat.to_bytes(target_bucket_name)))
)
result = bucket.process_object(source_image_name, process)
print(result)

```

Generate a signed object URL that includes IMG parameters

URLs of private objects must be signed. IMG parameters cannot be added to the end of a signed URL. If you want to process a private object, add IMG parameters to the signature. The following code provides an example on how to add IMG parameters to a signature:

```
# -*- coding: utf-8 -*-
import oss2
# Specify the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com.
endpoint = 'https://oss-cn-hangzhou.aliyuncs.com'
# The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to access OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
access_key_id = 'yourAccessKeyId'
access_key_secret = 'yourAccessKeySecret'
# Specify the bucket name. Example: examplebucket.
bucket_name = 'examplebucket'
# Specify the name of the bucket in which the source image is stored. If the image is not stored in the root directory of the bucket, you must add the full path of the image. Example: exampledir/example.jpg.
key = 'exampledir/example.jpg'
# Specify the bucket instance. You must use the bucket instance to call all object-related methods.
bucket = oss2.Bucket(oss2.Auth(access_key_id, access_key_secret), endpoint, bucket_name)
# If the image does not exist in the specified bucket, you must upload the image to the bucket.
# bucket.put_object_from_file(key, 'D:\\localpath\\example.jpg')
# After you resize the image to a height and width of 100 pixels, rotate the image 90 degrees.
style = 'image/resize,m_fixed,w_100,h_100/rotate,90'
# Generate a signed object URL that includes IMG parameters. Set the validity period to 10 minutes. Unit: seconds.
url = bucket.sign_url('GET', key, 10 * 60, params={'x-oss-process': style})
print(url)
```

5.17. Chinese character and time

This topic describes the Chinese characters and time used by OSS Python SDK.

Chinese character

Errors may occur if Chinese characters are used in the Python code. Therefore, you must declare the Chinese character encoding at the beginning of the code. For example:

```
# -*- coding: utf-8 -*-
```

- Data type

Python 2.x supports the following two data types:

Data type	Description
str	Character strings, corresponding to the bytes type in Python 3.x

Data type	Description
unicode	Unicode streams, of which the length is measured in characters. For example, the length of <code>u'中文'</code> is 2.

Python 3.x supports the following two data types:

Data type	Description
str	Character strings, corresponding to the unicode type in Python 2.x
bytes	Byte streams, of which the length is measured in bytes. For example, the length of <code>b'中文'</code> depends on the encoding, which is 6 if it is encoded with UTF-8.

- Input and output conventions

The following table describes the conventions of input parameter types for OSS Python SDK.

Input	Data type	Remarks
OSS object name	str	If the input type is bytes, it must be encoded with UTF-8.
Local file name	str, unicode	If the input type is bytes, it must be encoded with UTF-8, such as the <code>yourLocalFile</code> parameter in <code>bucket.get_object_to_file</code> .
Input data stream	bytes	For example, the data parameter in <code>bucket.put_object</code> .

The following table describes the conventions of output parameter types for OSS Python SDK.

Output	Data type	Remarks
Results parsed from the XML	str	For example, a string in the result obtained by <code>bucket.list_object</code> .
Downloaded content	bytes	Python SDK regards that the bytes type is encoded with UTF-8 by default. Therefore, make sure that the Python source file is also encoded with UTF-8.

- Type conversion functions

OSS Python SDK provides three functions for type conversion.

Function	Description
to_bytes	- Converts the unicode type to the str type in Python 2.x. Other types are returned without changes. - Converts the str type to the bytes type in Python 3.x. Other types are returned without changes.
to_unicode	- Converts the str type to the unicode type in Python 2.x. Other types are returned without changes. - Converts the bytes type to the str type in Python 3.x. Other types are returned without changes.
to_string	Acts as to_bytes in Python 2.x. Acts as to_unicode in Python 3.x.

Time

OSS Python SDK converts all timestamp strings retrieved from the server to **Unix Time**, that is, the number of seconds since UTC 00:00 on January 1, 1970. For example, the last_modified time in the results returned by bucket.get_object is the Unix Time of the int type.

You can use the datetime.datetime.fromtimestamp() method to convert the time, obtaining the timestamp string.

5.18. Exception handling

OSS Python SDK has three types of exceptions: ClientError, RequestError, and ServerError, which are defined in the oss2.exceptions sub-module.

The following table lists the member variables of exceptions.

Variable	Type	Description
status	int	• ServerError: HTTP status code • ClientError and RequestError: fixed values
request_id	str	• ServerError: request ID returned by the OSS server • ClientError and RequestError: empty string
Code and message	str	Correspond to text text in the Code and Message XML Tags in OSS error response .

Example for handling exceptions

The following code demonstrates how to handle exceptions when downloading an object that does not exist, and how to print the HTTP status code and request ID of the error messages.

```
# -*- coding: utf-8 -*-
import oss2
# It is highly risky to log on with AccessKey of an Alibaba Cloud account because the account has permissions on all the APIs in OSS. We recommend that you log on as a RAM user to access APIs or perform routine operations and maintenance. To create a RAM account, log on to https://ram.console.aliyun.com.
auth = oss2.Auth('<yourAccessKeyId>', '<yourAccessKeySecret>')
# This example uses endpoint China East 1 (Hangzhou). Specify the actual endpoint based on your requirements.
bucket = oss2.Bucket(auth, 'http://oss-cn-hangzhou.aliyuncs.com', '<yourBucketName>')
try:
    stream = bucket.get_object('random-key.txt')
except oss2.exceptions.NoSuchKey as e:
    print('status={0}, request_id={1}'.format(e.status, e.request_id))
```

ClientError

ClientError is caused by input error on the client side. For example, a ClientError is thrown when an empty object name list is received when the bucket.batch_delete_objects method is used. The status value of ClientError objects is oss2.exceptions.OSS_CLIENT_ERROR_STATUS.

RequestError

When an exception is thrown by the underlying HTTP library, the Python SDK will convert the exception to RequestError. The status value of ClientError objects is fixed to oss2.exceptions.OSS_CLIENT_ERROR_STATUS.

ServerError

When an HTTP error code is returned by the OSS server, Python SDK converts it into a ServerError. ServerError is divided into multiple subclasses based on the HTTP status code and OSS error code. The NotFound subclass corresponds to all 404 exceptions, and the Conflict subclass corresponds to all 409 exceptions.

The following table lists common error codes.

Exception class	HTTP status code	OSS error code	Description
NotModified	304	Null	It is not modified.
AccessDenied	403	AccessDenied	The access is denied.
NoSuchBucket	404	NoSuchBucket	The bucket does not exist.
NoSuchKey	404	NoSuchKey	The file does not exist.
NoSuchUpload	404	NoSuchUpload	The multipart upload does not exist.

Exception class	HTTP status code	OSS error code	Description
NoSuchWebsite	404	NoSuchWebsiteConfiguration	The static website hosting is not configured.
NoSuchLifecycle	404	NoSuchLifecycle	The lifecycle management is not configured.
NoSuchCors	404	NoSuchCORSConfiguration	CORS is not configured.
BucketNotEmpty	409	BucketNotEmpty	The bucket is not empty.
PositionNotEqualToLength	409	PositionNotEqualToLength	The Append position is not equal to the object length.
ObjectNotAppendable	409	ObjectNotAppendable	The object is not appendable.

5.19. FAQ

This topic describes the causes and solutions for common errors you may encounter when you use OSS SDK for Python.

Multipart upload tasks initiated by OSS SDK for Python failed.

Try the following solutions if parts failed to upload by using OSS SDK for Python:

- First, determine whether the object is directly uploaded to OSS or uploaded to OSS through a proxy such as CDN. If the object is uploaded to OSS through CDN, the ETag of the object must be exposed and the following headers must be configured in the request: `Access-Control-Allow-Origin`, `Access-Control-Allow-Methods`, and `Access-Control-Allow-Headers`. For more information, see [PutBucket cors](#).
- If the upload failed because of timeout, we recommend that you use resumable upload to upload the object. By using resumable upload, you can split the objects into parts of a custom size and then upload the parts concurrently. When exceptions occur, view and analyze the error message returned by OSS SDK for Python.
- Delete the parts generated by the failed upload task and upload the object again.

If the problem persists, submit a ticket that includes the following information:

- The request ID included in the exception returned by OSS SDK.
- The packets captured by running the following tcpdump command when you upload the object again.

```
tcpdump -i <Outbound port name> -s0 host <OSS endpoint> -w faild.pcap
```

The upload and download speeds of ossutil are far better than those of OSS SDK for Python.

- Cause

ossutil is developed based on OSS SDK for Go and provides better performance in concurrent upload scenarios. If the download speed of OSS SDK for Python is far slower than that of ossutil, it may be because that crcmod is not properly installed.

- Solution

For more information about how to install crcmode, see [Installation](#). If the problem persists, submit a ticket.

The multipart upload function provided by OSS SDK for Python can be called in CentOS systems. However, a 403 error is returned when the multipart upload function is called on Ubuntu systems.

Use tcpdump to capture packets on the client. Analyze the packets to determine whether the calculated signature is different from that on the OSS server because of incorrect header information in the request.

```

POST /ttsservice%2Fpasswd? uploadId=D468E486D1D94D90A1AB8885A4E32AE0 HTTP/1.1
Host: rokid.oss-cn-hangzhou.aliyuncs.com
Accept-Encoding: identity
Accept: text/html
Content-Length: 137
date: Sat, 29 Dec 2018 07:32:34 GMT
authorization: OSS LTAIknFr:r2KPR0y4E0G5tnU/MYdcvXHP****
Content-Type: application/x-www-form-urlencoded
User-Agent: aliyun-sdk-python/2.6.0(Linux/4.4.0-31-generic/x86_64;3.4.3)
<CompleteMultipartUpload><Part><PartNumber>1</PartNumber><ETag>"3195544E19D99658706D5****"<
/ETag></Part></CompleteMultipartUpload>HTTP/1.1 403 Forbidden
Server: AliyunOSS
Date: Sat, 29 Dec 2018 07:33:43 GMT
Content-Type: application/xml
Content-Length: 1122
Connection: keep-alive
x-oss-request-id: 5C2723573183****
x-oss-server-time: 0
<? xml version="1.0" encoding="UTF-8"? >
<Error>
  <Code>SignatureDoesNotMatch</Code>
  <Message>The request signature we calculated does not match the signature you provided. Check your key and signing method. </Message>
  <RequestId> 5C2723573183A12D </RequestId>
  <HostId>rokid.oss-cn-hangzhou.aliyuncs.com</HostId>
  <OSSAccessKeyId> LTAXXX </OSSAccessKeyId>
  <SignatureProvided>r2KPR0y4E0G5tnU/MYdc****</SignatureProvided>
  <StringToSign>POST
application/x-www-form-urlencoded
Sat, 29 Dec 2018 07:32:34 GMT
/rokid/ttsservice/passwd? uploadId=D468E486D1D94D90A1AB8885A4E3****</StringToSign>
  <StringToSignBytes>50 4F 53 54 0A 0A 61 70 70 6C 69 63 61 74 69 6F 6E 2F 78 2D 77 77 77 2D
66 6F 72 6D 2D 75 72 6C 65 6E 63 6F 64 65 64 0A 53 61 74 2C 20 32 39 20 44 65 63 20 32 30 3
1 38 20 30 37 3A 33 32 3A 33 34 20 47 4D 54 0A 2F 72 6F 6B 69 64 2D 6F 70 73 2D 6D 6F 64 65
6C 2F 74 74 73 73 65 72 76 69 63 65 2F 70 61 73 73 77 64 3F 75 70 6C 6F 61 64 49 64 3D 44 3
4 36 38 45 34 38 36 44 31 44 39 34 44 39 30 41 31 41 42 38 38 38 35 41 34 45 33 32 41 45 30
</StringToSignBytes>
</Error>

```

If the signature received by the server and the signature calculated on the client are inconsistent, the content of the request is modified. We recommend that you upload the object by using HTTPS.

The preceding code shows the packets captured by tcpdump on the client. Use the captured header information in the following script to calculate the signature and check whether the calculated signature is consistent with the signature in the captured packets.

```

import base64
import hmac
import sha

mac = hmac.new("<Secretkey>", "POST\n\napplication/x-www-form-urlencoded\nSat, 29 Dec 2018 0
7:32:34 GMT\n/rokid/ttsservice/passwd? uploadId=D468E486D1D94D90A1AB8885A4E3****", sha)
Signature = base64.b64encode(mac.digest())
print(Signature)

```


If the signatures are consistent, it indicates that the signature in the request generated by OSS SDK is correct. If the calculated signature is not consistent with the signature in the captured packets, it may be because that the MD5 hash calculated based on the signature is different in Ubuntu systems due to compatibility issues.

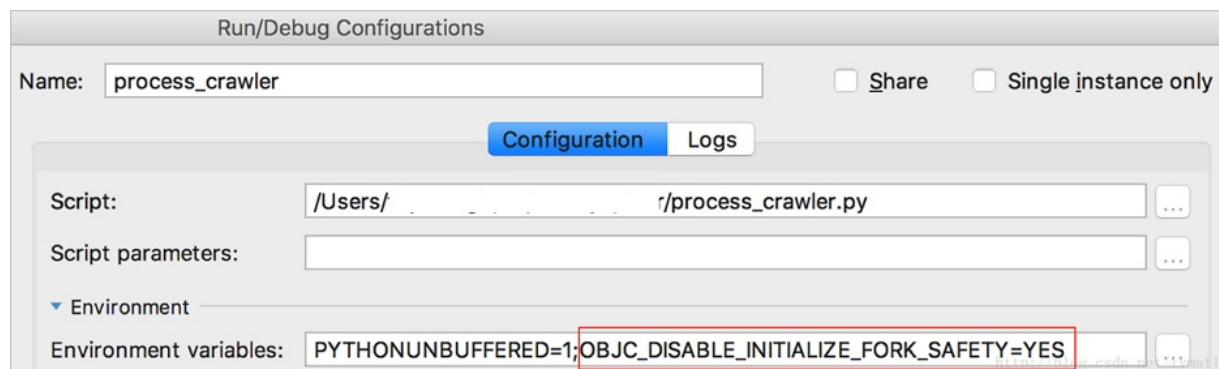
When OSS SDK for Python is used in macOS systems to start multiple threads and perform operations on OSS resources in a subthread, an error occurs if tensorflow is imported. If tensorflow is not imported, no errors occur. In addition, tensorflow can be imported without errors if you do not start multiple threads.

When you use OSS SDK for Python to start multiple threads, the following error occurs:

```
objc[2483]: +[__NSPlaceholderDate initialize] may have been in progress in another thread w
hen fork() was called.
objc[2483]: +[__NSPlaceholderDate initialize] may have been in progress in another thread w
hen fork() was called. We cannot safely call it or ignore it in the fork() child process. C
rashing instead. Set a breakpoint on objc_initializeAfterForkError to debug.
```

Solution

Add an environment variable `OBJC_DISABLE_INITIALIZE_FORK_SAFETY=YES`. The following figure shows how to add the environment variable.



For more information, visit [Workarounds for compatibility](#).

Add retry policies.

OSS SDK for Python does not provide retry mechanisms. Requests may fail in poor network conditions. In this case, we recommend that you add retry policies to your code. The following code provides an example on how to add a retry policy:

```
def test_get_object():
    MAX_RETRIES = 3
    retry_count = 0
    while True:
        try:
            retry_count += 1
            # yourObjectName indicates the complete path of the object excluding the bucket
            # name. Example: abc/efg/example.jpg.
            # yourFileName indicates the complete path to which you want to download the ob
            # ject. Example: /users/local/example.jpg.
            bucket.get_object_to_file("yourObjectName", "yourFileName")
            break
        except Exception:
            if retry_count >= MAX_RETRIES:
                raise
```

6.Browser.js

6.1. Installation

This topic describes how to install Object Storage Service (OSS) SDK for Browser.js.

Preparations

- A Resource Access Management (RAM) user or a Security Token Service (STS) credential is created to access OSS.

The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Therefore, we recommend that you follow the best practices of Alibaba Cloud for security. If you deploy your service on the server, you can use a RAM user or an STS credential to call API operations or perform routine O&M. If you deploy your service on the client, use an STS credential to call API operations. For more information, see [Overview](#).

- Cross-origin resource sharing (CORS) rules are configured.

For more information, see [Configure CORS rules](#).

To access OSS by using a browser, configure the CORS rule based on the following requirements:

- **Sources:** Specify an accurate domain name, such as `www.aliyun.com` or a domain name that contains an asterisk (*) as the wildcard, such as `*.aliyun.com`.
- **Allowed Methods:** Select the methods based on your requirements. For example, you can select **PUT** to perform multipart upload and select **DELETE** to delete objects.
- **Allowed Headers:** Set the value to `*`.
- **Exposed Headers:** Set the values to `ETag`, `x-oss-request-id`, and `x-oss-version-id`.

Create Rule

Sources *

www.aliyun.com

You can set multiple sources. Each line can contain one source and up to one wildcard (*).

Allowed Methods *

☐ GET ☐ POST ☒ PUT ☐ DELETE ☐ HEAD

Allowed Headers

*

You can set multiple allowed headers. Each line can contain one header and up to one wildcard (*).

Exposed Headers

Etag
x-oss-request-id
x-oss-version-id

You can set multiple exposed headers. Each line can contain one header and cannot contain wildcards (*).

Cache Timeout (Seconds)

0

Vary: Origin

☐

Configure whether to return the Vary: Origin header. If both CORS and non-CORS requests are sent to OSS, select this option to avoid errors. If Vary: Origin is selected, access through the browser or CDN back-to-origin requests may increase. [Learn more](#).

OK

Cancel

Usage notes

OSS SDK for Browser.js uses [Browserify](#) and [Babel](#) to generate and compile code for your browser. However, the following features are not supported in browsers because of the limits of the browser environment:

- Streaming upload: Chunked encoding cannot be configured in browsers. We recommend that you use multipart upload instead of streaming upload.
- Local file management: Local file systems cannot be managed in browsers. We recommend that you

use signed URLs to download objects.

- OSS does not support bucket-related requests across origins. We recommend that you perform bucket-related operations in the OSS console.

Download OSS SDK for Browser.js

Demos in documents on the official website are based on OSS SDK V6.X. For more information about versions earlier than 6.X, visit [oss-nodejs-sdk](#). To upgrade your SDK version to 6.X, see [What is OSS?](#).

- [GitHub](#)
- [API operations](#)
- [ChangeLog](#)

Install OSS SDK for Browser.js

- OSS SDK for Browser.js supports the following browsers:
 - Internet Explorer 10 and later versions
 - Microsoft Edge
 - Major versions of Google Chrome, Firefox, and Safari
 - Default browsers of Android, iOS, and Windows Phone of major versions

- Installation methods

You can install OSS SDK for Browser.js by using one of the following methods:

- Use a browser to import OSS SDK for Browser.js

 **Notice** Some browsers do not provide native support for promise. In this case, you must import a promise library. For example, you must import the promise-polyfill library for Internet Explorer 10 and 11.

```
<!-- Import online resources -->
<script src="https://gosspublic.alicdn.com/aliyun-oss-sdk-x.x.x.min.js"></script>
```

```
<!-- Import local resources -->
<script src="./aliyun-oss-sdk-x.x.x.min.js"></script>
```

Note

- If you import online resources, the speed depends on the stability of Alibaba Cloud Content Delivery Network (CDN) servers. We recommend that you import the SDK from local resources or build the SDK by yourself.
- If you import the SDK from local resources in a sample program, set src to the path of the local resources relative to the path of the sample program.
- The aliyun-oss-sdk-x.x.x.min.js part in src indicates the name of the local resource, in which x.x.x indicates the version number of the SDK. For more information about the versions, visit [oss-js-sdk](#).

The following code provides an example on how to use an OSS object :

 **Note** For more information about how to use STS, see [Use a temporary credential provided by STS to access OSS](#) in OSS User Guide. You can call the [AssumeRole](#) operation or use STS SDKs for various programming languages to obtain temporary access credentials. For more information, see [STS SDK overview](#). The temporary access credentials consist of an AccessKey pair and a security token. The AccessKey pair contains an AccessKey ID and an AccessKey secret. The minimum validity period of temporary access credentials is 900 seconds. The maximum validity period of temporary access credentials is the maximum session duration specified for the current role. For more information, see [Specify the maximum session duration for a RAM role](#).

```
<script type="text/javascript">
  const client = new OSS({
    // Set yourRegion to the endpoint of the region in which the bucket is located. For
    // example, if your bucket is located in the China (Hangzhou) region, set yourRegion to os
    // s-cn-hangzhou.
    region: 'yourRegion',
    // Obtain a temporary AccessKey pair from STS.
    accessKeyId: 'yourAccessKeyId',
    accessKeySecret: 'yourAccessKeySecret',
    // Obtain a security token from STS.
    stsToken: 'yourSecurityToken',
    refreshSTSToken: async () => {
      // Obtain a temporary access credential from the STS that you set up.
      const info = await fetch('your_sts_server');
      return {
        accessKeyId: info.accessKeyId,
        accessKeySecret: info.accessKeySecret,
        stsToken: info.stsToken
      }
    },
    // Set the time interval at which to refresh the temporary access credential. Unit:
    // milliseconds.
    refreshSTSTokenInterval: 300000,
    // Specify the bucket name.
    bucket: 'examplebucket'
  });
</script>
```

- Use **npm** to install the package of OSS SDK for Browser.js

```
npm install ali-oss
```

After the package is installed, you can use the `import` or `require` syntax to use OSS SDK for Browser.js. Browsers do not provide native support for the `require` syntax. Therefore, you must use a packaging tool such as `webpack` or `browserify` in the development environment.

```
const OSS = require('ali-oss');
const client = new OSS({
  // Set yourRegion to the endpoint of the region in which the bucket is located. For
  // example, if your bucket is located in the China (Hangzhou) region, set yourRegion to os
  // s-cn-hangzhou.
  region: 'yourRegion',
  // Obtain a temporary AccessKey pair from STS, which consists of an AccessKey ID an
  // d an AccessKey secret.
  accessKeyId: 'yourAccessKeyId',
  accessKeySecret: 'yourAccessKeySecret',
  // Obtain a security token from STS.
  stsToken: 'yourSecurityToken',
  refreshSTSToken: async () => {
    // Obtain a temporary access credential from the STS that you set up.
    const info = await fetch('your_sts_server');
    return {
      accessKeyId: info.accessKeyId,
      accessKeySecret: info.accessKeySecret,
      stsToken: info.stsToken
    }
  },
  // Set the time interval at which to refresh the temporary access credential. Unit:
  // milliseconds.
  refreshSTSTokenInterval: 300000,
  // Specify the bucket name.
  bucket: 'examplebucket'
});
```

Implementation methods

You can use the synchronous or asynchronous mode when you use OSS SDK for Browser.js.

- Synchronous mode: The `async/await` method defined in JavaScript ES7 is used to synchronize asynchronous operations.
- Asynchronous mode: Asynchronous operations are performed in a way similar to callback. An API operation returns a **Promise** object for a request. The `then()` method is used to process the returned result, and the `catch()` method is used to handle errors.

 **Note** `new OSS()` is used to create OSS clients when the synchronous or asynchronous mode is used.

The following code provides examples on how to upload and download an object in synchronous and asynchronous modes:

- Synchronous mode

```
// Create an OSSClient instance.
const client = new OSS(...);
async function put () {
  try {
    // object specifies the name of the uploaded object.
    // file specifies the name of the file that you want to upload from the browser. Files of the HTML5 and Blob types are supported.
    const r1 = await client.put('object', file);
    console.log('put success: %j', r1);
    const r2 = await client.get('object');
    console.log('get success: %j', r2);
  } catch (e) {
    console.error('error: %j', e);
  }
}
put();
```

- **Asynchronous mode**

```
// Create an OSSClient.
const client = new OSS(...);
// object specifies the name of the uploaded object.
// file specifies the name of the file that you want to upload from the browser. Files of the HTML5 and Blob types are supported.
client.put('object', file).then(function (r1) {
  console.log('put success: %j', r1);
  return client.get('object');
}).then(function (r2) {
  console.log('get success: %j', r2);
}).catch(function (err) {
  console.error('error: %j', err);
});
```

6.2. Parameters

This topic describes the parameters that you can configure when you use OSS SDK for Browser.js.

OSS configuration item descriptions

- [accessKeyId] {String}: the AccessKey ID that you created in the Alibaba Cloud Management Console.
- [accessKeySecret] {String}: the AccessKey secret that you created in the Alibaba Cloud Management Console.
- [stsToken] {String}: the temporary STS token used to access OSS. For more information, see the "[Use STS to authorize temporary access](#)" section of the Authorized access topic.
- [bucket] {String}: the bucket that you create in the console.
- [endpoint] {String}: the endpoint used to access your OSS bucket.
- [region] {String}: the region where your bucket is located. Default value: oss-cn-hangzhou.
- [internal] {Boolean}: indicates whether to use the Alibaba Cloud internal network to access OSS. Default value: false. Example: Set this parameter to true if you use an ECS instance to access OSS. The ECS instance accesses OSS through the Alibaba Cloud internal network, which saves costs.
- [cname] {Boolean}: indicates whether CNAME can be used to access OSS. Default value: false. If you

set this parameter to true, you must bind a CNAME to your bucket before you use the CNAME to access the bucket.

- `[isRequestPay]` {Boolean}: indicates whether the pay-by-requester mode is enabled for your bucket. Default value: false.
- `[secure]` {Boolean}: indicates whether HTTPS is used to access OSS. The value of true indicates that HTTPS is used to access OSS. The value of false indicates that HTTP is used to access OSS. For more information, see [FAQ](#).
- `[timeout]` {String|Number}: the timeout period in seconds. Default value: 60.

Examples:

```
// Assume that the OSS object is imported to the browser environment. The introduction method can be 'script' or 'npm'.
let store = new OSS({
  accessKeyId: 'your access key',
  accessKeySecret: 'your access secret',
  bucket: 'your bucket name',
  region: 'oss-cn-hangzhou'
});
```

6.3. Quick start

This topic describes how to use OSS SDK for Browser.js to perform operations such as uploading, downloading, and viewing objects.

Prerequisites

A cross-origin resource sharing (CORS) rule is configured for the bucket. For more information, see [Installation](#).

Background information

Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log on to OSS because the account has permissions on all API operations. We recommend that you use STS to authorize the client. You can use STS to grant a third-party application or your RAM user an access credential that has custom permissions and a custom validity period. For more information, see [Use a temporary credential provided by STS to access OSS](#).

For more information about how to use browsers, see [Browser](#).

For more information about sample projects, see [example](#).

Use OSS SDK for Browser.js

Currently, browsers do not support bucket-related operations, such as GetBucket (ListObjects), GetBucketLogging, and GetBucketReferer. However, you can use a browser to perform object-related operations, such as PutObject and GetObject.



Notice The environment of a browser is untrusted. Security risks may arise if you store AccessKey IDs and AccessKey secrets in a browser. We recommend that you use plaintext only for tests and use the STS authentication mode or self-signed mode for business application. For more information, see [Authorized access](#) and [Set up direct data transfer for mobile apps](#).

- Import OSS SDK for Browser.js

Include the following information in the `<head>` field of the HTML file:

```
<script src="http://gosspublic.alicdn.com/aliyun-oss-sdk-x.x.x.min.js"></script>
```

For more information about the methods that you can use to import the SDK, see [Installation](#).

You can use the `new OSS` method to create a client. This method provides an asynchronous API operation similar to the callback method to process the returned results in `.then()` and handle errors in `.catch()`.

- Build an STS server and obtain temporary authorization information from the client
 - i. Use the STS-based signature services exclusive to SDKs or API operations. To help you get started with the STS service, OSS provides SDK demos in the following programming languages:

- [Node.js](#)
- [PHP](#)
- [Java](#)
- [Ruby](#)

 **Note** The preceding demos are for reference only. You must develop required code for authentication and other features in the production environment.

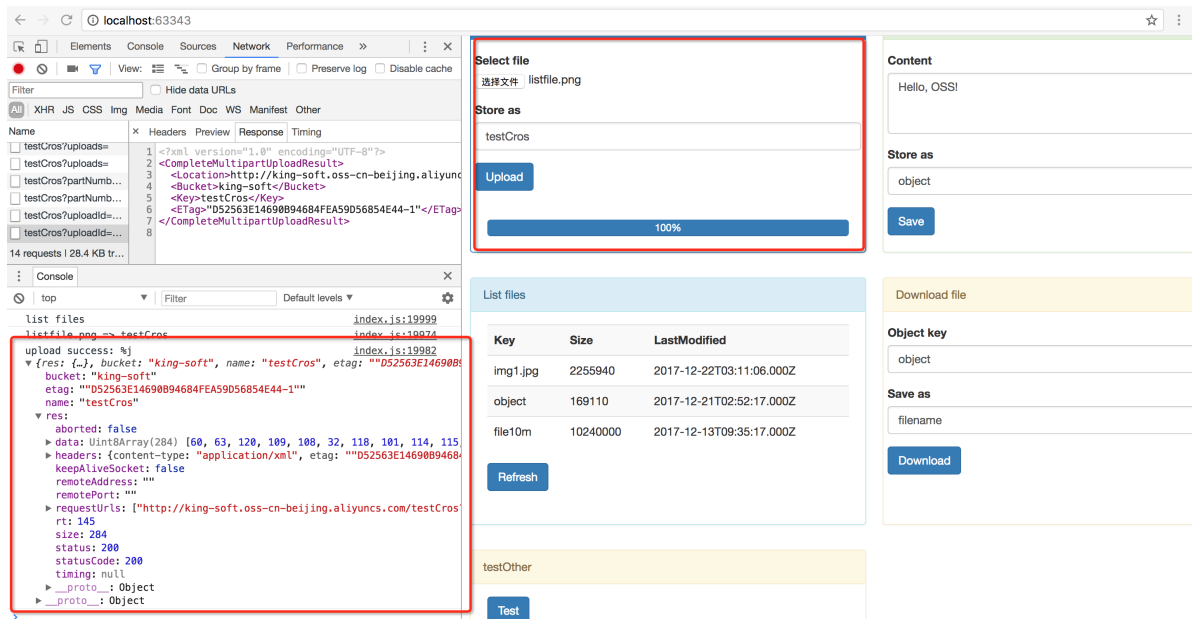
- ii. Initiate requests to the STS service built in the frontend (browser) to obtain the STS temporary authorization information.

```
<script type="text/javascript">
  // OSS.urlLib is the logic of sending requests encapsulated within the SDK. Developers
  // can use a library that sends requests to send requests to the STS server.
  OSS.urlLib.request("http://your_sts_server/", {method: 'GET'}, (err, response) => {
    if (err) {
      return alert(err);
    }
    try {
      result = JSON.parse(response);
    } catch (e) {
      errmsg = 'parse sts response info error: ' + e.message;
      return alert(errmsg);
    }
    // console.log(result)
  })
</script>
```

- Upload objects

You can call the `multipartUpload` operation to upload objects. For more information about `multipartUpload`, see [Overview](#).

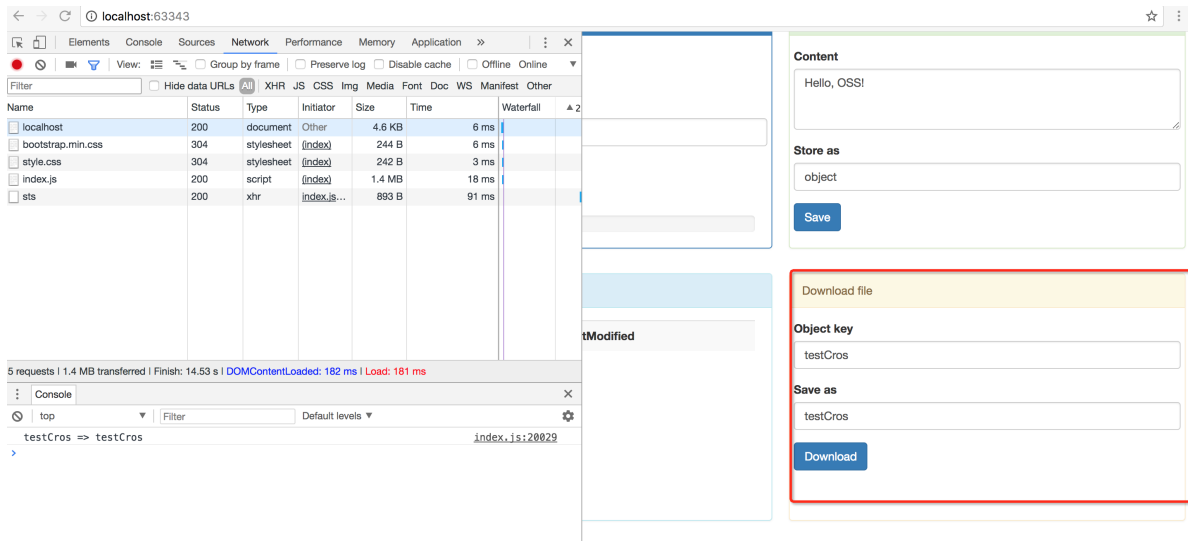
```
<body>
  <input type="file" id="file" />
  <script type="text/javascript">
    document.getElementById('file').addEventListener('change', function (e) {
      let file = e.target.files[0];
      let storeAs = 'upload-file';
      console.log(file.name + ' => ' + storeAs);
      // OSS.urlLib is the logic of sending requests encapsulated within the SDK. Developers
      // can use a library that sends requests to send requests to the STS server.
      OSS.urlLib.request("http://your_sts_server/", {method: 'GET'}, (err, response) => {
        if (err) {
          return alert(err);
        }
        try {
          result = JSON.parse(response);
        } catch (e) {
          return alert('parse sts response info error: ' + e.message);
        }
        let client = new OSS({
          accessKeyId: result.AccessKeyId,
          accessKeySecret: result.AccessKeySecret,
          stsToken: result.SecurityToken,
          // region indicates the region from which OSS services are provided. Example:
          // oss-cn-hangzhou.
          region: '<Your region>',
          bucket: '<Your bucket name>'
        });
        // You can set storeAs to an object name such as file.txt or a directory such as
        // abc/test/file.txt to upload the object to the root folder or the specified folder in
        // the bucket.
        // You can set file to file objects, Blob data, or OSS buffers.
        client.multipartUpload(storeAs, file).then(function (result) {
          console.log(result);
        }).catch(function (err) {
          console.log(err);
        });
      });
    });
  </script>
</body>
```



- Download objects

OSS generates a signed URL as the download link of an object. You can use the link to download the object.

```
<body>
  <input type="button" id="download" value="Download" />
  <script type="text/javascript">
    document.getElementById('download').addEventListener('click', function (e) {
      let objectKey = 'test/download_file';
      let saveAs = 'download_file';
      console.log(objectKey + ' => ' + saveAs);
      // OSS.urlLib is the logic of sending requests encapsulated within the SDK. Developers
      // can use a library that sends requests to send requests to the STS server.
      OSS.urlLib.request("http://your_sts_server/", {method: 'GET'}, (err, response) => {
        if (err) {
          return alert(err);
        }
        try {
          result = JSON.parse(response);
        } catch (e) {
          return alert('parse sts response info error: ' + e.message);
        }
        let client = new OSS({
          accessKeyId: result.AccessKeyId,
          accessKeySecret: result.AccessKeySecret,
          stsToken: result.SecurityToken,
          // region indicates the region from which OSS services are provided. Example:
          // oss-cn-hangzhou.
          region: '<Your region>',
          bucket: '<Your bucket name>'
        });
        let result = client.signatureUrl(objectKey, {
          expires: 3600,
          response: {
            'content-disposition': 'attachment; filename="' + saveAs + '"'
          }
        });
        console.log(result);
        window.location = result;
      });
    });
  </script>
</body>
```

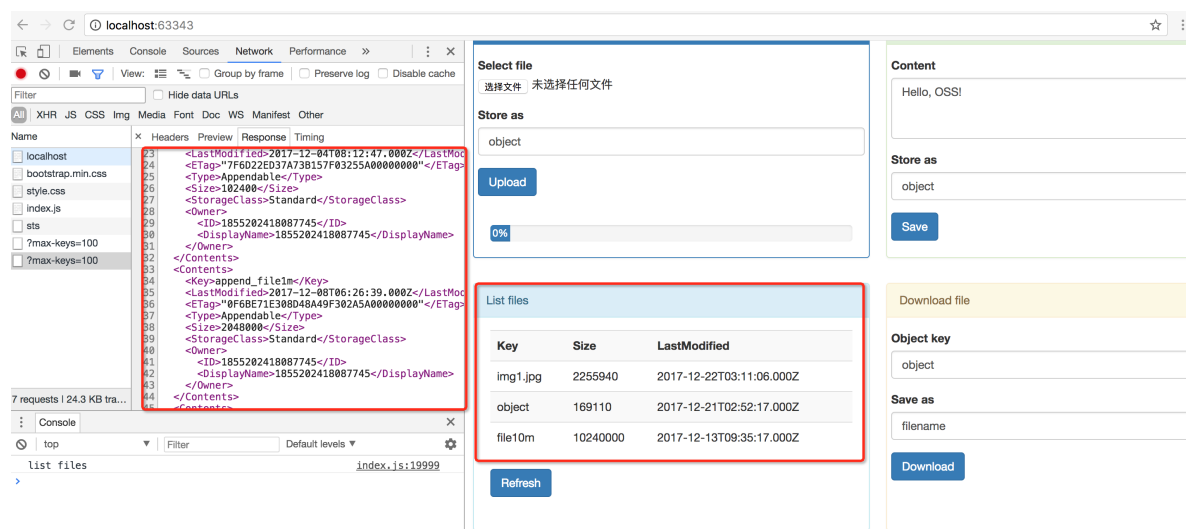


- List objects in a bucket

The following code provides an example on how to list objects in a bucket:

```
<script type="text/javascript">
OSS.urllib.request("http://your_sts_server/", {method: 'GET'}, (err, response) => {
  if (err) {
    return alert(err);
  }
  try {
    result = JSON.parse(response);
  } catch (e) {
    return alert('parse sts response info error: ' + e.message);
  }
  let client = new OSS({
    accessKeyId: result.AccessKeyId,
    accessKeySecret: result.AccessKeySecret,
    stsToken: result.SecurityToken,
    // region indicates the region from which OSS services are provided. Example: oss-cn-
    hangzhou.
    region: '<Your region>',
    bucket: '<Your bucket name>'
  });
  client.list({
    'max-keys': 10
  }).then(function (result) {
    console.log(result);
  }).catch(function (err) {
    console.log(err);
  });
});
</script>
```

You can open the HTML file in Google Chrome and view the listed objects in Chrome DevTools.



6.4. Upload objects

6.4.1. Overview

This topic describes the object upload methods that are supported by Object Storage Service (OSS) SDK for Browser.js. You can choose a method based on your requirements.

OSS SDK for Browser.js supports the following object upload methods.

- You can use simple upload to upload file objects, Blob data, and OSS buffers to OSS. For more information, see [Simple upload](#).
- To upload objects larger than 100 MB and smaller than 48.8 TB, you can use multipart upload. Multipart upload is an object upload method that splits a large object into multiple parts to upload. After the parts are uploaded, CompleteMultipartUpload is called to combine the parts together into a complete object. This way, you can upload an object in a resumable manner. For more information, see [Multipart upload](#).
- To resume the upload of an object that fails to be uploaded to OSS due to a network exception or program error, you can use resumable upload. For more information, see [Resumable upload](#).
- To configure upload callback for your application server, you can use upload callback. For more information, see [Upload callback](#).

6.4.2. Simple upload

This topic describes how to use putObject to upload file objects, Blob data, or Object Storage Service (OSS) buffers to OSS. Progress functions cannot be used in simple upload.

Usage notes

- When you use packaging tools such as Webpack and Browserify, install OSS SDK for Browser.js by running the `npm install ali-oss` command.
- In most cases, OSS SDK for Browser.js is used in browsers. To prevent your AccessKey pair from being exposed, we recommend that you use temporary access credentials obtained from Security Token Service (STS) to access OSS.

For more information about how to use STS, see [Use a temporary credential provided by STS to access OSS](#) in OSS Developer Guide. You can call the [AssumeRole](#) operation or use STS SDKs for various programming languages to obtain temporary access credentials. For more information, see [STS SDK overview](#). Temporary access credentials include a security token and a temporary AccessKey pair that consists of an AccessKey ID and an AccessKey secret.

Sample code

 **Note** Blob indicates large binary objects. For more information, see [Blob](#).

The following code provides an example on how to use simple upload to upload data to an object named `exampleobject.txt` in the `exampledir` directory of the `examplebucket` bucket:

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <title>Document</title>
  </head>
  <body>
    <button id="upload">Upload An Object</button>
    <input id="file" type="file" />
    <!-- Import an SDK file -->
    <script
      type="text/javascript"
      src="https://gosspublic.alicdn.com/aliyun-oss-sdk-6.16.0.min.js"
    ></script>
    <script type="text/javascript">
      const client = new OSS({
        // Specify the region in which the bucket is located. For example, if the bucket is
        // located in the China (Hangzhou) region, set the region to oss-cn-hangzhou.
        region: 'yourRegion',
        // Specify the temporary AccessKey pair obtained from Security Token Service (STS).
        // The AccessKey pair consists of an AccessKey ID and an AccessKey secret.
        accessKeyId: 'yourAccessKeyId',
        accessKeySecret: 'yourAccessKeySecret',
        // Specify the security token obtained from STS.
        stsToken: 'yourSecurityToken',
        // Specify the bucket name.
        bucket: 'examplebucket'
      });
      // Select the local file from the drop-down list. Example: <input type="file" id="file" />.
      let data;
      // Create and specify the Blob data.
      //const data = new Blob(['Hello OSS']);
      // Create and specify the content of the OSS buffer.
      //const data = new OSS.Buffer(['Hello OSS']);
      const upload = document.getElementById("upload");
      const headers = {
        // Specify the caching behavior of the web page when the object is downloaded.
        // 'Cache-Control': 'no-cache',
        // Specify the name of the object when the object is downloaded.
        // 'Content-Disposition': 'attachment; filename=exampleobject.txt'
```



```
// 'Content-Disposition': 'OSS_download.txt',
// Specify the content encoding format of the object when the object is downloaded.

// 'Content-Encoding': 'UTF-8',
// Specify the expiration time of the upload request.
// 'Expires': 'Wed, 08 Jul 2022 16:57:01 GMT',
// Specify the storage class of the object.
// 'x-oss-storage-class': 'Standard',
// Specify the access control list (ACL) of the object.
// 'x-oss-object-acl': 'private',
// Configure tagging for the object. You can specify multiple tags for the object a
t a time.
// 'x-oss-tagging': 'Tag1=tag&Tag2=2',
// Specify whether the CopyObject operation is allowed to overwrite the existing ob
ject that has the same name as the object to copy. In this example, this parameter is set t
o true, which indicates that the operation is not allowed to overwrite the existing object
that has the same name as the object to copy.
// 'x-oss-forbid-overwrite': 'true',
};

async function putObject (data) {
  try {
    // Specify the full path of the object. The full path cannot contain the bucket
name.

    // You can specify an object name such as exampleobject.txt to upload data to th
e current bucket. You can also specify a directory named exampledir/exampleobject.txt to up
load data to the specified directory in the bucket.
    // You can set the data to file objects, Blob data, or OSS buffers.
    const result = await client.put(
      "exampledir/exampleobject.txt",
      data
      //{headers}
    );
    console.log(result);
  } catch (e) {
    console.log(e);
  }
}

upload.addEventListener("click", () => {
  data = document.getElementById("file").files[0];
  putObject(data);
});
</script>
</body>
</html>
```

References

- For the complete sample code that is used to perform simple upload, visit [GitHub](#).
- For more information about the API operation that you can call to perform simple upload, see [Put Object](#).

6.4.3. Multipart upload

Object Storage Service (OSS) provides the multipart upload feature. Multipart upload allows you to split a large object into multiple parts to upload. After the parts are uploaded, you can call `CompleteMultipartUpload` to combine the parts together into a complete object. This way, you can upload an object in a resumable manner.

Usage notes

- When you use packaging tools such as Webpack and Browserify, install OSS SDK for Browser.js by running the `npm install ali-oss` command.
- In most cases, OSS SDK for Browser.js is used in browsers. To prevent your AccessKey pair from being exposed, we recommend that you use temporary access credentials obtained from Security Token Service (STS) to access OSS.

For more information about how to use STS, see [Use a temporary credential provided by STS to access OSS](#) in OSS Developer Guide. You can call the `AssumeRole` operation or use STS SDKs for various programming languages to obtain temporary access credentials. For more information, see [STS SDK overview](#). Temporary access credentials include a security token and a temporary AccessKey pair that consists of an AccessKey ID and an AccessKey secret.

Complete sample code

To upload a large object, you can call `MultipartUpload` to perform multipart upload. In multipart upload, an object is split into multiple parts that are separately uploaded. If some parts fail to be uploaded, you can reupload only these parts based on the upload progress. To upload an object larger than 100 MB, we recommend that you use multipart upload. This improves the upload success rate.

The following code provides an example on how to use multipart upload to upload a local file to an object named `exampleobject.txt` in the `examplebucket` bucket:

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <title>Document</title>
  </head>
  <body>
    <button id="submit">Upload An Object</button>
    <input id="file" type="file" />
    <!-- import an SDK file -->
    <script
      type="text/javascript"
      src="https://gosspublic.alicdn.com/aliyun-oss-sdk-6.16.0.min.js"
    ></script>
    <script type="text/javascript">
      const client = new OSS({
        // Set yourRegion to the region in which the bucket is located. For example, if you
        // bucket is located in the China (Hangzhou) region, set yourRegion to oss-cn-hangzhou.
        region: "yourRegion",
        // Specify the temporary AccessKey pair obtained from STS.
        accessKeyId: "yourAccessKeyId",
        accessKeySecret: "yourAccessKeySecret",
        // Specify the security token obtained from STS.
        stsToken: "yourSecurityToken",
        // Specify the name of the bucket. Example: examplebucket.
        bucket: "examplebucket",
```

```

});
const headers = {
  // Specify the caching behavior of the web page when the object is downloaded.
  "Cache-Control": "no-cache",
  // Specify the name of the object when the object is downloaded.
  "Content-Disposition": "example.txt",
  // Specify the content encoding format of the object when the object is downloaded.

  "Content-Encoding": "utf-8",
  // Specify the validity period of the request. Unit: milliseconds.
  Expires: "1000",
  // Specify the storage class of the object.
  "x-oss-storage-class": "Standard",
  // Specify the tag for the object. You can specify multiple tags for the object at
  the same time.
  "x-oss-tagging": "Tag1=1&Tag2=2",
  // Specify whether existing objects are overwritten by objects with the same names
  when the multipart upload task is initiated. In this example, this parameter is set to true
  , which indicates that the object with the same name cannot be overwritten.
  "x-oss-forbid-overwrite": "true",
};
// Specify the name of the object that is uploaded to the examplebucket bucket. Exam
le: exampleobject.txt.
const name = "exampleobject.txt";
// Query DOM.
const submit = document.getElementById("submit");
const data = document.getElementById("file").files[0];
const options = {
  // Query the progress, the checkpoint, and the return value of the multipart upload
  task.
  progress: (p, cpt, res) => {
    console.log(p);
  },
  // Specify the number of parts that can be uploaded in parallel.
  parallel: 4,
  // Set the size available for each part. Default value: 1 MB. Minimum value: 100 KB
  .
  partSize: 1024 * 1024,
  // headers,
  // Specify the user metadata of the object. You can call the HeadObject operation t
  o query the object metadata.
  meta: { year: 2020, people: "test" },
  mime: "text/plain",
};
// Create a button that is used to start the multipart upload task.
submit.addEventListener("click", async () => {
  try {
    // Start the multipart upload task.
    const res = await client.multipartUpload(name, data, {
      // Configure upload callback.
      callback: {
        // Specify the address of the server that receives the callback request.
        url: "http://examplebucket.aliyuncs.com:23450",
        // Specify the Host header in the callback request.

```

```

        host: "yourHost",
        /* eslint no-template-curly-in-string: [0] */
        // Specify the value of the body field included in the callback request.
        body: "bucket=${bucket}&object=${object}&var1=${x:var1}",
        // Specify Content-Type in the callback request.
        contentType: "application/x-www-form-urlencoded",
        customValue: {
            // Configure the custom parameters for the callback request.
            var1: "value1",
            var2: "value2",
        },
    },
    });
} catch (err) {
    console.log(err);
}
});
</script>
</body>
</html>

```

If the `ConnectionTimeoutError` occurs when you call the `MultipartUpload` operation, you must handle the error. To fix the error, you can reduce the size of each part, extend the timeout period, or resend the request. You can also capture `ConnectionTimeoutError` to analyze the specific cause. For more information, see [Network connection timeout handling](#).

Cancel a multipart upload task

You can use the `client.abortMultipartUpload` method to cancel a multipart upload task. If you cancel a multipart upload task, you cannot use the upload ID to upload parts. The uploaded parts are deleted.

The following code provides an example on how to cancel a multipart upload task:

```

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8" />
    <title>Document</title>
</head>
<body>
    <button id='submit'>Upload An Object</button>
    <button id='abort'>Cancel Upload</button>
    <!-- import an SDK file -->
    <script type="text/javascript" src="https://gosspublic.alicdn.com/aliyun-oss-sdk-6.16.0.min.js"></script>
    <script type="text/javascript">
        const client = new OSS({
            // Set yourRegion to the region in which the bucket is located. For example, if your
            bucket is located in the China (Hangzhou) region, set yourRegion to oss-cn-hangzhou.
            region: 'yourRegion',
            // Specify the temporary AccessKey pair obtained from STS.
            accessKeyId: 'yourAccessKeyId',
            accessKeySecret: 'yourAccessKeySecret',

```

```

    // Specify the security token obtained from STS.
    stsToken: 'yourSecurityToken',
    // Specify the name of the bucket. Example: examplebucket.
    bucket: "examplebucket",
  });
  // Create a local file that you want to upload to the bucket by using multipart upload.
  The size of the file is 100 MB.
  const fileContent = Array(1024 * 1024 * 100).fill('a').join('')
  const file = new File([fileContent], 'multipart-upload-file')
  // Specify the name of the object that is uploaded to the examplebucket bucket. Example
  : exampleobject.txt.
  const name = "exampleobject.txt";
  // Set the checkpoint at which the multipart upload task is canceled.
  let abortCheckpoint
  // Query DOM.
  const submit = document.getElementById('submit');
  const abort = document.getElementById('abort')
  // Create a button that is used to start the multipart upload task.
  submit.addEventListener('click', async () => {
    try {
      const res = await client.multipartUpload(name, file, {
        progress: (p, cpt, res) => {
          // Assign a value to the checkpoint.
          abortCheckpoint = cpt
          // Query the progress of the multipart upload task.
          console.log(p)
        }
      })
    } catch (err) {
      console.log(err)
    }
  })
  // Create a button that is used to cancel the multipart upload task.
  abort.addEventListener('click', () => {
    // Cancel the multipart upload task.
    client.abortMultipartUpload(abortCheckpoint.name, abortCheckpoint.uploadId)
  })
</script>
</body>
</html>

```

List uploaded parts

You can call the `client.listParts` method to list all parts that are uploaded by using the specified upload ID.

The following code provides an example on how to list uploaded parts:

```

<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8" />
  <title>Document</title>
</head>
<body>

```

```

<body>
  <button id='check'>List Uploaded Parts</button>
  <!-- import an SDK file -->
  <script type="text/javascript" src="https://gosspublic.alicdn.com/aliyun-oss-sdk-6.16.0.min.js"></script>
  <script type="text/javascript">
    const client = new OSS({
      // Set yourRegion to the region in which the bucket is located. For example, if your
      bucket is located in the China (Hangzhou) region, set yourRegion to oss-cn-hangzhou.
      region: 'yourRegion',
      // Specify the temporary AccessKey pair obtained from STS.
      accessKeyId: 'yourAccessKeyId',
      accessKeySecret: 'yourAccessKeySecret',
      // Specify the security token obtained from STS.
      stsToken: 'yourSecurityToken',
      // Specify the name of the bucket. Example: examplebucket.
      bucket: "examplebucket",
    });
    // Create a local file that you want to upload to the bucket by using multipart upload.
    The size of the file is 100 MB.
    const fileContent = Array(1024 * 1024 * 100).fill('a').join('')
    const file = new File([fileContent], 'multipart-upload-file')
    // Specify the name of the object that is uploaded to the examplebucket bucket. Example
    : exampleobject.txt.
    const name = "exampleobject.txt";
    // Set the checkpoint at which the multipart upload task is canceled.
    let abortCheckpoint
    // Query DOM.
    const submit = document.getElementById('submit');
    const check = document.getElementById('check')
    // Create a button that is used to start the multipart upload task.
    submit.addEventListener('click', async () => {
      try {
        const res = await client.multipartUpload(name, file, {
          progress: (p, cpt, res) => {
            // Assign a value to the checkpoint.
            abortCheckpoint = cpt
            // Query the progress of the multipart upload task.
            console.log('progress====', p)
          }
        })
      } catch (err) {
        console.log(err)
      }
    })
    // Create a button that is used to list the uploaded parts.
    check.addEventListener('click', async () => {
      // List the uploaded parts.
      const result = await client.listParts(name, abortCheckpoint.uploadId)
      console.log(result)
    })
  </script>
</body>
</html>

```

List multipart upload tasks

You can use the `client.listUploads` method to list all ongoing multipart upload tasks, which include tasks that have been initiated but are not completed or are canceled.

The following code provides an example on how to list multipart upload tasks:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8" />
  <title>Document</title>
</head>
<body>
  <button id='check'>List Multipart Upload Tasks</button>
  <!-- import an SDK file -->
  <script type="text/javascript" src="https://gosspublic.alicdn.com/aliyun-oss-sdk-6.16.0.min.js"></script>
  <script type="text/javascript">
    const client = new OSS({
      // Set yourRegion to the region in which the bucket is located. For example, if your bucket is located in the China (Hangzhou) region, set yourRegion to oss-cn-hangzhou.
      region: 'yourRegion',
      // Specify the temporary AccessKey pair obtained from STS.
      accessKeyId: 'yourAccessKeyId',
      accessKeySecret: 'yourAccessKeySecret',
      // Specify the security token obtained from STS.
      stsToken: 'yourSecurityToken',
      // Specify the name of the bucket. Example: examplebucket.
      bucket: "examplebucket",
    });
    // Query DOM.
    const check = document.getElementById('check')
    // Create a button that is used to list multipart upload tasks.
    check.addEventListener('click', async () => {
      // Query multipart upload tasks that have been initiated but are not completed or are canceled.
      const result = await client.listUploads({ 'max-uploads': 100 })
      console.log('uploads', result.uploads)
    })
  </script>
</body>
</html>
```

References

- For more information about the complete sample code of multipart upload, visit [GitHub](#).
- The `multipartUpload` method that is used by OSS SDK for Browser.js to perform multipart upload contains the following three API operations:
 - The API operation that you can call to initiate a multipart upload task. For more information, see [InitiateMultipartUpload](#).
 - The API operation that you can call to upload data by part. For more information, see [UploadPart](#).

- The API operation that you can call to complete a multipart upload task. For more information, see [CompleteMultipartUpload](#).
- For more information about the API operation that you can call to cancel a multipart upload task, see [AbortMultipartUpload](#).
- For more information about the API operation that you can call to list uploaded parts, see [ListParts](#).
- For more information about the API operation that you can call to list ongoing multipart upload tasks, see [ListMultipartUploads](#). Ongoing multipart upload tasks include tasks that have been initiated but are not completed or are canceled.

6.4.4. Resumable upload

Before you upload an object to Object Storage Service (OSS) by using resumable upload, you can specify a folder for the checkpoint file that stores resumable upload records. If an object fails to upload because a network exception or program error, the upload task is resumed from the position recorded in the checkpoint file to upload the remaining bytes.

Usage notes

- When you use packaging tools such as Webpack and Browserify, install OSS SDK for Browser.js by running the `npm install ali-oss` command.
- OSS SDK for Browser.js is usually used in browsers. To prevent your AccessKey pair from being exposed, we recommend that you use a temporary access credential generated by Security Token Service (STS) to access OSS.

For more information about how to set up STS, see [Use a temporary credential provided by STS to access OSS](#) in OSS Developer Guide. You can call the [AssumeRole](#) operation or use STS SDKs for various programming languages to obtain a temporary access credential. For more information, see [STS SDK overview](#). A temporary access credential contains a security token and a temporary AccessKey pair that consists of an AccessKey ID and an AccessKey secret.

Sample code

The following code provides an example on how to upload an object by using resumable upload:

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <title>Document</title>
  </head>
  <body>
    <button id="submit">Upload An Object</button>
    <button id="pause">Pause Upload</button>
    <button id="resume">Resume Upload</button>
    <!-- import an SDK file -->
    <script
      type="text/javascript"
      src="https://gosspublic.alicdn.com/aliyun-oss-sdk-6.16.0.min.js"
    ></script>
    <script type="text/javascript">
      const client = new OSS({
        // Set yourRegion to the endpoint of the region in which the bucket is located. For
        // example, if your bucket is located in the China (Hangzhou) region, set yourRegion to oss-cn
```



```

-hangzhou.
    region: 'yourRegion',
    // Specify the temporary AccessKey pair obtained from STS.
    accessKeyId: 'yourAccessKeyId',
    accessKeySecret: 'yourAccessKeySecret',
    // Specify the security token obtained from STS.
    stsToken: 'yourSecurityToken',
    // Specify the name of the bucket. Example: examplebucket.
    bucket: "examplebucket",
  });
  // Create the object to upload by using resumable upload. The size of the object is 1
  00 MB.
  const fileContent = Array(1024 * 1024 * 100)
    .fill("a")
    .join("");
  const file = new File([fileContent], "multipart-upload-file");
  // Specify the full path of the object that is uploaded to the bucket. Example: exam
  ledir/exampleobject.txt.
  const name = "test.txt";
  // Define the checkpoint at which the object upload pauses.
  let abortCheckpoint;
  // Obtain the DOM elements of the upload and the checkpoint.
  const submit = document.getElementById("submit");
  const pause = document.getElementById("pause");
  // Obtain the DOM elements of the resumable upload task.
  const resume = document.getElementById("resume");
  // Create a button that is used to upload the object.
  submit.addEventListener("click", () => {
    client
      .multipartUpload(name, file, {
        progress: (p, cpt, res) => {
          // Assign a value to the checkpoint.
          abortCheckpoint = cpt;
          console.log(abortCheckpoint);
          // Obtain the upload progress.
          console.log(p * 100);
        },
      })
      .then((r) => console.log(r));
  });
  // Create a button that is used to pause the upload of the object.
  pause.addEventListener("click", () => {
    // Pause the upload.
    client.cancel();
  });
  const resumeUpload = async () => {
    // Set the number of upload retry operations to 5.
    try {
      const result = await client.multipartUpload(name, file, {
        checkpoint: abortCheckpoint,
        progress: (p, cpt, res) => {
          // To implement resumable upload, you can save the checkpoint information dur
          ing the upload. This way, after an upload exception occurs, the checkpoint is passed to mul
          tipartUpload as a parameter. Then, the upload is resumed from the byte at which last upload

```

```
is paused.
        abortCheckpoint = cpt;
        // Obtain the upload progress.
        console.log(p);
    },
    });
    console.log(result);
} catch (e) {
    console.log(e);
}
};
// Create a button that is used to resume the upload.
resume.addEventListener("click", async () => {
    await resumeUpload();
});
</script>
</body>
</html>
```

References

For more information about the complete sample code for resumable upload, visit [GitHub](#).

6.4.5. Upload callback

To configure upload callback, you need only to include the corresponding parameters in the request that is sent to Object Storage Service (OSS).

Sample code

The following code provides an example on how to configure upload callback:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8" />
  <title>Document</title>
</head>
<body>
  <button id='submit'>Upload Callback</button>
  <!-- import an SDK file -->
  <script type="text/javascript" src="https://gosspublic.alicdn.com/aliyun-oss-sdk-6.16.0.min.js"></script>
  <script type="text/javascript">
    const client = new OSS({
      // Set yourRegion to the endpoint of the region in which the bucket is located. For example, if your bucket is located in the China (Hangzhou) region, set yourRegion to oss-cn-hangzhou.
      region: 'yourRegion',
      // Specify the temporary AccessKey pair obtained from STS.
      accessKeyId: 'yourAccessKeyId',
      accessKeySecret: 'yourAccessKeySecret',
      // Specify the security token obtained from STS.
      stsToken: 'yourSecurityToken',
      // Specify the name of the bucket. Example: examplebucket.
      bucket: "examplebucket",
    });
    const options = {
      callback: {
        // Set the address of the application server that receives the callback request.
        url: 'http://examplebucket.aliyuncs.com:23450',
        // Set the value of the Host field included in the callback request header.
        // host: 'oss-cn-hangzhou.aliyuncs.com',
        // Set the value of the body field included in the callback request.
        body: "bucket=${bucket}&object=${object}&etag=${etag}&size=${size}&mimeType=${mimeType}&imageInfo.height=${imageInfo.height}&imageInfo.width=${imageInfo.width}&imageInfo.format=${imageInfo.format}&my_var=${x:my_var}",
        // Set Content-Type for the callback request.
        // contentType: 'application/x-www-form-urlencoded',
        // Configure the custom parameters for the callback request.
        customValue: {
          var1: 'value1',
          var2: 'value2'
        }
      }
    }
    // Obtain the DOM elements of the upload callback configuration.
    const submit = document.getElementById('submit')
    // Implement upload callback.
    submit.addEventListener('click', () => {
      client.put('test.txt', new Blob(['1234']), options).then(r => console.log(r))
    })
  </script>
</body>
</html>
```

References

For more information about the API operation that is called to configure upload callback, see [Callback](#).

6.5. Download objects

6.5.1. Overview

This topic describes the object download methods that are supported by Object Storage Service (OSS) SDK for Browser.js. You can choose a download method based on your business requirements.

OSS SDK for Browser.js provides the following object download methods:

- You can use the `signatureUrl` method in a browser to generate the URL to download or preview an object. For more information, see [Preview or download an object](#).
- You can use range download to download parts of the content of an object. For more information, see [Range download](#).
- You can use conditional download to download objects that meet multiple specified conditions. For example, you can specify conditions to download only objects that are last modified before the specified time or objects whose Etags match the specified values. For more information, see [Conditional download](#).

6.5.2. Preview or download an object

You can use the `signatureUrl` method in a browser to generate the URL to download or preview an object. You can use the `download` attribute of the `<a>` HTML element or the `window.open` API to obtain the URL of an object.

Obtain the URL of an object for previews

The following code provides an example on how to obtain the URL of an object named `exampleobject.txt` in a bucket named `examplebucket` for previews.

 **Note** To preview an object in a browser, set `Content-Disposition` to `inline` and use the custom domain name that is mapped to the bucket to access the object. For more information about specific operations, see [Configure object metadata](#) and [Map custom domain names](#).

```
const OSS = require('ali-oss');
const client = new OSS({
  // Set yourRegion to the endpoint of the region in which the bucket is located. For example, if your bucket is located in the China (Hangzhou) region, set yourRegion to oss-cn-hangzhou.
  region: 'yourRegion',
  // Specify the temporary AccessKey pair obtained from STS.
  accessKeyId: 'yourAccessKeyId',
  accessKeySecret: 'yourAccessKeySecret',
  // Specify the security token obtained from STS.
  stsToken: 'yoursecurityToken',
  // Specify the bucket name.
  bucket: 'examplebucket',
});
let url;
// Specify the full path of the object. The full path of the object cannot contain the bucket name. By default, the validity period of the object URL is 1,800 seconds, which equals to 30 minutes.
url = client.signatureUrl('exampleobject.txt');
console.log(url);
// Set the validity period of the URL to 3600 seconds. If you do not set the validity period, the default value 1800 is used.
//url = client.signatureUrl('exampleobject.txt', {expires: 3600});
//console.log(url);
```

Obtain the URL to download an object

The following code provides an example on how to obtain the URL to download an object named `exampleobject.txt` in a bucket named `examplebucket`. By default, the validity period of the URL is 1,800 seconds.

```
const OSS = require('ali-oss');
const client = new OSS({
  // Set yourRegion to the endpoint of the region in which the bucket is located. For example, if your bucket is located in the China (Hangzhou) region, set yourRegion to oss-cn-hangzhou.
  region: 'yourRegion',
  // Specify the temporary AccessKey pair obtained from STS.
  accessKeyId: 'yourAccessKeyId',
  accessKeySecret: 'yourAccessKeySecret',
  // Specify the security token obtained from STS.
  stsToken: 'yoursecurityToken',
  // Specify the bucket name.
  bucket: 'examplebucket',
});
// Set the response header to automatically download an object by using the URL, and set the name of the local file after the object is downloaded.
const filename = 'examplefile.txt'
const response = {
  'content-disposition': `attachment; filename=${encodeURIComponent(filename)}`
}
// Specify the full path of the object. The full path of the object cannot contain the bucket name.
const url = client.signatureUrl('exampleobject.txt', { response });
console.log(url);
```

6.5.3. Range download

You can use range download to download parts of the content of an object.

Sample code

The following code provides an example on how to perform range download:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8" />
  <title>Document</title>
</head>
<body>
  <button id='download'>Download</button>
  <!-- import an SDK file -->
  <script type="text/javascript" src="https://gosspublic.alicdn.com/aliyun-oss-sdk-6.16.0.min.js"></script>
  <script type="text/javascript">
    const client = new OSS({
      // Set yourRegion to the endpoint of the region in which the bucket is located. For example, if your bucket is located in the China (Hangzhou) region, set yourRegion to oss-cn-hangzhou.
      region: 'yourRegion',
      // Specify the temporary AccessKey pair obtained from STS.
      accessKeyId: 'yourAccessKeyId',
      accessKeySecret: 'yourAccessKeySecret',
```

```
// Specify the security token obtained from STS.
stsToken: 'yourSecurityToken',
// Specify the name of the bucket. Example: examplebucket.
bucket: "examplebucket",
});
// Obtain the DOM elements of the range download operation.
const download = document.getElementById('download')
// For an object that is 1,000 bytes in size, the valid range is within the range from
byte 0 to byte 999.
// If the specified value range is invalid, for example, the specified range includes a
negative number, or the specified value is larger than the object size, the entire object i
s downloaded.
const start = 0, end = 999
// Create a button that is used to start the range download operation.
download.addEventListener('click', () => {
  client.get('example.txt',{
    headers: {
      Range: `bytes=${start}-${end}`,
    },
  }).then(r => {
    // Convert the returned data to a Blob object that can be downloaded.
    const newBlob = new Blob([r.content], { type: r.res.headers['content-type'] });
    // Create an <a> tag.
    const link = document.createElement('a')
    // Add the href attribute to the tag.
    link.href = window.URL.createObjectURL(newBlob)
    // Specify the name of the object after it is downloaded to your local device.
    link.download = 'example.txt'
    // Download the object.
    link.click()
    // Remove the URL added to the tag.
    window.URL.revokeObjectURL(link.href)
  })
})
</script>
</body>
</html>
```

References

- For more information about the complete sample code for range download, visit [GitHub](#).
- For more information about the API operation that is called to perform range download, see [Get Object](#).

6.5.4. Conditional download

You can specify one or more conditions when you download objects. If the specified conditions are met, the object is downloaded. If the specified conditions are not met, an error is returned and the object is not downloaded.

Sample code

The following code provides an example on how to perform conditional download:

```

<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8" />
  <title>Document</title>
</head>
<body>
  <button id='upload'>Upload</button>
  <button id='download'>Download</button>
  <!-- import an SDK file -->
  <script type="text/javascript" src="https://gosspublic.alicdn.com/aliyun-oss-sdk-6.16.0.min.js"></script>
  <script type="text/javascript">
    const client = new OSS({
      // Set yourRegion to the endpoint of the region in which the bucket is located. For example, if your bucket is located in the China (Hangzhou) region, set yourRegion to oss-cn-hangzhou.
      region: 'yourRegion',
      // Specify the temporary AccessKey pair obtained from STS.
      accessKeyId: 'yourAccessKeyId',
      accessKeySecret: 'yourAccessKeySecret',
      // Specify the security token obtained from STS.
      stsToken: 'yourSecurityToken',
      // Specify the name of the bucket. Example: examplebucket.
      bucket: "examplebucket",
    });
    const download = document.getElementById('download')
    const upload = document.getElementById('upload')
    // Upload the object.
    upload.addEventListener('click', () => {
      // Specify the object to upload.
      const file = new Blob(['examplecontent'])
      // Specify the full path of the object. Example: exampledir/exampleobject.txt.
      const fileName = 'exampledir/exampleobject.txt'
      const result = client.put(fileName, file).then(r => console.log(r))
    })
    // Download the object.
    download.addEventListener('click', () => {
      // Specify the data range of the object to download.
      const start = 1, end = 5
      client.get('exampledir/exampleobject.txt', {
        headers: {
          // Set the If-Modified-Since header in the request. If the value of this header is earlier than the time when the uploaded object is last modified, the object is downloaded. // If the value of the If-Modified-Since header is equal to or later than the time when the uploaded object is last modified, OSS returns 304 Not Modified.
          "If-Modified-Since": new Date("1970-01-01").toGMTString()
          // Set the If-Unmodified-Since header in the request. If the value of this header is equal to or later than the time when the uploaded object is last modified, the object is downloaded. If the value of the If-Unmodified-Since header is earlier than the time when the uploaded object is last modified, OSS returns 412 Precondition Failed.
          //"If-Unmodified-Since": new Date(1970-01-01).toGMTString()
          // Set the If-Match header to an ETag value. If the specified ETag value matches the ETag of the object, the object is downloaded. If the specified ETag value does not match

```



```
h the ETag of the object, OSS returns 412 Precondition Failed.
    // "If-Match": '5B3C1A2E0563E1B002CC607C****'

    // Set the If-Match header to an ETag value. If the specified ETag value does not
    match the ETag of the object, the object is downloaded. If the specified ETag value matches
    the ETag of the object, OSS returns 304 Not Modified.
    // "If-None-Match": '5B3C1A2E0563E1B002CC607C****'
  },
}).then(r => {
  if (r.content.length > 0) {
    const newBlob = new Blob([r.content], { type: r.res.headers['content-type'] });
    const link = document.createElement('a')
    link.href = window.URL.createObjectURL(newBlob)
    link.download = 'foo.txt'
    link.click()
    window.URL.revokeObjectURL(link.href)
  } else {
    console.log ('Error code', r.res.status)
    console.log ('No eligible objects to download')
  }
})
})
</script>
</body>
</html>
```

References

- For more information about the complete sample code for conditional download, visit [GitHub](#).
- For the API operation that is called to perform conditional download, see [Get Object](#).

6.6. Manage objects

6.6.1. Overview

Object Storage Service (OSS) SDK for Browser.js supports multiple operations to manage your buckets.

You can choose a method based on your business requirements.

- You can list the information about all objects in a bucket or objects and subdirectories in a specified directory. For more information, see [List objects](#).
- You can copy and replicate an object from a source bucket to the same bucket or another bucket that is located in the same region. For more information, see [Copy an object](#).
- To prevent an object from overwriting objects that have the same name during upload or replication, you can disable overwriting objects with the same names. For more information, see [Prevent objects from being overwritten by objects with the same names](#).
- To access an Archive or Cold Archive object, you must first restore the object. For more information, see [Restore an object](#).
- To quickly access objects that are frequently accessed in a bucket, you can use symbolic links. For more information, see [Manage symbolic links](#).
- To identify the purpose or attributes of objects, you can set the object metadata during uploads. For more information, see [Manage object metadata](#).

6.6.2. List objects

This topic describes how to list all objects in a bucket and objects and subdirectories within a specified directory of a bucket.

List all objects in a bucket

The following code provides an example on how to use the `list` function to list all objects in the current bucket:

```
const OSS = require('ali-oss');
const client = new OSS({
  // Set yourRegion to the endpoint of the region in which the bucket is located. For exam
  // ple, if your bucket is located in the China (Hangzhou) region, set yourRegion to oss-cn-han
  // gzhou.
  region: 'yourRegion',
  // Specify the temporary AccessKey pair obtained from STS.
  accessKeyId: 'yourAccessKeyId',
  accessKeySecret: 'yourAccessKeySecret',
  // Specify the security token obtained from STS.
  stsToken: 'yourSecurityToken',
  // Specify the name of the bucket. Example: examplebucket.
  bucket: "examplebucket",
});
async function list (dir) {
  try {
    // By default, a maximum of 1,000 objects are listed.
    let result = await client.list();
    console.log(result);
    // The list operation continues from the last object that was stopped in the previous l
    ist operation.
    if (result.isTruncated) {
      let result = await client.list({ marker : result.nextMarker });
    }
    // List the objects whose names start with the prefix my-.
    let result = await client.list({
      prefix: 'my-'
    });
    console.log(result);
    // List all objects whose names start with the prefix my- and come after the marker my-
    object.
    let result = await client.list({
      prefix: 'my-',
      marker: 'my-object'
    });
    console.log(result);
  } catch (e) {
    console.log(e);
  }
}
list();
```

List objects and subfolders in a specified folder

OSS does not use a hierarchical structure for objects but instead uses a flat structure. All elements are stored as objects in buckets. However, OSS supports directories as a concept to group objects and simplify management. A directory is an object whose name ends with a forward slash (/) and whose size is 0. This object can be uploaded and downloaded. The console shows any object whose name ends with a forward slash (/) as a directory. You can use the `Delimiter` and `Prefix` parameters together to simulate directories in the following ways:

- If `Prefix` is set to a directory name in the request, the objects whose names contain this prefix are listed, including all objects and subdirectories in the directory.
- If you also set the `Delimiter` parameter to a forward slash (/) in the request, only objects and subdirectories in the directory are listed. The subdirectories are listed in the `SubDir` element, excluding recursive objects and directories in these subdirectories.

Assume that the following objects are stored in a specific bucket:

```
foo/x
foo/y
foo/bar/a
foo/bar/b
foo/hello/C/1
foo/hello/C/2
...
foo/hello/C/9999
```

The following code provides an example on how to use the `listDir` function to list objects and subdirectories in a specific directory:

```
const OSS = require('ali-oss');
const client = new OSS({
  // Set yourRegion to the endpoint of the region in which the bucket is located. For example, if your bucket is located in the China (Hangzhou) region, set yourRegion to oss-cn-hangzhou.
  region: 'yourRegion',
  // Specify the temporary AccessKey pair obtained from STS.
  accessKeyId: 'yourAccessKeyId',
  accessKeySecret: 'yourAccessKeySecret',
  // Specify the security token obtained from STS.
  stsToken: 'yourSecurityToken',
  // Specify the name of the bucket. Example: examplebucket.
  bucket: "examplebucket",
});
async function listDir (dir) {
  try {
    let result = await client.list({
      prefix: dir,
      // Set the delimiter to a forward slash (/).
      delimiter: '/'
    });
    result.prefixes.forEach(function (subDir) {
      console.log('SubDir: %s', subDir);
    });
    result.objects.forEach(function (obj) {
      console.log('Object: %s', obj.name);
    });
  } catch (e) {
    console.log(e);
  }
}
listDir()
```

The following code shows the returned list results:

```
> listDir('foo/')
=> SubDir: foo/bar/
    SubDir: foo/hello/
    Object: foo/x
    Object: foo/y
> listDir('foo/bar/')
=> Object: foo/bar/a
    Object: foo/bar/b
> listDir('foo/hello/C/')
=> Object: foo/hello/C/1
    Object: foo/hello/C/2
    ...
    Object: foo/hello/C/9999
```

References

For more information about the API operation that is called to used to list objects, see [GetBucket \(ListObjects\)](#).

6.6.3. Copy an object

Copy an object within a bucket

The following code provides an example on how to copy an object within the same bucket:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8" />
  <title>Document</title>
</head>
<body>
  <button id='upload'>Upload An Object</button>
  <button id='copy'>Copy An Object</button>
  <!-- import an SDK file -->
  <script type="text/javascript" src="https://gosspublic.alicdn.com/aliyun-oss-sdk-6.16.0.min.js"></script>
  <script type="text/javascript">
    const client = new OSS({
      // Set yourRegion to the endpoint of the region in which the bucket is located. For
      // example, if your bucket is located in the China (Hangzhou) region, set yourRegion to oss-cn-
      // hangzhou.
      region: 'yourRegion',
      // Specify the temporary AccessKey pair obtained from STS.
      accessKeyId: 'yourAccessKeyId',
      accessKeySecret: 'yourAccessKeySecret',
      // Specify the security token obtained from STS.
      stsToken: 'yourSecurityToken',
      // Specify the name of the bucket. Example: examplebucket.
      bucket: "examplebucket",
    });
    const upload = document.getElementById('upload')
    const copy = document.getElementById('copy')
    // Specify the object to upload.
    const file = new Blob(['examplecontent'])
    // Specify the full path of the object. Example: exampledir/exampleobject.txt.
    const fileName = 'exampledir/exampleobject.txt'
    // Upload the object.
    upload.addEventListener('click', () => {
      const result = client.put(fileName, file).then(r => console.log(r))
    })
    // Copy the object.
    copy.addEventListener('click', () => {
      // Specify the name of the destination object.
      client.copy('newexampleobject.txt', fileName
        // Set HTTP headers and metadata for the destination object.
        // {
          // Specify the headers parameter to set HTTP headers for the destination object
          // . If the headers parameter is not specified, the HTTP headers of the destination object are
          // the same as those of the source object. The HTTP headers of the source object are copied.
```

```

        // headers:{
        // 'Cache-Control': 'no-cache',
        // If the ETag value of the source object is the same as the ETag value specified in the request, OSS copies the object and returns 200 OK.
        // 'if-match': '5B3C1A2E053D763E1B002CC607C5***',
        // If the ETag value of the source object is different from the ETag value specified in the request, OSS copies the object and returns 200 OK.
        // 'if-none-match': '5B3C1A2E053D763E1B002CC607C5***',
        // If the specified time is earlier than the modified time of the source object, OSS copies the object and returns 200 OK.
        // 'if-modified-since': '2021-12-09T07:01:56.000Z',
        // If the specified time is later than the modified time of the source object, OSS copies the object and returns 200 OK.
        // 'if-unmodified-since': '2021-12-09T07:01:56.000Z',
        // Set the access control list (ACL) of the destination object to private. This indicates that only the object owner and authorized users can read and write the object. Other users have no permissions on the object.
        // 'x-oss-object-acl': 'private',
        // Add tags to the destination object. You can add multiple tags to the object at the same time.
        // 'x-oss-tagging': 'Tag1=1&Tag2=2',
        // Specifies whether the CopyObject operation overwrites the object of the same name. Set x-oss-forbid-overwrite to true, which indicates that the object with the same name in the bucket cannot be overwritten.
        // 'x-oss-forbid-overwrite': 'true',
    },
    // Specify the meta parameter to set the metadata for the destination object. If the meta parameter is not specified, the metadata of the object is the same as that of the source object. The metadata of the source object is copied.
    // meta:{
    //   location: 'hangzhou',
    //   year: 2015,
    //   people: 'mary'
    // }
  }).then(r => {
    console.log(r.res.status)
  })
})
</script>
</body>
</html>

```

Copy an object across buckets

The following code provides an example on how to copy the `srcobject.txt` object from the source bucket named `srcbucket` to the `destobject.txt` object in the destination bucket named `destbucket` :

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <title>Document</title>
  </head>
  <body>
    <button id="copy">Copy An Object</button>
    <!-- import an SDK file -->
    <script
      type="text/javascript"
      src="https://gosspublic.alicdn.com/aliyun-oss-sdk-6.16.0.min.js"
    ></script>
    <script type="text/javascript">
      const client = new OSS({
        // Set yourRegion to the endpoint of the region in which the bucket is located. For
        // example, if your bucket is located in the China (Hangzhou) region, set yourRegion to oss-
        // cn-hangzhou.
        region: 'yourRegion',
        // Specify the temporary AccessKey pair obtained from STS.
        accessKeyId: 'yourAccessKeyId',
        accessKeySecret: 'yourAccessKeySecret',
        // Specify the security token obtained from STS.
        stsToken: 'yourSecurityToken',
        // Specify the name of the destination bucket.
        bucket: "destbucket",
      });
      const copy = document.getElementById("copy");
      const fileContent = Array(1024 * 1024 * 5)
        .fill("a")
        .join("");
      const file = new File([fileContent], fileName);
      copy.addEventListener("click", () => {
        client
          .copy(
            // Specify the name of the destination object.
            "destobject.txt",
            // Specify the name of the source object.
            "srcobject.txt",
            // Specify the name of the source bucket.
            "sourcebucket"
          )
          .then((r) => console.log(r));
      });
    </script>
  </body>
</html>
```

References

- For more information about the complete sample code that is used to copy an object, visit [GitHub](#).
- For more information about the API operation that is called to copy an object, see [CopyObject](#).

6.6.4. Set ACLs

Object Storage Service (OSS) allows you to set access control lists (ACLs) for objects. This way, you can conveniently control access to your resources.

For more information about access control, see [Overview](#).

You can specify one of the following ACLs for an object:

- **Default:** The object inherits the ACL of the bucket to which the object belongs. The ACL of the object is the same as the ACL of the bucket in which the object is stored.
- **Public read/write:** All users, including anonymous users, can perform read and write operations on the object.
- **Public read:** Only the bucket owner can perform write operations on the object. Other users, including anonymous users, can perform only read operations on the object.
- **Private:** Only the owner and authorized users of the bucket in which the object is stored can perform read and write operations on the object. Other users, including anonymous users, cannot access the object without authorization.

By default, the ACL of the object is the same as that of the bucket where the object is stored. After you upload an object, you can use `putACL` to set the ACL for the object.

```
let OSS = require('ali-oss')
let client = new OSS({
  region: '<Your region>'
  accessKeyId: '<Your AccessKeyId>',
  accessKeySecret: '<Your AccessKeySecret>',
  bucket: '<Your bucket name>'
});
async function getACL () {
  try {
    let result = await client.getACL('my-object');
    console.log(result.acl); // default
    await client.putACL('my-object', 'public-read');
    let result = await client.getACL('my-object');
    console.log(result.acl); // public-read
  } catch (e) {
    console.log(e);
  }
}
getACL();
```



6.6.5. Prevent objects from being overwritten by objects with the same names

By default, if you upload an object that has the same name as an existing object, the existing object is overwritten by the uploaded object. This topic describes how to configure the `x-oss-forbid-overwrite` request header to prevent objects from being overwritten by objects with the same names when you copy objects or perform simple upload or multipart upload.

Simple upload

The following code provides an example on how to prevent objects from being overwritten by objects with the same names in simple upload:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8" />
  <title>Document</title>
</head>
<body>
  <button id='upload'>Upload An Object</button>
  <!-- import an SDK file -->
  <script type="text/javascript" src="https://gosspublic.alicdn.com/aliyun-oss-sdk-6.16.0.min.js"></script>
  <script type="text/javascript">
    const client = new OSS({
      // Set yourRegion to the endpoint of the region in which the bucket is located. For
      // example, if your bucket is located in the China (Hangzhou) region, set yourRegion to oss-cn-
      // hangzhou.
      region: 'yourRegion',
      // Specify the temporary AccessKey pair obtained from STS.
      accessKeyId: 'yourAccessKeyId',
      accessKeySecret: 'yourAccessKeySecret',
      // Specify the security token obtained from STS.
      stsToken: 'yourSecurityToken',
      // Specify the name of the bucket. Example: examplebucket.
      bucket: "examplebucket",
    });
    const upload = document.getElementById('upload')
    // Specify the local file to upload.
    const fileContent = Array(1024 * 1024 * 10).fill('a').join('')
    const fileName = 'example.txt'
    const file = new Blob([fileContent])
    // Specify whether existing objects are overwritten by objects with the same names when
    // you upload the object. Set x-oss-forbid-overwrite to true, which indicates that the object
    // with the same name in the bucket cannot be overwritten. If an object with the same name exists
    // in the bucket, OSS returns an error.
    const headers = {
      'x-oss-forbid-overwrite': true
    }
    upload.addEventListener('click', () => {
      // Upload the local file.
      client.put(fileName, file, { headers }).then(r => console.log(r))
    })
  </script>
</body>
</html>
```

Copy an object

The following code provides an example on how to prevent objects from being overwritten by objects with the same names when you copy an object:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8" />
  <title>Document</title>
</head>
<body>
  <button id='upload'>Upload</button>
  <!-- import an SDK file -->
  <script type="text/javascript" src="https://gosspublic.alicdn.com/aliyun-oss-sdk-6.16.0.min.js"></script>
  <script type="text/javascript">
    const client = new OSS({
      // Set yourRegion to the endpoint of the region in which the bucket is located. For
      // example, if your bucket is located in the China (Hangzhou) region, set yourRegion to oss-cn-
      // hangzhou.
      region: 'yourRegion',
      // Specify the temporary AccessKey pair obtained from STS.
      accessKeyId: 'yourAccessKeyId',
      accessKeySecret: 'yourAccessKeySecret',
      // Specify the security token obtained from STS.
      stsToken: 'yourSecurityToken',
      // Specify the name of the bucket. Example: examplebucket.
      bucket: "examplebucket",
    });
    const upload = document.getElementById('upload')
    // Specify the object to copy.
    const fileContent = Array(1024 * 1024 * 5).fill('a').join('')
    const fileName = 'example.txt'
    const file = new Blob([fileContent])
    // Specify whether existing objects are overwritten by objects with the same names when
    // you upload the object. Set x-oss-forbid-overwrite to true, which indicates that the object
    // with the same name in the bucket cannot be overwritten. If an object with the same name exists
    // in the bucket, OSS reports an error.
    const headers = {
      'x-oss-forbid-overwrite': true
    }
    upload.addEventListener('click', () => {
      // Copy the object.
      client.copy('newName', fileName, {headers}).then(r=>console.log(r))
    })
  </script>
</body>
</html>
```

Multipart upload

The following code provides an example on how to prevent objects from being overwritten by objects with the same names in multipart upload:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8" />
  <title>Document</title>
</head>
<body>
  <button id='upload'>Upload An Object</button>
  <!-- import an SDK file -->
  <script type="text/javascript" src="https://gosspublic.alicdn.com/aliyun-oss-sdk-6.16.0.min.js"></script>
  <script type="text/javascript">
    const client = new OSS({
      // Set yourRegion to the endpoint of the region in which the bucket is located. For
      // example, if your bucket is located in the China (Hangzhou) region, set yourRegion to oss-cn-
      // hangzhou.
      region: 'yourRegion',
      // Specify the temporary AccessKey pair obtained from STS.
      accessKeyId: 'yourAccessKeyId',
      accessKeySecret: 'yourAccessKeySecret',
      // Specify the security token obtained from STS.
      stsToken: 'yourSecurityToken',
      // Specify the name of the bucket. Example: examplebucket.
      bucket: "examplebucket",
    });
    const upload = document.getElementById('upload')
    // Specify the object to upload.
    const fileContent = Array(1024 * 1024 * 5).fill('a').join('')
    const fileName = 'example.txt'
    const file = new Blob([fileContent])
    // Specify whether existing objects are overwritten by objects with the same names when
    // you upload the object. Set x-oss-forbid-overwrite to true, which indicates that the object
    // with the same name in the bucket cannot be overwritten. If an object with the same name exists
    // in the bucket, OSS reports an error.
    const headers = {
      'x-oss-forbid-overwrite': true
    }
    upload.addEventListener('click', () => {
      // Upload the object by using multipart upload.
      // client.multipartUpload('newName', file, {headers}).then(r=>console.log(r))
    })
  </script>
</body>
</html>
```

References

- For more information about the API operation that you can call to perform simple upload, see [Put Object](#).
- For more information about the API operation that you can call to copy an object, see [Copy Object](#).

- The `multipartUpload` method called by OSS SDK for Browser.js encapsulates three API operations.
 - For more information about `InitiateMultipartUpload`, see [InitiateMultipartUpload](#).
 - For more information about `UploadPart`, see [UploadPart](#).
 - For more information about `CompleteMultipartUpload`, see [CompleteMultipartUpload](#).

6.6.6. Convert the storage classes of an object

OSS provides the following storage classes to cover various data storage scenarios from hot data to cold data: Standard, Infrequent Access (IA), Archive and Cold Archive. This topic describes how to convert the storage class of an object.

Sample code

- Convert the storage class of an object from Standard or Infrequent Access (IA) to Archive or Cold Archive

The following code provides an example on how to convert the storage class of an object from Standard or IA to Archive:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8" />
  <title>Document</title>
</head>
<body>
  <button id='upload'>Upload An Object</button>
  <button id='copy'>Convert Storage Class</button>
  <!-- import an SDK file -->
  <script type="text/javascript" src="https://gosspublic.alicdn.com/aliyun-oss-sdk-6.16.0.min.js"></script>
  <script type="text/javascript">
    const client = new OSS({
      // Set yourRegion to the endpoint of the region in which the bucket is located. For example, if your bucket is located in the China (Hangzhou) region, set yourRegion to oss-cn-hangzhou.
      region: 'yourRegion',
      // Specify the temporary AccessKey pair obtained from STS.
      accessKeyId: 'yourAccessKeyId',
      accessKeySecret: 'yourAccessKeySecret',
      // Specify the security token obtained from STS.
      stsToken: 'yourSecurityToken',
      // Specify the name of the bucket. Example: examplebucket.
      bucket: "examplebucket",
    });
    const upload = document.getElementById('upload')
    const copy = document.getElementById('copy')
    // Specify the object to upload.
    const file = new Blob(['examplecontent'])
    // Specify the full path of the object to upload. Example: exampledir/exampleobject.txt.
    const fileName = 'exampledir/exampleobject.txt'
    // Upload the object.
```

```

upload.addEventListener('click', () => {
  client.put(fileName, file).then(r => console.log(r))
})
// Convert the storage class of the object by copying the object.
copy.addEventListener('click', () => {
  // Specify the name of the destination object. Example: newexampleobject.txt.
  client.copy('newexampleobject.txt', fileName, {
    headers: {
      // Specify the storage class of the destination object as Archive. If you want
      // to convert the storage class of the source object to Cold Archive, replace Archive with C
      // old Archive.
      'x-oss-storage-class': 'Archive'
    }
  }).then(r => {
    console.log(r.res.status)
  })
})
</script>
</body>
</html>

```

- Convert the storage class of an object from Archive or Cold Archive to Standard or IA

The following code provides an example of how to convert the storage class of an object from Archive or Cold Archive to Standard or IA:

```

<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <title>Document</title>
  </head>
  <body>
    <button id="check">Check Storage Class</button>
    <button id="restore">Restore An Object</button>
    <button id="change">Convert Storage Class</button>
    <!-- import an SDK file -->
    <script
      type="text/javascript"
      src="https://gosspublic.alicdn.com/aliyun-oss-sdk-6.16.0.min.js"
    ></script>
    <script type="text/javascript">
      const client = new OSS({
        // Set yourRegion to the endpoint of the region in which the bucket is located. F
        // or example, if your bucket is located in the China (Hangzhou) region, set yourRegion to o
        // ss-cn-hangzhou.
        region: 'yourRegion',
        // Specify the temporary AccessKey pair obtained from STS.
        accessKeyId: 'yourAccessKeyId',
        accessKeySecret: 'yourAccessKeySecret',
        // Specify the security token obtained from STS.
        stsToken: 'yourSecurityToken',
        // Specify the name of the bucket. Example: examplebucket.
        bucket: "examplebucket",

```

```

});
const check = document.getElementById("check");
const change = document.getElementById("change");
const restore = document.getElementById("restore");
check.addEventListener("click", () => {
    console.log ("Query the storage class of the object")
    // Check the header of the object in the developer tools of your browser to query
the storage class of the object.
    client.head("newName.txt").then((r) => console.log(r));
});
// Restore the object.
restore.addEventListener("click", () => {
    // Specify the name of the destination object. Example: newexampleobject.txt.
    client.restore("newexampleobject.txt").then((r) => {
        console.log(r);
        console.log ("Start to restore the object")
    });
});
// Start to convert the storage class of the object after the object is restored.
change.addEventListener("click", () => {
    // The duration of the restoration is determined by the object size.
    console.log ("start to convert the storage class of the object")
    client
        .copy("newexampleobject.txt", "newName.txt", {
            // Specify that the storage class of the object is converted to IA. If you wa
nt to convert the storage class of the object to Standard, replace IA with Standard.
            headers: { "x-oss-storage-class": "IA" },
        })
        .then((r) => console.log(r))
        // The common reason why an error is reported when you convert the storage clas
s is that the object is still being restored.
        .catch((e) => console.log ("conversion error:", e))
    );
});
</script>
</body>
</html>

```

References

- For more information about the complete sample code that is used to convert the storage class of an object, visit [GitHub](#).
- For more information about the API operation that is called to convert the storage class of an object, see [CopyObject](#).

6.6.7. Single-connection bandwidth throttling

When you access objects in Object Storage Service (OSS) by using a client, especially a client on which traffic throttling is difficult to implement, a large amount of bandwidth of OSS may be occupied, which obstructs other applications from accessing OSS. To prevent this issue, you can use the single-connection bandwidth throttling feature that is provided by OSS to throttle the bandwidth for object downloads. This way, sufficient bandwidth can be reserved for other applications to access OSS.

Sample code

The following code provides an example on how to configure bandwidth throttling when you download an object by using a signed URL:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8" />
  <title>Document</title>
</head>
<body>
  <button id="download">Download</button>
  <!-- Import the SDK file -->
  <script type="text/javascript" src="https://gosspublic.alicdn.com/aliyun-oss-sdk-6.16.0.min.js"></script>
  <script type="text/javascript">
    const client = new OSS({
      // Set yourRegion to the region in which the bucket is located. For example, if your bucket is located in the China (Hangzhou) region, set yourRegion to oss-cn-hangzhou.
      region: 'yourRegion',
      // Specify the temporary AccessKey pair that is obtained from Security Token Service (STS). The AccessKey pair consists of an AccessKey ID and an AccessKey secret.
      accessKeyId: 'yourAccessKeyId',
      accessKeySecret: 'yourAccessKeySecret',
      // Specify the security token that is obtained from STS.
      stsToken: 'yourSecurityToken',
      // Specify the name of the bucket. Example: examplebucket.
      bucket: "examplebucket",
    });
    const download = document.getElementById("download");
    download.addEventListener("click", () => {
      console.log(client.signatureUrl('bigData', {
        // Set the download speed limit to 100 KB/s.
        trafficLimit: 1024 * 8 * 100,
        responsel: {
          'content-type': 'attachment'
        }
      }));
    });
  </script>
</body>
</html>
```

References

For more information about the complete sample code of single-connection bandwidth throttling, visit [GitHub](#).

6.6.8. Restore an object

You must restore an Archive or Cold Archive object before you can read it. This topic describes how to restore an Archive or Cold Archive object.

Usage notes

- When you use packaging tools such as Webpack and Browserify, install OSS SDK for Browser.js by running the `npm install ali-oss` command.
- OSS SDK for Browser.js is usually used in browsers. To prevent your AccessKey pair from being exposed, we recommend that you use a temporary access credential generated by Security Token Service (STS) to access OSS.

For more information about how to set up STS, see [Use a temporary credential provided by STS to access OSS](#) in OSS Developer Guide. You can call the [AssumeRole](#) operation or use STS SDKs for various programming languages to obtain a temporary access credential. For more information, see [STS SDK overview](#). A temporary access credential contains a security token and a temporary AccessKey pair that consists of an AccessKey ID and an AccessKey secret.

Restoration

If you want to read an Archive object or a Cold Archive object, you must restore the object in advance. It takes several minutes to restore an Archive object and takes several hours to restore a Cold Archive object.

The following section describes the status of an Archive object or a Cold Archive object throughout the restoration process:

1. By default, an Archive or a Cold Archive object is in the frozen state before restoration.
2. After you submit a restore request, the object is in the restoring state.
3. After the server completes the restore task, the object enters the restored state and you can read the object.
 - Archive objects

For Archive objects, the restored state lasts 24 hours by default. During the 24 hours, if you call `RestoreObject` again, the restored state is extended by 24 hours. You can extend the restored state to seven days by calling `RestoreObject` six times during the first 24 hours of restored state. You can also configure the duration of the restored state in days by calling `RestoreObject` once. You can specify a duration of at most seven days.
 - Cold Archive objects

For Cold Archive objects, you can specify the duration of restored state and restoration priority. The duration of the restored state must be at least one day and at most 365 days. The time required to restore a Cold Archive object to the readable state is determined based on the restoration priority of the object:

 - Expedited: The object is restored within one hour.
 - Standard: The object is restored within two to five hours. If the `JobParameters` element is not passed in, the default restoration priority is Standard.
 - Bulk: The object is restored within five to twelve hours.
4. After the restored state expires, the object returns to the frozen state.

Restore an Archive object

The following code provides an example on how to restore an Archive object:


```

<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8" />
  <title>Document</title>
</head>
<body>
  <button id='upload'>Upload</button>
  <button id='restore'>Restore Object</button>
  <!-- Import the SDK file -->
  <script type="text/javascript" src="https://gosspublic.alicdn.com/aliyun-oss-sdk-6.16.0.min.js"></script>
  <script type="text/javascript">
    const client = new OSS({
      // Set yourRegion to the endpoint of the region in which the bucket is located. For
      // example, if your bucket is located in the China (Hangzhou) region, set yourRegion to oss-cn-
      // hangzhou.
      region: 'yourRegion',
      // Specify the temporary AccessKey pair obtained from STS. The AccessKey pair consists
      // of an AccessKey ID and an AccessKey secret.
      accessKeyId: 'yourAccessKeyId',
      accessKeySecret: 'yourAccessKeySecret',
      // Specify the security token obtained from STS.
      stsToken: 'yourSecurityToken',
      // Specify the bucket name. Example: examplebucket.
      bucket: "examplebucket",
    });
    const upload = document.getElementById('upload')
    const restore = document.getElementById('restore')
    // Specify the content of the object to upload.
    const file = new Blob(['examplecontent'])
    // Specify the name of the object that is uploaded to the bucket.
    const fileName = 'example.txt'
    // Upload the object and specify its storage class as Archive.
    upload.addEventListener('click', () => {
      client.put(fileName, file, {
        headers: {
          'x-oss-storage-class': 'Archive'
        }
      }).then(r => console.log(r))
    })
    // Restore the Archive object.
    restore.addEventListener('click', () => {
      client.restore(fileName).then(r => console.log(r))
    })
  </script>
</body>
</html>

```

Restore a Cold Archive object

The following code provides an example on how to restore a Cold Archive object:

```

<!DOCTYPE html>

```

```

<:DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8" />
  <title>Document</title>
</head>
<body>
  <button id='upload'>Upload</button>
  <button id='restore'>Restore Object</button>
  <!-- Import the SDK file -->
  <script type="text/javascript" src="https://gosspublic.alicdn.com/aliyun-oss-sdk-6.16.0.min.js"></script>
  <script type="text/javascript">
    const client = new OSS({
      // Set yourRegion to the endpoint of the region in which the bucket is located. For
      // example, if your bucket is located in the China (Hangzhou) region, set yourRegion to oss-cn-
      // hangzhou.
      region: 'yourRegion',
      // Specify the temporary AccessKey pair obtained from STS. The AccessKey pair consists
      // of an AccessKey ID and an AccessKey secret.
      accessKeyId: 'yourAccessKeyId',
      accessKeySecret: 'yourAccessKeySecret',
      // Specify the security token obtained from STS.
      stsToken: 'yourSecurityToken',
      // Specify the bucket name. Example: examplebucket.
      bucket: "examplebucket",
    });
    const upload = document.getElementById('upload')
    const restore = document.getElementById('restore')
    // Specify the content of the object to upload.
    const file = new Blob(['examplecontent'])
    // Specify the name of the object that is uploaded to the bucket.
    const fileName = 'example.txt'
    // Upload the object and specify its storage class as Cold Archive.
    upload.addEventListener('click', () => {
      client.put(fileName, file, {
        headers: {
          'x-oss-storage-class': 'ColdArchive'
        }
      }).then(r => console.log(r))
    })
    // type is used to specify the storage class of the object as Cold Archive.
    // JobParameters is used to specify the restoration priority of the object.
    // Expedited: The object is restored within one hour.
    // Standard: The object is restored within two to five hours. If the JobParameters element
    // is not passed in, the default restoration priority is Standard.
    // Bulk: The object is restored within five to twelve hours.
    // Days is used to specify the duration in which the object can remain in the restored state.
    // Unit: days. Valid values: 1 to 7.
    // Restore the Cold Archive object.
    restore.addEventListener('click', () => {
      client.restore(fileName, {type: 'ColdArchive', JobParameters: 'Standard', Days: 2}).then(r
=> console.log(r))
    })
  </script>

```

```
</body>
</html>
```

References

- For more information about the complete sample code that is used to restore an Archive or Cold Archive object, visit [GitHub](#).
- For more information about the API operation that you can call to restore an Archive or Cold Archive object, see [RestoreObject](#).

6.6.9. Manage object metadata

Object metadata includes HTTP headers and user metadata. You can configure the metadata of an object when you upload the object to Object Storage Service (OSS). The metadata of an object is uploaded and downloaded together with the object.

Sample code

The following code provides an example on how to specify object metadata during uploads and how to update and view object metadata after uploads:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8" />
  <title>Document</title>
</head>
<body>
  <button id='upload'>Upload An Object</button>
  <button id='update'>Modify Object Metadata</button>
  <button id='check'>View Object Metadata</button>
  <!-- import an SDK file -->
  <script type="text/javascript" src="https://gosspublic.alicdn.com/aliyun-oss-sdk-6.16.0.min.js"></script>
  <script type="text/javascript">
    const client = new OSS({
      // Set yourRegion to the endpoint of the region in which the bucket is located. For
      // example, if your bucket is located in the China (Hangzhou) region, set yourRegion to oss-cn-
      // hangzhou.
      region: 'yourRegion',
      // Specify the temporary AccessKey pair obtained from STS.
      accessKeyId: 'yourAccessKeyId',
      accessKeySecret: 'yourAccessKeySecret',
      // Specify the security token obtained from STS.
      stsToken: 'yourSecurityToken',
      // Specify the name of the bucket. Example: examplebucket.
      bucket: "examplebucket",
    });
    const upload = document.getElementById('upload')
    const update = document.getElementById('update')
    const check = document.getElementById('check')
    // Specify the local file to upload.
    const file = new Blob(['examplecontent'])
```

```
// Specify the name of the object that is uploaded to the bucket.
const fileName = 'exampleobject.txt'
// Upload the object.
upload.addEventListener('click', () => {
  client.put(fileName, file, {
    // Configure the user metadata of the object. You can configure multiple metadata attributes for an object. However, the total size of the object metadata cannot exceed 8 KB.

    meta: {
      year: 2020,
      people: 'eliot'
    }
  }).then(r => console.log(r))
})
// Modify the object metadata.
update.addEventListener('click', () => {
  client.putMeta(fileName, {
    year: 2021,
    people: 'evan'
  })
  .then(r => {
    console.log(r)
  })
})
// View the object metadata.
check.addEventListener('click', () => {
  client.head(fileName).then(m => console.log(m))
})
</script>
</body>
</html>
```

References

- For more information about the complete sample code for object metadata management, visit [GitHub](#).
- For more information about the operation that is called to configure or modify object metadata during uploads, see [Put Object](#).
- For more information about the API operation that is called to view object metadata, see [Head Object](#).

6.6.10. Manage symbolic links

Symbolic links can be used to access objects that are frequently used in buckets. After you create a symbolic link for an object, you can use the symbolic link to access the object.

Create a symbolic link

The following code provides an example on how to create a symbolic link:

```

<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8" />
  <title>Document</title>
</head>
<body>
  <button id='upload'>Upload An Object</button>
  <button id='symlink'>Create A Symbolic Link</button>
  <!-- import an SDK file -->
  <script type="text/javascript" src="https://gosspublic.alicdn.com/aliyun-oss-sdk-6.16.0.min.js"></script>
  <script type="text/javascript">
    const client = new OSS({
      // Set yourRegion to the endpoint of the region in which the bucket is located. For
      // example, if your bucket is located in the China (Hangzhou) region, set yourRegion to oss-cn-
      // hangzhou.
      region: 'yourRegion',
      // Specify the temporary AccessKey pair obtained from STS.
      accessKeyId: 'yourAccessKeyId',
      accessKeySecret: 'yourAccessKeySecret',
      // Specify the security token obtained from STS.
      stsToken: 'yourSecurityToken',
      // Specify the name of the bucket. Example: examplebucket.
      bucket: "examplebucket",
    });
    const upload = document.getElementById('upload')
    const symlink = document.getElementById('symlink')
    const getSymlink = document.getElementById("getSymlink")
    // Specify the object to upload.
    const file = new Blob(['examplecontent'])
    // Specify the name of the object after it is uploaded to the bucket.
    const fileName = 'exampleobject.txt'
    // Upload the object.
    upload.addEventListener('click', () => {
      client.put(fileName, file).then(r => console.log(r))
    })
    // Create a symbolic link named symlink.txt.
    symlink.addEventListener('click', () => {
      client.putSymlink('symlink.txt', fileName).then(r => console.log(r))
    })
  </script>
</body>
</html>

```

Query the name of the object to which a symbolic link points

To query the name of the object to which a symbolic link points, you must have read permissions on the symbolic link. The following code provides an example on how to query the name of the object to which the symbolic link points:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8" />
  <title>Document</title>
</head>
<body>
  <button id='symlink'>Query A Symbolic Link</button>
  <!-- import an SDK file -->
  <script type="text/javascript" src="https://gosspublic.alicdn.com/aliyun-oss-sdk-6.16.0.min.js"></script>
  <script type="text/javascript">
    const client = new OSS({
      // Set yourRegion to the endpoint of the region in which the bucket is located. For
      // example, if your bucket is located in the China (Hangzhou) region, set yourRegion to oss-cn-
      // hangzhou.
      region: 'yourRegion',
      // Specify the temporary AccessKey pair obtained from STS.
      accessKeyId: 'yourAccessKeyId',
      accessKeySecret: 'yourAccessKeySecret',
      // Specify the security token obtained from STS.
      stsToken: 'yourSecurityToken',
      // Specify the name of the bucket. Example: examplebucket.
      bucket: "examplebucket",
    });
    const symlink = document.getElementById('symlink')
    const getSymlink = document.getElementById("getSymlink")
    // Obtain the name of the object to which the symbolic link points.
    getSymlink.addEventListener('click', () => {
      client.getSymlink('symlink.txt').then(r => console.log(r))
    })
  </script>
</body>
</html>
```

References

- Create a symbolic link
 - For more information about the complete sample code for creating a symbolic link, visit [GitHub](#).
 - For more information about the API operation that is called to create a symbolic link, see [PutSymlink](#).
- Obtain the object to which a symbolic link points
 - For more information about the complete sample code that is used to obtain the object to which a symbolic link points, visit [GitHub](#).
 - For more information about the API operation that is called to query the object to which a symbolic link points, see [GetSymlink](#).

6.6.11. Delete objects

You can delete a single or multiple objects that you no longer need.

 **Warning** Deleted objects cannot be recovered. Exercise caution when you delete objects.

Delete a single object

The following code provides an example on how to delete a single object:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8" />
  <title>Document</title>
</head>
<body>
  <button id="delete">Delete An Object</button>
  <!-- import an SDK file -->
  <script type="text/javascript" src="https://gosspublic.alicdn.com/aliyun-oss-sdk-6.16.0.min.js"></script>
  <script type="text/javascript">
    const client = new OSS({
      // Set yourRegion to the endpoint of the region in which the bucket is located. For example, if your bucket is located in the China (Hangzhou) region, set yourRegion to oss-cn-hangzhou.
      region: 'yourRegion',
      // Specify the temporary AccessKey pair obtained from STS.
      accessKeyId: 'yourAccessKeyId',
      accessKeySecret: 'yourAccessKeySecret',
      // Specify the security token obtained from STS.
      stsToken: 'yourSecurityToken',
      // Specify the name of the bucket. Example: examplebucket.
      bucket: "examplebucket",
    });
    const deleteSingle = document.getElementById("delete");
    // Delete a single object.
    deleteSingle.addEventListener("click", async () => {
      // Specify the name of the object to delete. // Specify the full path of the object.
      // The path of the object cannot contain the bucket name. Example: exampledir/exampleobject.txt.
      let result = await client.delete('exampledir/exampleobject.txt');
      console.log(result);
    });
  </script>
</body>
</html>
```

Deletes multiple objects

The following code provides an example on how to delete multiple objects:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8" />
  <title>Document</title>
</head>
<body>
  <button id="deleteAll">Delete Objects</button>
  <!-- import an SDK file -->
  <script type="text/javascript" src="https://gosspublic.alicdn.com/aliyun-oss-sdk-6.16.0.min.js"></script>
  <script type="text/javascript">
    const client = new OSS({
      // Set yourRegion to the endpoint of the region in which the bucket is located. For example, if your bucket is located in the China (Hangzhou) region, set yourRegion to oss-cn-hangzhou.
      region: 'yourRegion',
      // Specify the temporary AccessKey pair obtained from STS.
      accessKeyId: 'yourAccessKeyId',
      accessKeySecret: 'yourAccessKeySecret',
      // Specify the security token obtained from STS.
      stsToken: 'yourSecurityToken',
      // Specify the name of the bucket. Example: examplebucket.
      bucket: "examplebucket",
    });
    const deleteAll = document.getElementById('deleteAll')
    // Delete multiple objects.
    deleteAll.addEventListener('click', async () => {
      // Specify the names of the objects to delete. Specify the full path of the objects.
      // The object names cannot contain the bucket name.
      let result = await client.deleteMulti(['object1.txt', 'object2.txt', 'object3.txt']);
      console.log(result);
      let result = await client.deleteMulti(['object1.txt', 'object2.txt', 'object3.txt'], {
        // Set the quiet parameter to specify whether to list all objects that are deleted.
        // If you set quiet to true, OSS does not return the list of the objects that are deleted. If you set quiet to false, OSS returns the list of the objects that are deleted.
        quiet: true
      });
    })
  </script>
</body>
</html>
```

References

- Delete a single object
 - For more information about the complete sample code that is used to delete an object, visit [GitHub](#).
 - For more information about the API operation that is called to delete an object, see [DeleteObject](#).
- Delete multiple objects

- For more information about the complete sample code that is used to delete multiple objects, visit [GitHub](#).
- For more information about the API operation that is called to delete multiple objects, see [DeleteMultipleObjects](#).

6.7. Map custom domain names

This topic describes how to map a custom domain name to a bucket.

Object Storage Service (OSS) allows you to map a custom domain name to an OSS bucket based on a CNAME record. You can use this method to migrate data to OSS. For example, you have the domain name `example.com`, and all of your images can be accessed by using URLs in the following format:

`http://img.example.com/x.jpg`. To access your images by using original URLs after you migrate the images to an OSS bucket, perform the following steps:

- Activate OSS and create a bucket.
- Modify the Domain Name System (DNS) configurations of the domain name. Add a CNAME record to map `img.example.com` to an OSS endpoint such as `my-bucket.oss-cn-hangzhou.aliyuncs.com`.
- Log on to the [OSS console](#) to map `img.example.com` to the created bucket.
- Upload an image to the OSS bucket.

You can use the original URL `http://img.example.com/x.jpg` to access the image that is stored in the OSS bucket. For more information about how to map custom domain names, see [Map custom domain names](#).

To use your custom domain name as the endpoint of a bucket when you use OSS SDK for Browser.js, set the value of `cname` to `true`.

```
let OSS = require('ali-oss')
let client = new OSS({
  endpoint: '<Your endpoint>',
  accessKeyId: '<Your AccessKeyId>',
  accessKeySecret: '<Your AccessKeySecret>',
  cname: true
});
```

 **Notice** If a custom domain name is mapped to a bucket, the domain name cannot be used to call the `list_buckets` operation.

6.8. Authorized access

This topic describes how to authorize temporary access to Object Storage Service (OSS) by using Security Token Service (STS) or a signed URL.

Use STS for temporary access authorization

You can use Alibaba Cloud STS to authorize temporary access to OSS. STS is a web service that provides temporary access tokens for users. You can use STS to grant a set of temporary access credentials that have a custom validity period and custom permissions to a third-party application or a RAM user managed by you. For more information about STS, see [What is STS?](#)

STS provides the following benefits:

- You need only to generate an access token and send the access token to a third-party application, instead of exposing your AccessKey pair to the third-party application. You can define the access permissions and validity period of this token.
- The token automatically expires after the validity period. Therefore, you do not need to manually revoke the access permissions of a token.

 **Note** For more information about how to use STS, see [Use a temporary credential provided by STS to access OSS](#) in OSS User Guide. You can call the [AssumeRole](#) operation or use STS SDKs for various programming languages to obtain temporary access credentials. For more information, see [STS SDK overview](#). The temporary access credentials consist of an AccessKey pair and a security token. The AccessKey pair contains an AccessKey ID and an AccessKey secret. The minimum validity period of temporary access credentials is 900 seconds. The maximum validity period of temporary access credentials is the maximum session duration specified for the current role. For more information, see [Specify the maximum session duration for a RAM role](#).

The following code provides an example on how to use Security Token Service (STS) to create a signed request:

```
// Obtain temporary access credentials from the STS that you set up.
fetch('http://your_sts_server/')
  .then(resp => resp.json())
  .then(result => {
    const store = new OSS({
      // Specify the temporary AccessKey pair obtained from STS. The AccessKey pair consists of an AccessKey ID and an AccessKey secret.
      accessKeyId: result.AccessKeyId,
      accessKeySecret: result.AccessKeySecret,
      // Specify the security token obtained from STS.
      stsToken: result.SecurityToken,
      // Specify the region of the bucket that you want to access. For example, if the requested bucket is located in the China (Hangzhou) region, set the region to oss-cn-hangzhou.
      region: 'oss-cn-hangzhou',
      // Specify the bucket name. Example: examplebucket.
      bucket: 'examplebucket'
    });
    // Generate a signed URL.
    // Specify the full path of the object that you want to access. Example: ossdemo.txt. The full path of the object cannot contain the bucket name.
    const url = store.signatureUrl('ossdemo.txt');
    console.log(url);
  })
```

Use a signed URL for temporary access authorization

 **Notice** To use a signed URL that contains custom parameters to access an object from a browser, make sure that the value of the Content-Type parameter contained in the URL is the same as the Content-Type value specified in the request. Otherwise, Object Storage Service (OSS) may report the SignatureDoesNotMatch error. For more information about how to configure Content-Type, see [How do I configure the Content-Type of objects?](#).

- **Generate a signed URL**

You can generate a signed URL and provide the URL to a visitor for temporary access. When you generate a signed URL, you can specify the validity period of the URL to limit the period of access from visitors. By default, the validity period of a signed URL is 1,800 seconds. The maximum validity period of a signed URL is 32,400 seconds.

- Generate a signed URL for an object

 **Note** name {String} specifies the name of the object stored in OSS. [expires] {Number} specifies the validity period of the URL. By default, the validity period is 1,800 seconds. For more information about other parameters, visit [GitHub](#).

The following code provides an example on how to generate a signed URL for an object:

```
const url = store.signatureUrl('exampleobject.txt');
console.log(url);
// Obtain the signed URL that is used to upload the exampleobject.txt object.
const url = store.signatureUrl('exampleobject.txt', {
  // Set the validity period of the URL to 3,600 seconds.
  expires: 3600,
  // Set the request method to PUT. By default, the request method is GET.
  method: 'PUT'
});
console.log(url);
const url = store.signatureUrl('exampleobject.txt', {
  expires: 3600,
  method: 'PUT',
  // Specify Content-Type.
  'Content-Type': 'text/plain; charset=UTF-8',
});
console.log(url);
// Obtain the signed URL that is used to download the exampleobject.txt object. By default, if you use the signed URL to access the object in a browser, the object is previewed but not downloaded.
const url = store.signatureUrl('exampleobject.txt', {
  expires: 3600,
  response: {
    'content-type': 'text/custom',
    // Obtain the signed URL that is used to download the exampleobject.txt object, and set the Content-Disposition header to attachment. This way, if you use the signed URL to access the object in a browser, the object is automatically downloaded, and you can specify the name of the downloaded object.
    // To preview the object when you use the signed URL to access the object in a browser, set the Content-Disposition header to inline and use the custom domain name that is mapped to the bucket to access the object.
    'content-disposition': 'attachment'
  }
});
console.log(url);
```

- Generate a signed object URL that includes Image Processing (IMG) parameters

```
// Obtain the signed URL that is used to access an image object named exampleobject.png
.
// Specify the full path of the object. The full path cannot contain the bucket name.
const url = store.signatureUrl('exampleobject.png', {
  // Set IMG parameters that are used to process the image.
  process: 'image/resize,w_200'
});
console.log(url);
// Set the validity period of the URL to 3,600 seconds.
const url = store.signatureUrl('exampleobject.png', {
  expires: 3600,
  process: 'image/resize,w_200'
});
console.log(url);
```

- Use a signed URL

After a signed URL is generated, you can use the URL to upload, preview, or download an object.

- Use a signed URL to upload an object

```
// Specify signatureUrl to the generated signed URL.
const url = 'signatureUrl';
var xhr = new XMLHttpRequest();
xhr.open('PUT', url, true);
xhr.onload = function () {
  // Add code for further operations.
};
xhr.send(null);
// xhr.send('string');
// xhr.send(new Blob());
// xhr.send(new Int8Array());
// xhr.send({ form: 'data' });
// xhr.send(document);
```

- Use a signed URL to preview or download an object

You can use the download attribute of the <a> HTML element or the window.open API to obtain the URL of an object.

References

- For the complete code that is used to authorize temporary access by using STS, visit [GitHub](#).
- For the complete code that is used to authorize temporary access by using a signed URL, visit [GitHub](#).

6.9. IMG

Image Processing (IMG) provided by Object Storage Service (OSS) is a secure, cost-effective, and highly reliable image processing service that can process large amounts of data. After you upload source images to OSS, you can call RESTful APIs to process the images anytime, anywhere, and on all Internet devices.

 **Note** For more information about IMG parameters, see [Parameters](#).

Use IMG parameters to process images

- Use a single IMG parameter to process images

```
const OSS = require('ali-oss');
const client = new OSS({
  // Obtain a temporary AccessKey pair from Security Token Service (STS).
  accessKeyId: 'yourAccessKeyId',
  accessKeySecret: 'yourAccessKeySecret',
  // Obtain a security token from STS.
  stsToken: 'yourSecurityToken',
  refreshSTSToken: async () => {
    // Obtain a temporary access credential from the STS you set up.
    const info = await fetch('your_sts_server');
    return {
      accessKeyId: info.accessKeyId,
      accessKeySecret: info.accessKeySecret,
      stsToken: info.stsToken
    }
  },
  // Set the time interval at which to refresh the temporary access credential. Unit: milliseconds.
  refreshSTSTokenInterval: 300000,
  // Specify the name of the bucket that contains the object you want to access. Example: examplebucket.
  bucket: 'examplebucket',
  // Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to http://oss-cn-hangzhou.aliyuncs.com.
  endpoint: 'yourEndpoint'
});
// Resize the image to the height and width of 100 pixels.
async function scale () {
  try {
    // Specify the full path of the object. Example: exampledir/exampleobject.jpg. The full path of the object cannot contain bucket names.
    const result = await client.signatureUrl('exampledir/exampleobject.jpg', {expires: 3600, process: 'image/resize,m_fixed,w_100,h_100'})
  } catch (e) {
    console.log(e);
  }
}
scale();
```

- Use multiple IMG parameters to process images

When you use multiple IMG parameters to process the image, separate these parameters with forward slashes (/).

```
const OSS = require('ali-oss');
const client = new OSS({
  // Obtain a temporary AccessKey pair from STS.
  accessKeyId: 'yourAccessKeyId',
  accessKeySecret: 'yourAccessKeySecret',
  // Obtain a security token from STS.
  stsToken: 'yourSecurityToken',
  refreshSTSToken: async () => {
    // Obtain a temporary access credential from the STS you set up.
    const info = await fetch('your_sts_server');
    return {
      accessKeyId: info.accessKeyId,
      accessKeySecret: info.accessKeySecret,
      stsToken: info.stsToken
    }
  },
  // Set the time interval at which to refresh the temporary access credential. Unit: milliseconds.
  refreshSTSTokenInterval: 300000,
  // Specify the name of the bucket that contains the object you want to access. Example: examplebucket.
  bucket: 'examplebucket',
  // Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com.
  endpoint: 'yourEndpoint',
});
// After you resize the image to the height and width of 100 pixels, rotate the image 90 degrees.
async function resize () {
  try {
    // Specify the full path of the object. Example: exampledir/exampleobject.jpg. The full path of the object cannot contain bucket names.
    const result = await client.signatureUrl('exampledir/exampleobject.jpg', {expires: 3600, process: 'image/resize,m_fixed,w_100,h_100/rotate,90'})
  } catch (e) {
    console.log(e);
  }
}
resize();
```

Use image styles to process images

You can create an image style in the OSS console and encapsulate multiple IMG parameters in the style. Then, you can use the style to process an image. For more information, see [Image styles](#).

```

const OSS = require('ali-oss');
const client = new OSS({
  // Obtain a temporary AccessKey pair from STS.
  accessKeyId: 'yourAccessKeyId',
  accessKeySecret: 'yourAccessKeySecret',
  // Obtain a security token from STS.
  stsToken: 'yourSecurityToken',
  refreshSTSToken: async () => {
    // Obtain a temporary access credential from the STS you set up.
    const info = await fetch('your_sts_server');
    return {
      accessKeyId: info.accessKeyId,
      accessKeySecret: info.accessKeySecret,
      stsToken: info.stsToken
    }
  },
  // Set the time interval at which to refresh the temporary access credential. Unit: milli
seconds.
  refreshSTSTokenInterval: 300000,
  // Specify the name of the bucket that contains the object you want to access. Example: e
xamplebucket.
  bucket: 'examplebucket',
  // Set yourEndpoint to the endpoint of the region in which the bucket is located. For exa
mple, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://
oss-cn-hangzhou.aliyuncs.com.
  endpoint: 'yourEndpoint',
});
// Use the specified image style to process the image.
async function style () {
  try {
    // Specify the full path of the object. Example: exampledir/exampleobject.jpg. The full
path of the object cannot contain bucket names.
    // Set yourCustomStyleName to the name of the image style you create in the OSS console
    .
    const result = await client.signatureUrl('exampledir/exampleobject.jpg', {expires: 3600
, process: 'style/yourCustomStyleName'});
  } catch (e) {
    console.log(e);
  }
}
style();

```

Save processed images

By default, IMG does not save processed images. You can call the Save As operation to save the images to the bucket where the source images are stored.

The following code provides an example on how to save a processed image:


```

const OSS = require('ali-oss');
const client = new OSS({
  // Obtain a temporary AccessKey pair from STS.
  accessKeyId: 'yourAccessKeyId',
  accessKeySecret: 'yourAccessKeySecret',
  // Obtain a security token from STS.
  stsToken: 'yourSecurityToken',
  refreshSTSToken: async () => {
    // Obtain a temporary access credential from the STS you set up.
    const info = await fetch('your_sts_server');
    return {
      accessKeyId: info.accessKeyId,
      accessKeySecret: info.accessKeySecret,
      stsToken: info.stsToken
    }
  },
  // Set the time interval at which to refresh the temporary access credential. Unit: milli
seconds.
  refreshSTSTokenInterval: 300000,
  // Specify the name of the bucket that contains the object you want to access. Example: e
xamplebucket.
  bucket: 'examplebucket',
  // Set yourEndpoint to the endpoint of the region in which the bucket is located. For exa
mple, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://
oss-cn-hangzhou.aliyuncs.com.
  endpoint: 'yourEndpoint'
});
// Specify the full path of the source object. Example: exampledir/exampleobject.jpg. The f
ull path of the object cannot contain bucket names.
const sourceImage = 'exampledir/exampleobject.jpg';
// Specify the full path of the processed object. Example: exampledir/exampleobject-resize.
jpg. The full path of the object cannot contain bucket names.
const targetImage = 'exampledir/exampleobject-resize.jpg';
async function processImage(processStr, targetBucket) {
  const result = await client.processObjectSave(
    sourceImage,
    targetImage,
    processStr,
    targetBucket
  );
  console.log(result.res.status);
}
// Resize the source image to a height and width of 100 pixels and save the image to the bu
cket that contains the source image.
processImage("image/resize,m_fixed,w_100,h_100", "examplebucket")

```

Generate a signed object URL that includes IMG parameters

URLs of private objects must be signed. IMG parameters cannot be added to a signed URL. If you want to process a private object, add IMG parameters to the signature. The following code provides an example on how to add IMG parameters to the signature:

```
const OSS = require('ali-oss');
const client = new OSS({
  // Obtain a temporary AccessKey pair from STS.
  accessKeyId: 'yourAccessKeyId',
  accessKeySecret: 'yourAccessKeySecret',
  // Obtain a security token from STS.
  stsToken: 'yourSecurityToken',
  refreshSTSToken: async () => {
    // Obtain a temporary access credential from the STS you set up.
    const info = await fetch('your_sts_server');
    return {
      accessKeyId: info.accessKeyId,
      accessKeySecret: info.accessKeySecret,
      stsToken: info.stsToken
    }
  },
  // Set the time interval at which to refresh the temporary access credential. Unit: milli
seconds.
  refreshSTSTokenInterval: 300000,
  // Specify the name of the bucket that contains the object you want to access. Example: e
xamplebucket.
  bucket: 'examplebucket',
  // Set yourEndpoint to the endpoint of the region in which the bucket is located. For exa
mple, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://
oss-cn-hangzhou.aliyuncs.com.
  endpoint: 'yourEndpoint'
});
// Generate a signed object URL that includes IMG parameters. Set the validity period of th
e signed URL to 600 seconds.
const signUrl = client.signatureUrl('example.jpg', {expires: 600, 'process' : 'image/resize
,w_300'});
console.log("signUrl="+signUrl);
```

6.10. Application in browsers

This topic describes the application of OSS SDK for Browser.js in browsers.

OSS SDK for Browser.js supports the following browsers:

- Internet Explorer 10 and later versions and Microsoft Edge.
- Mainstream Chrome/Firefox/Safari versions
- Default browsers of mainstream versions of Android/iOS/Windows Phone systems.



Notice To use OSS SDK for Browser.js in Internet Explorer, you must import the promise library after the SDK is imported.

Application in browsers of earlier versions

OSS SDK for Browser.js uses the File API to perform operations on objects. Therefore, problems may occur when you use the SDK in browsers of earlier versions. We recommend that you use third-party plug-ins to call OSS API operations to perform operations on objects, such as upload and download objects. For detailed examples, see [Add signatures on the client by using JavaScript and upload data to OSS](#).

- Related settings

- Bucket settings

To access a bucket from a browser, you must configure a [CORS](#) rule for the bucket in the following method.

- Set Sources to `*`.
- Set Allowed Methods to `GET, POST, PUT, DELETE, HEAD`.
- Set the Allowed Headers to `*`.
- Set Exposed Headers to `etag`.

 **Notice** Specify this rule as the first of all CORS rules.

- STS settings

To prevent the AccessKey ID and AccessKey secret from being exposed, STS is used for temporary authorization. The authorization server is maintained by users. Only authenticated users can apply for a temporary access from the authorization server. For more information about how to set a policy for RAM users and STS, visit [A simple app server using STS](#) to build your own authorization server.

- Examples

Use OSS SDK for Browser.js to develop a web application that can perform the following operations: upload objects, upload text, list objects, and download objects. For the complete sample code, visit [oss-in-browser](#).

The following example shows how to develop a web application to upload an object.

- i. Create a web page.

- Create a selection box to choose the object to upload.
- Create a text box to enter the name of the object to upload.
- Create a button used to start the upload.
- Create a progress bar that displays the upload progress.

The following code provides an example on how to create the preceding four controls, in which

`class="xxx"` is included to use the [Bootstrap](#) style.

```

<div class="form-group">
  <label>Select file</label>
  <input type="file" id="file" />
</div>
<div class="form-group">
  <label>Store as</label>
  <input type="text" class="form-control" id="object-key-file" value="object" />
</div>
<div class="form-group">
  <input type="button" class="btn btn-primary" id="file-button" value="Upload" />
</div>
<div class="form-group">
  <div class="progress">
    <div id="progress-bar"
      class="progress-bar"
      role="progressbar"
      aria-valuenow="0"
      aria-valuemin="0"
      aria-valuemax="100" style="min-width: 2em;">
      0%
    </div>
  </div>
</div>

```

ii. Add a style.

Bootstrap is used to add the style. The following code provides an example on how to add the style in the `<head>` field:

```

<head>
  <title>OSS in Browser</title>
  <link rel="stylesheet" href="bootstrap.min.css" />
</head>

```

The `bootstrap.min.css` style can be download from the official website of [Bootstrap](#).

iii. Add scripts.

The web page is still static after the preceding two steps. You must add JavaScript scripts to the web page to perform operations on objects, such as upload, download, and list objects.

Import the package of OSS SDK for Browser.js in the `<head>` field.

```

<head>
  <title>OSS in Browser</title>
  <link rel="stylesheet" href="bootstrap.min.css" />
  <script type="text/javascript" src="http://gosspublic.alicdn.com/aliyun-oss-sdk-6.1
7.1.min.js"></script>
  <script type="text/javascript" src="app.js"></script>
</head>

```

The `app.js` script includes the following code used to upload objects:

```
const appServer = 'http://localhost:3000';
const bucket = 'js-sdk-bucket-sts';
const region = 'oss-cn-hangzhou';
const urlLib = OSS.urlLib;
const applyTokenDo = function (func) {
  const url = appServer;
  return urlLib.request(url, {
    method: 'GET'
  }).then(function (result) {
    const creds = JSON.parse(result.data);
    const client = new OSS({
      region: region,
      accessKeyId: creds.AccessKeyId,
      accessKeySecret: creds.AccessKeySecret,
      stsToken: creds.SecurityToken,
      bucket: bucket
    });
    return func(client);
  });
};

let currentCheckpoint;
const progress = async function progress(p, checkpoint) {
  currentCheckpoint = checkpoint;
  const bar = document.getElementById('progress-bar');
  bar.style.width = `${Math.floor(p * 100)}%`;
  bar.innerHTML = `${Math.floor(p * 100)}%`;
};

let uploadFileClient;
const uploadFile = function (client) {
  if (! uploadFileClient || Object.keys(uploadFileClient).length === 0) {
    uploadFileClient = client;
  }
  const file = document.getElementById('file').files[0];
  const key = document.getElementById('object-key-file').value.trim() || 'object';
  console.log(`${file.name} => ${key}`);
  const options = {
    progress,
    partSize: 100 * 1024,
    meta: {
      year: 2017,
      people: 'test',
    },
  };
  return client.multipartUpload(key, file, options).then( (res) => {
    console.log('upload success: %j', res);
    currentCheckpoint = null;
    uploadFileClient = null;
  }).catch((err) => {
    if (uploadFileClient && uploadFileClient.isCancel()) {
      console.log('stop-upload!') ;
    } else {
      console.error(err);
    }
  });
};
```

```
};  
window.onload = function () {  
    document.getElementById('file-button').onclick = function () {  
        applyTokenDo(uploadFile);  
    }  
};
```

The following steps are performed to upload an object:

- a. Apply for a temporary access from the authorization server. The authorization server is a service built by website developers to grant temporary access to end users. Developers can provide different permissions for different users. To build an authorization server, visit [GitHub](#). In the example, the authorization server returns a temporary access credential to users.
- b. Use the temporary key to create an OSSClient.
- c. Use `multipartUpload` to upload the selected objects and use `progress` to configure the progress bar.

For more information about the sample code of uploading text, listing objects, and downloading objects, visit [OSS in Browser](#).

6.11. Exception handling

If a request error occurs during SDK usage, an exception is thrown containing the following information:

- status: the HTTP status code of the error request
- code: OSS error code
- message: OSS error message
- requestId: the UUID that uniquely identifies the request. Seek help from OSS development engineers by providing this RequestId if necessary.

For more information about common error messages in OSS, see [OSS Error Response](#).

Debugging

In a browser environment, set the `localStorage.debug` variable in the console to enable debugging:

```
localStorage.debug = 'ali-oss'
```

6.12. FAQ

This topic describes common questions and answers.

How to call STS

A browser is an untrusted environment. It brings an extremely high risk if you store an AccessKeyID and AccessKeySecret in a browser.

for OSS interface calls in a browser environment.

After obtaining the STS token, you can initialize the SDK.

```
<script type="text/javascript">
$.ajax("http://your_sts_server/",{method: 'GET'},function (err, result) {
    let client = new OSS({
        accessKeyId: result.AccessKeyId,
        accessKeySecret: result.AccessKeySecret,
        stsToken: result.SecurityToken,
        endpoint: '<oss endpoint>',
        bucket: '<Your bucket name>'
    });
});
</script>
```

How to enable HTTPS access

When you initialize the SDK, you can input the following parameters:

- **region**: This parameter is the region used when you apply for the OSS service, such as `oss-cn-hangzhou`. Complete region lists can be viewed in [Regions and endpoints](#).
- **internal**: Used in combination with `region`. If `internal` is set to `true`, visit the intranet node.
- **secure**: Used in combination with `region`. If `secure` is set to `true`, use the HTTPS for access.
- **endpoint**: For example, `http://oss-cn-hangzhou.aliyuncs.com`. If `endpoint` is specified, the `region` is ignored. You can specify the HTTPS or IP address when using `endpoint`.

How to solve the cross-domain issues when using a browser

Set CORS attributes of buckets before using SDK in a browser. For more information, see bucket settings in [Quick start](#).

How to set the user-defined data (meta), file type (mime), and request header (header) of the file to be uploaded

For more information, see [Multipart upload](#).

Instruction for resumable upload on browser side

Save the checkpoint to a browser localStorage. You can directly use the checkpoint parameter next time to achieve resumable upload.

How to obtain the upload progress

You can obtain upload process when using multipart upload. For more information, see [Upload objects](#).

How to obtain the download progress

The browser does not provide download progress. You can call the `signatureUrl` method to obtain the download address. For more information, see [Download objects](#).

How to upload files to a specified directory

Add the specified directory prefix to the object name to be uploaded. For more information, see [Comparison between OSS and file system](#).

```
let OSS = require('ali-oss')
let client = new OSS({
  region: '<Your region>',
  accessKeyId: '<Your AccessKeyId>',
  accessKeySecret: '<Your AccessKeySecret>',
  bucket: 'Your bucket name'
});
client.multipartUpload('base-dir/' + 'object-key', 'local-file', {
  progress: async function (p) {
    console.log('Progress: ' + p);
  }
});
console.log(result);
}).catch((err) => {
  console.log(err);
});
```

How to upload a Base64-encoded image

Base64 first transcodes images into specified formats, and then calls OSS Upload interface to upload the images. For more information, see [Github examples](#).


```
/**
 * base64 to file
 * @param dataurl    base64 content
 * @param filename   set up a meaningful suffix, or you can set mime type in options
 * @returns {File|*}
 */
const dataURLtoFile = function dataURLtoFile(dataurl, filename) {
  const arr = dataurl.split(',');
  const mime = arr[0].match(/:(.*?);/)[1];
  const bstr = atob(arr[1]);
  let n = bstr.length;
  const u8arr = new Uint8Array(n);
  while (n--) {
    u8arr[n] = bstr.charCodeAt(n);
  }
  return new Blob([u8arr], { type: mime }); // if env support File, also can use this: return new File([u8arr], filename, { type: mime });
};

// client indicates OSS client instances
const uploadBase64Img = function uploadBase64Img(client) {
  // base64 format content
  const base64Content = 'data:image:xxxxxxxxxxxxx';
  const filename = 'img.png';
  const imgfile = dataURLtoFile(base64Content, filename);
  //key indicates uploaded object key and imgFile indicates the returned images after processing by dataURLtoFile
  client.multipartUpload(key, imgfile).then((res) => {
    console.log('upload success: %j', res);
  }).catch((err) => {
    console.error(err);
  });
};
```

How to limit the size of the uploaded file

The size (in bytes) of the uploaded file can be obtained in the browser according to `document.getElementById( "file" ).files[0].size`. See the web post request.

How to obtain the signed URL of an object

You can call the `signatureUrl` method to obtain the download address. For more information, see [Download objects](#).

How to use a signed URL generated by SDK to upload resources

Signed URLs are often used to authorize third parties to download and upload resources. For the download operations, see the answer for the next question. The `signatureUrl` API provided by SDK is used to return a signed URL and users can use this URL to upload or download resources directly. For more information about how to upload resources using signed URLs, see [Upload resources using signed URLs](#).

How to upload resources to OSS servers using Form-based File Upload

For more information, see [Overview of direct transfer on Web client](#).

Common errors

- [JS-SDK exception](#)
- [OSS error response](#)

6.13. Log collection

This topic describes how to collect logs.

Applications developed by OSS JS SDK runs in different mobile terminals and are operated in different ways. Logs are required to locate the root causes of various problems. Therefore, Alibaba Cloud OSS provides a solution for developers to collect logs. For detailed usage, see [How to collect browser logs by using OSS JS SDK](#).

7.. NET

7.1. Preface

The OSS C# SDK supports .NET Framework 2.0 and later. This document is written based on OSS C# SDK 2.8.0.

Compatibility

- For SDK series of 2.x.x
 - API: compatible
 - Namespace: compatible
- For SDK series of 1.0.x
 - API: compatible
 - Namespace: incompatible. Aliyun.OpenServices.OpenStorageService is changed to Aliyun.OSS

SDK source code and API documents

For the SDK source code, see [GitHub](#). For more information, see [API documents](#).

Code samples

OSS SDK for C# provides a variety of code samples for your reference or use. You can obtain the code samples from [GitHub](#). The following table describes the content of code samples.

Sample file	Content
PutObjectSample.cs	Upload objects
AppendObjectSample.cs	Append upload
DoesObjectExistSample.cs	Determine whether an object exists
DeleteObjectsSample.cs	Delete objects
CopyObjectSample.cs	Copy objects
ModifyObjectMetaSample.cs	Manage Object Meta
MultipartUploadSample.cs	Multipart upload
ResumableSample.cs	Resumable upload
GetObjectSample.cs	Download objects
GetObjectByRangeSample.cs	Range download
GetObjectAclSample.cs	Manage ACL for an object
SetObjectAclSample.cs	Manage ACL for an object

Sample file	Content
ListObjectsSample.cs	List objects
UrlSignatureSample.cs	Authorized access
UploadCallbackSample.cs	Upload callback
ProgressSample.cs	Upload progress bars and Download progress bars
CNameSample.cs	Use custom domain name to access OSS (CNAME)
PostPolicySample.cs	Form upload
CreateBucketSample.cs	Create a bucket
DeleteBucketSample.cs	Delete a bucket
Doesbucketexistsample.cs	Determine whether a bucket exists
ListBucketsSample.cs	List buckets
SetBucketAclSample.cs	Configure an ACL for a bucket
SetBucketLifecycleSample.cs	Lifecycle management
SetBucketLoggingSample.cs	Access log files
SetBucketRefererSample.cs	Hotlink protection
SetBucketWebsiteSample.cs	Static website hosting
SetBucketCorsSample.cs	CORS
ImageProcessSample.cs	Image processing

7.2. Installation

This topic describes how to install OSS SDK for .NET.

Prerequisites

- Windows
 - .NET 2.0 or later is required.
 - Visual Studio 2010 or later is required.
- Linux/macOS
 - Mono 3.12 or later is required.

Download OSS SDK for .NET

- [Download SDK installation package](#)

- [Download from GitHub](#)
- [Download previous versions](#)

Install OSS SDK for .NET

- Install OSS SDK for .NET in Windows
 - Install by using NuGet
 - a. If NuGet is not installed for your Visual Studio, you must first install NuGet. For more information, visit [Install NuGet client tools](#).
 - b. After you install NuGet, create a project or open an existing project in Visual Studio, and then choose **Tools > NuGet Package Manager > Manage NuGet Packages for Solution**.
 - c. Enter aliyun.oss.sdk in the search box and click Search. Find Aliyun.OSS.SDK (applies to .NET Framework) or Aliyun.OSS.SDK.NetCore (applies to .Net Core) in the search results. Select the latest version, and click **Install**.
 - Install by using a DLL file
 - a. Download and decompress the installation package of OSS SDK for .NET.
 - b. In Visual Studio, go to **Solution Explorer** and find your project. Right-click the project name and choose **Reference > Add Reference**. In the dialog box that appears, select **Browse**.
 - c. Find the directory to which the installation package is decompressed and select the *Aliyun.OSS.dll* in the bin folder. Click **OK**.
 - Install by importing a project

If you download the SDK installation package or download the source code from GitHub and use the source code to install the SDK, perform the following operations:

 - a. In Visual Studio, right-click and select **Solution** from the menu that appears. Select **Add > Existing projects**.
 - b. In the dialog box that appears, select the *aliyun-oss-sdk.csproj* file. Click **Open**.
 - c. Right-click the name of your project and choose **Reference > Add Reference**. In the dialog box that appears, click the **Projects** tab, select the aliyun-oss-sdk project, and click **OK**.
- Install OSS SDK for .NET in Unix and macOS
 - Install by using NuGet
 - a. In Xamarin, create a project or open an existing project. Select **Add NuGet** from Tools.
 - b. Search and find Aliyun.OSS.SDK or Aliyun.OSS.SDK.NetCore. Select the latest version, and click **Add Package** to add the package to the project application.
 - Install by using a DLL file
 - a. Download and decompress the installation package of OSS SDK for .NET.
 - b. In Xamarin, select your project in **Solution**. Right click and select **Reference**. In the menu that appears, select **Edit References**.
 - c. In the Edit References dialog box, select *.Net Assembly* and click **Browse**. Find the directory to which the installation package is decompressed and select *Aliyun.OSS.dll* in the bin folder. Click **Open**.

7.3. Initialization

OssClient serves as the OSS .NET client to manage OSS resources such as buckets and objects.

Create an OssClient instance

To create an OssClient instance, you must specify yourEndpoint. For more information about endpoints, see [Regions and endpoints](#) and [Map custom domain names](#).

- Create the OssClient instance by using an OSS endpoint

The following code provides an example on how to create an OssClient instance based on an OSS endpoint:

```
using Aliyun.OSS;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to
// access OSS because the account has permissions on all API operations. We recommend that y
// ou use a RAM user to call API operations or perform routine operations and maintenance. T
// o create a RAM user, log on to the RAM console.
const string accessKeyId = "yourAccessKeyId";
const string accessKeySecret = "yourAccessKeySecret";
// Specify the endpoint of the region in which the bucket is located. For example, if the
// bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hang
// zhou.aliyuncs.com.
const string endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
// Create a new OssClient instance.
var ossClient = new OssClient(endpoint, accessKeyId, accessKeySecret);
```

- Create an OssClient by using a custom domain name

The following code provides an example on how to create an OssClient instance by using a custom domain name:

```
using Aliyun.OSS;
using Aliyun.OSS.Common;
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to
// access OSS because the account has permissions on all API operations. We recommend that y
// ou use a RAM user to call API operations or perform routine operations and maintenance. T
// o create a RAM user, log on to the RAM console.
const string accessKeyId = "yourAccessKeyId";
const string accessKeySecret = "yourAccessKeySecret";
// Specify the custom domain name.
const string endpoint = "yourDomain";
// Create a ClientConfiguration instance. Modify parameters as required.
var conf = new ClientConfiguration();
// Specify that a CNAME can be used as an endpoint to create the OssClient instance. A CN
// AME record specifies the mapping relationship between a custom domain name and a bucket.
conf.IsCname = true;
// Create an OssClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret, conf);
```



Notice ossClient.listBuckets is unavailable when a CNAME is used.

Configure an OssClient instance

Client Configuration is a configuration class of OssClient. You can use Client Configuration to configure parameters such as host proxies, connection timeouts, and the maximum number of connections. You can configure the following parameters.

Parameter	Description	Default value
ConnectionLimit	The maximum number of HTTP connections that can be enabled.	512
MaxErrorRetry	The maximum number of retry attempts in the case of a request error.	3
ConnectionTimeout	The timeout period for connections. Unit: millisecond.	-1, which indicates that connections do not time out.
IsCname	Specifies whether a CNAME can be used as an endpoint to create the OssClient instance. A CNAME record specifies the mapping relationship between a custom domain name and a bucket.	false
ProgressUpdateInterval	The update interval of the progress bar. Unit: bytes.	8096

The following code provides an example on how to configure an OssClient instance:

```
using Aliyun.OSS;
using Aliyun.OSS.Common;
var conf = new ClientConfiguration();
// Specify the maximum number of concurrent connections to the instance.
conf.ConnectionLimit = 512;
// Specify the maximum number of retry attempts in the case of a request error.
conf.MaxErrorRetry = 3;
// Specify the timeout period of the connections.
conf.ConnectionTimeout = 300;
var client = new OssClient(endpoint, accessKeyId, accessKeySecret, conf);
```

- Data verification

The following code provides an example on how to enable MD5 verification:

```
using Aliyun.OSS;
using Aliyun.OSS.Common;
var conf = new ClientConfiguration();
// Enable MD5 verification. By default, MD5 verification is disabled. If this parameter is
// set to true, MD5 verification is automatically performed on the uploaded and downloaded
// data.
conf.EnableMD5Check = true;
var client = new OssClient(endpoint, accessKeyId, accessKeySecret, conf);
```

 **Notice** The system performance may degrade if MD5 verification is enabled.

- Proxy network

The following table describes the parameters that you can configure when you use proxy networks.

Parameter	Description	Default value
ProxyHost	The address of the proxy server. Example: <code>example.aliyundoc.com</code> .	None
ProxyPort	The port of the proxy server. Example: <code>3128</code> or <code>8080</code> .	None
ProxyUserName	The account used to log on to the proxy server. This parameter is optional.	None
ProxyPassword	The password used to log on to the proxy server. This parameter is optional.	None

The following code provides an example on how to configure proxy networks that do not require an account and a password:

```
using Aliyun.OSS;
using Aliyun.OSS.Common;
var conf = new ClientConfiguration();
// Specify the proxy server.
conf.ProxyHost = "example.aliyundoc.com";
// Specify the port of the proxy server.
conf.ProxyPort = 3128;
var client = new OssClient(endpoint, accessKeyId, accessKeySecret, conf);
```

The following code provides an example on how to configure proxy networks that require an account and a password:

```
using Aliyun.OSS;
using Aliyun.OSS.Common;
var conf = new ClientConfiguration();
// Specify the proxy server.
conf.ProxyHost = "example.aliyundoc.com";
// Specify the port of the proxy server.
conf.ProxyPort = 3128;
// Specify the account used to access the proxy network.
conf.ProxyUserName = "user";
// Specify the password used to access the proxy network.
conf.ProxyPassword = "password";
var client = new OssClient(endpoint, accessKeyId, accessKeySecret, conf);
```

7.4. Quick start

This topic describes how to use Object Storage Service (OSS) SDK for .NET to complete routine operations such as creating a bucket, upload objects, and download objects.

Create a bucket

```
using Aliyun.OSS;
var endpoint = "<yourEndpoint>";
var accessKeyId = "<yourAccessKeyId>";
var accessKeySecret = "<yourAccessKeySecret>";
var bucketName = "<yourBucketName>";
// Create an OSSClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
try
{
    // Create the bucket.
    var bucket = client.CreateBucket(bucketName);
    Console.WriteLine("Create bucket succeeded, {0} ", bucket.Name);
}
catch (Exception ex)
{
    Console.WriteLine("Create bucket failed, {0}", ex.Message);
}
```

Upload an object

The following code provides an example on how to upload objects to OSS by using stream upload:

```
using Aliyun.OSS;
var endpoint = "<yourEndpoint>";
var accessKeyId = "<yourAccessKeyId>";
var accessKeySecret = "<yourAccessKeySecret>";
var bucketName = "<yourBucketName>";
var objectName = "<yourObjectName>";
var localFilename = "<yourLocalFilename>";
// Create an OSSClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
try
{
    // Upload the local file.
    var result = client.PutObject(bucketName, objectName, localFilename);
    Console.WriteLine("Put object succeeded, ETag: {0} ", result.ETag);
}
catch (Exception ex)
{
    Console.WriteLine("Put object failed, {0}", ex.Message);
}
```

For more information, see [Overview](#).

Download an object

The following code provides an example on how to download a specified object to a local file:

```
using Aliyun.OSS;
using Aliyun.OSS.Common;
var endpoint = "<yourEndpoint>";
var accessKeyId = "<yourAccessKeyId>";
var accessKeySecret = "<yourAccessKeySecret>";
var bucketName = "<yourBucketName>";
var objectName = "<yourObjectName>";
var localFilename = "<yourLocalFilename>";
var downloadFilename = "<yourDownloadFilename>";
// Create an OSSClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
try
{
    // Download the object.
    var result = client.GetObject(bucketName, objectName);
    using (var requestStream = result.Content)
    {
        using (var fs = File.Open(downloadFilename, FileMode.OpenOrCreate))
        {
            int length = 4 * 1024;
            var buf = new byte[length];
            do
            {
                length = requestStream.Read(buf, 0, length);
                fs.Write(buf, 0, length);
            } while (length != 0);
        }
        Console.WriteLine("Get object succeeded");
    }
}
catch (OssException ex)
{
    Console.WriteLine("Failed with error code: {0}; Error info: {1}. \nRequestId:{2}\tHostID:{3}",
        ex.ErrorCode, ex.Message, ex.RequestId, ex.HostId);
}
catch (Exception ex)
{
    Console.WriteLine("Failed with error info: {0}", ex.Message);
}
```

List objects

The following code provides an example on how to list objects stored in a bucket named examplebucket. By default, a maximum of 100 objects are listed.

```
using Aliyun.OSS;
using Aliyun.OSS.Common;
var endpoint = "<yourEndpoint>";
var accessKeyId = "<yourAccessKeyId>";
var accessKeySecret = "<yourAccessKeySecret>";
var bucketName = "<yourBucketName>";
// Create an OSSClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
try
{
    var objects = new List<string>();
    ObjectListing result = null;
    string nextMarker = string.Empty;
    do
    {
        var listObjectsRequest = new ListObjectsRequest(bucketName)
        {
            Marker = nextMarker,
        };
        // List objects in the bucket.
        result = client.ListObjects(listObjectsRequest);
        foreach (var summary in result.ObjectSummaries)
        {
            Console.WriteLine(summary.Key);
            objects.Add(summary.Key);
        }
        nextMarker = result.NextMarker;
    } while (result.IsTruncated);
    Console.WriteLine("List objects of bucket:{0} succeeded ", bucketName);
}
catch (OssException ex)
{
    Console.WriteLine("Failed with error code: {0}; Error info: {1}. \nRequestId:{2}\tHostID:{3}",
        ex.ErrorCode, ex.Message, ex.RequestId, ex.HostId);
}
catch (Exception ex)
{
    Console.WriteLine("Failed with error info: {0}", ex.Message);
}
```

Delete an object

The following code provides an example on how to delete an object:

```
using Aliyun.OSS;
var endpoint = "<yourEndpoint>";
var accessKeyId = "<yourAccessKeyId>";
var accessKeySecret = "<yourAccessKeySecret>";
var bucketName = "<yourBucketName>";
var objectName = "<yourObjectName>";
// Create an OSSClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
try
{
    // Delete the object.
    client.DeleteObject(bucketName, objectName);
    Console.WriteLine("Delete object succeeded");
}
catch (Exception ex)
{
    Console.WriteLine("Delete object failed, {0}", ex.Message);
}
```

7.5. Buckets

7.5.1. Create buckets

A bucket is a container used to store objects in Object Storage Service (OSS). Each object is contained in a bucket. This topic describes how to create a bucket.

Sample code

The following code provides an example on how to create a bucket named examplebucket:

```
using Aliyun.OSS;

// Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com.
var endpoint = "yourEndpoint";

// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
var accessKeyId = "yourAccessKeyId";
var accessKeySecret = "yourAccessKeySecret";

// Specify the bucket name.
var bucketName = "examplebucket";

// Initialize an OSSClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);

// Create the bucket.
public void CreateBucket(string bucketName)
{
    try
    {
        var request = new CreateBucketRequest(bucketName);
        // Set the access control list (ACL) of the bucket to public-read (PublicRead). The default value is private.
        request.ACL = CannedAccessControlList.PublicRead;
        // Set the redundancy type of the bucket to zone-redundant storage (ZRS).
        request.DataRedundancyType = DataRedundancyType.ZRS;
        client.CreateBucket(request);
        Console.WriteLine("Create bucket succeeded");
    }
    catch (Exception ex)
    {
        Console.WriteLine("Create bucket failed. {0}", ex.Message);
    }
}
```

References

- For more information about the complete sample code that is used to create a bucket, visit [Git Hub](#).
- For more information about the API operation that you can call to create a bucket, see [PutBucket](#).

7.5.2. List buckets

Buckets are listed in alphabetical order in Object Storage Service (OSS). This topic describes how to list all buckets that belong to the current Alibaba Cloud account.

Sample code

The following code provides an example on how to list all buckets that belong to the current Alibaba Cloud account:

```
using Aliyun.OSS;
// Specify the endpoint of the region in which the bucket is located. For example, if the b
ucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzho
u.aliyuncs.com.
var endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to ac
cess OSS because the account has permissions on all API operations. We recommend that you u
se a RAM user to call API operations or perform routine O&M. To create a RAM user, log on t
o the RAM console.
var accessKeyId = "yourAccessKeyId";
var accessKeySecret = "yourAccessKeySecret";
// Initialize an OSSClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
// List all buckets that belong to the current Alibaba Cloud account.
public void ListBuckets()
{
    try
    {
        var buckets = client.ListBuckets();
        Console.WriteLine("List bucket succeeded");
        foreach (var bucket in buckets)
        {
            Console.WriteLine("Bucket name:{0}, Location:{1}, Owner:{2}", bucket.Name, buck
et.Location, bucket.Owner);
        }
    }
    catch (Exception ex)
    {
        Console.WriteLine("List bucket failed. {0}", ex.Message);
    }
}
```

References

- For more information about the complete sample code that is used to list buckets, visit [Git Hub](#).
-

7.5.3. Determine whether buckets exist

A bucket is a container used to store objects in Object Storage Service (OSS). Every object is contained in a bucket. This topic describes how to determine whether a bucket exists.

For the complete code used to determine whether a bucket exists, visit [Git Hub](#).

```
using Aliyun.OSS;
// Initialize an OSSClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
// Determine whether the bucket exists.
public void DoesBucketExist(string bucketName)
{
    try
    {
        var exist = client.DoesBucketExist(bucketName);
        Console.WriteLine("Check object Exist succeeded");
        Console.WriteLine("exist ? {0}", exist);
    }
    catch (Exception ex)
    {
        Console.WriteLine("Check object Exist failed. {0}", ex.Message);
    }
}
```

7.5.4. Query the regions of buckets

A bucket is a container for objects stored in Object Storage Service (OSS). Every object is contained in a bucket. This topic describes how to query the region of a bucket.

Sample code

The following code provides an example on how to query the region or location of a bucket:

```
using Aliyun.OSS;
using Aliyun.OSS.Common;
// Specify the endpoint of the region in which the bucket is located. For example, if the b
ucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzho
u.aliyuncs.com.
var endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to ac
cess OSS, because the account has permissions on all API operations. We recommend that you
use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on
to the RAM console.
var accessKeyId = "yourAccessKeyId";
var accessKeySecret = "yourAccessKeySecret";
// Set yourBucketName to the name of the bucket.
var bucketName = "yourBucketName";

// Create an OSSClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
try
{
    // Query the region of the bucket.
    var result = client.GetBucketLocation(bucketName);
    Console.WriteLine("Get bucket:{0} Info succeeded ", bucketName);
    Console.WriteLine("bucket Location: {0}", result.Location);
}
catch (OssException ex)
{
    Console.WriteLine("Failed with error code: {0}; Error info: {1}. \nRequestId:{2}\tHostI
D:{3}",
        ex.ErrorCode, ex.Message, ex.RequestId, ex.HostId);
}
```

References

For more information about the API operation that you can call to query the region of a bucket, see [GetBucketLocation](#).

7.5.5. Query bucket information

A bucket is a container for objects stored in OSS. Every object is contained in a bucket. This topic describes how to query information about a bucket.

The following code provides an example on how to query the information about a bucket, including the region, creation date, and ACL of the bucket.


```
using Aliyun.OSS;
using Aliyun.OSS.Common;
// The endpoint of the China (Hangzhou) region is used in this example. Specify the actual
endpoint.
var endpoint = "<https://oss-cn-hangzhou.aliyuncs.com>";
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to lo
g on to OSS because the account has permissions on all API operations. We recommend that yo
u use your RAM user's credentials to call API operations or perform routine operations and
maintenance. To create a RAM user, log on to the RAM console.
var accessKeyId = "<yourAccessKeyId>";
var accessKeySecret = "<yourAccessKeySecret>";
var bucketName = "<yourBucketName>";

// Create an OSSClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
try
{
    // Information about a bucket includes the region (Region or Location), creation date (
CreationDate), owner (Owner), and ACL (Grants) of the bucket.
    var bucketInfo = client.GetBucketInfo(bucketName);
    Console.WriteLine("Get bucket:{0} Info succeeded ", bucketName);
    // Query the region of the bucket.
    Console.WriteLine("bucketInfo Location: {0}", bucketInfo.Bucket.Location);
    // Query the creation time of the bucket.
    Console.WriteLine("bucketInfo CreationDate: {0}", bucketInfo.Bucket.CreationDate);
    // Query the type of disaster recovery for a bucket.
    Console.WriteLine("bucketInfo DataRedundancyType: {0}", bucketInfo.Bucket.DataRedundanc
yType);
    // Query the ACL information of the bucket.
    Console.WriteLine("bucketInfo Grant: {0}", bucketInfo.Bucket.AccessControlList.Grant);
}
catch (OssException ex)
{
    Console.WriteLine("Failed with error code: {0}; Error info: {1}. \nRequestId:{2}\tHostI
D:{3}",
        ex.ErrorCode, ex.Message, ex.RequestId, ex.HostId);
}
```

7.5.6. Manage bucket ACLs

A bucket is a container for objects stored in Object Storage Service (OSS). Every object is contained in a bucket. This topic describes how to configure and query the access control list (ACL) of a bucket.

Configure the ACL of a bucket

The following table describes the bucket ACLs.

ACL	Description	Method
-----	-------------	--------

ACL	Description	Method
private	Only the bucket owner and authorized users can perform read and write operations on objects in the bucket. Other users cannot access the objects in the bucket.	<code>CannedAccessControlList.Private</code>
public read	Only the bucket owner and authorized users of this bucket can perform read and write operations on objects in the bucket. Other users, including anonymous users, can only read objects in the bucket. Exercise caution when you set the bucket ACL to this value.	<code>CannedAccessControlList.PublicRead</code>
public read/write	All users, including anonymous users, can perform read and write operations on objects in the bucket. Exercise caution when you set the bucket ACL to this value.	<code>CannedAccessControlList.PublicReadWrite</code>

The following code provides an example on how to configure ACL of a bucket:

```
using Aliyun.OSS;
// Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com.
var endpoint = "yourEndpoint";
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
var accessKeyId = "yourAccessKeyId";
var accessKeySecret = "yourAccessKeySecret";
// Specify the name of the bucket.
var bucketName = "yourBucketName";
// Initialize an OSSClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
public void SetBucketAcl(string bucketName)
{
    try
    {
        // Set the ACL of the bucket to public read.
        client.SetBucketAcl(bucketName, CannedAccessControlList.PublicRead);
        Console.WriteLine("Set bucket ACL succeeded");
    }
    catch (Exception ex)
    {
        Console.WriteLine("Set bucket ACL failed. {0}", ex.Message);
    }
}
```

For more information about how to configure the ACL of a bucket, see [PutBucketAcl](#).

Query the ACL of a bucket

The following code provides an example on how to query the ACL of a bucket:

```
using Aliyun.OSS;
// Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com.
var endpoint = "yourEndpoint";
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
var accessKeyId = "yourAccessKeyId";
var accessKeySecret = "yourAccessKeySecret";
// Initialize an OSSClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
// Query the ACL of the bucket.
public void GetBucketAcl(string bucketName)
{
    try
    {
        // Specify the name of the bucket.
        string bucketName = "yourBucketName";
        var acl = client.GetBucketAcl(bucketName);
        Console.WriteLine("Get bucket ACL success", acl.ACL.ToString());
    }
    catch (Exception ex)
    {
        Console.WriteLine("Get bucket ACL failed. {0}", ex.Message);
    }
}
```

For more information about how to query the ACL of a bucket, see [GetBucketAcl](#).

7.5.7. Delete a bucket

If you no longer use a bucket, delete the bucket to stop unnecessary charges.

 **Warning** Deleted buckets cannot be recovered. Exercise caution when you perform this operation.

Prerequisites

- All objects in the bucket are deleted.
 - For more information about how to manually delete a small number of objects, see [删除文件](#).
 - If you want to delete a large number of objects, we recommend that you configure lifecycle rules to delete the objects in batches. For more information about how to delete a large number of objects, see [Configure lifecycle rules](#).

 Notice

If versioning is enabled for the bucket that you want to delete, make sure that all versions of objects in the bucket are deleted. For more information about how to delete all versions of objects in a bucket, see [Configure versioning](#).

- Parts generated by multipart upload or resumable upload tasks in the bucket are deleted. For more information about how to delete parts in a bucket, see [Manage parts](#).
- All LiveChannels in the bucket are deleted. For more information about how to delete LiveChannels, see [DeleteLiveChannel](#).

Sample code

The following sample code shows how to delete a bucket named examplebucket:

```
using Aliyun.OSS;
// Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com.
var endpoint = "yourEndpoint";
// The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to access OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine Q&M. To create a RAM user, log on to the RAM console.
var accessKeyId = "yourAccessKeyId";
var accessKeySecret = "yourAccessKeySecret";
// Specify the name of the bucket. Example: examplebucket.
var bucketName = "examplebucket";
// Delete the bucket.
public void DeleteBucket(string bucketName)
{
    try
    {
        client.DeleteBucket(bucketName);
        Console.WriteLine("Delete bucket succeeded");
    }
    catch (Exception ex)
    {
        Console.WriteLine("Delete bucket failed. {0}", ex.Message);
    }
}
```

References

- [GitHub](#)
- For more information about the API operation that you can call to delete a bucket, see [DeleteBucket](#).

7.5.8. Bucket tagging

Object Storage Service (OSS) allows you to classify and manage buckets by means of bucket tagging. For example, the bucket tagging feature allows you to add tags to buckets that are used for different purposes, list buckets that have specific tags, and configure access control lists (ACLs) for these buckets.

Usage notes

The bucket tagging feature uses a key-value pair to identify a bucket. You can add tags to buckets that are used for different purposes and manage the buckets by tags.

- Only the bucket owner and users who are granted the `oss:PutBucketTagging` permission can add tags to the bucket or modify the tags of the bucket. If other users try to add tags to the bucket or modify the tags of the bucket, 403 Forbidden is returned with the `AccessDenied` error code.
- You can configure up to 20 tags for a bucket.
- The key and value of a tag must be UTF-8-encoded.
- The key of a tag can be up to 64 characters in length and cannot start with `http://`, `https://`, or `Aliyun`. In addition, the key of a tag cannot be empty.
- The value of a tag can be up to 128 characters in length and can be empty.

Configure tags for a bucket

The following code provides an example on how to configure tags for a bucket:

```
using Aliyun.OSS;
using Aliyun.OSS.Common;
// Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com.
var endpoint = "yourEndpoint";
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access Object Storage Service (OSS) because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
var accessKeyId = "yourAccessKeyId";
var accessKeySecret = "yourAccessKeySecret";
// Specify the bucket name. Example: examplebucket.
var bucketName = "examplebucket";
// Create an OSSClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
try
{
    // Configure tags for the bucket.
    var setRequest = new SetBucketTaggingRequest(bucketName);
    var tag1 = new Tag
    {
        Key = "project",
        Value = "projectone"
    };
    var tag2 = new Tag
    {
        Key = "user",
        Value = "jsmith"
    };
    setRequest.AddTag(tag1);
    setRequest.AddTag(tag2);
    client.SetBucketTagging(setRequest);
    Console.WriteLine("Set bucket:{0} Tagging succeeded ", bucketName);
}
catch (OssException ex)
{
    Console.WriteLine("Failed with error code: {0}; Error info: {1}. \nRequestId:{2}\tHostID:{3}",
        ex.ErrorCode, ex.Message, ex.RequestId, ex.HostId);
}
catch (Exception ex)
{
    Console.WriteLine("Failed with error info: {0}", ex.Message);
}
```

Query tags of a bucket

The following code provides an example on how to query tags of a bucket:

```
using Aliyun.OSS;
using Aliyun.OSS.Common;
// Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com.
var endpoint = "yourEndpoint";
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
var accessKeyId = "yourAccessKeyId";
var accessKeySecret = "yourAccessKeySecret";
// Specify the bucket name. Example: examplebucket.
var bucketName = "examplebucket";
// Create an OSSClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
try
{
    // Query tags of the bucket.
    var result = client.GetBucketTagging(bucketName);
    Console.WriteLine("Get bucket:{0} tagging succeeded ", bucketName);
    for (var i = 0; i < result.Tags.Count; i++) {
        Console.WriteLine("bucket tagging key: {0}; bucket tagging value: {1}", result.Tags[i].Key, result.Tags[i].Value);
    }
}
catch (OssException ex)
{
    Console.WriteLine("Failed with error code: {0}; Error info: {1}. \nRequestId:{2}\tHostId:{3}",
        ex.ErrorCode, ex.Message, ex.RequestId, ex.HostId);
}
catch (Exception ex)
{
    Console.WriteLine("Failed with error info: {0}", ex.Message);
}
```

List buckets that have specific tags

The following code provides an example on how to list buckets that have specific tags:


```
using Aliyun.OSS;
using Aliyun.OSS.Common;
// Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com.
var endpoint = "yourEndpoint";
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
var accessKeyId = "yourAccessKeyId";
var accessKeySecret = "yourAccessKeySecret";
// Specify the bucket name. Example: examplebucket.
var bucketName = "examplebucket";
// Create an OSSClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
try
{
    // List buckets that have a specific tag.
    var tag = new Tag
    {
        Key = "tag",
        Value = "value"
    };
    var listRequest = new ListBucketsRequest();
    listRequest.Tag = tag;
    var result = client.ListBuckets(listRequest);
    Console.WriteLine("list bucket:{0} succeeded ", bucketName);
    foreach (var bucket in result.Buckets) {
        Console.WriteLine("bucket name: {0}", bucket.Name);
    }
}
catch (OssException ex)
{
    Console.WriteLine("Failed with error code: {0}; Error info: {1}. \nRequestId:{2}\tHostID:{3}",
        ex.ErrorCode, ex.Message, ex.RequestId, ex.HostId);
}
catch (Exception ex)
{
    Console.WriteLine("Failed with error info: {0}", ex.Message);
}
```

Delete all tags of a bucket

The following code provides an example on how to delete all tags of a bucket:

```
using Aliyun.OSS;
using Aliyun.OSS.Common;
// Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com.
var endpoint = "yourEndpoint";
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
var accessKeyId = "yourAccessKeyId";
var accessKeySecret = "yourAccessKeySecret";
// Specify the bucket name. Example: examplebucket.
var bucketName = "examplebucket";
// Create an OSSClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
try
{
    // Delete all tags of the bucket.
    client.DeleteBucketTagging(bucketName);
    Console.WriteLine("Delete bucket:{0} Tagging succeeded ", bucketName);
}
catch (OssException ex)
{
    Console.WriteLine("Failed with error code: {0}; Error info: {1}. \nRequestId:{2}\tHostID:{3}",
        ex.ErrorCode, ex.Message, ex.RequestId, ex.HostId);
}
catch (Exception ex)
{
    Console.WriteLine("Failed with error info: {0}", ex.Message);
}
```

Delete specific tags of a bucket

The following code provides an example on how to delete specific tags of a bucket:

```
using Aliyun.OSS;
using Aliyun.OSS.Common;
// Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com.
var endpoint = "yourEndpoint";
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
var accessKeyId = "yourAccessKeyId";
var accessKeySecret = "yourAccessKeySecret";
// Specify the bucket name. Example: examplebucket.
var bucketName = "examplebucket";
// Create an OSSClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
try
{
    var tag = new Tag
    {
        Key = "tag1",
        Value = "value1"
    };
    var tags = new List<Tag>();
    tags.Add(tag);
    // Delete specific tags of a bucket by specifying the tag keys.
    var request = new DeleteBucketTaggingRequest(bucketName, tags);
    var result = client.DeleteBucketTagging(request);
    Console.WriteLine("delete bucket:{0} Tagging by key succeeded ", bucketName);
}
catch (OssException ex)
{
    Console.WriteLine("Failed with error code: {0}; Error info: {1}. \nRequestId:{2}\tHostID:{3}",
        ex.ErrorCode, ex.Message, ex.RequestId, ex.HostId);
}
```

References

- For more information about the API operation that you can call to configure tags for a bucket, see [PutBucketTags](#).
- For more information about the API operation that you can call to query tags of a bucket, see [GetBucketTags](#).
- For more information about the API operation that you can call to delete tags of a bucket, see [DeleteBucketTags](#).

7.5.9. Bucket inventory

This topic describes how to add an inventory for a bucket and how to query, list, and delete inventories configured for a bucket.

Usage notes

- Make sure that you have permissions to create, view, list, and delete inventories for the bucket. By default, the bucket owner has permissions to perform these operations. If you do not have permissions to perform these operations, apply for the permissions from the bucket owner.
- You can configure up to 1,000 inventories for a bucket.
- Inventory lists must be stored in a bucket in the same region as the bucket for which the inventory feature is configured.

Create an inventory for a bucket

The following code provides an example on how to create an inventory for a bucket:

```
using Aliyun.OSS;
using Aliyun.OSS.Common;

// Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com.
var endpoint = "yourEndpoint";
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
var accessKeyId = "yourAccessKeyId";
var accessKeySecret = "yourAccessKeySecret";
// Specify the name of the bucket.
var bucketName = "examplebucket";
// Specify the account ID granted by the bucket owner to perform the PutBucketInventory operation.
var accountId = "yourDestinationBucketAccountId";
// Specify the RAM role for which you want to configure inventories. The RAM role must have permissions to read the source bucket for which you want to configure the inventory and permissions to write data to the destination bucket in which you want to store the generated inventory lists.
var roleArn = "yourDestinationBucketRoleArn";
// Specify the name of the bucket in which you want to store the generated inventory lists.

var destBucketName = "yourDestinationBucketName";
// Create an OSSClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
try
{
    // Create an inventory for the bucket.
    var config = new InventoryConfiguration();
    // Specify the inventory name.
    config.Id = "report1";
    // Specify whether inventory is enabled for the bucket. Valid values: true and false. If you set this parameter to true, the inventory is enabled.
    config.IsEnabled = true;
    // Specify the rule used to filter the objects to include in the inventory lists. The following code provides an example on how to filter the objects by prefix.
    config.Filter = new InventoryFilter("filterPrefix");
    // Configure the destination bucket used to store the generated inventory lists.
    config.Destination = new InventoryDestination();
    config.Destination.OSSBucketDestination = new InventoryOSSBucketDestination();
    // Specify the format of the inventory lists.
```

```

config.Destination.OSSBucketDestination.Format = InventoryFormat.CSV;
// Specify the ID of the Alibaba Cloud account that owns the destination bucket.
config.Destination.OSSBucketDestination.AccountId = accountId;
// Specify the role ARN of the destination bucket.
config.Destination.OSSBucketDestination.RoleArn = roleArn;
// Specify the name of the destination bucket in which the generated inventory lists are stored.
config.Destination.OSSBucketDestination.Bucket = destBucketName;
// Specify the prefix of the path in which you want to store the generated inventory lists.
config.Destination.OSSBucketDestination.Prefix = "prefix1";
// Specify whether to generate the inventory list on a daily or weekly basis. The following code provides an example on how to generate an inventory list on a weekly basis.
config.Schedule = new InventorySchedule(InventoryFrequency.Daily);
// Specify that the inventory list includes only the current version of objects. If you set the InventoryIncludedObjectVersions parameter to All, all versions of objects are included in the inventory list. This configuration takes effect only when versioning is enabled for the bucket.
config.IncludedObjectVersions = InventoryIncludedObjectVersions.All;
// Specify the object attributes to include in the inventory lists.
config.OptionalFields.Add(InventoryOptionalField.Size);
config.OptionalFields.Add(InventoryOptionalField.LastModifiedDate);
config.OptionalFields.Add(InventoryOptionalField.StorageClass);
config.OptionalFields.Add(InventoryOptionalField.IsMultipartUploaded);
config.OptionalFields.Add(InventoryOptionalField.EncryptionStatus);
config.OptionalFields.Add(InventoryOptionalField.ETag);
var req = new SetBucketInventoryConfigurationRequest(bucketName, config);
client.SetBucketInventoryConfiguration(req);
Console.WriteLine("Set bucket:{0} InventoryConfiguration succeeded", bucketName);
}
catch (OssException ex)
{
    Console.WriteLine("Failed with error code: {0}; Error info: {1}. \nRequestID:{2}\tHostID:{3}",
        ex.ErrorCode, ex.Message, ex.RequestId, ex.HostId);
}

```

Query an inventory configured for a bucket

The following code provides an example on how to query inventories configured for a bucket:

```
using Aliyun.OSS;
using Aliyun.OSS.Common;
// Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com.
var endpoint = "yourEndpoint";
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
var accessKeyId = "yourAccessKeyId";
var accessKeySecret = "yourAccessKeySecret";
// Specify the name of the bucket.
var bucketName = "examplebucket";
// Specify the inventory name.
var id = "BucketInventoryConfigurationId";
// Create an OSSClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
try
{
    // Query the inventory with a specified ID.
    var request = new GetBucketInventoryConfigurationRequest(bucketName, id);
    //Query the inventory.
    var result = client.GetBucketInventoryConfiguration(request);
    Console.WriteLine("Get bucket:{0} BucketInventoryConfiguration succeeded ", bucketName);
;
    // Display the information about the inventory.
    Console.WriteLine("bucket InventoryConfiguration id: {0}; bucket InventoryConfiguration IsEnabled: {1}", result.Configuration.Id, result.Configuration.IsEnabled);
}
catch (OssException ex)
{
    Console.WriteLine("Failed with error code: {0}; Error info: {1}. \nRequestId:{2}\tHostID:{3}",
        ex.ErrorCode, ex.Message, ex.RequestId, ex.HostId);
}
```

List multiple inventories configured for a bucket

 **Note** You can query up to 100 inventories by sending a request. To query more than 100 inventory rules, you must send multiple requests and use the token returned for each request as the parameter for the next request.

The following code provides an example on how to list multiple inventories configured for a bucket at a time:

```

using Aliyun.OSS;
using Aliyun.OSS.Common;
// Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com.
var endpoint = "yourEndpoint";
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
var accessKeyId = "yourAccessKeyId";
var accessKeySecret = "yourAccessKeySecret";
// Specify the name of the bucket.
var bucketName = "examplebucket";
// Specify the inventory name.
var id = "BucketInventoryConfigurationId";
// Create an OSSClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
try
{
    // List the inventories. By default, a maximum of 100 inventories are listed. If more than 100 inventories are configured, the inventories are listed in pages. You can use the continuationToken parameter to list the inventories on the subsequent page.
    string continuationToken = null;
    bool isTruncated = false;
    do {
        var request = new ListBucketInventoryConfigurationRequest(bucketName, continuationToken);
        var result = client.ListBucketInventoryConfiguration(request);
        Console.WriteLine("List bucket:{0} BucketInventoryConfiguration succeeded ", bucketName);
        // Display the information about the inventory.
        for (var i = 0; i < result.Configurations.Count; i++) {
            Console.WriteLine("bucket InventoryConfiguration id: {0}; bucket InventoryConfiguration IsEnabled: {1}", result.Configurations[i].Id, result.Configurations[i].IsEnabled);
        }
        continuationToken = result.NextContinuationToken;
        isTruncated = result.IsTruncated;
    } while (isTruncated)
}
catch (OssException ex)
{
    Console.WriteLine("Failed with error code: {0}; Error info: {1}. \nRequestId:{2}\tHostId:{3}",
        ex.ErrorCode, ex.Message, ex.RequestId, ex.HostId);
}

```

Delete an inventory configured for a bucket

The following code provides an example on how to delete an inventory configured for a bucket:

```
using Aliyun.OSS;
using Aliyun.OSS.Common;
// Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com.
var endpoint = "yourEndpoint";
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
var accessKeyId = "yourAccessKeyId";
var accessKeySecret = "yourAccessKeySecret";
// Specify the name of the bucket.
var bucketName = "examplebucket";
// Specify the inventory name.
var id = "BucketInventoryConfigurationId";
// Create an OSSClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
try
{
    // Delete the inventory with the specified ID.
    var request = new DeleteBucketInventoryConfigurationRequest(bucketName, id);
    var result = client.DeleteBucketInventoryConfiguration(request);
    Console.WriteLine("delete bucket:{0} BucketInventoryConfiguration succeeded ", bucketName);
}
catch (OssException ex)
{
    Console.WriteLine("Failed with error code: {0}; Error info: {1}. \nRequestId:{2}\tHostId:{3}",
        ex.ErrorCode, ex.Message, ex.RequestId, ex.HostId);
}
```

References

- For more information about the API operation that is called to create an inventory, see [PutBucketInventory](#).
- For more information about the API operation that is called to query inventories configured for a bucket, see [GetBucketInventory](#).
- For more information about the API operation that is called to list multiple inventories configured for a bucket, see [ListBucketInventory](#).
- For more information about the API operation that is called to delete inventories configured for a bucket, see [DeleteBucketInventory](#).

7.5.10. Lifecycle

Object Storage Service (OSS) allows you to configure lifecycle rules to delete expired objects and parts or convert the storage class of expired objects to Infrequent Access (IA) or Archive to reduce your storage costs. This topic describes how to manage lifecycle rules for buckets.

 **Note** For more information about lifecycle rules, see [Lifecycle rules based on the last modified time](#) in OSS Developer Guide.

Configure lifecycle rules

The following sample code provides an example on how to configure a lifecycle rule that matches objects by prefix and tag for the bucket named examplebucket:

```
using Aliyun.OSS;
using Aliyun.OSS.Common;
// Specify the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set Endpoint to https://oss-cn-hangzhou.aliyuncs.com.
var endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
var accessKeyId = "yourAccessKeyId";
var accessKeySecret = "yourAccessKeySecret";
// Specify the name of the bucket. Example: examplebucket.
var bucketName = "examplebucket";
// Create an OSSClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
try
{
    var setBucketLifecycleRequest = new SetBucketLifecycleRequest(bucketName);
    // Create the first lifecycle rule.
    LifecycleRule lcr1 = new LifecycleRule()
    {
        ID = "delete obsoleted files",
        Prefix = "obsoleted/",
        Status = RuleStatus.Enabled,
        ExpirationDays = 3,
        Tags = new Tag[1]
    };
    // Specify the tags.
    var tag1 = new Tag
    {
        Key = "project",
        Value = "projectone"
    };
    lcr1.Tags[0] = tag1;
    // Create the second lifecycle rule.
    LifecycleRule lcr2 = new LifecycleRule()
    {
        ID = "delete temporary files",
        Prefix = "temporary/",
        Status = RuleStatus.Enabled,
        ExpirationDays = 20,
        Tags = new Tag[1]
    };
    // Specify the tags.
```

```

var tag2 = new Tag
{
    Key = "user",
    Value = "jsmith"
};
lcr2.Tags[0] = tag2;
// Specify that parts expire 30 days after they are last modified.
lcr2.AbortMultipartUpload = new LifecycleRule.LifecycleExpiration()
{
    Days = 30
};
LifecycleRule lcr3 = new LifecycleRule();
lcr3.ID = "only NoncurrentVersionTransition";
lcr3.Prefix = "test1";
lcr3.Status = RuleStatus.Enabled;
lcr3.NoncurrentVersionTransitions = new LifecycleRule.LifecycleNoncurrentVersionTransit
ion[2]
{
    // Specify that the storage class of objects is converted to Infrequent Access (IA)
    90 days after they become previous versions.
    new LifecycleRule.LifecycleNoncurrentVersionTransition() {
        StorageClass = StorageClass.IA,
        NoncurrentDays = 90
    },
    // Specify that the storage class of objects is converted to Archive 180 days after
    they become previous versions.
    new LifecycleRule.LifecycleNoncurrentVersionTransition() {
        StorageClass = StorageClass.Archive,
        NoncurrentDays = 180
    }
};
setBucketLifecycleRequest.AddLifecycleRule(lcr1);
setBucketLifecycleRequest.AddLifecycleRule(lcr2);
setBucketLifecycleRequest.AddLifecycleRule(lcr3);
// Configure the lifecycle rules.
client.SetBucketLifecycle(setBucketLifecycleRequest);
Console.WriteLine("Set bucket:{0} Lifecycle succeeded ", bucketName);
}
catch (OssException ex)
{
    Console.WriteLine("Failed with error code: {0}; Error info: {1}. \nRequestID:{2}\tHostI
D:{3}",
        ex.ErrorCode, ex.Message, ex.RequestId, ex.HostId);
}
catch (Exception ex)
{
    Console.WriteLine("Failed with error info: {0}", ex.Message);
}

```

Query lifecycle rules

The following code provides an example on how to query the lifecycle rules configured for the bucket named `examplebucket`:

```
using Aliyun.OSS;
using Aliyun.OSS.Common;
// Specify the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set Endpoint to https://oss-cn-hangzhou.aliyuncs.com.
var endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
var accessKeyId = "yourAccessKeyId";
var accessKeySecret = "yourAccessKeySecret";
// Specify the name of the bucket. Example: examplebucket.
var bucketName = "examplebucket";
// Create an OSSClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
try
{
    // Query the lifecycle rules configured for the bucket.
    var rules = client.GetBucketLifecycle(bucketName);
    Console.WriteLine("Get bucket:{0} Lifecycle succeeded ", bucketName);
    foreach (var rule in rules)
    {
        Console.WriteLine("ID: {0}", rule.ID);
        Console.WriteLine("Prefix: {0}", rule.Prefix);
        Console.WriteLine("Status: {0}", rule.Status);
        // View the tags specified in the lifecycle rules.
        foreach (var tag in rule.Tags)
        {
            Console.WriteLine("key:{0}, value:{1}", tag.Key, tag.Value);
        }
        // View the expiration rule for the previous versions of objects.
        foreach (var version in rule.NoncurrentVersionTransitions)
        {
            Console.WriteLine("expiration day:{0}, storage class:{1}", version.NoncurrentDays, version.StorageClass);
        }
        if (rule.ExpirationDays.HasValue)
            Console.WriteLine("ExpirationDays: {0}", rule.ExpirationDays);
    }
}
catch (OssException ex)
{
    Console.WriteLine("Failed with error code: {0}; Error info: {1}. \nRequestID:{2}\tHostID:{3}",
        ex.ErrorCode, ex.Message, ex.RequestId, ex.HostId);
}
catch (Exception ex)
{
    Console.WriteLine("Failed with error info: {0}", ex.Message);
}
```

Delete lifecycle rules

The following code provides an example on how to delete lifecycle rules configured for the bucket named `examplebucket`:

```
using Aliyun.OSS;
using Aliyun.OSS.Common;
// Specify the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set Endpoint to https://oss-cn-hangzhou.aliyuncs.com.
var endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
var accessKeyId = "yourAccessKeyId";
var accessKeySecret = "yourAccessKeySecret";
// Specify the name of the bucket. Example: examplebucket.
var bucketName = "examplebucket";
// Create an OSSClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
try
{
    // Delete the lifecycle rules.
    client.DeleteBucketLifecycle(bucketName);
    Console.WriteLine("Delete bucket:{0} Lifecycle succeeded ", bucketName);
}
catch (OssException ex)
{
    Console.WriteLine("Failed with error code: {0}; Error info: {1}. \nRequestId:{2}\tHostID:{3}",
        ex.ErrorCode, ex.Message, ex.RequestId, ex.HostId);
}
catch (Exception ex)
{
    Console.WriteLine("Failed with error info: {0}", ex.Message);
}
```

7.5.11. Pay-by-requester

When pay-by-requester is enabled for a bucket in Object Storage Service (OSS), the requester is charged the requests and traffic fees instead of the bucket owner. The bucket owner is charged only the storage fees. You can enable pay-by-requester for a bucket to share data in the bucket without paying for the request and traffic fees incurred by the access to your bucket.

Enable pay-by-requester

The following code provides an example on how to enable pay-by-requester for a bucket:

```
using Aliyun.OSS;
using Aliyun.OSS.Common;
// Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com.
var endpoint = "yourEndpoint";
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS, because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
var accessKeyId = "yourAccessKeyId";
var accessKeySecret = "yourAccessKeySecret";
// Specify the bucket name.
var bucketName = "examplebucket";
// Create an OSSClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
try
{
    // Enable pay-by-requester for the bucket.
    var request = new SetBucketRequestPaymentRequest(bucketName, RequestPayer.Requester);
    client.SetBucketRequestPayment(request);
    Console.WriteLine("Set bucket:{0} RequestPayment succeeded ", bucketName);
}
catch (OssException ex)
{
    Console.WriteLine("Failed with error code: {0}; Error info: {1}. \nRequestId:{2}\tHostID:{3}",
        ex.ErrorCode, ex.Message, ex.RequestId, ex.HostId);
}
catch (Exception ex)
{
    Console.WriteLine("Failed with error info: {0}", ex.Message);
}
```

Query pay-by-requester configurations

The following code provides an example on how to query the pay-by-requester configurations of a bucket:

```
using Aliyun.OSS;
using Aliyun.OSS.Common;
// Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com.
var endpoint = "yourEndpoint";
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS, because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
var accessKeyId = "yourAccessKeyId";
var accessKeySecret = "yourAccessKeySecret";
// Specify the bucket name.
var bucketName = "examplebucket";
// Create an OSSClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
try
{
    // Query the pay-by-requester configurations of the bucket.
    var result = client.GetBucketRequestPayment(bucketName);
    Console.WriteLine("Set bucket:{0} RequestPayment succeeded; RequestPayment: {1}", bucketName, result.Payer);
}
catch (OssException ex)
{
    Console.WriteLine("Failed with error code: {0}; Error info: {1}. \nRequestId:{2}\tHostID:{3}",
        ex.ErrorCode, ex.Message, ex.RequestId, ex.HostId);
}
catch (Exception ex)
{
    Console.WriteLine("Failed with error info: {0}", ex.Message);
}
```

Specify that third parties are charged for access to objects

If you specify that third parties are charged for access to objects in a bucket, requesters must include the `x-oss-request-payer:requester` header in the HTTP requests to perform operations on your objects. If this header is not included, an error is returned.

The following code provides an example on how to specify that third parties are charged when they access objects by calling the `PutObject`, `GetObject`, and `DeleteObject` operations. You can use the method to specify that third parties are charged when they perform read and write operations on objects by calling other API operations in the similar way.

```

using Aliyun.OSS;
using Aliyun.OSS.Common;
// Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com.
var endpoint = "yourEndpoint";
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS, because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
var accessKeyId = "yourAccessKeyId";
var accessKeySecret = "yourAccessKeySecret";
// Specify the bucket name.
var bucketName = "examplebucket";
var objectContent = "More than just cloud.";
// Create an OSSClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
try
{
    byte[] binaryData = Encoding.ASCII.GetBytes(objectContent);
    MemoryStream requestContent = new MemoryStream(binaryData);
    // Specify the payer when a third party calls the PutObject operation to access objects
    .
    var putRequest = new PutObjectRequest(bucketName, objectName, requestContent);
    putRequest.RequestPayer = RequestPayer.Requester;
    var result = client.PutObject(putRequest);
    // Specify the payer when a third party calls the GetObject operation to access objects
    .
    var getRequest = new GetObjectRequest(bucketName, objectName);
    getRequest.RequestPayer = RequestPayer.Requester;
    var getResult = client.GetObject(getRequest);
    // Specify the payer when a third party calls the DeleteObject operation to access objects.
    var delRequest = new DeleteObjectRequest(bucketName, objectName);
    delRequest.RequestPayer = RequestPayer.Requester;
    client.DeleteObject(delRequest);
}
catch (OssException ex)
{
    Console.WriteLine("Failed with error code: {0}; Error info: {1}. \nRequestId:{2}\tHostID:{3}",
        ex.ErrorCode, ex.Message, ex.RequestId, ex.HostId);
}
catch (Exception ex)
{
    Console.WriteLine("Failed with error info: {0}", ex.Message);
}

```

References

- For more information about the API operation that you can call to configure pay-by-requester for a bucket, see [Put Bucket Request Payment](#).

- For more information about the API operation that you can call to query the pay-by-requester configurations of a bucket, see [GetBucketRequestPayment](#).

7.5.12. Bucket policy

This topic describes how to configure, query, and delete policies for a bucket.

Background information

Bucket policies provide resource-based authorization for users. Bucket policies apply to the following scenarios:

For more information about bucket policy configurations and use cases, see [Configure bucket policies to authorize other users to access OSS resources](#). For more information about bucket policy syntax, see [Policy structure and syntax](#).

Configure bucket policies

The following sample code shows how to configure a policy for a bucket:

```
using Aliyun.OSS;
using Aliyun.OSS.Common;
var endpoint = "<yourEndpoint>";
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log on to OSS because the account has permissions on all API operations. We recommend that you use your RAM user's credentials to call API operations or perform routine operations and maintenance. To create your RAM user, log on to the RAM console.
String accessKeyId = "<yourAccessKeyId>";
var accessKeyId = "<yourAccessKeyId>";
var accessKeySecret = "<yourAccessKeySecret>";
var bucketName = "<yourBucketName>";
// Create an OSSClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
try
{
    // Configure a bucket policy.
    string policy = "{\n\"Version\": \"1\", \"Statement\": [{\n\"Action\": [\"oss:PutObject\", \"oss:GetObject\"], \"Resource\": \"acs:oss:*:*:*\", \"Effect\": \"Deny\"}]\n}";
    var request = new SetBucketPolicyRequest(bucketName, policy);
    client.SetBucketPolicy(request);
    Console.WriteLine("Set bucket:{0} Policy succeeded ", bucketName);
}
catch (OssException ex)
{
    Console.WriteLine("Failed with error code: {0}; Error info: {1}. \nRequestId:{2}\tHostID:{3}",
        ex.ErrorCode, ex.Message, ex.RequestId, ex.HostId);
}
catch (Exception ex)
{
    Console.WriteLine("Failed with error info: {0}", ex.Message);
}
```

Query bucket policies

The following sample code shows how to query the policies configured for a bucket:

```
using Aliyun.OSS;
using Aliyun.OSS.Common;
var endpoint = "<yourEndpoint>";
var accessKeyId = "<yourAccessKeyId>";
var accessKeySecret = "<yourAccessKeySecret>";
var bucketName = "<yourBucketName>";
// Create an OSSClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
try
{
    // Query bucket policies.
    var result = client.GetBucketPolicy(bucketName);
    Console.WriteLine("Get bucket:{0} Policy succeeded ", bucketName);
    Console.WriteLine("Policy: {0}", result.Policy);
}
catch (OssException ex)
{
    Console.WriteLine("Failed with error code: {0}; Error info: {1}. \nRequestID:{2}\tHostID:{3}",
        ex.ErrorCode, ex.Message, ex.RequestId, ex.HostId);
}
catch (Exception ex)
{
    Console.WriteLine("Failed with error info: {0}", ex.Message);
}
```

Delete bucket policies

The following sample code shows how to delete the policies configured for a bucket:

```
using Aliyun.OSS;
using Aliyun.OSS.Common;
var endpoint = "<yourEndpoint>";
var accessKeyId = "<yourAccessKeyId>";
var accessKeySecret = "<yourAccessKeySecret>";
var bucketName = "<yourBucketName>";
// Create an OSSClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
try
{
    // Delete bucket policies.
    client.DeleteBucketPolicy(bucketName);
    Console.WriteLine("Delete bucket:{0} Policy succeeded ", bucketName);
}
catch (OssException ex)
{
    Console.WriteLine("Failed with error code: {0}; Error info: {1}. \nRequestID:{2}\tHostID:{3}",
        ex.ErrorCode, ex.Message, ex.RequestId, ex.HostId);
}
catch (Exception ex)
{
    Console.WriteLine("Failed with error info: {0}", ex.Message);
}
```

7.5.13. Logging

A large number of logs are generated when Object Storage Service (OSS) resources are accessed. After you enable and configure logging for a bucket, OSS generates logs every hour in accordance with predefined naming conventions and then stores the logs as objects in a specified bucket.

Enable logging for a bucket

The following code provides an example on how to enable logging for a bucket:

```
using Aliyun.OSS;
using Aliyun.OSS.Common;
// Specify the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com. Specify the actual endpoint.
var endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
// The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in Object Storage Service (OSS) is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
var accessKeyId = "yourAccessKeyId";
var accessKeySecret = "yourAccessKeySecret";
// Specify the name of the bucket for which you want to enable logging. Example: examplebucket.
var bucketName = "examplebucket";
// Specify the name of the destination bucket used to store the log objects. The source bucket and the destination bucket can be the same bucket or different buckets.
var targetBucketName = "destbucket";
// Create an OSSClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
try
{
    // Specify that the log objects are stored in the log/ directory. If you specify this parameter, log objects are stored in a specified directory of the destination bucket. If you do not specify this parameter, log objects are stored in the root directory of the destination bucket.
    var request = new SetBucketLoggingRequest(bucketName, targetBucketName, "log/");
    // Enable logging for the bucket.
    client.SetBucketLogging(request);
    Console.WriteLine("Set bucket:{0} Logging succeeded ", bucketName);
}
catch (OssException ex)
{
    Console.WriteLine("Failed with error info: {0}; Error info: {1}. \nRequestId:{2}\tHostID:{3}",
        ex.ErrorCode, ex.Message, ex.RequestId, ex.HostId);
}
catch (Exception ex)
{
    Console.WriteLine("Failed with error info: {0}", ex.Message);
}
```

View the logging configurations of a bucket

The following code provides an example on how to view the logging configurations of a bucket:

```
using Aliyun.OSS;
using Aliyun.OSS.Common;
// Specify the endpoint of the region in which the bucket is located. For example, if the b
ucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzho
u.aliyuncs.com. Specify the actual endpoint.
var endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
// The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Us
ing these credentials to perform operations in OSS is a high-risk operation. We recommend t
hat you use a RAM user to call API operations or perform routine O&M. To create a RAM user,
log on to the RAM console.
var accessKeyId = "yourAccessKeyId";
var accessKeySecret = "yourAccessKeySecret";
// Specify the bucket name. Example: examplebucket.
var bucketName = "examplebucket";
// Create an OSSClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
try
{
    // View the logging configurations of the bucket.
    var result = client.GetBucketLogging(bucketName);
    Console.WriteLine("Get bucket:{0} Logging succeeded, prefix:{1}", bucketName, result.Ta
rgetPrefix);
}
catch (OssException ex)
{
    Console.WriteLine("Failed with error info: {0}; Error info: {1}. \nRequestID:{2}\tHostI
D:{3}",
        ex.ErrorCode, ex.Message, ex.RequestId, ex.HostId);
}
catch (Exception ex)
{
    Console.WriteLine("Failed with error info: {0}", ex.Message);
}
```

Disable logging for a bucket

The following code provides an example on how to disable logging for a bucket:

```

using Aliyun.OSS;
using Aliyun.OSS.Common;
// Specify the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com. Specify the actual endpoint.
var endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
// The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
var accessKeyId = "yourAccessKeyId";
var accessKeySecret = "yourAccessKeySecret";
// Specify the bucket name. Example: examplebucket.
var bucketName = "examplebucket";
// Create an OSSClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
try
{
    // Disable logging for the bucket.
    client.DeleteBucketLogging(bucketName);
    Console.WriteLine("Delete bucket:{0} Logging succeeded ", bucketName);
}
catch (OssException ex)
{
    Console.WriteLine("Failed with error info: {0}; Error info: {1}. \nRequestId:{2}\tHostID:{3}",
        ex.ErrorCode, ex.Message, ex.RequestId, ex.HostId);
}
catch (Exception ex)
{
    Console.WriteLine("Failed with error info: {0}", ex.Message);
}

```

References

- For the complete sample code that is used to configure logging for a bucket, visit [GitHub](#).
- For more information about the API operation that you can call to enable logging for a bucket, see [PutBucketLogging](#).
- For more information about the API operation that you can call to view the logging configurations of a bucket, see [GetBucketLogging](#).
- For more information about the API operation that you can call to disable logging for a bucket, see [DeleteBucketLogging](#).

7.5.14. CORS

Cross-origin resource sharing (CORS) is a standard cross-origin solution provided by HTML5 to allow web application servers to control cross-origin access and secure cross-origin data transmission.

Configure CORS rules

The following code provides an example on how to configure CORS rules for a specific bucket:

```

using Aliyun.OSS;
using Aliyun.OSS.Common;
// Set endpoint to the endpoint of the region in which the bucket is located. For example,
if the bucket is located in the China (Hangzhou) region, set endpoint to https://oss-cn-han
gzhou.aliyuncs.com.
var endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
// The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Us
ing these credentials to access OSS is a high-risk operation. We recommend that you use a R
AM user to call API operations or perform routine O&M. To create a RAM user, log on to the
RAM console.
var accessKeyId = "yourAccessKeyId";
var accessKeySecret = "yourAccessKeySecret";
// Specify the name of the bucket. Example: examplebucket.
var bucketName = "examplebucket";
// Create an OSSClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
try
{
    var request = new SetBucketCorsRequest(bucketName);
    var rule1 = new CORSRule();
    // Specify the origins from which cross-origin requests are allowed.
    rule1.AddAllowedOrigin("http://example.com");
    // Specify the HTTP methods that cross-origin requests are allowed to use, including PU
T, POST, GET, DELETE, and HEAD.
    rule1.AddAllowedMethod("POST");
    // AllowedHeaders and ExposeHeaders do not support wildcard characters.
    rule1.AddAllowedHeader("*");
    // Specify the response headers for allowed access requests from applications.
    rule1.AddExposeHeader("x-oss-test");
    // You can configure up to 10 CORS rules for the bucket.
    request.AddCORSRule(rule1);
    var rule2 = new CORSRule();
    // You can use only one asterisk (*) as the wildcard character for AllowedOrigins and A
llowedMethods in a CORS rule. The asterisk (*) wildcard character specifies that all origin
s or operations are allowed.
    rule2.AddAllowedOrigin("http://example.net");
    rule2.AddAllowedMethod("GET");
    // Specify whether the headers that are specified by Access-Control-Request-Headers in
the OPTIONS preflight request are allowed.
    rule2.AddExposeHeader("x-oss-test2");
    // Specify the period of time in which the browser can cache the response to an OPTIONS
preflight request for specific resources. Unit: seconds.
    rule2.MaxAgeSeconds = 100;
    request.AddCORSRule(rule2);
    // Configure the CORS rules.
    client.SetBucketCors(request);
    Console.WriteLine("Set bucket:{0} Cors succeeded ", bucketName);
}
catch (OssException ex)
{
    Console.WriteLine("Failed with error info: {0}; Error info: {1}. \nRequestID:{2}\tHostI
D:{3}",
        ex.ErrorCode, ex.Message, ex.RequestId, ex.HostId);
}
catch (Exception ex)

```

```

catch (Exception ex)
{
    Console.WriteLine("Failed with error info: {0}", ex.Message);
}

```

Query CORS rules

The following code provides an example on how to query CORS rules that are configured for a specific bucket:

```

using Aliyun.OSS;
using Aliyun.OSS.Common;
// Set endpoint to the endpoint of the region in which the bucket is located. For example,
// if the bucket is located in the China (Hangzhou) region, set endpoint to https://oss-cn-han
// gzhou.aliyuncs.com.
var endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
// The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Us
// ing these credentials to access OSS is a high-risk operation. We recommend that you use a R
// AM user to call API operations or perform routine O&M. To create a RAM user, log on to the
// RAM console.
var accessKeyId = "yourAccessKeyId";
var accessKeySecret = "yourAccessKeySecret";
// Specify the name of the bucket. Example: examplebucket.
var bucketName = "examplebucket";
// Create an OSSClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
try
{
    // Query the CORS rules.
    var result = client.GetBucketCors(bucketName);
    Console.WriteLine("Get bucket:{0} Cors succeeded ", bucketName);
    foreach (var rule in result)
    {
        foreach (var origin in rule.AllowedOrigins)
        {
            Console.WriteLine("Allowed origin:{0}", origin);
        }
    }
}
catch (OssException ex)
{
    Console.WriteLine("Failed with error info: {0}; Error info: {1}. \nRequestID:{2}\tHostID:
    {3}",
        ex.ErrorCode, ex.Message, ex.RequestId, ex.HostId);
}
catch (Exception ex)
{
    Console.WriteLine("Failed with error info: {0}", ex.Message);
}

```

Delete CORS rules

The following code provides an example on how to delete the CORS rules that are configured for a specific bucket:

```
using Aliyun.OSS;
using Aliyun.OSS.Common;
// Set endpoint to the endpoint of the region in which the bucket is located. For example,
// if the bucket is located in the China (Hangzhou) region, set endpoint to https://oss-cn-hangzhou.aliyuncs.com.
var endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
// The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to access OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
var accessKeyId = "yourAccessKeyId";
var accessKeySecret = "yourAccessKeySecret";
// Specify the name of the bucket. Example: examplebucket.
var bucketName = "examplebucket";
// Create an OSSClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
try
{
    // Delete the CORS rules.
    client.DeleteBucketCors(bucketName);
    Console.WriteLine("Delete bucket:{0} Cors succeeded ", bucketName);
}
catch (OssException ex)
{
    Console.WriteLine("Failed with error info: {0} ; Error info: {1}. \nRequestId:{2}\tHostID:{3}",
        ex.ErrorCode, ex.Message, ex.RequestId, ex.HostId);
}
catch (Exception ex)
{
    Console.WriteLine("Failed with error info: {0}", ex.Message);
}
```

References

- For more information about the complete sample code of CORS, visit [GitHub](#).
- For more information about the API operation that you can call to configure CORS rules, see [PutBucketCors](#).
- For more information about the API operation that you can call to query CORS rules, see [GetBucketCors](#).
- For more information about the API operation that you can call to delete CORS rules, see [DeleteBucketCors](#).

7.5.15. Static website hosting

You can enable static website hosting for buckets. After you host a static website on a bucket, you can access the bucket to visit the website. You are automatically redirected to the specified default homepage or default 404 page.

Configure static website hosting

The following code provides an example on how to configure static website hosting:

```
using Aliyun.OSS;
using Aliyun.OSS.Common;
// Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com.
var endpoint = "yourEndpoint";
// The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in Object Storage Service (OSS) is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
var accessKeyId = "yourAccessKeyId";
var accessKeySecret = "yourAccessKeySecret";
// Specify the bucket name.
var bucketName = "examplebucket";
// Create an OSSClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
try
{
    // Set the default homepage to index.html and the default 404 page to error.html for the static website hosted on the bucket.
    var request = new SetBucketWebsiteRequest(bucketName, "index.html", "error.html");
    client.SetBucketWebsite(request);
    Console.WriteLine("Set bucket:{0} Website succeeded ", bucketName);
}
catch (OssException ex)
{
    Console.WriteLine("Failed with error info: {0}; Error info: {1}. \nRequestId:{2}\tHostID:{3}",
        ex.ErrorCode, ex.Message, ex.RequestId, ex.HostId);
}
catch (Exception ex)
{
    Console.WriteLine("Failed with error info: {0}", ex.Message);
}
```

Query static website hosting configurations

The following code provides an example on how to query static website hosting configurations:

```
using Aliyun.OSS;
using Aliyun.OSS.Common;
// Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com.
var endpoint = "yourEndpoint";
// The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
var accessKeyId = "yourAccessKeyId";
var accessKeySecret = "yourAccessKeySecret";
// Specify the bucket name.
var bucketName = "examplebucket";
// Create an OSSClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
try
{
    // Query the static website hosting configurations.
    var result = client.GetBucketWebsite(bucketName);
    Console.WriteLine("Get bucket:{0} Website succeeded, index doc:{1}, error doc:{2}",
        bucketName, result.IndexDocument, result.ErrorDocument);
}
catch (OssException ex)
{
    Console.WriteLine("Failed with error info: {0}; Error info: {1}. \nRequestId:{2}\tHostID:{3}",
        ex.ErrorCode, ex.Message, ex.RequestId, ex.HostId);
}
catch (Exception ex)
{
    Console.WriteLine("Failed with error info: {0}", ex.Message);
}
```

Delete static website hosting configurations

The following code provides an example on how to delete static website hosting configurations:

```
using Aliyun.OSS;
using Aliyun.OSS.Common;
// Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com.
var endpoint = "yourEndpoint";
// The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
var accessKeyId = "yourAccessKeyId";
var accessKeySecret = "yourAccessKeySecret";
// Specify the bucket name.
var bucketName = "examplebucket";
// Create an OSSClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
try
{
    // Delete the static website hosting configurations.
    client.DeleteBucketWebsite(bucketName);
    Console.WriteLine("Delete bucket:{0} Website succeeded ", bucketName);
}
catch (OssException ex)
{
    Console.WriteLine("Failed with error info: {0}; Error info: {1}. \nRequestId:{2}\tHostID:{3}",
        ex.ErrorCode, ex.Message, ex.RequestId, ex.HostId);
}
catch (Exception ex)
{
    Console.WriteLine("Failed with error info: {0}", ex.Message);
}
```

References

- For the complete sample code that is used to configure static website hosting, visit [GitHub](#).
- For more information about the API operation that you can call to configure static website hosting, see [PutBucketWebsite](#).
- For more information about the API operation that you can call to query static website hosting configurations, see [GetBucketWebsite](#).
- For more information about the API operation that you can call to delete static website hosting configurations, see [DeleteBucketWebsite](#).

7.5.16. Hotlink protection

This topic describes how to use hotlink protection.

To prevent your data on OSS from being leached, OSS supports hotlink protection through the referer field settings in the HTTP header, including the following parameters:

- Referrer whitelist: Used to allow access only for specified domains to OSS data.
- Empty referer: Determines whether the referer can be empty. If it is not allowed, only requests with

the referer filed in their HTTP or HTTPS headers can access OSS data.

For more information about hot link protection, see [Hotlink protection](#) in OSS Developer Guide.

For the complete code on how to configure hot link protection for a bucket, visit [GitHub](#).

Configure a Referer whitelist for a bucket

The following code provides an example on how to configure a Referer whitelist for a bucket:

```
using Aliyun.OSS;
using Aliyun.OSS.Common;
var endpoint = "<yourEndpoint>";
var accessKeyId = "<yourAccessKeyId>";
var accessKeySecret = "<yourAccessKeySecret>";
var bucketName = "<yourBucketName>";
// Create an OSSClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
try
{
    var refererList = new List<string>();
    // Add a Referer whitelist. You can use asterisks (*) and question marks (?) as wildcards in Referers.
    refererList.Add(" http://www.aliyun.com");
    refererList.Add(" http://www.*.com");
    refererList.Add(" http://www.?.aliyuncs.com");
    var srq = new SetBucketRefererRequest(bucketName, refererList);
    // Configure the Referer whitelist.
    client.SetBucketReferer(srq);
    Console.WriteLine("Set bucket:{0} Referer succeeded ", bucketName);
}
catch (OssException ex)
{
    Console.WriteLine("Failed with error code: {0}; Error info: {1}. \nRequestId:{2}\tHostID:{3}",
        ex.ErrorCode, ex.Message, ex.RequestId, ex.HostId);
}
catch (Exception ex)
{
    Console.WriteLine("Failed with error info: {0}", ex.Message);
}
```

Query a Referer whitelist of a bucket

For the complete code used to query a Referer whitelist of a bucket, visit [GitHub](#).

The following code provides an example on how to query a Referer whitelist of a bucket:

```
using Aliyun.OSS;
using Aliyun.OSS.Common;
var endpoint = "<yourEndpoint>";
var accessKeyId = "<yourAccessKeyId>";
var accessKeySecret = "<yourAccessKeySecret>";
var bucketName = "<yourBucketName>";
// Create an OSSClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
try
{
    // Query the Referer whitelist of a bucket.
    var rc = client.GetBucketReferer(bucketName);
    Console.WriteLine("Get bucket:{0} Referer succeeded ", bucketName);
    Console.WriteLine("allow?" + (rc.AllowEmptyReferer ? "yes" : "no"));
    if (rc.RefererList.Referers != null)
    {
        for (var i = 0; i < rc.RefererList.Referers.Length; i++)
            Console.WriteLine(rc.RefererList.Referers[i]);
    }
    else
    {
        Console.WriteLine("Empty Referer List");
    }
}
catch (OssException ex)
{
    Console.WriteLine("Failed with error code: {0}; Error info: {1}. \nRequestID:{2}\tHostID:{3}",
        ex.ErrorCode, ex.Message, ex.RequestId, ex.HostId);
}
catch (Exception ex)
{
    Console.WriteLine("Failed with error info: {0}", ex.Message);
}
finally
{
    client.SetBucketReferer(new SetBucketRefererRequest(bucketName));
}
```

Clear a Referer whitelist of a bucket

The following code provides an example on how to clear a Referer whitelist of a bucket:

```
using Aliyun.OSS;
using Aliyun.OSS.Common;
var endpoint = "<yourEndpoint>";
var accessKeyId = "<yourAccessKeyId>";
var accessKeySecret = "<yourAccessKeySecret>";
var bucketName = "<yourBucketName>";
// Create an OSSClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
try
{
    // The Referer whitelist of a bucket cannot be directly cleared. You must create a Referer whitelist that allows empty Referer fields to overwrite the existing Referer whitelist.

    var srq = new SetBucketRefererRequest(bucketName);
    client.SetBucketReferer(srq);
    Console.WriteLine("Set bucket:{0} Referer succeeded ", bucketName);
}
catch (OssException ex)
{
    Console.WriteLine("Failed with error code: {0}; Error info: {1}. \nRequestId:{2}\tHostID:{3}",
        ex.ErrorCode, ex.Message, ex.RequestId, ex.HostId);
}
catch (Exception ex)
{
    Console.WriteLine("Failed with error info: {0}", ex.Message);
}
```

7.6. Upload objects

7.6.1. Overview

In OSS, an object is the basic unit for data operations. OSS .NET SDK provides the following file upload methods:

- **Simple upload**: supports the upload of a file up to 5 GB in size.
- **Append upload**: supports the upload of a file up to 5 GB in size.
- **Resumable upload**: supports concurrent upload and you can define the size of each part. It applies to the upload of large files. Resumable upload supports the upload of a file up to 48.8 TB in size.
- **Multipart upload**: supports the upload of a file up to 48.8 TB in size. It applies to the upload of large files.

 **Note** For more information about the usage scenarios of each upload method, see “Upload files” in OSS Developer Guide.

During the file upload, you can set **Object Meta**, and view the **upload progress**. After you complete the file upload, you can perform **upload callback**.

7.6.2. Simple upload

To implement simple upload, you can use the `PutObject` method to upload a single object. You can use simple upload to upload a string or local file in synchronous or asynchronous mode. You can also perform MD5 verification to ensure data integrity in simple upload.

Upload a string

 **Note** For the complete code that is used to upload a string, visit [GitHub](#).

The following sample code provides an example on how to upload a string to an object named `exampleobject.txt` in a bucket named `examplebucket`:

```
using System.Text;
using Aliyun.OSS;
// Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if your bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com.
var endpoint = "yourEndpoint";
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
var accessKeyId = "yourAccessKeyId";
var accessKeySecret = "yourAccessKeySecret";
// Specify the name of the bucket.
var bucketName = "examplebucket";
// Specify the full path of the object. The full path cannot contain the bucket name.
var objectName = "exampleobject.txt";
// Specify the string that you want to upload.
var objectContent = "More than just cloud.";
// Create an OSSClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
try
{
    byte[] binaryData = Encoding.ASCII.GetBytes(objectContent);
    MemoryStream requestContent = new MemoryStream(binaryData);
    // Upload the string to the object.
    client.PutObject(bucketName, objectName, requestContent);
    Console.WriteLine("Put object succeeded");
}
catch (Exception ex)
{
    Console.WriteLine("Put object failed, {0}", ex.Message);
}
```

Upload a local file

 **Note** For the complete code that is used to upload a local file, visit [GitHub](#).

The following sample code provides an example on how to upload a local file named `examplefile.txt` to a bucket named `examplebucket`. After the local file is uploaded, the name of the object stored in OSS is `exampleobject.txt`.


```
using Aliyun.OSS;
// Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com.
var endpoint = "yourEndpoint";
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
var accessKeyId = "yourAccessKeyId";
var accessKeySecret = "yourAccessKeySecret";
// Specify the name of the bucket.
var bucketName = "examplebucket";
// Specify the full path of the object. The full path cannot contain the bucket name.
var objectName = "exampleobject.txt";
// Specify the full path of the local file that you want to upload. By default, if you do not specify the path of the local file, the local file is uploaded from the path of the project to which the sample program belongs.
var localFilename = "D:\\localpath\\examplefile.txt";
// Create an OSSClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
try
{
    // Upload the file.
    client.PutObject(bucketName, objectName, localFilename);
    Console.WriteLine("Put object succeeded");
}
catch (Exception ex)
{
    Console.WriteLine("Put object failed, {0}", ex.Message);
}
```

Upload a local file and perform MD5 verification

 **Note** For the complete code that is used to upload a local file and perform MD5 verification, visit [GitHub](#).

To ensure that the data sent by the client is the same as that received by the OSS server, you can add the Content-MD5 header in the request as object metadata. The OSS server uses the value of this header to perform MD5 verification.

 **Notice** Take note that MD5 verification causes performance degradation.

The following sample code provides an example on how to perform MD5 verification when you upload a local file:

```
using System;
using System.IO;
using Aliyun.OSS;
using Aliyun.OSS.Util;
// Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com.
var endpoint = "yourEndpoint";
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
var accessKeyId = "yourAccessKeyId";
var accessKeySecret = "yourAccessKeySecret";
// Specify the name of the bucket.
var bucketName = "examplebucket";
// Specify the full path of the object. The full path cannot contain the bucket name.
var objectName = "exampleobject.txt";
// Specify the full path of the local file that you want to upload. By default, if you do not specify the path of the local file, the local file is uploaded from the path of the project to which the sample program belongs.
var localFilename = "D:\\localpath\\examplefile.txt";
// Create an OSSClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
try
{
    // Calculate the MD5 hash.
    string md5;
    using (var fs = File.Open(localFilename, FileMode.Open))
    {
        md5 = OssUtils.ComputeContentMd5(fs, fs.Length);
    }
    var objectMeta = new ObjectMetadata
    {
        ContentMd5 = md5
    };
    // Upload the file.
    client.PutObject(bucketName, objectName, localFilename, objectMeta);
    Console.WriteLine("Put object succeeded");
}
catch (Exception ex)
{
    Console.WriteLine("Put object failed, {0}", ex.Message);
}
```

Upload a local file in asynchronous mode

 **Note** For the complete code that is used to upload a local file in asynchronous mode, visit [GitHub](#).

The following sample code provides an example on how to upload a local file named `examplefile.txt` to a bucket named `examplebucket` in asynchronous mode. After the local file is uploaded, the name of the object stored in OSS is `exampleobject.txt`. When you upload an object in asynchronous mode, you must implement a function to process callback information.

```
using System;
using System.IO;
using System.Threading;
using Aliyun.OSS;
using Aliyun.OSS.Common;
// Upload a local file in asynchronous mode.
namespace AsyncPutObject
{
    class Program
    {
        // Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if your bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com.
        static string endpoint = "yourEndpoint";
        // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
        static string accessKeyId = "yourAccessKeyId";
        static string accessKeySecret = "yourAccessKeySecret";
        // Specify the name of the bucket.
        static string bucketName = "examplebucket";
        // Specify the full path of the object. The full path cannot contain the bucket name.
        static string objectName = "exampleobject.txt";
        // Specify the full path of the local file that you want to upload. By default, if you do not specify the path of the local file, the local file is uploaded from the path of the project to which the sample program belongs.
        static string localFilename = "D:\\\\localpath\\\\examplefile.txt";
        static AutoResetEvent _event = new AutoResetEvent(false);
        // Create an OSSClient instance.
        static OssClient client = new OssClient(endpoint, accessKeyId, accessKeySecret);
        private static void PutObjectCallback(IAsyncResult ar)
        {
            try
            {
                client.EndPutObject(ar);
                Console.WriteLine(ar.AsyncState as string);
                Console.WriteLine("Put object succeeded");
            }
            catch (Exception ex)
            {
                Console.WriteLine(ex.Message);
            }
            finally
            {
                _event.Set();
            }
        }
    }
}
```

```

public static void AsyncPutObject()
{
    try
    {
        using (var fs = File.Open(localFilename, FileMode.Open))
        {
            var metadata = new ObjectMetadata();
            // Configure the user metadata.
            metadata.UserMetadata.Add("mykey1", "myval1");
            metadata.UserMetadata.Add("mykey2", "myval2");
            metadata.CacheControl = "No-Cache";
            metadata.ContentType = "text/txt";
            string result = "Notice user: put object finish";
            // Upload the local file in asynchronous mode.
            client.BeginPutObject(bucketName, objectName, fs, metadata, PutObjectCa
llback, result.ToCharArray());
            _event.WaitOne();
        }
    }
    catch (OssException ex)
    {
        Console.WriteLine("Failed with error code: {0}; Error info: {1}. \nRequestID: {2} \t HostID: {3}",
            ex.ErrorCode, ex.Message, ex.RequestId, ex.HostId);
    }
    catch (Exception ex)
    {
        Console.WriteLine("Failed with error info: {0}", ex.Message);
    }
}

static void Main(string[] args)
{
    Program.AsyncPutObject();
}
}

```

7.6.3. Append upload

You can call the `AppendObject` operation to append content to existing appendable objects.

Limits

- If the object to which you want to append content does not exist, an appendable object is created when you call the `AppendObject` operation.
- When you call the `AppendObject` operation to append content to an object, you may encounter a few situations: If the object is an appendable object, and the specified position from which the append operation starts is equal to the current object length, the content is appended to the end of the object. If the specified position from which the append operation starts is not equal to the current object length, a `PositionNotEqualToLength` exception is thrown. If the object is not an appendable object, an `ObjectNotAppendable` exception is thrown.
- The `CopyObject` operation cannot be performed on appendable objects.

Sample code

The following code provides an example on how to perform append upload operation:

```
using Aliyun.OSS;
// Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com.
var endpoint = "yourEndpoint";
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
var accessKeyId = "yourAccessKeyId";
var accessKeySecret = "yourAccessKeySecret";
// Specify the name of the bucket. Example: examplebucket.
var bucketName = "examplebucket";
// Specify the full path of the object. The full path cannot contain the bucket name. Example: exampledir/exampleobject.txt.
var objectName = "exampledir/exampleobject.txt";
// Specify the full path of the local file. Example: D:\\localpath\\examplefile.txt. By default, if you do not specify the full path of the local file, the local file is uploaded from the path of the project to which the sample program belongs.
var localFilename = "D:\\localpath\\examplefile.txt";
// Create an OSSClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
// If the object is appended for the first time, set the position from which the append operation starts to 0. The position for the next append operation is included in the response. The position from which the next append operation starts is the current length of the object.
long position = 0;
try
{
    var metadata = client.GetObjectMetadata(bucketName, objectName);
    position = metadata.ContentLength;
}
catch (Exception) { }
try
{
    using (var fs = File.Open(localFilename, FileMode.Open))
    {
        var request = new AppendObjectRequest(bucketName, objectName)
        {
            ObjectMetadata = new ObjectMetadata(),
            Content = fs,
            Position = position
        };
        // Perform the append operation.
        var result = client.AppendObject(request);
        // Specify the position from which the append operation starts.
        position = result.NextAppendPosition;
        Console.WriteLine("Append object succeeded, next append position:{0}", position);
    }
    // Obtain the position for the next append operation and perform the second append operation.
```

```

        action.
        using (var fs = File.Open(localFilename, FileMode.Open))
        {
            var request = new AppendObjectRequest(bucketName, objectName)
            {
                ObjectMetadata = new ObjectMetadata(),
                Content = fs,
                Position = position
            };
            var result = client.AppendObject(request);
            position = result.NextAppendPosition;
            Console.WriteLine("Append object succeeded, next append position:{0}", position);
        }
    }
    catch (Exception ex)
    {
        Console.WriteLine("Append object failed, {0}", ex.Message);
    }
}

```

References

- For more information about the complete sample code that is used to perform append upload operation, visit [GitHub](#).
- For more information about the API operation that is called to perform append upload operation, see [AppendObject](#).

7.6.4. Resumable upload

Before you upload an object to Object Storage Service (OSS) by using resumable upload, you can specify a directory for the checkpoint file that stores resumable upload records. If an object fails to upload due to a network exception or program error, the upload task is resumed from the position recorded in the checkpoint file to upload the remaining data.

The progress of the current upload is recorded in the checkpoint file during the upload process. If a part fails to be uploaded during the process, the next upload attempt starts from the recorded position in the checkpoint file. The checkpoint file is deleted after the resumable upload task is completed.

 **Note** For the complete code that is used to perform resumable upload, visit [GitHub](#).

The following code provides an example on how to perform resumable upload by using multipart upload:

```

using Aliyun.OSS;
using Aliyun.OSS.Common;
// Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com.
var endpoint = "yourEndpoint";
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a Resource Access Management (RAM) user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.

```

```

var accessKeyId = "yourAccessKeyId";
var accessKeySecret = "yourAccessKeySecret";
// Specify the name of the bucket to which you want to upload the object. Example: examplebucket.
var bucketName = "examplebucket";
// Specify the full path of the object. Example: exampledir/exampleobject.txt. The full path of the object cannot contain the bucket name.
var objectName = "exampledir/exampleobject.txt";
// Specify the full path of the local file to upload. Example: D:\\localpath\\examplefile.txt.
// By default, if you specify only the name of the local file such as examplefile.txt without specifying the local path, the local file is uploaded from the path of the project to which the sample program belongs.
var localFilename = "D:\\localpath\\examplefile.txt";
// Specify the checkpoint file that records the upload result of each part. The file stores the progress information generated in the upload process.
string checkpointDir = "yourCheckpointDir";
// Create an OSSClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
try
{
    // Configure parameters by using UploadFileRequest.
    UploadObjectRequest request = new UploadObjectRequest(bucketName, objectName, localFilename)
    {
        // Specify the size of each part to upload.
        PartSize = 8 * 1024 * 1024,
        // Specify the number of concurrent threads.
        ParallelThreadCount = 3,
        // Set the checkpointDir parameter to store the state of the resumable upload, which is used to resume the upload when the upload fails.
        CheckpointDir = checkpointDir,
    };
    // Start resumable upload.
    client.ResumableUploadObject(request);
    Console.WriteLine("Resumable upload object:{0} succeeded", objectName);
}
catch (OssException ex)
{
    Console.WriteLine("Failed with error code: {0}; Error info: {1}. \nRequestId:{2}\tHostId:{3}",
        ex.ErrorCode, ex.Message, ex.RequestId, ex.HostId);
}
catch (Exception ex)
{
    Console.WriteLine("Failed with error info: {0}", ex.Message);
}

```

7.6.5. Multipart upload

By using the multipart upload feature provided by OSS, you can split a large object into multiple parts and upload them separately. After all parts are uploaded, call the `CompleteMultipartUpload` operation to combine these parts into a single object to implement resumable upload.

Process

To upload an object by using multipart upload, perform the following steps:

1. Initialize a multipart upload task.

Call the `InitiateMultipartUploadRequest` method to obtain a unique upload ID.

2. Upload parts.

Call the `UploadPartRequest` method to upload parts.

Note

- For parts that are uploaded by a multipart upload task with a specific upload ID, part numbers identify their relative positions in an object. If you upload a part with the same part number as an existing part, the existing part is overwritten by the uploaded part.
- OSS includes the MD5 hash of part data in the ETag header and returns the MD5 hash to the user.
- OSS calculates the MD5 hash of uploaded data and compares the MD5 hash with the MD5 hash calculated by the SDK. If the two hashes are different, the `InvalidDigest` error code is returned.

3. Complete the multipart upload task.

After all parts are uploaded, call the `CompleteMultipartUploadRequest` method to combine the parts into a complete object.

Complete sample code

The following code provides a complete example that describes the process of multipart upload:

```
using Aliyun.OSS;
// Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com.
var endpoint = "yourEndpoint";
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access Object Storage Service (OSS) because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
var accessKeyId = "yourAccessKeyId";
var accessKeySecret = "yourAccessKeySecret";
// Specify the name of the bucket.
var bucketName = "examplebucket";
// Specify the full path of the object. The full path cannot contain the name of the bucket.
var objectName = "exampleobject.txt";
// Specify the full path of the local file that you want to upload. By default, if you do not specify the full path of the local file, the local file is uploaded from the path of the project to which the sample program belongs.
var localFilename = "D:\\localpath\\examplefile.txt";
```



```

// Create an OSSClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
// Initiate the multipart upload task and obtain the upload ID in the response.
var uploadId = "";
try
{
    // Specify the name of the uploaded object and the bucket for the object. You can configure object metadata in InitiateMultipartUploadRequest. However, you do not need to specify ContentLength.
    var request = new InitiateMultipartUploadRequest(bucketName, objectName);
    var result = client.InitiateMultipartUpload(request);
    uploadId = result.UploadId;
    // Display the upload ID.
    Console.WriteLine("Init multi part upload succeeded");
    Console.WriteLine("Upload Id:{0}", result.UploadId);
}
catch (Exception ex)
{
    Console.WriteLine("Init multi part upload failed, {0}", ex.Message);
}
// Calculate the total number of parts.
var partSize = 100 * 1024;
var fi = new FileInfo(localFilename);
var fileSize = fi.Length;
var partCount = fileSize / partSize;
if (fileSize % partSize != 0)
{
    partCount++;
}
// Start the multipart upload task. partETags is a list of PartETags. OSS verifies the validity of each part after it receives the list of parts. After all parts are verified, OSS combines these parts into a complete object.
var partETags = new List<PartETag>();
try
{
    using (var fs = File.Open(localFilename, FileMode.Open))
    {
        for (var i = 0; i < partCount; i++)
        {
            var skipBytes = (long)partSize * i;
            // Find the start position of the current multipart upload task.
            fs.Seek(skipBytes, 0);
            // Calculate the part size in this upload. The size of the last part is the size of the remainder after the object is split by the calculated part size.
            var size = (partSize < fileSize - skipBytes) ? partSize : (fileSize - skipBytes);

            var request = new UploadPartRequest(bucketName, objectName, uploadId)
            {
                InputStream = fs,
                PartSize = size,
                PartNumber = i + 1
            };
            // Call UploadPart to upload parts. The returned results contain the ETag values of parts.

```

```
        var result = client.UploadPart(request);
        partETags.Add(result.PartETag);
        Console.WriteLine("finish {0}/{1}", partETags.Count, partCount);
    }
    Console.WriteLine("Put multi part upload succeeded");
}
}
catch (Exception ex)
{
    Console.WriteLine("Put multi part upload failed, {0}", ex.Message);
}
// Complete the multipart upload task.
try
{
    var completeMultipartUploadRequest = new CompleteMultipartUploadRequest(bucketName, objectName, uploadId);
    foreach (var partETag in partETags)
    {
        completeMultipartUploadRequest.PartETags.Add(partETag);
    }
    var result = client.CompleteMultipartUpload(completeMultipartUploadRequest);
    Console.WriteLine("complete multi part succeeded");
}
catch (Exception ex)
{
    Console.WriteLine("complete multi part failed, {0}", ex.Message);
}
```

Cancel a multipart upload task

You can call the `client.AbortMultipartUpload` method to cancel a multipart upload task. If you cancel the multipart upload task, the upload ID cannot be used to upload any parts. The parts that are uploaded by using the multipart upload task are deleted.

The following code provides an example on how to cancel a multipart upload task:

```
using Aliyun.OSS;
// Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com.
var endpoint = "yourEndpoint";
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
var accessKeyId = "yourAccessKeyId";
var accessKeySecret = "yourAccessKeySecret";
// Specify the name of the bucket.
var bucketName = "examplebucket";
// Specify the full path of the object. The full path cannot contain the name of the bucket.
var objectName = "exampleobject.txt";
// Specify the upload ID.
var uploadId = "yourUploadId";
// Create an OSSClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
// Initiate a multipart upload task.
try
{
    var request = new InitiateMultipartUploadRequest(bucketName, objectName);
    var result = client.InitiateMultipartUpload(request);
    uploadId = result.UploadId;
    // Display the upload ID.
    Console.WriteLine("Init multi part upload succeeded");
    Console.WriteLine("Upload Id:{0}", result.UploadId);
}
catch (Exception ex)
{
    Console.WriteLine("Init multi part upload failed, {0}", ex.Message);
}
// Cancel the multipart upload task.
try
{
    var request = new AbortMultipartUploadRequest(bucketName, objectName, uploadId);
    client.AbortMultipartUpload(request);
    Console.WriteLine("Abort multi part succeeded, {0}", uploadId);
}
catch (Exception ex)
{
    Console.WriteLine("Abort multi part failed, {0}", ex.Message);
}
```

List uploaded parts

The following code provides an example on how to list uploaded parts:

```
using Aliyun.OSS;
// Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com.
var endpoint = "yourEndpoint";
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
var accessKeyId = "yourAccessKeyId";
var accessKeySecret = "yourAccessKeySecret";
// Specify the name of the bucket.
var bucketName = "examplebucket";
// Specify the full path of the object. The full path cannot contain the name of the bucket.
var objectName = "exampleobject.txt";
// Specify the upload ID.
var uploadId = "yourUploadId";
// Create an OSSClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
try
{
    PartListing listPartsResult = null;
    var nextMarker = 0;
    do
    {
        var listPartsRequest = new ListPartsRequest(bucketName, objectName, uploadId)
        {
            PartNumberMarker = nextMarker,
        };
        // List uploaded parts.
        listPartsResult = client.ListParts(listPartsRequest);
        Console.WriteLine("List parts succeeded");
        // Display the information about all uploaded parts.
        var parts = listPartsResult.Parts;
        foreach (var part in parts)
        {
            Console.WriteLine("partNumber: {0}, ETag: {1}, Size: {2}", part.PartNumber, part.ETag, part.Size);
        }
        nextMarker = listPartsResult.NextPartNumberMarker;
    } while (listPartsResult.IsTruncated);
}
catch (Exception ex)
{
    Console.WriteLine("List parts failed, {0}", ex.Message);
}
```

List multipart upload tasks

You can call the `ossClient.listMultipartUploads` method to list all ongoing multipart upload tasks. Ongoing multipart upload tasks are tasks that have been initiated but are not completed and tasks that have been canceled. The following table describes the parameters that you can configure when you list all ongoing multipart upload tasks.

Parameter	Description	Method
prefix	Specifies that the returned object name must contain a specified prefix. Note that if you use a prefix for a query, the returned object name contains the prefix.	<code>ListMultipartUploadsRequest.setPrefix(String prefix)</code>
delimiter	Groups objects by name. Specifies the character used to group multipart upload tasks by name. Multipart upload tasks whose names contain the same string that starts with the prefix and ends with the first occurrence of the specified delimiter are grouped as a single result element.	<code>ListMultipartUploadsRequest.setDelimiter(String delimiter)</code>
maxUploads	Specifies the maximum number of multipart upload tasks to return each time. Default value: 1000. Maximum value: 1000.	<code>ListMultipartUploadsRequest.setMaxUploads(Integer maxUploads)</code>
keyMarker	Specifies the position from which the list operation starts. All ongoing multipart upload tasks initiated for objects whose names are alphabetically after the keyMarker value are listed. You can use this parameter with the <code>uploadIdMarker</code> parameter to specify the start position from which to list the returned results.	<code>ListMultipartUploadsRequest.setKeyMarker(String keyMarker)</code>

Parameter	Description	Method
uploadIdMarker	Specifies the start position from which to list returned results. This parameter is used together with the keyMarker parameter. If the keyMarker parameter is not configured, the uploadIdMarker parameter is invalid. If the keyMarker parameter is configured, the query results include the following items: • All objects whose names are alphabetically after the keyMarker value. • All multipart upload tasks with the same object name as the keyMarker value and an upload ID that is greater than the uploadIdMarker value.	ListMultipartUploadsRequest.setUploadIdMarker(String uploadIdMarker)

The following code provides an example on how to use simple list to list multipart upload tasks:

```
using Aliyun.OSS;
// Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com.
var endpoint = "yourEndpoint";
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
var accessKeyId = "yourAccessKeyId";
var accessKeySecret = "yourAccessKeySecret";
// Specify the name of the bucket.
var bucketName = "examplebucket";
// Create an OSSClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
try
{
    MultipartUploadListing multipartUploadListing = null;
    var nextMarker = string.Empty;
    do
    {
        // List multipart upload tasks.
        var request = new ListMultipartUploadsRequest(bucketName)
        {
            KeyMarker = nextMarker,
        };
        multipartUploadListing = client.ListMultipartUploads(request);
        Console.WriteLine("List multi part succeeded");
        // List the information about each multipart upload task.
        foreach (var mu in multipartUploadListing.MultipartUploads)
        {
            Console.WriteLine("Key: {0}, UploadId: {1}", mu.Key, mu.UploadId);
        }
        // If the value of the isTruncated field in the returned result is false, values of nextKeyMarker and nextUploadIdMarker are returned and used as the start point for the next read.
        nextMarker = multipartUploadListing.NextKeyMarker;
    } while (multipartUploadListing.IsTruncated);
}
catch (Exception ex)
{
    Console.WriteLine("List multi part uploads failed, {0}", ex.Message);
}
```

List multipart upload tasks by specifying a prefix and the maximum number of records to return

The following code provides an example on how to list multipart upload tasks by specifying a prefix and the maximum number of records to return:

```
using Aliyun.OSS;
// Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com.
var endpoint = "yourEndpoint";
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
var accessKeyId = "yourAccessKeyId";
var accessKeySecret = "yourAccessKeySecret";
// Specify the name of the bucket.
var bucketName = "examplebucket";
// Specify the prefix.
var prefix = "yourObjectPrefix";
// Specify the maximum number of records to return as 100.
var maxUploads = 100;
// Create an OSSClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
try
{
    MultipartUploadListing multipartUploadListing = null;
    var nextMarker = string.Empty;
    do
    {
        // List multipart upload tasks. By default, 1,000 multipart upload tasks are listed.

        var request = new ListMultipartUploadsRequest(bucketName)
        {
            KeyMarker = nextMarker,
            // Specify the prefix.
            Prefix = prefix,
            // Specify the maximum number of records to return.
            MaxUploads = maxUploads,
        };
        multipartUploadListing = client.ListMultipartUploads(request);
        Console.WriteLine("List multi part succeeded");
        // List the information about each multipart upload task.
        foreach (var mu in multipartUploadListing.MultipartUploads)
        {
            Console.WriteLine("Key: {0}, UploadId: {1}", mu.Key, mu.UploadId);
        }
        nextMarker = multipartUploadListing.NextKeyMarker;
    } while (multipartUploadListing.IsTruncated);
}
catch (Exception ex)
{
    Console.WriteLine("List multi part uploads failed, {0}", ex.Message);
}
```

References

- For more information about the complete sample code for multipart upload, visit [GitHub](#).

- The following API operations are required to perform multipart upload:
 - The API operation that you can call to initiate a multipart upload task. For more information, see [InitiateMultipartUpload](#).
 - The API operation that you can call to upload data by part. For more information, see [UploadPart](#).
 - The API operation that you can call to complete a multipart upload task. For more information, see [CompleteMultipartUpload](#).
- For more information about the API operation that you can call to cancel a multipart upload task, see [AbortMultipartUpload](#).
- For more information about the API operation that you can call to list the uploaded parts, see [ListParts](#).
- For more information about the API operation that you can call to list all ongoing multipart upload tasks, see [ListMultipartUploads](#). Ongoing multipart upload tasks include tasks that have been initiated but are not completed and tasks that are canceled.

7.6.6. Progress bars

You can configure a progress bar to display the progress of an upload or download task. This topic describes how to configure a progress bar when you call the PutObject operation to upload an object.

The following code provides an example on how to configure a progress bar to display the progress of an upload task:

```
using System;
using System.IO;
using System.Text;
using Aliyun.OSS;
using Aliyun.OSS.Common;
namespace PutObjectProgress
{
    class Program
    {
        static void Main(string[] args)
        {
            Program.PutObjectProgress();
            Console.ReadKey();
        }
        public static void PutObjectProgress()
        {
            // Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com.
            var endpoint = "<yourEndpoint>";
            // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in Object Storage Service (OSS) is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
            var accessKeyId = "<yourAccessKeyId>";
            var accessKeySecret = "<yourAccessKeySecret>";
            // Specify the bucket name.
            var bucketName = "<yourBucketName>";
            // Specify the full path of the object. The path cannot contain bucket names.
            var objectName = "<yourObjectName>";
```

```

        var objectName = <yourObjectName>;

        // Specify the full path of the local file to upload. In this example, the path
        // is set to D:\\localpath\\examplefile.txt to upload a local file named examplefile.txt in th
        // e D:\\localpath directory.
        var localFilename = "<yourLocalFilename>";
        // Create an OSSClient instance.
        var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
        // Configure a progress bar while uploading the object.
        try
        {
            using (var fs = File.Open(localFilename, FileMode.Open))
            {
                var putObjectRequest = new PutObjectRequest(bucketName, objectName, fs)

;

                putObjectRequest.StreamTransferProgress += streamProgressCallback;
                client.PutObject(putObjectRequest);
            }
            Console.WriteLine("Put object:{0} succeeded", objectName);
        }
        catch (OssException ex)
        {
            Console.WriteLine("Failed with error code: {0}; Error info: {1}. \nRequestID: {2}\\tHostID: {3}",
                ex.ErrorCode, ex.Message, ex.RequestId, ex.HostId);
        }
        catch (Exception ex)
        {
            Console.WriteLine("Failed with error info: {0}", ex.Message);
        }
    }

    // Display the progress of the upload task.
    private static void streamProgressCallback(object sender, StreamTransferProgressArgs args)
    {
        System.Console.WriteLine("ProgressCallback - Progress: {0}%, TotalBytes:{1}, TransferredBytes:{2} ",
            args.TransferredBytes * 100 / args.TotalBytes, args.TotalBytes, args.TransferredBytes);
    }
}

```

For the complete code that is used to configure progress bars, visit [GitHub](#).

7.6.7. Upload callback

After an object is uploaded, Object Storage Service (OSS) can start a callback process for the application server. To implement callback, you need only to send a request that contains relevant callback parameters to OSS.

Sample code

The following code provides an example on how to configure upload callback:

```

using System;
using System.IO;
using System.Text;
using Aliyun.OSS;
using Aliyun.OSS.Common;
using Aliyun.OSS.Util;
namespace Callback
{
    class Program
    {
        static void Main(string[] args)
        {
            Program.PutObjectCallback();
            Console.ReadKey();
        }
        public static void PutObjectCallback()
        {
            // Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com.
            var endpoint = "yourEndpoint";
            // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
            var accessKeyId = "yourAccessKeyId";
            var accessKeySecret = "yourAccessKeySecret";
            // Specify the name of the bucket. Example: examplebucket.
            var bucketName = "examplebucket";
            // Specify the full path of the object. The full path of the object cannot contain the bucket name. Example: exampledir/exampleobject.txt.
            var objectName = "exampledir/exampleobject.txt";
            // Specify the full path of the local file to upload. Example: D:\\localpath\\examplefile.txt. If you do not specify the full path of the local file, the local file is uploaded from the path of the project to which the sample program belongs.
            var localFilename = "D:\\localpath\\examplefile.txt";
            // Set the address of the application server that receives the callback request.
            const string callbackUrl = "http://oss-demo.aliyuncs.com:23450";
            // Set the value of the body field included in the callback request.
            const string callbackBody = "bucket=${bucket}&object=${object}&etag=${etag}&size=${size}&mimeType=${mimeType}&" +
                "my_var1=${x:var1}&my_var2=${x:var2}";
            // Create an OSSClient instance.
            var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
            try
            {
                string responseContent = "";
                var metadata = BuildCallbackMetadata(callbackUrl, callbackBody);
                using (var fs = File.Open(localFilename, FileMode.Open))
                {
                    var putObjectRequest = new PutObjectRequest(bucketName, objectName, fs, metadata);
                    var result = client.PutObject(putObjectRequest);
                    responseContent = GetCallbackResponse(result);
                }
            }
            catch { }
        }
    }
}

```

```

        responseContent = GetCallbackResponse(result);
    }
    Console.WriteLine("Put object:{0} succeeded, callback response content:{1}"
, objectName, responseContent);
}
catch (OssException ex)
{
    Console.WriteLine("Failed with error code: {0}; Error info: {1}. \nRequestID: {2}\tHostID: {3}",
        ex.ErrorCode, ex.Message, ex.RequestId, ex.HostId);
}
catch (Exception ex)
{
    Console.WriteLine("Failed with error info: {0}", ex.Message);
}
}
// Configure upload callback.
private static ObjectMetadata BuildCallbackMetadata(string callbackUrl, string callbackBody)
{
    string callbackHeaderBuilder = new CallbackHeaderBuilder(callbackUrl, callbackBody).Build();
    string CallbackVariableHeaderBuilder = new CallbackVariableHeaderBuilder().AddCallbackVariable("x:var1", "x:value1").AddCallbackVariable("x:var2", "x:value2").Build();
    var metadata = new ObjectMetadata();
    metadata.AddHeader(HttpHeaders.Callback, callbackHeaderBuilder);
    metadata.AddHeader(HttpHeaders.CallbackVar, CallbackVariableHeaderBuilder);
    return metadata;
}
// Read the response for the upload callback request.
private static string GetCallbackResponse(PutObjectResult putObjectResult)
{
    string callbackResponse = null;
    using (var stream = putObjectResult.ResponseStream)
    {
        var buffer = new byte[4 * 1024];
        var bytesRead = stream.Read(buffer, 0, buffer.Length);
        callbackResponse = Encoding.Default.GetString(buffer, 0, bytesRead);
    }
    return callbackResponse;
}
}
}

```

References

- For more information about the complete sample code that is used to configure upload callback, visit [GitHub](#).
- For more information about the API operations that you can call to configure upload callback, see [Callback](#).

7.7. Download objects

7.7.1. Overview

OSS .NET SDK provides the following download methods:

- [Streaming download](#)
- [Range download](#)
- [Resumable download](#)

You can view the download progress shown in [download progress bars](#).

7.7.2. Streaming download

If you have a large object to download or it is time-consuming to download the entire object at a time, you can use streaming download. Streaming download enables you to download part of the object each time until you have downloaded the entire object.

For the complete code of streaming download, see [GitHub](#).

Run the following code to download a specified OSS object to a stream:

```
using Aliyun.OSS;
var endpoint = "<yourEndpoint>";
var accessKeyId = "<yourAccessKeyId>";
var accessKeySecret = "<yourAccessKeySecret>";
var bucketName = "<yourBucketName>";
var objectName = "<yourObjectName>";
var downloadFilename = "<yourDownloadFilename>";
// Create an OSSClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
try
{
    // Download an object to a stream. OssObject includes object information, such as the bucket that contains the object, object name, metadata, and an input stream.
    var obj = client.GetObject(bucketName, objectName);
    using (var requestStream = obj.Content)
    {
        byte[] buf = new byte[1024];
        var fs = File.Open(downloadFilename, FileMode.OpenOrCreate);
        var len = 0;
        // Use the input stream to read the object content into a file or the memory.
        while ((len = requestStream.Read(buf, 0, 1024)) != 0)
        {
            fs.Write(buf, 0, len);
        }
        fs.Close();
    }
    Console.WriteLine("Get object succeeded");
}
catch (Exception ex)
{
    Console.WriteLine("Get object failed. {0}", ex.Message);
}
```

7.7.3. Range download

If you only need part of the data in an object, you can use range downloads to download specified content.

For the complete code of range download, see [GitHub](#).

Run the following code for range download:

```
using Aliyun.OSS;
var endpoint = "<yourEndpoint>";
var accessKeyId = "<yourAccessKeyId>";
var accessKeySecret = "<yourAccessKeySecret>";
var bucketName = "<yourBucketName>";
var objectName = "<yourObjectName>";
var downloadFilename = "<yourDownloadFilename>";
// Create an OSSClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
try
{
    var getObjectRequest = new GetObjectRequest(bucketName, objectName);
    // Set the range, which is from the 20th byte to the 100th byte.
    getObjectRequest.SetRange(20, 100);
    // Start range download. The setRange of getObjectRequest can be used to implement multipart download and resumable download.
    var obj = client.GetObject(getObjectRequest);
    // Download the data and write it into a file.
    using (var requestStream = obj.Content)
    {
        byte[] buf = new byte[1024];
        var fs = File.Open(downloadFilename, FileMode.OpenOrCreate);
        var len = 0;
        while ((len = requestStream.Read(buf, 0, 1024)) != 0)
        {
            fs.Write(buf, 0, len);
        }
        fs.Close();
    }
    Console.WriteLine("Get object succeeded");
}
catch (Exception ex)
{
    Console.WriteLine("Get object failed. {0}", ex.Message);
}
```

You can configure the following parameters of `GetObjectRequest`.

Parameter	Description
Range	Specifies the transmission range of an object.
ModifiedSinceConstraint	If the specified time is earlier than the actual modification time of the file, the file is transmitted normally. Otherwise, a 304 Not Modified exception is thrown.
UnmodifiedSinceConstraint	If the time passed to the parameter is equal to or later than the actual modification time of the file, the file is transmitted normally. Otherwise, a 412 precondition failed exception is thrown.

Parameter	Description
MatchingETagConstraints	Passes in a group of ETags. If the ETags match the ETag of the file, the file is transmitted normally. Otherwise, a 412 precondition failed exception is thrown.
NonmatchingETagConstraints	Passes in a group of ETags. If the ETags do not match the ETag of the file, the file is transmitted normally. Otherwise, a 304 Not Modified exception is thrown.
ResponseHeaderOverrides	Customizes some headers in requests returned by OSS.

7.7.4. Resumable download

You may fail to download a large object if the network is unstable or other exceptions occur. In some cases, you may still fail to download the object even after multiple attempts. To resolve this issue, Object Storage Service (OSS) SDK for .NET provides the resumable download feature. In resumable download, the object that you want to download is split into multiple parts and each part is separately downloaded. After all parts are downloaded, these parts are combined into a complete object.

Sample code

The following code provides an example on how to perform resumable download:

```
using Aliyun.OSS;
using Aliyun.OSS.Common;
// Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com.
var endpoint = "yourEndpoint";
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
var accessKeyId = "yourAccessKeyId";
var accessKeySecret = "yourAccessKeySecret";
// Specify the name of the bucket. Example: examplebucket.
var bucketName = "examplebucket";
// Specify the full path of the object. The full path cannot contain the bucket name. Example: exampledir/exampleobject.txt.
var objectName = "exampleobject.txt";
// Download the object to D:\\localpath. After the object is downloaded, the local file is named examplefile.txt. If the specified local file exists, the object to download overwrites the file. Otherwise, a file is created.
// If the path for the object is not specified, the downloaded object is saved to the path of the project to which the sample program belongs.
var downloadFilename = "D:\\localpath\\examplefile.txt";
// Specify the full path of the checkpoint file. Example: D:\\localpath\\examplefile.txt.dcp.
// After the checkpoint file is generated because the object download is interrupted, you m
```



```

ust set the full path of the checkpoint file only when you want to continue the download of
the object. After the object is downloaded, the checkpoint file is deleted.
var checkpointDir = "D:\\localpath\\examplefile.txt.dcp";
// Create an OSSClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
try
{
    // Configure multiple parameters by using DownloadObjectRequest.
    DownloadObjectRequest request = new DownloadObjectRequest(bucketName, objectName, downl
oadFilename)
    {
        // Specify the size of each part to download. Unit: bytes.
        PartSize = 8 * 1024 * 1024,
        // Specify the number of concurrent threads.
        ParallelThreadCount = 3,
        // checkpointDir is a file used to store the information about the resumable downlo
ad progress. If a part fails to be downloaded, the download can be continued based on the p
rogress information recorded in this file. If you set checkpointDir to null, resumable downl
oad does not take effect and objects are downloaded again when they fail to be downloaded.

        CheckpointDir = checkpointDir,
    };
    // Start the resumable download.
    client.ResumableDownloadObject(request);
    Console.WriteLine("Resumable download object:{0} succeeded", objectName);
}
catch (OssException ex)
{
    Console.WriteLine("Failed with error code: {0}; Error info: {1}. \nRequestId:{2}\tHostID:
{3}",
        ex.ErrorCode, ex.Message, ex.RequestId, ex.HostId);
}
catch (Exception ex)
{
    Console.WriteLine("Failed with error info: {0}", ex.Message);
}

```

References

For more information about the API operation that you can call to perform resumable download, see [GetObject](#).

7.7.5. Download progress bars

You can use progress bars to indicate the upload or download progress.

For the complete code of download progress bar, see [GitHub](#).

The following code is used as an example to describe how to view progress information with `GetObject`:

```

using System;
using System.IO;
using Aliyun.OSS;
using Aliyun.OSS.Common;

```

```

namespace GetObjectProgress
{
    class Program
    {
        static void Main(string[] args)
        {
            Program.GetObjectProgress();
            Console.ReadKey();
        }
        public static void GetObjectProgress()
        {
            var endpoint = "<yourEndpoint>";
            var accessKeyId = "<yourAccessKeyId>";
            var accessKeySecret = "<yourAccessKeySecret>";
            var bucketName = "<yourBucketName>";
            var objectName = "<yourObjectName>";
            // Create an OSSClient instance.
            var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
            try
            {
                var getObjectRequest = new GetObjectRequest(bucketName, objectName);
                getObjectRequest.StreamTransferProgress += streamProgressCallback;
                // Download an object.
                var ossObject = client.GetObject(getObjectRequest);
                using (var stream = ossObject.Content)
                {
                    var buffer = new byte[1024 * 1024];
                    var bytesRead = 0;
                    while ((bytesRead = stream.Read(buffer, 0, buffer.Length)) > 0)
                    {
                        // Process the read data (detailed code is not listed here).
                    }
                }
                Console.WriteLine("Get object:{0} succeeded", objectName);
            }
            catch (OssException ex)
            {
                Console.WriteLine("Failed with error code: {0}; Error info: {1}. \nRequestID:{2}\tHostID:{3}",
                    ex.ErrorCode, ex.Message, ex.RequestId, ex.HostId);
            }
            catch (Exception ex)
            {
                Console.WriteLine("Failed with error info: {0}", ex.Message);
            }
        }
        private static void streamProgressCallback(object sender, StreamTransferProgressArgs args)
        {
            System.Console.WriteLine("ProgressCallback - Progress: {0}%, TotalBytes:{1}, TransferredBytes:{2} ",
                args.TransferredBytes * 100 / args.TotalBytes, args.TotalBytes, args.TransferredBytes);
        }
    }
}

```

```
}  
}
```

7.8. Manage objects

7.8.1. Overview

This topic is the overview of object management in OSS .NET SDK.

You can perform the following operations on objects in a bucket with APIs:

- [Determine whether an object exists](#)
- [Manage ACL for an object](#)
- [Manage Object Meta](#)
- [List objects](#)
- [Delete an object](#)
- [Copy an object](#)
- [Restore an archive object](#)
- [Manage a symbolic link](#)

7.8.2. Determine whether a specified object exists

Run the following code to determine whether a specified object exists:

```

using Aliyun.OSS;
using Aliyun.OSS.Common;
var endpoint = "<yourEndpoint>";
var accessKeyId = "<yourAccessKeyId>";
var accessKeySecret = "<yourAccessKeySecret>";
var bucketName = "<yourBucketName>";
var objectName = "<yourObjectName>";
// Create an OSSClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
try
{
    // Determine whether a specified object exists.
    var exist = client.DoesObjectExist(bucketName, objectName);
    Console.WriteLine("Object exist ? " + exist);
}
catch (OssException ex)
{
    Console.WriteLine("Failed with error code: {0}; Error info: {1}. \nRequestId:{2}\tHostID:{3}",
        ex.ErrorCode, ex.Message, ex.RequestId, ex.HostId);
}
catch (Exception ex)
{
    Console.WriteLine("Failed with error info: {0}", ex.Message);
}

```

7.8.3. Manage object ACLs

This topic describes how to manage the access control list (ACL) of an object.

The following table describes the ACLs that you can configure for an object.

Permission	Description	Value
Inherited from bucket	The ACL of an object is the same as that of the bucket in which the object is stored.	CannedAccessControlList.Default
Private	Only the object owner or authorized users can read and write the object.	CannedAccessControlList.Private
Public read	Only the object owner or authorized users can write the object. Other users, including anonymous users can only read the object. Exercise caution when you configure this ACL.	CannedAccessControlList.PublicRead
Public	Any users, including anonymous users can read and write the object. Exercise caution when you configure this ACL.	CannedAccessControlList.PublicReadWrite

For the complete code used to configure the ACL of an object, visit [Git Hub](#). For the complete code used to query the ACL of an object, visit [Git Hub](#).

Run the following code to configure and query the ACL of the object.

```
using Aliyun.OSS;
using Aliyun.OSS.Common;
var endpoint = "<yourEndpoint>";
var accessKeyId = "<yourAccessKeyId>";
var accessKeySecret = "<yourAccessKeySecret>";
var bucketName = "<yourBucketName>";
var objectName = "<yourObjectName>";
// Create an OSSClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
// Configure the object ACL.
try
{
    // Use SetObjectAcl to configure the ACL of the object.
    client.SetObjectAcl(bucketName, objectName, CannedAccessControlList.PublicRead);
    Console.WriteLine("Set Object:{0} ACL succeeded ", objectName);
}
catch (Exception ex)
{
    Console.WriteLine("Set Object ACL failed with error info: {0}", ex.Message);
}
// Query the ACL of the object.
try
{
    // Use GetObjectAcl to query the ACL of the object.
    var result = client.GetObjectAcl(bucketName, objectName);
    Console.WriteLine("Get Object ACL succeeded, Id: {0}  ACL: {1}",
        result.Owner.Id, result.ACL.ToString());
}
catch (OssException ex)
{
    Console.WriteLine("Failed with error code: {0}; Error info: {1}. \nRequestID: {2}\tHost ID: {3}",
        ex.ErrorCode, ex.Message, ex.RequestId, ex.HostId);
}
catch (Exception ex)
{
    Console.WriteLine("Failed with error info: {0}", ex.Message);
}
```

For more information about object ACLs, see [Overview](#).

7.8.4. Manage object metadata

Object metadata includes HTTP headers and user metadata. This topic describes how to configure, modify, and query object metadata.

Sample code

The following code provides an example on how to configure, modify, and query object metadata:

```
using Aliyun.OSS;
using Aliyun.OSS.Common;
// Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com.
var endpoint = "yourEndpoint";
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access Object Storage Service (OSS) because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
var accessKeyId = "yourAccessKeyId";
var accessKeySecret = "yourAccessKeySecret";
// Specify the bucket name.
var bucketName = "examplebucket";
// Specify the full path of the object. The full path cannot contain the bucket name.
var objectName = "exampleobject.txt";
// Specify the full path of the local file that you want to upload. By default, if you do not specify the full path of the local file, the local file is uploaded from the path of the project to which the sample program belongs.
var localFilename = "D:\\localpath\\examplefile.txt";
// Create an OSSClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
try
{
    using (var fs = File.Open(localFilename, FileMode.Open))
    {
        // Create the metadata of the object. You can configure the HTTP headers of the object.
        var metadata = new ObjectMetadata()
        {
            // Specify the object type.
            ContentType = "text/html",
            // Specify the expiration time of the cache in GMT.
            ExpirationTime = DateTime.Parse("2025-10-12T00:00:00.000Z"),
        };
        // Specify the length of the object content to upload. If the actual object length is greater than the specified length, the object is uploaded based on the specified length. If the object is smaller than the specified length, the object is uploaded based on the actual length.
        metadata.ContentLength = fs.Length;
        // Specify the caching behavior of the website when the content is downloaded.
        metadata.CacheControl = "No-Cache";
        // Set the value of mykey1 for user metadata to myval1.
        metadata.UserMetadata.Add("mykey1", "myval1");
        // Set the value of mykey2 for user metadata to myval2.
        metadata.UserMetadata.Add("mykey2", "myval2");
        var saveAsFilename = "Filetest123.txt";
        var contentDisposition = string.Format("attachment;filename*=utf-8''{0}", HttpUtils.EncodeUri(saveAsFilename, "utf-8"));
        // Provide a default object name when the requested content is stored as a object.
        metadata.ContentDisposition = contentDisposition;
        // Upload the object and configure object metadata.
        client.PutObject(bucketName, objectName, fs, metadata);
        Console.WriteLine("Put object succeeded");
    }
}
```

```
// Query object metadata.
var oldMeta = client.GetObjectMetadata(bucketName, objectName);
// Configure the metadata of the destination object.
var newMeta = new ObjectMetadata()
{
    ContentType = "application/octet-stream",
    ExpirationTime = DateTime.Parse("2035-11-11T00:00:00.000Z"),
    // Specify the encoding format that is used when the object is downloaded.
    ContentEncoding = null,
    CacheControl = ""
};
// Configure user metadata.
newMeta.UserMetadata.Add("author", "oss");
newMeta.UserMetadata.Add("flag", "my-flag");
newMeta.UserMetadata.Add("mykey2", "myval2-modified-value");
// Modify object metadata by using the ModifyObjectMeta method.
client.ModifyObjectMeta(bucketName, objectName, newMeta);
}
}
catch (Exception ex)
{
    Console.WriteLine("Put object failed, {0}", ex.Message);
}
```

References

- For the complete sample code that is used to configure, modify, and query object metadata, visit [GitHub](#).
- For more information about the API operation that you can call to configure object metadata in simple upload, see [Put Object](#).
- For more information about the API operation that you can call to modify object metadata in a copy task, see [CopyObject](#).
- For more information about the API operation that you can call to query object metadata, see [GetObjectMeta](#).

7.8.5. List objects

This topic describes how to list all objects in a bucket, a specified number of objects in a bucket, and objects whose names contain a specified prefix in a bucket.

Simple list

The following code provides an example on how to list 100 objects in a specified bucket:

```
using Aliyun.OSS;
// Set yourEndpoint to the endpoint of the bucket. For example, if the bucket is located in
the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com. Spec
ify the endpoint based on your business requirements.
var endpoint = "yourEndpoint";
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to ac
cess Object Storage Service (OSS) because the account has permissions on all API operations
. We recommend that you use a RAM user to call API operations or perform routine O&M. To cr
eate a RAM user, log on to the RAM console.
var accessKeyId = "yourAccessKeyId";
var accessKeySecret = "yourAccessKeySecret";
// Specify the name of the bucket. Example: examplebucket.
var bucketName = "examplebucket";
// Create an OSSClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
try
{
    var listObjectsRequest = new ListObjectsRequest(bucketName);
    // Provide a simple list of objects in the specified bucket. By default, 100 objects ar
e returned.
    var result = client.ListObjects(listObjectsRequest);
    Console.WriteLine("List objects succeeded");
    foreach (var summary in result.ObjectSummaries)
    {
        Console.WriteLine("File name:{0}", summary.Key);
    }
}
catch (Exception ex)
{
    Console.WriteLine("List objects failed. {0}", ex.Message);
}
```

List a specified number of objects in a bucket

The following code provides an example on how to list a specified number of objects:


```
using Aliyun.OSS;
// Set yourEndpoint to the endpoint of the bucket. For example, if the bucket is located in
the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com. Spec
ify the endpoint based on your business requirements.
var endpoint = "yourEndpoint";
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to ac
cess OSS because the account has permissions on all API operations. We recommend that you u
se a RAM user to call API operations or perform routine O&M. To create a RAM user, log on t
o the RAM console.
var accessKeyId = "yourAccessKeyId";
var accessKeySecret = "yourAccessKeySecret";
// Specify the name of the bucket. Example: examplebucket.
var bucketName = "examplebucket";
// Create an OSSClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
try
{
    var listObjectsRequest = new ListObjectsRequest(bucketName)
    {
        // Specify MaxKeys to set the maximum number of listed objects to 200. The default
value of MaxKeys is 100. The maximum value of MaxKeys is 1000.
        MaxKeys = 200,
    };
    var result = client.ListObjects(listObjectsRequest);
    Console.WriteLine("List objects succeeded");
    foreach (var summary in result.ObjectSummaries)
    {
        Console.WriteLine(summary.Key);
    }
}
catch (Exception ex)
{
    Console.WriteLine("List objects failed, {0}", ex.Message);
}
```

List objects whose names contain a specified prefix in a bucket

The following code provides an example on how to list objects whose names contain a specified prefix:

```
using Aliyun.OSS;
using Aliyun.OSS.Common;
// Set yourEndpoint to the endpoint of the bucket. For example, if the bucket is located in
the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com. Spec
ify the endpoint based on your business requirements.
var endpoint = "yourEndpoint";
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to ac
cess OSS because the account has permissions on all API operations. We recommend that you u
se a RAM user to call API operations or perform routine O&M. To create a RAM user, log on t
o the RAM console.
var accessKeyId = "yourAccessKeyId";
var accessKeySecret = "yourAccessKeySecret";
// Specify the name of the bucket. Example: examplebucket.
var bucketName = "examplebucket";
// List objects whose names contain the specified prefix "test". By default, a maximum of 1
00 objects are listed.
var prefix = "test";
// Create an OSSClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
try
{
    var keys = new List<string>();
    ObjectListing result = null;
    string nextMarker = string.Empty;
    do
    {
        var listObjectsRequest = new ListObjectsRequest(bucketName)
        {
            Marker = nextMarker,
            MaxKeys = 100,
            Prefix = prefix,
        };
        result = client.ListObjects(listObjectsRequest);
        foreach (var summary in result.ObjectSummaries)
        {
            Console.WriteLine(summary.Key);
            keys.Add(summary.Key);
        }
        nextMarker = result.NextMarker;
    } while (result.IsTruncated);
    Console.WriteLine("List objects of bucket:{0} succeeded ", bucketName);
}
catch (OssException ex)
{
    Console.WriteLine("Failed with error code: {0}; Error info: {1}. \nRequestID:{2}\tHostI
D:{3}",
        ex.ErrorCode, ex.Message, ex.RequestId, ex.HostId);
}
catch (Exception ex)
{
    Console.WriteLine("Failed with error info: {0}", ex.Message);
}
```

List objects in a bucket by marker

You can configure the marker parameter to specify the name of the object after which the list operation starts. The following code provides an example on how to list objects whose names are alphabetically after the object specified by marker:

```

using Aliyun.OSS;
using Aliyun.OSS.Common;
// Set yourEndpoint to the endpoint of the bucket. For example, if the bucket is located in
the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com. Spec
ify the endpoint based on your business requirements.
var endpoint = "yourEndpoint";
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to ac
cess OSS because the account has permissions on all API operations. We recommend that you u
se a RAM user to call API operations or perform routine O&M. To create a RAM user, log on t
o the RAM console.
var accessKeyId = "yourAccessKeyId";
var accessKeySecret = "yourAccessKeySecret";
// Specify the name of the bucket. Example: examplebucket.
var bucketName = "examplebucket";
// Specify the marker parameter. Example: exampleobject.txt.
var marker = "exampleobject.txt";
// Create an OSSClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
try
{
    var keys = new List<string>();
    ObjectListing result = null;
    string nextMarker = marker;
    do
    {
        var listObjectsRequest = new ListObjectsRequest(bucketName)
        // You can modify the value of MaxKeys to return more objects or list objects by pa
ge.
        {
            Marker = nextMarker,
            MaxKeys = 100,
        };
        result = client.ListObjects(listObjectsRequest);
        foreach (var summary in result.ObjectSummaries)
        {
            Console.WriteLine(summary.Key);
            keys.Add(summary.Key);
        }
        nextMarker = result.NextMarker;
        // If the value of IsTruncated is true, the next read starts from NextMarker.
    } while (result.IsTruncated);
    Console.WriteLine("List objects of bucket:{0} succeeded ", bucketName);
}
catch (OssException ex)
{
    Console.WriteLine("Failed with error code: {0}; Error info: {1}. \nRequestId:{2}\tHostI
D:{3}",
        ex.ErrorCode, ex.Message, ex.RequestId, ex.HostId);
}
catch (Exception ex)
{
    Console.WriteLine("Failed with error info: {0}", ex.Message);
}

```

List all objects in a bucket

The following code provides an example on how to list all objects in a specified bucket:

```
using Aliyun.OSS;

// Set yourEndpoint to the endpoint of the bucket. For example, if the bucket is located in
// the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com. Spec
// ify the endpoint based on your business requirements.
var endpoint = "yourEndpoint";
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to ac
// cess OSS because the account has permissions on all API operations. We recommend that you u
// se a RAM user to call API operations or perform routine O&M. To create a RAM user, log on t
// o the RAM console.
var accessKeyId = "yourAccessKeyId";
var accessKeySecret = "yourAccessKeySecret";
// Specify the name of the bucket. Example: examplebucket.
var bucketName = "examplebucket";
// Initialize OssClient.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
// List all objects in the bucket.
public void ListObject(string bucketName)
{
    try
    {
        ObjectListing result = null;
        string nextMarker = string.Empty;
        do
        {
            // You can configure the MaxKeys parameter to specify the maximum number of obj
            // ects that can be listed on each page. If the number of returned objects exceeds the value o
            // f MaxKeys, the objects are listed by page.
            var listObjectsRequest = new ListObjectsRequest(bucketName)
            {
                Marker = nextMarker,
                MaxKeys = 100
            };
            result = client.ListObjects(listObjectsRequest);
            Console.WriteLine("File:");
            foreach (var summary in result.ObjectSummaries)
            {
                Console.WriteLine("Name:{0}", summary.Key);
            }
            nextMarker = result.NextMarker;
        } while (result.IsTruncated);
    }
    catch (Exception ex)
    {
        Console.WriteLine("List object failed. {0}", ex.Message);
    }
}
```

Asynchronously list objects in a bucket

The following code provides an example on how to asynchronously list objects in a specified bucket:

```
using System;
using System.IO;
using System.Threading;
using Aliyun.OSS;
namespace AsyncListObjects
{
    class Program
    {
        // Set yourEndpoint to the endpoint of the bucket. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com. Specify the endpoint based on your business requirements.
        static string endpoint = "yourEndpoint";
        // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
        static string accessKeyId = "yourAccessKeyId";
        static string accessKeySecret = "yourAccessKeySecret";
        // Specify the name of the bucket. Example: examplebucket.
        static string bucketName = "yourBucketName";
        static AutoResetEvent _event = new AutoResetEvent(false);
        // Create an OSSClient instance.
        static OssClient client = new OssClient(endpoint, accessKeyId, accessKeySecret);
        static void Main(string[] args)
        {
            Program.AsyncListObjects();
            Console.ReadKey();
        }
        public static void AsyncListObjects()
        {
            try
            {
                var listObjectsRequest = new ListObjectsRequest(bucketName);
                client.BeginListObjects(listObjectsRequest, ListObjectCallback, null);
                _event.WaitOne();
            }
            catch (Exception ex)
            {
                Console.WriteLine("Async list objects failed. {0}", ex.Message);
            }
        }
        // Call the ListObjectCallback method to implement callback after the objects are asynchronously listed. If you call asynchronous operations to list objects, the operations must implement callback in similar methods.
        private static void ListObjectCallback(IAsyncResult ar)
        {
            try
            {
                var result = client.EndListObjects(ar);
                foreach (var summary in result.ObjectSummaries)
                {
                    Console.WriteLine("Object name: {0}", summary.Key);
                }
            }
            catch { }
        }
    }
}
```

```
        }
        _event.Set();
        Console.WriteLine("Async list objects succeeded");
    }
    catch (Exception ex)
    {
        Console.WriteLine("Async list objects failed. {0}", ex.Message);
    }
}
}
```

References

- For more information about the complete sample code that is used to synchronously list objects, see [GitHub Sample](#).
- For more information about the complete sample code that is used to asynchronously list objects, see [GitHub Sample](#).
- For more information about the API operations that are called to list objects, see [Get Bucket \(List Objects\)](#).

7.8.6. Delete objects

This topic describes how to delete a single object or multiple objects.



Warning Deleted objects cannot be recovered. Exercise caution when you delete objects.

Delete a single object

The following code provides an example on how to delete an object named *exampleobject.txt* from a bucket named *examplebucket*:

```
using Aliyun.OSS;
// Specify the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com.
var endpoint = "yourEndpoint";
// The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in Object Storage Service (OSS) is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
var accessKeyId = "yourAccessKeyId";
var accessKeySecret = "yourAccessKeySecret";
var bucketName = "examplebucket";
var objectName = "exampleobject.txt";
// Create an OSSClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
try
{
    // Delete the object.
    client.DeleteObject(bucketName, objectName);
    Console.WriteLine("Delete object succeeded");
}
catch (Exception ex)
{
    Console.WriteLine("Delete object failed. {0}", ex.Message);
}
```

Delete multiple objects

You can delete up to 1,000 objects at a time.

The result can be returned in one of the following modes:

- **verbose**: If `quietMode` is not specified or is set to `false`, a list of all deleted objects is returned. This is the default return mode.
- **quiet**: If `quietMode` is set to `true`, a list of objects that fail to be deleted is returned.

The following code provides an example on how to delete multiple specified objects from a bucket named `examplebucket`:


```
using Aliyun.OSS;
// Specify the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com.
var endpoint = "yourEndpoint";
// The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
var accessKeyId = "yourAccessKeyId";
var accessKeySecret = "yourAccessKeySecret";
var bucketName = "examplebucket";
// Create an OSSClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
try
{
    // Specify the paths of multiple objects that you want to delete. The full paths of the objects cannot contain the bucket name.
    var keys = new List<string>();
    keys.Add("exampleobject.txt");
    keys.Add("testdir/sampleobject.txt");
    // Set quietMode to false to return the list of all deleted objects.
    var quietMode = false;
    var request = new DeleteObjectsRequest(bucketName, keys, quietMode);
    // Delete multiple objects.
    var result = client.DeleteObjects(request);
    if ((!quietMode) && (result.Keys != null))
    {
        foreach (var obj in result.Keys)
        {
            Console.WriteLine("Delete successfully : {0} ", obj.Key);
        }
    }
    Console.WriteLine("Delete objects succeeded");
}
catch (Exception ex)
{
    Console.WriteLine("Delete objects failed. {0}", ex.Message);
}
```

For the complete code that is used to delete multiple objects, visit [GitHub](#).

7.8.7. Copy objects

This topic describes how to copy an object within a bucket or across buckets in the same region.

Precautions

- To copy an object, you must have read permissions on the source object and read and write permissions on the destination bucket.
- The source bucket and the destination bucket must have no retention policies configured. Otherwise, the error message `The object you specified is immutable.` is returned.

- The source bucket and the destination bucket must be in the same region. For example, objects cannot be copied from a bucket located in the China (Hangzhou) region to another bucket located in the China (Qingdao) region.

Copy a small object

The following code provides an example on how to copy a small object :

```

using Aliyun.OSS;
using Aliyun.OSS.Common;
// Specify the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com.
var endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access Object Storage Service (OSS) because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
var accessKeyId = "yourAccessKeyId";
var accessKeySecret = "yourAccessKeySecret";
// Specify the name of the source bucket. Example: srcexamplebucket.
var sourceBucket = "srcexamplebucket";
// Specify the full path of the source object. The full path cannot contain the bucket name. Example: srcdir/scrobject.txt.
var sourceObject = "srcdir/scrobject.txt";
// Specify the name of the destination bucket. The destination bucket must be located in the same region as the source bucket. Example: destbucket.
var targetBucket = "destbucket";
// Specify the full path of the destination object. The full path cannot contain the bucket name. Example: destdir/destobject.txt.
var targetObject = "destdir/destobject.txt";
// Create an OssClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
try
{
    var metadata = new ObjectMetadata();
    metadata.AddHeader("mk1", "mv1");
    metadata.AddHeader("mk2", "mv2");
    var req = new CopyObjectRequest(sourceBucket, sourceObject, targetBucket, targetObject)
    {
        // If the value of NewObjectMetadata is null, the COPY mode is used and the metadata of the source object is copied to the destination object. If the value of NewObjectMetadata is not null, the REPLACE mode is used and the metadata of the destination object is overwritten by the metadata specified in the request.
        NewObjectMetadata = metadata
    };
    // Copy the object.
    client.CopyObject(req);
    Console.WriteLine("Copy object succeeded");
}
catch (OssException ex)
{
    Console.WriteLine("Failed with error code: {0}; Error info: {1}. \nRequestId: {2} \tHostID: {3}",
        ex.ErrorCode, ex.Message, ex.RequestId, ex.HostId);
}
catch (Exception ex)
{
    Console.WriteLine("Failed with error info: {0}", ex.Message);
}

```

Copy a large object

- Copy objects by using multipart copy

To copy an object that is larger than 1 GB in size, you must split the object into parts and sequentially copy the parts by using `UploadPartCopy`. To implement multipart copy, perform the following steps:

- i. Call the `InitiateMultipartUploadRequest` method to initiate a multipart copy task.
- ii. Call the `UploadPartCopy` method to perform multipart copy.
- iii. Call the `CompleteMultipartUpload` method to complete the multipart copy task.

The following code provides an example on how to perform multipart copy:

```
using Aliyun.OSS;

// Specify the endpoint of the region in which the bucket is located. For example, if the
// bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hang
// zhou.aliyuncs.com.
var endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to
// access OSS because the account has permissions on all API operations. We recommend that y
// ou use a RAM user to call API operations or perform routine O&M. To create a RAM user, lo
// g on to the RAM console.
var accessKeyId = "yourAccessKeyId";
var accessKeySecret = "yourAccessKeySecret";
// Specify the name of the source bucket. Example: srcexamplebucket.
var sourceBucket = "srcexamplebucket";
// Specify the full path of the source object. The full path cannot contain the bucket na
// me. Example: srcdir/scrobject.txt.
var sourceObject = "srcdir/scrobject.txt";
// Specify the name of the destination bucket. The destination bucket must be located in
// the same region as the source bucket. Example: destbucket.
var targetBucket = "destbucket";
// Specify the full path of the destination object. The full path cannot contain the buck
// et name. Example: destdir/destobject.txt.
var targetObject = "destdir/destobject.txt";
var uploadId = "";
var partSize = 50 * 1024 * 1024;
// Create an OssClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
try
{
    // Initiate a multipart copy task. You can use InitiateMultipartUploadRequest to spec
    // ify the metadata of the destination object.
    var request = new InitiateMultipartUploadRequest(targetBucket, targetObject);
    var result = client.InitiateMultipartUpload(request);
    // Display the upload ID.
    uploadId = result.UploadId;
    Console.WriteLine("Init multipart upload succeeded, Upload Id: {0}", result.UploadId)
;
    // Calculate the total number of parts.
    var metadata = client.GetObjectMetadata(sourceBucket, sourceObject);
    var fileSize = metadata.ContentLength;
    var partCount = (int)fileSize / partSize;
    if (fileSize % partSize != 0)
    {
```

```

        partCount++;
    }
    // Start the multipart copy task.
    var partETags = new List<PartETag>();
    for (var i = 0; i < partCount; i++)
    {
        var skipBytes = (long)partSize * i;
        var size = (partSize < fileSize - skipBytes) ? partSize : (fileSize - skipBytes);
        // Create an UploadPartCopyRequest object. You can use UploadPartCopyRequest to s
        pecify conditions.
        var uploadPartCopyRequest = new UploadPartCopyRequest(targetBucket, targetObject,
            sourceBucket, sourceObject, uploadId)
        {
            PartSize = size,
            PartNumber = i + 1,
            // Use BeginIndex to find the start position to upload the part.
            BeginIndex = skipBytes
        };
        // Call the UploadPartCopy method to copy each part.
        var uploadPartCopyResult = client.UploadPartCopy(uploadPartCopyRequest);
        Console.WriteLine("UploadPartCopy : {0}", i);
        partETags.Add(uploadPartCopyResult.PartETag);
    }
    // Complete the multipart copy task.
    var completeMultipartUploadRequest =
        new CompleteMultipartUploadRequest(targetBucket, targetObject, uploadId);
    // partETags is a list of partETag. OSS verifies each part after it receives partETag
    s. After all parts are verified, OSS combines these parts into a complete object.
    foreach (var partETag in partETags)
    {
        completeMultipartUploadRequest.PartETags.Add(partETag);
    }
    var completeMultipartUploadResult = client.CompleteMultipartUpload(completeMultipartU
        ploadRequest);
    Console.WriteLine("CompleteMultipartUpload succeeded");
}
catch (OssException ex)
{
    Console.WriteLine("Failed with error code: {0}; Error info: {1}. \nRequestID: {2} \tH
        ostID: {3}",
        ex.ErrorCode, ex.Message, ex.RequestId, ex.HostId);
}
catch (Exception ex)
{
    Console.WriteLine("Failed with error info: {0}", ex.Message);
}

```

- Resumable copy

If a resumable copy task is interrupted, you can continue the copy task.

The following code provides an example on how to copy an object by using resumable copy:

```

using Aliyun.OSS;
// Specify the endpoint of the region in which the bucket is located. For example, if the
bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hang
zhou.aliyuncs.com.
var endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to
access OSS because the account has permissions on all API operations. We recommend that y
ou use a RAM user to call API operations or perform routine O&M. To create a RAM user, lo
g on to the RAM console.
var accessKeyId = "yourAccessKeyId";
var accessKeySecret = "yourAccessKeySecret";
// Specify the name of the source bucket. Example: srcexamplebucket.
var sourceBucket = "srcexamplebucket";
// Specify the full path of the source object. The full path cannot contain the bucket na
me. Example: srcdir/scrobject.txt.
var sourceObject = "srcdir/scrobject.txt";
// Specify the name of the destination bucket. The destination bucket must be located in
the same region as the source bucket. Example: destbucket.
var targetBucket = "destbucket";
// Specify the full path of the destination object. The full path cannot contain the buck
et name. Example: destdir/destobject.txt.
var targetObject = "destdir/destobject.txt";
// Specify the checkpoint file that records the results of the resumable copy task. This
file stores information about the copy progress. If an object fails to be copied, the nex
t copy operation continues based on the recorded progress. After the object is copied, th
e checkpoint file is deleted.
var checkpointDir = @"yourCheckpointDir";
// Create an OssClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
try
{
    var request = new CopyObjectRequest(sourceBucket, sourceObject, targetBucket, targetO
bject);
    // Specify the checkpointDir directory in which the checkpoint file is stored. The ch
eckpoint file records the state of the resumable copy task and can be used to resume the
task in case of failures. If you set checkpointDir to null, resumable copy does not take
effect, and the object must be copied again from the start if it fails to be copied.
    client.ResumableCopyObject(request, checkpointDir);
    Console.WriteLine("Resumable copy new object:{0} succeeded", request.DestinationKey);
}
catch (Exception ex)
{
    Console.WriteLine("Resumable copy new object failed, {0}", ex.Message);
}

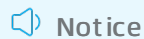
```

References

- Copy a small object
 - For the complete sample code that is used to copy a small object, visit [GitHub](#).
 - For more information about the API operation that you can call to copy a small object, see [CopyObject](#).
- Copy a large object

- For the complete sample code that is used to copy a large object, visit [GitHub](#).
- For more information about the API operation that you can call to copy a large object, see [UploadPartCopy](#).
- Resumable copy
For the complete sample code that is used to perform resumable copy, visit [GitHub](#).

7.8.8. Restore objects



Restoration

If you want to read an Archive object or a Cold Archive object, you must restore the object in advance. It takes several minutes to restore an Archive object and takes several hours to restore a Cold Archive object.

The following section describes the status of an Archive object or a Cold Archive object throughout the restoration process:

1. By default, an Archive or a Cold Archive object is in the frozen state before restoration.
2. After you submit a restore request, the object is in the restoring state.
3. After the server completes the restore task, the object enters the restored state and you can read the object.
 - Archive objects
For Archive objects, the restored state lasts 24 hours by default. During the 24 hours, if you call `RestoreObject` again, the restored state is extended by 24 hours. You can extend the restored state to seven days by calling `RestoreObject` six times during the first 24 hours of restored state. You can also configure the duration of the restored state in days by calling `RestoreObject` once. You can specify a duration of at most seven days.
 - Cold Archive objects
For Cold Archive objects, you can specify the duration of restored state and restoration priority. The duration of the restored state must be at least one day and at most 365 days. The time required to restore a Cold Archive object to the readable state is determined based on the restoration priority of the object:
 - Expedited: The object is restored within one hour.
 - Standard: The object is restored within two to five hours. If the `JobParameters` element is not passed in, the default restoration priority is Standard.
 - Bulk: The object is restored within five to twelve hours.
4. After the restored state expires, the object returns to the frozen state.

Restore an Archive object

```
using Aliyun.OSS;  
using Aliyun.OSS.Model;  
// Set yourEndpoint to the endpoint of the region in which the bucket is located. For example,  
// if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-  
//-cn-hangzhou.aliyuncs.com
```

```

s-cn-hangzhou.aliyuncs.com.
var endpoint = "yourEndpoint";
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS, because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
var accessKeyId = "yourAccessKeyId";
var accessKeySecret = "yourAccessKeySecret";
// Specify the name of the bucket.
var bucketName = "examplebucket";
// Specify the full path of the object. The path cannot contain the bucket name.
var objectName = "yourObjectName";
// Specify the content of the object.
var objectContent = "More than just cloud.";
int maxWaitTimeInSeconds = 600;
// Create an OSSClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
try
{
    // Create an Archive bucket.
    var bucket = client.CreateBucket(bucketName, StorageClass.Archive);
    Console.WriteLine("Create Archive bucket succeeded, {0} ", bucket.Name);
}
catch (Exception ex)
{
    Console.WriteLine("Create Archive bucket failed, {0}", ex.Message);
}
// Upload the object to the bucket and set the storage class of the object to Archive.
try
{
    byte[] binaryData = Encoding.ASCII.GetBytes(objectContent);
    MemoryStream requestContent = new MemoryStream(binaryData);
    client.PutObject(bucketName, objectName, requestContent);
    Console.WriteLine("Put object succeeded, {0}", objectName);
}
catch (Exception ex)
{
    Console.WriteLine("Put object failed, {0}", ex.Message);
}
var metadata = client.GetObjectMetadata(bucketName, objectName);
string storageClass = metadata.HttpMetadata["x-oss-storage-class"] as string;
if (storageClass != "Archive")
{
    Console.WriteLine("StorageClass is {0}", storageClass);
    return;
}
// Restore the Archive object.
RestoreObjectResult result = client.RestoreObject(bucketName, objectName);
Console.WriteLine("RestoreObject result HttpStatusCode : {0}", result.HttpStatusCode);
if (result.HttpStatusCode != HttpStatusCode.Accepted)
{
    throw new OssException(result.RequestId + ", " + result.HttpStatusCode + " ,");
}
while (maxWaitTimeInSeconds > 0)
{

```



```
var meta = client.GetObjectMetadata(bucketName, objectName);
string restoreStatus = meta.HttpMetadata["x-oss-restore"] as string;
if (restoreStatus != null && restoreStatus.StartsWith("ongoing-request=\"false\"", StringComparison.InvariantCultureIgnoreCase))
{
    break;
}
Thread.Sleep(1000);
maxWaitTimeInSeconds--;
}
if (maxWaitTimeInSeconds == 0)
{
    Console.WriteLine("RestoreObject is timeout. ");
    throw new TimeoutException();
}
else
{
    Console.WriteLine("RestoreObject is successful. ");
}
```

Restore a Cold Archive object

The following code provides an example on how to restore a Cold Archive object:

```
using Aliyun.OSS;
using Aliyun.OSS.Model;
// Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com.
var endpoint = "yourEndpoint";
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS, because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
var accessKeyId = "yourAccessKeyId";
var accessKeySecret = "yourAccessKeySecret";
// Specify the name of the bucket.
var bucketName = "examplebucket";
// Specify the full path of the object. The path cannot contain the bucket name.
var objectName = "yourObjectName";
// Create an OSSClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
// Restore the Cold Archive object.
var request = new RestoreObjectRequest(bucketName, objectName);
// Set the restoration priority of the Cold Archive object.
// TierType.Expedited: The object is restored within 1 hour.
// TierType.Standard: The object is restored within 2 to 5 hours.
// TierType.Bulk: The object is restored within 5 to 12 hours.
// Configure parameters. For example, set the restoration priority of the object to Expedited, and set the duration for which the object can remain in the restored state to two days.

// The Days parameter indicates the duration for which the object can remain in the restored state. The default value is one day. This parameter applies to Archive and Cold Archive objects.
// The Tier parameter indicates the restoration priority of the object and applies only to Cold Archive objects.
request.Days = 2;
request.Tier = TierType.Expedited;
RestoreObjectResult result = client.RestoreObject(request);
Console.WriteLine("RestoreObject result HttpStatusCode : {0}", result.HttpStatusCode);
```

References

- For more information about the complete sample code that is used to restore Archive and Cold Archive objects, visit [GitHub](#).
- For more information about the API operation that you can call to restore Archive and Cold Archive objects, see [RestoreObject](#).

7.8.9. Manage symbolic links

This topic describes how to create a symbolic link and obtain the name of the object to which the symbolic link points.

A symbolic link is a special object that maps to an object similar to a shortcut used in Windows. You can configure user-defined Object Meta for symbolic links.

 **Note** To obtain a symbolic link, you must have the read permissions on it. For more information about symbolic links, see [GetSymlink](#).

The following code provides an example on how to create a symbolic link and obtain the name of the object to which the symbolic link points:

```
using Aliyun.OSS;
var endpoint = "<yourEndpoint>";
var accessKeyId = "<yourAccessKeyId>";
var accessKeySecret = "<yourAccessKeySecret>";
var bucketName = "<yourBucketName>";
var targetObjectName = "<yourTargetObjectName>";
var symlinkObjectName = "<yourSymlinkObjectName>";
var objectContent = "More than just cloud." ;
// Create an OSSClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
try
{
    // Upload the object to which the symbolic link points.
    byte[] binaryData = Encoding.ASCII.GetBytes(objectContent);
    MemoryStream requestContent = new MemoryStream(binaryData);
    client.PutObject(bucketName, targetObjectName, requestContent);
    // Create the symbolic link.
    client.CreateSymlink(bucketName, symlinkObjectName, targetObjectName);
    // Obtain the name of the object to which the symbolic link points.
    var ossSymlink = client.GetSymlink(bucketName, symlinkObjectName);
    Console.WriteLine("Target object is {0}", ossSymlink.Target);
}
catch (Exception ex)
{
    Console.WriteLine("Failed with error info: {0}", ex.Message);
}
```

7.9. Versioning

7.9.1. Versioning

The versioning state of a bucket applies to all objects in the bucket. If you enable versioning for a bucket, you can recover any previous version of an object in the bucket when you accidentally overwrite or delete the object.

A bucket can be in one of the following versioning states: unversioned (default), versioning-enabled, or versioning-suspended. For more information about versioning, see [Overview](#).

Configure versioning for a bucket

The following code provides an example on how to set the versioning state of a bucket to Enabled or Suspended:

```
using Aliyun.OSS;
var endpoint = "<yourEndpoint>";
var accessKeyId = "<yourAccessKeyId>";
var accessKeySecret = "<yourAccessKeySecret>";
var bucketName = "<yourBucketName>";
// Initialize an OSSClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
try
{
    // Set the versioning status of the bucket to Enabled.
    client.SetBucketVersioning(new SetBucketVersioningRequest(bucketName, VersioningStatus.Enabled));
    Console.WriteLine("Create bucket Version succeeded");
}
catch (Exception ex)
{
    Console.WriteLine("Create bucket Version failed. {0}", ex.Message);
}
```

For more information about how to configure versioning for a bucket, see [PutBucketVersioning](#).

Query the versioning status of a bucket

The following code provides an example on how to query information about the versioning state of a bucket:

```
using Aliyun.OSS;
using Aliyun.OSS.Common;
var endpoint = "<yourEndpoint>";
var accessKeyId = "<yourAccessKeyId>";
var accessKeySecret = "<yourAccessKeySecret>";
var bucketName = "<yourBucketName>";
// Create an OSSClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
try
{
    // Query the versioning status of the bucket.
    var result = client.GetBucketVersioning(bucketName);
    Console.WriteLine("Get bucket:{0} Version succeeded ", bucketName);
    Console.WriteLine("bucket version status: {0}", result.Status);
}
catch (OssException ex)
{
    Console.WriteLine("Failed with error code: {0}; Error info: {1}. \nRequestId:{2}\tHostID:{3}",
        ex.ErrorCode, ex.Message, ex.RequestId, ex.HostId);
}
catch (Exception ex)
{
    Console.WriteLine("Failed with error info: {0}", ex.Message);
}
```

For more information about how to obtain the versioning state of a bucket, see [GetBucketVersioning](#).

List the version information of all objects in a bucket

The following code provides an example on how to list the version information of all objects in a specified bucket, including delete markers:

```
using Aliyun.OSS;
using Aliyun.OSS.Common;
var endpoint = "<yourEndpoint>";
var accessKeyId = "<yourAccessKeyId>";
var accessKeySecret = "<yourAccessKeySecret>";
var bucketName = "<yourBucketName>";
// Create an OSSClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
try
{
    ObjectVersionList result = null;
    var request = new ListObjectVersionsRequest(bucketName);
    do {
        result = client.ListObjectVersions(request);
        Console.WriteLine("ListObjectVersions succeeded");
        // View the version information of all listed objects.
        foreach (var objectversion in result.ObjectVersionSummaries)
        {
            Console.WriteLine("objectversion key: {0}; objectversion versionid: {1}", objectversion.Key, objectversion.VersionId);
        }
        // View the version information of all listed delete markers.
        foreach (var deletemarker in result.DeleteMarkerSummaries)
        {
            Console.WriteLine("deletemarker key: {0}; deletemarker versionid: {1}", deletemarker.Key, deletemarker.VersionId);
        }
        request.KeyMarker = result.NextKeyMarker;
        request.NextVersionIdMarker = result.NextVersionIdMarker;
    } while (result.IsTruncated)
}
catch (OssException ex)
{
    Console.WriteLine("Failed with error code: {0}; Error info: {1}. \nRequestId:{2}\tHostId:{3}",
        ex.ErrorCode, ex.Message, ex.RequestId, ex.HostId);
}
catch (Exception ex)
{
    Console.WriteLine("Failed with error info: {0}", ex.Message);
}
```

7.9.2. Upload objects

This topic describes how to upload objects to a versioning-enabled bucket.

Simple upload

When you upload an object to a versioning-enabled bucket, OSS automatically adds a unique version ID for the uploaded object, and returns the version ID as the x-oss-version-id field in the response header. In a versioning-suspended bucket, the version ID added to an upload object is null. If you upload an object with the same name as an existing object to a versioning-suspended bucket, the existing object is overwritten. In this case, each object only has one version of which the ID is null.

The following code provides an example for simple upload:

```
using System.Text;
using Aliyun.OSS;
var endpoint = "<yourEndpoint>";
var accessKeyId = "<yourAccessKeyId>";
var accessKeySecret = "<yourAccessKeySecret>";
var bucketName = "<yourBucketName>";
var objectName = "<yourObjectName>";
var objectContent = "More than just cloud." ;
// Create an OSSClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
try
{
    byte[] binaryData = Encoding.ASCII.GetBytes(objectContent);
    MemoryStream requestContent = new MemoryStream(binaryData);
    // Upload the object.
    var result = client.PutObject(bucketName, objectName, requestContent);
    Console.WriteLine("Put object succeeded versionid : {0}", result.VersionId);
}
catch (Exception ex)
{
    Console.WriteLine("Put object failed, {0}", ex.Message);
}
```

For more information about simple upload, see [PutObject](#).

Append upload

In a versioning-enabled bucket, the AppendObject operation can be performed only on objects of which the current version is an appendable object.

Note

- When you perform the AppendObject operation on an object of which the current version is an appendable object, OSS does not generate a previous version for the appendable object.
- When you perform the PutObject or DeleteObject operation on an object of which the current version is an appendable object, OSS stores the appendable object as a previous version and prevents the object from being further appended.
- You cannot perform the AppendObject operation on objects of which the current version is not an appendable object such as a normal object or a delete marker.

The following code provides an example for append upload:

```
using Aliyun.OSS;
var endpoint = "<yourEndpoint>";
```

```

var accessKeyId = "<yourAccessKeyId>";
var accessKeySecret = "<yourAccessKeySecret>";
var bucketName = "<yourBucketName>";
var objectName = "<yourObjectName>";
var localFilename = "<yourLocalFilename>";
// Create an OSSClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
// If this is the first time an object is appended, the append position is 0. The returned
value is the position for the next append. The position to start the next append is the tot
al length of appended bytes and the append object.
long position = 0;
try
{
    var metadata = client.GetObjectMetadata(bucketName, objectName);
    position = metadata.ContentLength;
}
catch (Exception) { }
try
{
    using (var fs = File.Open(localFilename, FileMode.Open))
    {
        var request = new AppendObjectRequest(bucketName, objectName)
        {
            ObjectMetadata = new ObjectMetadata(),
            Content = fs,
            Position = position
        };
        // Append the object.
        var result = client.AppendObject(request);
        // Configure the position where the object is appended based on the previous object
length.
        position = result.NextAppendPosition;
        Console.WriteLine("Append object succeeded, next append position:{0}, vesionid:{1}"
, position, result.VersionId);
    }
    // Query the position for the next append and start the next append.
    using (var fs = File.Open(localFilename, FileMode.Open))
    {
        var request = new AppendObjectRequest(bucketName, objectName)
        {
            ObjectMetadata = new ObjectMetadata(),
            Content = fs,
            Position = position
        };
        var result = client.AppendObject(request);
        position = result.NextAppendPosition;
        Console.WriteLine("Append object succeeded, next append position:{0}, vesionid:{1}"
, position, result.VersionId);
    }
}
catch (Exception ex)
{
    Console.WriteLine("Append object failed, {0}", ex.Message);
}

```

For more information about append upload, see [AppendObject](#).

Multipart upload

If you call `CompleteMultipartUpload` to complete the Multipart Upload task for an object in a versioning-enabled bucket, OSS generates a unique version ID for the object and returns it as the `x-oss-version-id` field in the response header.

The following code provides an example for multipart upload:

```
using Aliyun.OSS;
var endpoint = "<yourEndpoint>";
var accessKeyId = "<yourAccessKeyId>";
var accessKeySecret = "<yourAccessKeySecret>";
var bucketName = "<yourBucketName>";
var objectName = "<yourObjectName>";
var localFilename = "<yourLocalFilename>";
// Create an OSSClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
// Initialize a multipart upload task.
var uploadId = "";
try
{
    // Specify the name of the object to upload and the bucket to which the object belongs.
    You can configure object metadata when you use InitiateMultipartUploadRequest without specifying the ContentLength parameter.
    var request = new InitiateMultipartUploadRequest(bucketName, objectName);
    var result = client.InitiateMultipartUpload(request);
    uploadId = result.UploadId;
    // Display the upload ID.
    Console.WriteLine("Init multi part upload succeeded");
    Console.WriteLine("Upload Id:{0}", result.UploadId);
}
catch (Exception ex)
{
    Console.WriteLine("Init multi part upload failed, {0}", ex.Message);
}
// Calculate the total number of parts.
var partSize = 100 * 1024;
var fi = new FileInfo(localFilename);
var fileSize = fi.Length;
var partCount = fileSize / partSize;
if (fileSize % partSize != 0)
{
    partCount++;
}
// Start the multipart upload task. partETags is a list of part ETags. OSS verifies the validity of each part after it receives the part list. After all parts are verified, OSS combines these parts into a complete object.
var partETags = new List<PartETag>();
try
{
    using (var fs = File.Open(localFilename, FileMode.Open))
    {
        for (var i = 0; i < partCount; i++)
```



```

        {
            var skipBytes = (long)partSize * i;
            // Find the start position of the upload.
            fs.Seek(skipBytes, 0);
            // Calculate the size of the part to upload. The size of the last part is the s
            ize of the remaining data.
            var size = (partSize < fileSize - skipBytes) ? partSize : (fileSize - skipBytes
        );

            var request = new UploadPartRequest(bucketName, objectName, uploadId)
            {
                InputStream = fs,
                PartSize = size,
                PartNumber = i + 1
            };
            // Call the UploadPart operation to perform part upload. The returned results c
            ontain the value of the part ETag.
            var result = client.UploadPart(request);
            partETags.Add(result.PartETag);
            Console.WriteLine("finish {0}/{1}", partETags.Count, partCount);
        }
        Console.WriteLine("Put multi part upload succeeded");
    }
}
catch (Exception ex)
{
    Console.WriteLine("Put multi part upload failed, {0}", ex.Message);
}
// Complete the multipart upload task.
try
{
    var completeMultipartUploadRequest = new CompleteMultipartUploadRequest(bucketName, obj
    ectName, uploadId);
    foreach (var partETag in partETags)
    {
        completeMultipartUploadRequest.PartETags.Add(partETag);
    }
    var result = client.CompleteMultipartUpload(completeMultipartUploadRequest);
    Console.WriteLine("CompleteMultipartUpload succeeded, vesionid:{0}", result.VersionId);
}
catch (Exception ex)
{
    Console.WriteLine("complete multi part failed, {0}", ex.Message);
}
}

```

For more information about multipart upload, see [CompleteMultipartUpload](#).

7.9.3. Download objects

By default, calling the GetObject operation on an object in a versioning-enabled bucket returns only the current version of the object.

When you perform the GetObject operation on an object in a bucket:

- If the current version of the object is a delete marker, the 4040 Not Found error is returned.

- If you specify the versionId of the target object in the request, the specified version of the object is returned. For example, if the versionId is specified as null in the request, the version of which the versionId is null is returned.
- If the version specified in the request is a delete marker, the 405 Method Not Allowed error is returned.

The following code provides an example on how to download an object:

```
using Aliyun.OSS;
var endpoint = "<yourEndpoint>";
var accessKeyId = "<yourAccessKeyId>";
var accessKeySecret = "<yourAccessKeySecret>";
var bucketName = "<yourBucketName>";
// You must set objectName to the name of the object to download. Example: abc/efg/123.jpg.
var objectName = "<yourObjectName>";
var downloadFilename = "<yourDownloadFilename>";
var versionid = "<yourArchiveObjectVersionid>";
// Create an OSSClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
try
{
    // Download an object to a stream. OssObject includes object information such as the bucket where the object is stored, object name, object metadata, and an input stream.
    var request = new GetObjectRequest(bucketName, objectName)
    {
        // Specify the version ID of the object.
        VersionId = versionid
    };
    var obj = client.GetObject(request);
    using (var requestStream = obj.Content)
    {
        byte[] buf = new byte[1024];
        var fs = File.Open(downloadFilename, FileMode.OpenOrCreate);
        var len = 0;
        // The input stream can be used to read and download the object content to a local file or to the buffer.
        while ((len = requestStream.Read(buf, 0, 1024)) != 0)
        {
            fs.Write(buf, 0, len);
        }
        fs.Close();
    }
    Console.WriteLine("Get object succeeded, vesionid:{0}", obj.VersionId);
}
catch (Exception ex)
{
    Console.WriteLine("Get object failed. {0}", ex.Message);
}
```

7.9.4. Copy objects

This topic describes how to copy objects in a versioned bucket. You can use `CopyObject` to copy an object that is up to 1 GB in size and use `UploadPartCopy` to copy an object larger than 1 GB.

Copy a small object

You can use `CopyObject` to copy an object that is up to 1 GB in size from a source bucket to a destination bucket within the same region.

- By default, the `x-oss-copy-source` header in a `CopyObject` request specifies the current version of the object to copy. If the current version of the object to copy is a delete marker, 404 Not Found is returned to indicate that the object does not exist. You can add a version ID in the `x-oss-copy-source` header to copy a specified object version. Delete markers cannot be copied.
- You can copy a previous version of an object to the same bucket. The copied previous version becomes the current version of the object. This way, you can restore an object to a specified previous version.
- If versioning is enabled for the destination bucket, OSS generates a unique version ID for the destination object. The version ID is returned in the response as the `x-oss-version-id` header value. If versioning is disabled or suspended for the destination bucket, OSS generates a version whose version ID is null for the destination object and overwrites the original version whose version ID is null.
- Append objects in a bucket with versioning enabled or suspended cannot be copied.

The following code provides an example on how to copy a small object:

```

using Aliyun.OSS;
using Aliyun.OSS.Common;
var endpoint = "<yourEndpoint>";
var accessKeyId = "<yourAccessKeyId>";
var accessKeySecret = "<yourAccessKeySecret>";
var sourceBucket = "<yourSourceBucketName>";
var sourceObject = "<yourSourceObjectName>";
var targetBucket = "<yourDestBucketName>";
var targetObject = "<yourDestObjectName>";
var versionid = "<yourArchiveObjectVersionid>";
// Create an OSSClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
try
{
    var metadata = new ObjectMetadata();
    metadata.AddHeader("mk1", "mv1");
    metadata.AddHeader("mk2", "mv2");
    var req = new CopyObjectRequest(sourceBucket, sourceObject, targetBucket, targetObject)
    {
        // If the value of NewObjectMetadata is null, the COPY mode is used and the metadata
        // of the source object is copied to the destination object. If the value of NewObjectMetadata
        // is not null, the REPLACE mode is used and the metadata of the source object overwrites that
        // of the destination object.
        NewObjectMetadata = metadata,
        // Specify the version ID of the source object.
        SourceVersionId = versionid
    };
    // Copy the object.
    var result = client.CopyObject(req);
    Console.WriteLine("Copy object succeeded, versionid:{0}", result.VersionId);
}
catch (OssException ex)
{
    Console.WriteLine("Failed with error code: {0}; Error info: {1}. \nRequestId: {2} \tHostID: {3}",
        ex.ErrorCode, ex.Message, ex.RequestId, ex.HostId);
}
catch (Exception ex)
{
    Console.WriteLine("Failed with error info: {0}", ex.Message);
}

```

Copy a large object

To copy an object that is larger than 1 GB in size, you must split the object into parts and copy the parts by using UploadPartCopy.

By default, the UploadPartCopy operation uploads a part by copying data from the current version of an existing object. You can add versionId to the x-oss-copy-source request header to copy the specified version of the object. Example: x-oss-copy-source : /SourceBucketName/SourceObjectName?versionId=CAEQMxiBgICAof2D0BYiIDJhMGE3N2M1YTl1NDQzOGY5NTkyNTI3MGYyMzJm****.

 **Note** The value of SourceObjectName must be URL-encoded. The version ID of the copied object is returned in the response as the value of the x-oss-copy-source-version-id header.

If versionId is not specified in the request and the current version of the source object is a delete marker, OSS returns 404 Not Found. If versionId is specified in the request and the specified version of the source object is a delete marker, OSS returns 400 Bad Request.

The following code provides an example on how to use multipart copy to copy large objects:

```
using Aliyun.OSS;
var endpoint = "<yourEndpoint>";
var accessKeyId = "<yourAccessKeyId>";
var accessKeySecret = "<yourAccessKeySecret>";
var sourceBucket = "<yourSourceBucketName>";
var sourceObject = "<yourSourceObjectName>";
var targetBucket = "<yourDestBucketName>";
var targetObject = "<yourDestObjectName>";
var sourceObjectVersionid = "<yourSourceObjectVersionid>";
var uploadId = "";
var partSize = 50 * 1024 * 1024;
// Create an OSSClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
try
{
    // Initiate a multipart copy task. You can call the InitiateMultipartUploadRequest operation to specify the metadata of the destination object.
    var request = new InitiateMultipartUploadRequest(targetBucket, targetObject);
    var result = client.InitiateMultipartUpload(request);
    // Display the value of UploadId.
    uploadId = result.UploadId;
    Console.WriteLine("Init multipart upload succeeded, Upload Id: {0}", result.UploadId);
    // Calculate the total number of parts.
    var request = new GetObjectMetadataRequest(sourceBucket, sourceObject)
    {
        // Specify the version ID of the source object.
        VersionId = sourceObjectVersionid
    };
    var metadata = client.GetObjectMetadata(request);
    var fileSize = metadata.ContentLength;
    var partCount = (int)fileSize / partSize;
    if (fileSize % partSize != 0)
    {
        partCount++;
    }
    // Start the multipart copy task.
    var partETags = new List<PartETag>();
    for (var i = 0; i < partCount; i++)
    {
        var skipBytes = (long)partSize * i;
        var size = (partSize < fileSize - skipBytes) ? partSize : (fileSize - skipBytes);
        // Create an UploadPartCopyRequest request. You can call the UploadPartCopyRequest operation to specify conditions.
        var uploadPartCopyRequest = new UploadPartCopyRequest(targetBucket, targetObject, sourceBucket, sourceObject, uploadId)
```

```

        {
            PartSize = size,
            PartNumber = i + 1,
            // Use BeginIndex to find the start position to copy the part.
            BeginIndex = skipBytes,
            // Specify the version ID of the source object.
            VersionId = sourceObjectVersionId
        };
        // Call the uploadPartCopy operation to copy each part.
        var uploadPartCopyResult = client.UploadPartCopy(uploadPartCopyRequest);
        Console.WriteLine("UploadPartCopy : {0}", i);
        partETags.Add(uploadPartCopyResult.PartETag);
    }
    // Complete the multipart copy task.
    var completeMultipartUploadRequest =
        new CompleteMultipartUploadRequest(targetBucket, targetObject, uploadId);
    // partETags is a list of partETag. OSS verifies the validity of each part after it receives partETags. After all parts are verified, OSS combines these parts into a complete object.
    foreach (var partETag in partETags)
    {
        completeMultipartUploadRequest.PartETags.Add(partETag);
    }
    var completeMultipartUploadResult = client.CompleteMultipartUpload(completeMultipartUploadRequest);
    Console.WriteLine("CompleteMultipartUpload succeeded, versionid:{0}", completeMultipartUploadResult.VersionId);
}
catch (OssException ex)
{
    Console.WriteLine("Failed with error code: {0}; Error info: {1}. \nRequestId: {2} \tHostID: {3}",
        ex.ErrorCode, ex.Message, ex.RequestId, ex.HostId);
}
catch (Exception ex)
{
    Console.WriteLine("Failed with error info: {0}", ex.Message);
}

```

For more information about multipart copy, see [UploadPartCopy](#).

7.9.5. Delete objects

This topic describes how to delete a single object, multiple objects, or objects whose names contain a specified prefix from a versioned bucket.

Delete operations in a versioned bucket

When you delete an object from a versioned bucket, you must determine whether to specify a version ID in the request.

- Delete an object without specifying a version ID (temporary deletion)

By default, if you do not specify the version ID of the object that you want to delete in the request, Object Storage Service (OSS) does not delete the current version of the object but adds a delete marker to the object as the new current version. If you perform the `GetObject` operation on the object, OSS identifies the current version of the object as a delete marker and returns `404 Not Found`. Additionally, the `x-oss-delete-marker` header and the `x-oss-version-id` header that indicates the version ID of the created delete marker are included in the response to the `DeleteObject` request.

The value of the `x-oss-delete-marker` header in the response is true, which indicates that the `x-oss-version-id` value is the version ID of a delete marker.

- Delete an object by specifying a version ID (permanent deletion)

If you specify the version ID of the object that you want to delete in the request, OSS permanently deletes the version specified by the `versionId` field in the `params` parameter. To delete a version whose ID is null, add `params['versionId'] = "null"` to `params` in the request. OSS identifies the string "null" as the ID of the version to delete and deletes the version whose ID is null.

Delete a single object

The following examples describe how to permanently or temporarily delete a single object from a versioned bucket.

- Permanently delete an object from a versioned bucket

The following code provides an example on how to permanently delete an object with a specified version ID from a versioned bucket:

```
using Aliyun.OSS;
var endpoint = "<yourEndpoint>";
var accessKeyId = "<yourAccessKeyId>";
var accessKeySecret = "<yourAccessKeySecret>";
var bucketName = "<yourBucketName>";
var objectName = "<yourObjectName>";
var versionid = "<yourObjectVersionidOrDelMarkerVersionid>";
// Create an OSSClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
try
{
    // Specify the version ID of the object or delete marker you want to delete.
    var request = new DeleteObjectRequest(bucketName, key)
    {
        VersionId = versionid
    };
    client.DeleteObject(request);
    Console.WriteLine("Delete object succeeded");
}
catch (Exception ex)
{
    Console.WriteLine("Delete object failed. {0}", ex.Message);
}
```

- Temporarily delete an object from a versioned bucket

The following code provides an example on how to temporarily delete an object from a versioned bucket:

```
using Aliyun.OSS;
var endpoint = "<yourEndpoint>";
var accessKeyId = "<yourAccessKeyId>";
var accessKeySecret = "<yourAccessKeySecret>";
var bucketName = "<yourBucketName>";
var objectName = "<yourObjectName>";
// Create an OSSClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
try
{
    // Temporarily delete an object without specifying its version ID. A delete marker is
    // added to the object.
    var result = client.DeleteObject(bucketName, objectName);
    Console.WriteLine("Delete object succeeded, versionid: {0}, DeleteMarker: {1}", result.VersionId, result.DeleteMarker);
}
catch (Exception ex)
{
    Console.WriteLine("Delete object failed. {0}", ex.Message);
}
```

For more information about how to delete a single object, see [DeleteObject](#).

Delete multiple objects

The following examples describe how to permanently or temporarily delete multiple objects from a versioned bucket.

- Permanently delete multiple objects from a versioned bucket

The following code provides an example on how to permanently delete multiple objects or delete markers with specified version IDs from a versioned bucket:


```

using Aliyun.OSS;
using Aliyun.OSS.Common;
var endpoint = "<yourEndpoint>";
var accessKeyId = "<yourAccessKeyId>";
var accessKeySecret = "<yourAccessKeySecret>";
var bucketName = "<yourBucketName>";
// Create an OSSClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
try
{
    // Delete the objects with the specified version IDs or the objects whose current ver
    sions are delete markers with the specified version IDs.
    var obj1 = new ObjectIdentifier
    {
        Key = "yourObject1Name",
        VersionId = "yourObject1NameVersionid"
    };
    var obj2 = new ObjectIdentifier
    {
        Key = "yourObject2Name",
        VersionId = "yourObject2DelMarkerVersionid"
    };
    IList<ObjectIdentifier> objects = new List<ObjectIdentifier>();
    objects.Add(obj1);
    objects.Add(obj2);
    var request = new DeleteObjectVersionsRequest(bucketName, objects);
    // Initiate a deleteVersions request.
    client.DeleteObjectVersions(request);
    Console.WriteLine("DeleteObjectVersions succeeded ");
}
catch (OssException ex)
{
    Console.WriteLine("Failed with error code: {0}; Error info: {1}. \nRequestID:{2}\tHos
tID:{3}",
        ex.ErrorCode, ex.Message, ex.RequestId, ex.HostId);
}
catch (Exception ex)
{
    Console.WriteLine("Failed with error info: {0}", ex.Message);
}

```

For more information about how to delete multiple objects, see [DeleteMultipleObjects](#).

- Temporarily delete multiple objects from a versioned bucket

The following code provides an example on how to temporarily delete multiple objects from a versioned bucket:

```
using Aliyun.OSS;
var endpoint = "<yourEndpoint>";
var accessKeyId = "<yourAccessKeyId>";
var accessKeySecret = "<yourAccessKeySecret>";
var bucketName = "<yourBucketName>";
// Create an OSSClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
try
{
    var keys = new List<string>();
    var listResult = client.ListObjects(bucketName);
    foreach (var summary in listResult.ObjectSummaries)
    {
        keys.Add(summary.Key);
    }
    // Specify the value of quietMode. True indicates quiet. In this mode, a list of the
    objects that failed to be deleted is returned. False indicates verbose. In this mode, a l
    ist of the deleted objects is returned. Default value: false.
    var quietMode = false;
    // Specify the return mode by setting the value of quietMode in the DeleteObjectsRequ
    est request.
    var request = new DeleteObjectsRequest(bucketName, keys, quietMode);
    // Delete the objects without specifying their version IDs. Delete markers are added
    to the objects.
    var result = client.DeleteObjects(request);
    if ((! quietMode) && (result.Keys != null))
    {
        foreach (var obj in result.Keys)
        {
            Console.WriteLine("Delete successfully : {0} ", obj.Key);
        }
    }
    Console.WriteLine("Delete objects succeeded");
}
catch (Exception ex)
{
    Console.WriteLine("Delete objects failed. {0}", ex.Message);
}
```

Delete objects whose names contain a specified prefix

The following code provides an example on how to delete objects whose names contain a specified prefix:

```
using Aliyun.OSS;
using Aliyun.OSS.Common;
var endpoint = "<yourEndpoint>";
var accessKeyId = "<yourAccessKeyId>";
var accessKeySecret = "<yourAccessKeySecret>";
var bucketName = "<yourBucketName>";
var prefix = "<yourkeyPrefix>";
// Create an OSSClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
try
{
    ObjectVersionList result = null;
    var request = new ListObjectVersionsRequest(bucketName)
    {
        // Specify a prefix.
        Prefix = prefix;
    };
    // List the versions of all objects whose names contain the specified prefix and delete
    the versions.
    do {
        result = client.ListObjectVersions(request);
        Console.WriteLine("ListObjectVersions succeeded");
        foreach (var deleteversion in result.DeleteMarkerSummaries)
        {
            var request = new DeleteObjectRequest(bucketName, deleteversion.Key)
            {
                VersionId = deleteversion.VersionId
            };
            client.DeleteObject(request);
        }
        foreach (var objectversion in result.ObjectVersionSummaries)
        {
            var request = new DeleteObjectRequest(bucketName, objectversion.Key)
            {
                VersionId = objectversion.VersionId
            };
            client.DeleteObject(request);
        }
        request.KeyMarker = result.NextKeyMarker;
        request.NextVersionIdMarker = result.NextVersionIdMarker ;
    } while (result.IsTruncated)
}
catch (OssException ex)
{
    Console.WriteLine("Failed with error code: {0}; Error info: {1}. \nRequestId:{2}\tHostID:{3}",
        ex.ErrorCode, ex.Message, ex.RequestId, ex.HostId);
}
catch (Exception ex)
{
    Console.WriteLine("Failed with error info: {0}", ex.Message);
}
```

7.9.6. Restore objects

In a versioning-enabled bucket, different versions of the objects may have different storage classes. By default, the `RestoreObject` operation restores the current version of an object. To restore a specific version of the object, specify the version ID.

The following code provides an example on how to restore an object:

```
using Aliyun.OSS;
using Aliyun.OSS.Model;
var endpoint = "<yourEndpoint>";
var accessKeyId = "<yourAccessKeyId>";
var accessKeySecret = "<yourAccessKeySecret>";
var bucketName = "<yourBucketName>";
var objectName = "<yourObjectName>";
var objectContent = "More than just cloud.";
var versionid = "<yourArchiveObjectVersionid>";
int maxWaitTimeInSeconds = 600;
// Create an OSSClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
var getrequest = new GetObjectMetadataRequest(bucketName, objectName)
{
    // Specify the ID of the version that is of the Archive storage class.
    VersionId = versionid
};
var metadata = client.GetObjectMetadata(getrequest);
string storageClass = metadata.HttpMetadata["x-oss-storage-class"] as string;
if (storageClass != "Archive")
{
    Console.WriteLine("StorageClass is {0}", storageClass);
    return;
}
// Restore the object.
var request = new RestoreObjectRequest(bucketName, objectName)
{
    // Specify the ID of the version that is of the Archive storage class.
    VersionId = versionid
};
RestoreObjectResult result = client.RestoreObject(request);
Console.WriteLine("RestoreObject result HttpStatusCode : {0}, versionid: {1}", result.HttpS
tatusCode, result.VersionId);
if (result.HttpStatusCode != HttpStatusCode.Accepted)
{
    throw new OssException(result.RequestId + ", " + result.HttpStatusCode + " ,");
}
while (maxWaitTimeInSeconds > 0)
{
    var req = new GetObjectMetadataRequest(bucketName, objectName)
    {
        // Specify the ID of the version that is of the Archive storage class.
        VersionId = versionid
    };
    var meta = client.GetObjectMetadata(req);
    string restoreStatus = meta.HttpMetadata["x-oss-restore"] as string;
```

```
        if (restoreStatus != null && restoreStatus.StartsWith("ongoing-request=\"false\"", StringComparison.InvariantCultureIgnoreCase))
        {
            break;
        }
        Thread.Sleep(1000);
        maxWaitTimeInSeconds--;
    }
    if (maxWaitTimeInSeconds == 0)
    {
        Console.WriteLine("RestoreObject is timeout. ");
        throw new TimeoutException();
    }
    else
    {
        Console.WriteLine("RestoreObject is successful. ");
    }
}
```

7.9.7. Query object metadata

By default, calling the `HeadObject` operation on an object in a versioning-enabled bucket returns only metadata of the current version of the object.

 **Note** If the current version of the object is a delete marker, OSS returns 404 Not Found. If you specify the version ID in your request, OSS returns metadata of the object with the specified version ID.

The following code provides an example on how to obtain object metadata:

```
using Aliyun.OSS;
using Aliyun.OSS.Common;
var endpoint = "<yourEndpoint>";
var accessKeyId = "<yourAccessKeyId>";
var accessKeySecret = "<yourAccessKeySecret>";
var bucketName = "<yourBucketName>";
// Specify the complete name of the object excluding the bucket name. Example: example/test.jpg.
var objectName = "<yourObjectName>";
var versionid = "<yourArchiveObjectVersionid>";
// Create an OSSClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
try
{
    // Query the object metadata.
    var request = new GetObjectMetadataRequest(bucketName, objectName)
    {
        // Query the metadata of the specified version of the object.
        VersionId = versionid
    };
    // Query all metadata of the specified version of the object.
    var oldMeta = client.GetObjectMetadata(request);
    // Query partial metadata of the specified version of the object.
    var meta = client.GetSimplifiedObjectMetadata(request);
}
catch (Exception ex)
{
    Console.WriteLine("Put object failed, {0}", ex.Message);
}
```

For more information about how to obtain the metadata of an object, see [GetObjectMeta](#) and [HeadObject](#).

7.9.8. Manage the ACL of an object

This topic describes how to manage ACLs of objects in a versioning-enabled bucket.

Configure ACL for an object

The following code provides an example on how to configure the ACL of an object:

```
using Aliyun.OSS;
using Aliyun.OSS.Common;
var endpoint = "<yourEndpoint>";
var accessKeyId = "<yourAccessKeyId>";
var accessKeySecret = "<yourAccessKeySecret>";
var bucketName = "<yourBucketName>";
var objectName = "<yourObjectName>";
var versionid = "<yourArchiveObjectVersionid>";
// Create an OSSClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
// Configure the ACL of the object.
try
{
    // Use SetObjectAcl to configure the ACL of the object.
    var request = new SetObjectAclRequest(bucketName, objectName, CannedAccessControlList.PublicRead)
    {
        // Configure the ACL of the specified version of the object.
        VersionId = versionid
    };
    client.SetObjectAcl(request);
    Console.WriteLine("Set Object:{0} ACL succeeded ", objectName);
}
catch (Exception ex)
{
    Console.WriteLine("Set Object ACL failed with error info: {0}", ex.Message);
}
```

Obtain the ACL of an object

The following code provides an example on how to obtain the ACL of an object:

```
using Aliyun.OSS;
using Aliyun.OSS.Common;
var endpoint = "<yourEndpoint>";
var accessKeyId = "<yourAccessKeyId>";
var accessKeySecret = "<yourAccessKeySecret>";
var bucketName = "<yourBucketName>";
var objectName = "<yourObjectName>";
var versionid = "<yourArchiveObjectVersionid>";
// Create an OSSClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
// Obtain the ACL of the object.
try
{
    // Use GetObjectAcl to obtain the ACL of the object.
    var request = new GetObjectAclRequest(bucketName, objectName)
    {
        // Query the ACL of the specified version of the object.
        VersionId = versionid
    };
    var result = client.GetObjectAcl(request);
    Console.WriteLine("Get Object ACL succeeded, Id: {0}  ACL: {1}",
        result.Owner.Id, result.ACL.ToString());
}
catch (OssException ex)
{
    Console.WriteLine("Failed with error code: {0}; Error info: {1}. \nRequestId: {2}\tHost
ID: {3}",
        ex.ErrorCode, ex.Message, ex.RequestId, ex.HostId);
}
catch (Exception ex)
{
    Console.WriteLine("Failed with error info: {0}", ex.Message);
}
```

7.9.9. Manage symbolic links

Create a symbolic link

The following code provides an example on how to create a symbolic link:


```
using Aliyun.OSS;
var endpoint = "<yourEndpoint>";
var accessKeyId = "<yourAccessKeyId>";
var accessKeySecret = "<yourAccessKeySecret>";
var bucketName = "<yourBucketName>";
var targetObjectName = "<yourTargetObjectName>";
var symlinkObjectName = "<yourSymlinkObjectName>";
var objectContent = "More than just cloud." ;
// Create an OSSClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
try
{
    // Upload the target object.
    byte[] binaryData = Encoding.ASCII.GetBytes(objectContent);
    MemoryStream requestContent = new MemoryStream(binaryData);
    client.PutObject(bucketName, targetObjectName, requestContent);
    // Create the symbolic link.
    client.CreateSymlink(bucketName, symlinkObjectName, targetObjectName);
}
catch (Exception ex)
{
    Console.WriteLine("Failed with error info: {0}", ex.Message);
}
```

Obtain symbolic links

The following code provides an example on how to obtain a symbolic link:

```
using Aliyun.OSS;
var endpoint = "<yourEndpoint>";
var accessKeyId = "<yourAccessKeyId>";
var accessKeySecret = "<yourAccessKeySecret>";
var bucketName = "<yourBucketName>";
var symlinkObjectName = "<yourSymlinkObjectName>";
var versionid = "<yourArchiveObjectVersionid>";
// Create an OSSClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
try
{
    // Obtain the name of the object to which the symbolic link points.
    var request = new GetSymlinkRequest(bucketName, symlinkObjectName)
    {
        // Obtain the specified version of a symbolic link.
        VersionId = versionid
    };
    var ossSymlink = client.GetSymlink(request);
    Console.WriteLine("Target object is {0}", ossSymlink.Target);
}
catch (Exception ex)
{
    Console.WriteLine("Failed with error info: {0}", ex.Message);
}
```

7.10. Object tagging

7.10.1. Configure object tagging

Object Storage Service (OSS) allows you to configure object tags to classify objects. You can configure lifecycle rules and control access to objects based on tags.

 **Note** For more information about object tags, see [Object tagging](#).

Add tags to an object when you upload the object

The following examples describe how to add tags to objects in simple upload, multipart upload, append upload, and resumable upload:

- Add tags to an object when you upload the object by using simple upload

The following code provides an example on how to add tags to an object when you upload the object by using simple upload:

```
using System.Text;
using Aliyun.OSS;
using System.Text;
using Aliyun.OSS.Util;
// Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com.
var endpoint = "yourEndpoint";
// The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to access OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
var accessKeyId = "yourAccessKeyId";
var accessKeySecret = "yourAccessKeySecret";
// Specify the bucket name.
var bucketName = "examplebucket";
// Specify the full path of the object. The full path of the object cannot contain the bucket name.
var objectName = "exampleobject.txt";
var objectContent = "More than just cloud.";
String UriEncodeKey(String key)
{
    const string CharsetName = "utf-8";
    const char separator = '/';
    var segments = key.Split(separator);
    var encodedKey = new StringBuilder();
    encodedKey.Append(HttpUtils.EncodeUri(segments[0], CharsetName));
    for (var i = 1; i < segments.Length; i++)
        encodedKey.Append(separator).Append(HttpUtils.EncodeUri(segments[i], CharsetName));
    if (key.EndsWith(separator.ToString()))
    {
        // String#split ignores trailing empty strings, e.g., "a/b/" will be split as a 2-entries array,
        // so we have to append all the trailing slash to the uri.
    }
}
```

```

        foreach (var ch in key)
        {
            if (ch == separator)
                encodedKey.Append(separator);
            else
                break;
        }
    }
    return encodedKey.ToString();
}
// Create an OSSClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
try
{
    byte[] binaryData = Encoding.ASCII.GetBytes(objectContent);
    MemoryStream requestContent = new MemoryStream(binaryData);
    var meta = new ObjectMetadata();
    // Configure the tags in the HTTP header.
    string str = Uri.EscapeDataString("key1") + "=" + Uri.EscapeDataString("key1") + "&" + Uri.EscapeDataString("key2") + "=" + Uri.EscapeDataString("key2");
    meta.AddHeader("x-oss-tagging", str);
    var putRequest = new PutObjectRequest(bucketName, objectName, requestContent);
    putRequest.Metadata = meta;
    // Upload the object and add tags to the object.
    client.PutObject(putRequest);
    Console.WriteLine("Put object succeeded");
}
catch (Exception ex)
{
    Console.WriteLine("Put object failed, {0}", ex.Message);
}

```

- Add tags to an object when you upload the object by using multipart upload

The following code provides an example on how to add tags to an object when you upload it by using multipart upload:

```

using Aliyun.OSS;
using System.Text;
// Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com.
var endpoint = "yourEndpoint";
// The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to access OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
var accessKeyId = "yourAccessKeyId";
var accessKeySecret = "yourAccessKeySecret";
// Specify the bucket name.
var bucketName = "examplebucket";
// Specify the full path of the object. The full path of the object cannot contain the bucket name.
var objectName = "exampleobject.txt";
// Specify the full path of the local file. Example: D:\\localpath\\examplefile.txt. By d

```

```

default, if you do not specify the local file, the local file is uploaded from the path of
the project to which the sample program belongs.
var localFilename = "D:\\localpath\\examplefile.txt";
String UrlEncodeKey(String key)
{
    const string CharsetName = "utf-8";
    const char separator = '/';
    var segments = key.Split(separator);
    var encodedKey = new StringBuilder();
    encodedKey.Append(HttpUtils.EncodeUri(segments[0], CharsetName));
    for (var i = 1; i < segments.Length; i++)
        encodedKey.Append(separator).Append(HttpUtils.EncodeUri(segments[i], CharsetName));
    if (key.EndsWith(separator.ToString()))
    {
        // String#split ignores trailing empty strings, e.g., "a/b/" will be split as a 2
        // -entries array,
        // so we have to append all the trailing slash to the uri.
        foreach (var ch in key)
        {
            if (ch == separator)
                encodedKey.Append(separator);
            else
                break;
        }
    }
    return encodedKey.ToString();
}
// Create an OSSClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
// Initiate a multipart upload task.
var uploadId = "";
try
{
    // Specify the name of the object and the bucket to which the object is uploaded. You
    // can configure object metadata in InitiateMultipartUploadRequest, but you do not need to s
    // pecify ContentLength.
    var request = new InitiateMultipartUploadRequest(bucketName, objectName);
    var meta = new ObjectMetadata();
    // Configure the tags in the HTTP header.
    string str = UrlEncodeKey("key1") + "=" + UrlEncodeKey("key1") + "&" + UrlEncodeKey("
    key2") + "=" + UrlEncodeKey("key2");
    meta.AddHeader("x-oss-tagging", str);
    request.Metadata = meta;
    var result = client.InitiateMultipartUpload(request);
    uploadId = result.UploadId;
    // Display the upload ID.
    Console.WriteLine("Init multi part upload succeeded");
    Console.WriteLine("Upload Id:{0}", result.UploadId);
}
catch (Exception ex)
{
    Console.WriteLine("Init multi part upload failed, {0}", ex.Message);
}
// Calculate the total number of parts.

```

```

var partSize = 100 * 1024;
var fi = new FileInfo(localFilename);
var fileSize = fi.Length;
var partCount = fileSize / partSize;
if (fileSize % partSize != 0)
{
    partCount++;
}
// Start the multipart upload task. partETags is a list of part ETags. OSS verifies each
// part after it receives the list of parts. After all parts are verified, OSS combines the
// parts into a complete object.
var partETags = new List<PartETag>();
try
{
    using (var fs = File.Open(localFilename, FileMode.Open))
    {
        for (var i = 0; i < partCount; i++)
        {
            var skipBytes = (long)partSize * i;
            // Find the start position of the multipart upload task.
            fs.Seek(skipBytes, 0);
            // Calculate the part size in the current multipart upload task. The size of
            // the last part is the size of the remainder after the object is split by the calculated part
            // size.
            var size = (partSize < fileSize - skipBytes) ? partSize : (fileSize - skipBytes);

            var request = new UploadPartRequest(bucketName, objectName, uploadId)
            {
                InputStream = fs,
                PartSize = size,
                PartNumber = i + 1
            };
            // Call UploadPart to upload the parts. The returned results contain the ETag
            // values of the parts.
            var result = client.UploadPart(request);
            partETags.Add(result.PartETag);
            Console.WriteLine("finish {0}/{1}", partETags.Count, partCount);
        }
        Console.WriteLine("Put multi part upload succeeded");
    }
}
catch (Exception ex)
{
    Console.WriteLine("Put multi part upload failed, {0}", ex.Message);
}
// Complete the multipart upload task.
try
{
    var completeMultipartUploadRequest = new CompleteMultipartUploadRequest(bucketName, objectName, uploadId);
    foreach (var partETag in partETags)
    {
        completeMultipartUploadRequest.PartETags.Add(partETag);
    }
    var result = client.CompleteMultipartUpload(completeMultipartUploadRequest);
}

```

```

        var result = client.CompleteMultipartUpload(completeMultipartUploadRequest,
        Console.WriteLine("complete multi part succeeded");
    }
    catch (Exception ex)
    {
        Console.WriteLine("complete multi part failed, {0}", ex.Message);
    }
}

```

- Add tags to an object when you upload the object by using append upload

The following code provides an example on how to add tags to an object when you upload it by using append upload:

```

using Aliyun.OSS;
using System.Text;

// Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com.
var endpoint = "yourEndpoint";
// The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to access OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
var accessKeyId = "yourAccessKeyId";
var accessKeySecret = "yourAccessKeySecret";
// Specify the bucket name.
var bucketName = "examplebucket";
// Specify the full path of the object. The full path of the object cannot contain the bucket name.
var objectName = "exampleobject.txt";
// Specify the full path of the local file. Example: D:\\localpath\\examplefile.txt. By default, if you do not specify the local file, the local file is uploaded from the path of the project to which the sample program belongs.
var localFilename = "D:\\localpath\\examplefile.txt";
String UrlEncodeKey(String key)
{
    const string CharsetName = "utf-8";
    const char separator = '/';
    var segments = key.Split(separator);
    var encodedKey = new StringBuilder();
    encodedKey.Append(HttpUtils.EncodeUri(segments[0], CharsetName));
    for (var i = 1; i < segments.Length; i++)
        encodedKey.Append(separator).Append(HttpUtils.EncodeUri(segments[i], CharsetName));
    if (key.EndsWith(separator.ToString()))
    {
        // String#split ignores trailing empty strings, e.g., "a/b/" will be split as a 2-entries array,
        // so we have to append all the trailing slash to the uri.
        foreach (var ch in key)
        {
            if (ch == separator)
                encodedKey.Append(separator);
            else
                break;
        }
    }
}

```

```

    }
    return encodedKey.ToString();
}
// Create an OSSClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
// If the object is appended for the first time, the position from which the append operation starts is 0. The start position for the next append operation is included in the response. The position from which the next append upload starts is the current length of the object.
long position = 0;
try
{
    var metadata = client.GetObjectMetadata(bucketName, objectName);
    position = metadata.ContentLength;
}
catch (Exception) { }
try
{
    using (var fs = File.Open(localFilename, FileMode.Open))
    {
        var request = new AppendObjectRequest(bucketName, objectName)
        {
            ObjectMetadata = new ObjectMetadata(),
            Content = fs,
            Position = position
        };
        var meta = new ObjectMetadata();
        // Configure the tags in the HTTP header.
        string str = UriEncodeKey("key1") + "=" + UriEncodeKey("key1") + "&" + UriEncodeKey("key2") + "=" + UriEncodeKey("key2");
        meta.AddHeader("x-oss-tagging", str);
        request.Metadata = meta;
        // Perform the first append operation. Only the tags configured when the object is appended for the first time are added to the object.
        var result = client.AppendObject(request);
        // Specify the position from which the append operation starts.
        position = result.NextAppendPosition;
        Console.WriteLine("Append object succeeded, next append position:{0}", position);
    }
    // Obtain the position for the next append operation and perform the second append operation.
    using (var fs = File.Open(localFilename, FileMode.Open))
    {
        var request = new AppendObjectRequest(bucketName, objectName)
        {
            ObjectMetadata = new ObjectMetadata(),
            Content = fs,
            Position = position
        };
        var result = client.AppendObject(request);
        position = result.NextAppendPosition;
        Console.WriteLine("Append object succeeded, next append position:{0}", position);
    }
}
catch (Exception ex)

```

```
{
    Console.WriteLine("Append object failed, {0}", ex.Message);
}
```

- Add tags to an object when you upload the object by using resumable upload

The following code provides an example on how to add tags to an object when you upload it by using resumable upload:

```
using Aliyun.OSS;
using Aliyun.OSS.Common;
using System.Text;

// Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com.
var endpoint = "yourEndpoint";
// The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to access OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
var accessKeyId = "yourAccessKeyId";
var accessKeySecret = "yourAccessKeySecret";
// Specify the bucket name.
var bucketName = "examplebucket";
// Specify the full path of the object. The full path of the object cannot contain the bucket name.
var objectName = "exampleobject.txt";
// Specify the full path of the local file. Example: D:\\localpath\\examplefile.txt. By default, if you do not specify the local file, the local file is uploaded from the path of the project to which the sample program belongs.
var localFilename = "D:\\localpath\\examplefile.txt";
// Specify the checkpoint file that records the upload result of each part. The file stores the progress information generated in the upload process.
string checkpointDir = "yourCheckpointDir";
String UriEncodeKey(String key)
{
    const string CharsetName = "utf-8";
    const char separator = '/';
    var segments = key.Split(separator);
    var encodedKey = new StringBuilder();
    encodedKey.Append(HttpUtils.EncodeUri(segments[0], CharsetName));
    for (var i = 1; i < segments.Length; i++)
        encodedKey.Append(separator).Append(HttpUtils.EncodeUri(segments[i], CharsetName));
    if (key.EndsWith(separator.ToString()))
    {
        // String#split ignores trailing empty strings, e.g., "a/b/" will be split as a 2-entries array,
        // so we have to append all the trailing slash to the uri.
        foreach (var ch in key)
        {
            if (ch == separator)
                encodedKey.Append(separator);
            else
                break;
        }
    }
}
```



```

    }
    return encodedKey.ToString();
}
// Create an OSSClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
try
{
    // Configure parameters by using UploadFileRequest.
    UploadObjectRequest request = new UploadObjectRequest(bucketName, objectName, localFilename)
    {
        // Specify the size of each part to upload.
        PartSize = 8 * 1024 * 1024,
        // Specify the number of concurrent threads.
        ParallelThreadCount = 3,
        // The checkpoint file is used to record the result of multipart upload. This file stores information about upload progress. If a part fails to be uploaded, the task can be continued based on the progress recorded in the checkpoint file. If you set checkpointDir to null, resumable upload does not take effect and objects are uploaded again when they fail to be uploaded.
        CheckpointDir = checkpointDir,
    };
    var meta = new ObjectMetadata();
    // Configure the tags in the HTTP header.
    string str = Uri.EscapeDataString("key1") + "=" + Uri.EscapeDataString("key1") + "&" + Uri.EscapeDataString("key2") + "=" + Uri.EscapeDataString("key2");
    meta.AddHeader("x-oss-tagging", str);
    request.Metadata = meta;
    // Start resumable upload.
    client.ResumableUploadObject(request);
    Console.WriteLine("Resumable upload object:{0} succeeded", objectName);
}
catch (OssException ex)
{
    Console.WriteLine("Failed with error code: {0}; Error info: {1}. \nRequestId:{2}\tHostID:{3}",
        ex.ErrorCode, ex.Message, ex.RequestId, ex.HostId);
}
catch (Exception ex)
{
    Console.WriteLine("Failed with error info: {0}", ex.Message);
}

```

Add tags to an uploaded object or modify the tags of the object

The following code provides an example on how to add tags to an uploaded object or modify the tags of the object:

```

using Aliyun.OSS;
// Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com.
var endpoint = "yourEndpoint";
// The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to access OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
var accessKeyId = "yourAccessKeyId";
var accessKeySecret = "yourAccessKeySecret";
// Specify the bucket name.
var bucketName = "examplebucket";
// Specify the full path of the object. The full path of the object cannot contain the bucket name.
var objectName = "exampleobject.txt";
// Create an OSSClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
try
{
    // Configure the tags that you want to add to the object.
    var setRequest = new SetObjectTaggingRequest(bucketName, objectName);
    var tag1 = new Tag
    {
        Key = "project",
        Value = "projectone"
    };
    var tag2 = new Tag
    {
        Key = "user",
        Value = "jsmith"
    };
    setRequest.AddTag(tag1);
    setRequest.AddTag(tag2);
    client.SetObjectTagging(setRequest);
    Console.WriteLine("set object tagging succeeded");
}
catch (Exception ex)
{
    Console.WriteLine("set object tagging failed. {0}", ex.Message);
}

```

Add tags to an object when you copy the object

The following examples describe how to add tags to objects smaller than 1 GB in simple copy mode and larger than 1 GB in multipart copy mode:

- The following code provides an example on how to add tags to an object smaller than 1 GB in size when you copy the object by using simple copy:

```

using Aliyun.OSS;
using Aliyun.OSS.Common;
using System.Text;
// The AccessKey pair of an Alibaba Cloud account has permissions on all API operations.

```

Using these credentials to access OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.

```
var accessKeyId = "yourAccessKeyId";
var accessKeySecret = "yourAccessKeySecret";
// Specify the name of the source bucket. Example: srcexamplebucket.
var sourceBucket = "srcexamplebucket";
// Specify the full path of the source object. The full path cannot contain the bucket name. Example: srcdir/scrobject.txt.
var sourceObject = "srcdir/scrobject.txt";
// Specify the name of the destination bucket. The destination bucket must be located in the same region as the source bucket. Example: destbucket.
var targetBucket = "destbucket";
// Specify the full path of the destination object. The full path cannot contain the bucket name. Example: destdir/destobject.txt.
var targetObject = "destdir/destobject.txt";
String UrlEncodeKey(String key)
{
    const string CharsetName = "utf-8";
    const char separator = '/';
    var segments = key.Split(separator);
    var encodedKey = new StringBuilder();
    encodedKey.Append(HttpUtils.EncodeUri(segments[0], CharsetName));
    for (var i = 1; i < segments.Length; i++)
        encodedKey.Append(separator).Append(HttpUtils.EncodeUri(segments[i], CharsetName));
    if (key.EndsWith(separator.ToString()))
    {
        // String#split ignores trailing empty strings, e.g., "a/b/" will be split as a 2-entries array,
        // so we have to append all the trailing slash to the uri.
        foreach (var ch in key)
        {
            if (ch == separator)
                encodedKey.Append(separator);
            else
                break;
        }
    }
    return encodedKey.ToString();
}
// Create an OSSClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
try
{
    var metadata = new ObjectMetadata();
    metadata.AddHeader("mk1", "mv1");
    metadata.AddHeader("mk2", "mv2");
    // Configure the tags in the HTTP header.
    string str = UrlEncodeKey("key1") + "=" + UrlEncodeKey("key1") + "&" + UrlEncodeKey("key2") + "=" + UrlEncodeKey("key2");
    metadata.AddHeader("x-oss-tagging", str);
    var req = new CopyObjectRequest(sourceBucket, sourceObject, targetBucket, targetObject)
    {
```

```
// If the value of NewObjectMetadata is null, the Copy mode is used to copy the m
etadata of the source object. Otherwise, the Replace mode is used to overwrite the metada
ta of the source object.
    NewObjectMetadata = metadata
};
// Copy the object.
client.CopyObject(req);
Console.WriteLine("Copy object succeeded");
}
catch (OssException ex)
{
    Console.WriteLine("Failed with error code: {0}; Error info: {1}. \nRequestID: {2} \tH
ostID: {3}",
        ex.ErrorCode, ex.Message, ex.RequestId, ex.HostId);
}
catch (Exception ex)
{
    Console.WriteLine("Failed with error info: {0}", ex.Message);
}
```

- The following code provides an example on how to add tags to an object larger than 1 GB in size when you copy the object by using multipart copy:

```
using Aliyun.OSS;
using System.Text;
// The AccessKey pair of an Alibaba Cloud account has permissions on all API operations.
Using these credentials to access OSS is a high-risk operation. We recommend that you use
a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to
the RAM console.
var accessKeyId = "yourAccessKeyId";
var accessKeySecret = "yourAccessKeySecret";
// Specify the name of the source bucket. Example: srcexamplebucket.
var sourceBucket = "srcexamplebucket";
// Specify the full path of the source object. The full path cannot contain the bucket na
me. Example: srcdir/scrobject.txt.
var sourceObject = "srcdir/scrobject.txt";
// Specify the name of the destination bucket. The destination bucket must be located in
the same region as the source bucket. Example: destbucket.
var targetBucket = "destbucket";
// Specify the full path of the destination object. The full path cannot contain the buck
et name. Example: destdir/destobject.txt.
var targetObject = "destdir/destobject.txt";
var uploadId = "";
var partSize = 50 * 1024 * 1024;
String UrlEncodeKey(String key)
{
    const string CharSetName = "utf-8";
    const char separator = '/';
    var segments = key.Split(separator);
    var encodedKey = new StringBuilder();
    encodedKey.Append(HttpUtils.EncodeUri(segments[0], CharSetName));
    for (var i = 1; i < segments.Length; i++)
        encodedKey.Append(separator).Append(HttpUtils.EncodeUri(segments[i], CharSetName));
    if (key.EndsWith(separator.ToString()))
    {

```

```

        // String#split ignores trailing empty strings, e.g., "a/b/" will be split as a 2
        -entries array,
        // so we have to append all the trailing slash to the uri.
        foreach (var ch in key)
        {
            if (ch == separator)
                encodedKey.Append(separator);
            else
                break;
        }
    }
    return encodedKey.ToString();
}
// Create an OSSClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
try
{
    // Initiate a multipart copy task. You can call the InitiateMultipartUploadRequest op
    eration to specify the metadata of the destination object.
    var request = new InitiateMultipartUploadRequest(targetBucket, targetObject);
    var meta = new ObjectMetadata();
    // Configure the tags in the HTTP header.
    string str = UriEncodeKey("key1") + "=" + UriEncodeKey("key1") + "&" + UriEncodeKey("
key2") + "=" + UriEncodeKey("key2");
    meta.AddHeader("x-oss-tagging", str);
    request.Metadata = meta;
    var result = client.InitiateMultipartUpload(request);
    // Display the upload ID.
    uploadId = result.UploadId;
    Console.WriteLine("Init multipart upload succeeded, Upload Id: {0}", result.UploadId)
;

    // Calculate the total number of parts.
    var metadata = client.GetObjectMetadata(sourceBucket, sourceObject);
    var fileSize = metadata.ContentLength;
    var partCount = (int)fileSize / partSize;
    if (fileSize % partSize != 0)
    {
        partCount++;
    }
    // Start the multipart copy task.
    var partETags = new List<PartETag>();
    for (var i = 0; i < partCount; i++)
    {
        var skipBytes = (long)partSize * i;
        var size = (partSize < fileSize - skipBytes) ? partSize : (fileSize - skipBytes);
        // Create an UploadPartCopyRequest object. You can use UploadPartCopyRequest to s
        pecify conditions.
        var uploadPartCopyRequest = new UploadPartCopyRequest(targetBucket, targetObject,
sourceBucket, sourceObject, uploadId)
        {
            PartSize = size,
            PartNumber = i + 1,
            // Use BeginIndex to find the start position to copy the part.
            BeginIndex = skipBytes

```

```

        };
        // Call the uploadPartCopy operation to copy each part.
        var uploadPartCopyResult = client.UploadPartCopy(uploadPartCopyRequest);
        Console.WriteLine("UploadPartCopy : {0}", i);
        partETags.Add(uploadPartCopyResult.PartETag);
    }
    // Complete the multipart copy task.
    var completeMultipartUploadRequest =
        new CompleteMultipartUploadRequest(targetBucket, targetObject, uploadId);
    // partETags is a list of part ETags. OSS verifies each part after it receives the li
    st of parts. After all parts are verified, OSS combines these parts into a complete objec
    t.
    foreach (var partETag in partETags)
    {
        completeMultipartUploadRequest.PartETags.Add(partETag);
    }
    var completeMultipartUploadResult = client.CompleteMultipartUpload(completeMultipartU
    ploadRequest);
    Console.WriteLine("CompleteMultipartUpload succeeded");
}
catch (OssException ex)
{
    Console.WriteLine("Failed with error code: {0}; Error info: {1}. \nRequestID: {2} \tH
    ostID: {3}",
        ex.ErrorCode, ex.Message, ex.RequestId, ex.HostId);
}
catch (Exception ex)
{
    Console.WriteLine("Failed with error info: {0}", ex.Message);
}
}

```

Add tags to a symbolic link

The following code provides an example on how to add tags to a symbolic link:

```

using Aliyun.OSS;
using System.Text;
// Set yourEndpoint to the endpoint of the region in which the bucket is located. For exampl
e, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://os
s-cn-hangzhou.aliyuncs.com.
var endpoint = "yourEndpoint";
// The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Us
ing these credentials to access OSS is a high-risk operation. We recommend that you use a R
AM user to call API operations or perform routine O&M. To create a RAM user, log on to the
RAM console.
var accessKeyId = "yourAccessKeyId";
var accessKeySecret = "yourAccessKeySecret";
// Specify the bucket name.
var bucketName = "examplebucket";
// Specify the full path of the object. The full path of the object cannot contain the buck
et name.
var targetObjectName = "exampleobject.txt";
// Specify the name of the symbolic link. Example: symlink.txt.
var symlinkObjectName = "symlink.txt";

```

```

var objectContent = "More than just cloud.";
String UrlEncodeKey(String key)
{
    const string CharsetName = "utf-8";
    const char separator = '/';
    var segments = key.Split(separator);
    var encodedKey = new StringBuilder();
    encodedKey.Append(HttpUtils.EncodeUri(segments[0], CharsetName));
    for (var i = 1; i < segments.Length; i++)
        encodedKey.Append(separator).Append(HttpUtils.EncodeUri(segments[i], CharsetName));
    if (key.EndsWith(separator.ToString()))
    {
        // String#split ignores trailing empty strings, e.g., "a/b/" will be split as a 2-e
        ntries array,
        // so we have to append all the trailing slash to the uri.
        foreach (var ch in key)
        {
            if (ch == separator)
                encodedKey.Append(separator);
            else
                break;
        }
    }
    return encodedKey.ToString();
}
// Create an OSSClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
try
{
    // Upload the object to which the symbolic link points.
    byte[] binaryData = Encoding.ASCII.GetBytes(objectContent);
    MemoryStream requestContent = new MemoryStream(binaryData);
    client.PutObject(bucketName, targetObjectName, requestContent);
    var meta = new ObjectMetadata();
    // Configure the tags in the HTTP header.
    string str = UrlEncodeKey("key1") + "=" + UrlEncodeKey("key1") + "&" + UrlEncodeKey("ke
    y2") + "=" + UrlEncodeKey("key2");
    meta.AddHeader("x-oss-tagging", str);
    var request = new CreateSymlinkRequest(bucketName, symlinkObjectName, targetObjectName)
;
    request.ObjectMetadata = meta;
    // Create the symbolic link.
    client.CreateSymlink(request);
    // Query the name of the object to which the symbolic link points.
    var ossSymlink = client.GetSymlink(bucketName, symlinkObjectName);
    Console.WriteLine("Target object is {0}", ossSymlink.Target);
}
catch (Exception ex)
{
    Console.WriteLine("Failed with error info: {0}", ex.Message);
}

```

References

For more information about the API operation that you can call to configure object tagging, see [PutObjectTagging](#).

7.10.2. Query the tags added to an object

The object tagging feature uses a key-value pair to tag an object. This topic describes how to query the tags added to an object:

The following code provides an example on how to query the tags added to an object:

```
using Aliyun.OSS;
var endpoint = "<yourEndpoint>";
var accessKeyId = "<yourAccessKeyId>";
var accessKeySecret = "<yourAccessKeySecret>";
var bucketName = "<yourBucketName>";
var objectName = "<yourObjectName>";
// Create an OSSClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
try
{
    // Query the tags added to the object.
    var result = client.GetObjectTagging(bucketName, objectName);
    Console.WriteLine("get objects tagging succeeded");
    foreach (var tag in result.Tags)
    {
        Console.WriteLine("key:{0}, value:{1}", tag.Key, tag.Value);
    }
}
catch (Exception ex)
{
    Console.WriteLine("get objects tagging failed. {0}", ex.Message);
}
```

For more information about how to query the tags added to an object, see [GetObjectTagging](#). For more information about the object tagging feature, see [Object tagging](#).

7.10.3. Delete the tags added to an object

The object tagging feature uses a key-value pair to tag an object. This topic describes how to delete the tags added to an object.

The following code provides an example on how to delete the tags added to an object:


```
using Aliyun.OSS;
var endpoint = "<yourEndpoint>";
var accessKeyId = "<yourAccessKeyId>";
var accessKeySecret = "<yourAccessKeySecret>";
var bucketName = "<yourBucketName>";
var objectName = "<yourObjectName>";
// Create an OSSClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
try
{
    // Delete the tags added to the object.
    var result = client.DeleteObjectTagging(bucketName, objectName);
    Console.WriteLine("delete objects tagging succeeded");
}
catch (Exception ex)
{
    Console.WriteLine("delete objects failed. {0}", ex.Message);
}
```

For more information about how to delete the tags added to an object, see [DeleteObjectTagging](#). For more information about the object tagging feature, see [Object tagging](#).

7.10.4. Object tagging and lifecycle rules

Lifecycle rules can take effect on objects configured with specified prefixes or tags. You can also specify a combination of prefixes and tags as conditions for lifecycle rules to take effect.

Note

Add tagging configurations to a lifecycle rule

The following code provides an example on how to add tagging configurations to a lifecycle rule:

```
using Aliyun.OSS;
using Aliyun.OSS.Common;
var endpoint = "<yourEndpoint>";
var accessKeyId = "<yourAccessKeyId>";
var accessKeySecret = "<yourAccessKeySecret>";
var bucketName = "<yourBucketName>";
// Create an OSSClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
try
{
    var setBucketLifecycleRequest = new SetBucketLifecycleRequest(bucketName);
    // Create the first lifecycle rule.
    LifecycleRule lcr1 = new LifecycleRule()
    {
        ID = "delete obsoleted files",
        Prefix = "obsoleted/",
        Status = RuleStatus.Enabled,
        ExpirationDays = 3,
        Tags = new Tag[1]
```

```
};
// Configure tag1.
var tag1 = new Tag
{
    Key = "project",
    Value = "projectone"
};
lcr1.Tags[0] = tag1;
// Create the second lifecycle rule.
LifecycleRule lcr2 = new LifecycleRule()
{
    ID = "delete temporary files",
    Prefix = "temporary/",
    Status = RuleStatus.Enabled,
    ExpirationDays = 20,
    Tags = new Tag[1]
};
// Configure tag2.
var tag2 = new Tag
{
    Key = "user",
    Value = "jsmith"
};
lcr2.Tags[0] = tag2;
setBucketLifecycleRequest.AddLifecycleRule(lcr1);
setBucketLifecycleRequest.AddLifecycleRule(lcr2);
// Configure lifecycle rules.
client.SetBucketLifecycle(setBucketLifecycleRequest);
Console.WriteLine("Set bucket:{0} Lifecycle succeeded ", bucketName);
}
catch (OssException ex)
{
    Console.WriteLine("Failed with error code: {0}; Error info: {1}. \nRequestID:{2}\tHostID:{3}",
        ex.ErrorCode, ex.Message, ex.RequestId, ex.HostId);
}
catch (Exception ex)
{
    Console.WriteLine("Failed with error info: {0}", ex.Message);
}
```

View tagging configurations of a lifecycle rule.

The following code provides an example on how to view tagging configurations of a lifecycle rule:

```
using Aliyun.OSS;
using Aliyun.OSS.Common;
var endpoint = "<yourEndpoint>";
var accessKeyId = "<yourAccessKeyId>";
var accessKeySecret = "<yourAccessKeySecret>";
var bucketName = "<yourBucketName>";
// Create an OSSClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
try
{
    // View lifecycle rules.
    var rules = client.GetBucketLifecycle(bucketName);
    Console.WriteLine("Get bucket:{0} Lifecycle succeeded ", bucketName);
    foreach (var rule in rules)
    {
        Console.WriteLine("ID: {0}", rule.ID);
        Console.WriteLine("Prefix: {0}", rule.Prefix);
        Console.WriteLine("Status: {0}", rule.Status);
        // View tags.
        foreach (var tag in rule.Tags)
        {
            Console.WriteLine("key:{0}, value:{1}", tag.Key, tag.Value);
        }
        if (rule.ExpirationDays.HasValue)
            Console.WriteLine("ExpirationDays: {0}", rule.ExpirationDays);
    }
}
catch (OssException ex)
{
    Console.WriteLine("Failed with error code: {0}; Error info: {1}. \nRequestId:{2}\tHostID:{3}",
        ex.ErrorCode, ex.Message, ex.RequestId, ex.HostId);
}
catch (Exception ex)
{
    Console.WriteLine("Failed with error info: {0}", ex.Message);
}
```

7.11. Single-connection bandwidth throttling

This topic describes how to add parameters in an object upload and download request to set the limit of upload or download bandwidth. This way, you can ensure other applications have sufficient bandwidth.

Configure single-connection bandwidth throttling for simple upload and download

The following code provides an example on how to configure single-connection bandwidth throttling for simple upload and download:

```

using System.Text;
using Aliyun.OSS;
// Specify the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com.
var endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
var accessKeyId = "yourAccessKeyId";
var accessKeySecret = "yourAccessKeySecret";
// Specify the name of the bucket. Example: examplebucket.
var bucketName = "examplebucket";
// Specify the full path of the object. The path cannot contain the bucket name. Example: exampledir/exampleobject.txt.
var objectName = "exampledir/exampleobject.txt";
var objectContent = "More than just cloud.";
// Create an OSSClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
try
{
    byte[] binaryData = Encoding.ASCII.GetBytes(objectContent);
    MemoryStream requestContent = new MemoryStream(binaryData);
    // Set the bandwidth limit to 100 KB/s, which is equal to 819200 bit/s.
    var putRequest = new PutObjectRequest(bucketName, objectName, requestContent)
    {
        TrafficLimit = 100*1024*8
    };
    client.PutObject(putRequest);
    Console.WriteLine("Put object succeeded");
    // Set the bandwidth limit for object download to 100 KB/s, which is equal to 819200 bit/s.
    var getRequest = new GetObjectRequest(bucketName, objectName)
    {
        TrafficLimit = 100 * 1024 * 8
    };
    var getResult = client.GetObject(getRequest);
    Console.WriteLine("Get object succeeded");
}
catch (Exception ex)
{
    Console.WriteLine("Put object failed, {0}", ex.Message);
}

```

Configure single-connection bandwidth throttling for object uploads and downloads that use signed URLs

The following code provides an example on how to configure single-connection bandwidth throttling when you use signed URLs to upload or download objects:

```

using System.Text;
using Aliyun.OSS;

```

```
// Specify the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com.
var endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
var accessKeyId = "yourAccessKeyId";
var accessKeySecret = "yourAccessKeySecret";
// Specify the name of the bucket. Example: examplebucket.
var bucketName = "examplebucket";
// Specify the full path of the object. The path cannot contain the bucket name. Example: exampledir/exampleobject.txt.
var objectName = "exampledir/exampleobject.txt";
var objectContent = "More than just cloud.";
// Create an OSSClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
try
{
    // Generate a signed URL used to upload the object.
    var generatePresignedUriRequest = new GeneratePresignedUriRequest(bucketName, objectName, SignHttpMethod.Put)
    {
        Expiration = DateTime.Now.AddHours(1),
    };
    // Set the bandwidth limit to 100 KB/s, which is equal to 819,200 bit/s.
    generatePresignedUriRequest.AddQueryParam("x-oss-traffic-limit", "819200");
    var signedUrl = client.GeneratePresignedUri(generatePresignedUriRequest);
    // Use the signed URL to upload the object.
    var buffer = Encoding.UTF8.GetBytes(objectContent);
    using (var ms = new MemoryStream(buffer))
    {
        client.PutObject(signedUrl, ms);
    }
    Console.WriteLine("Put object by signature succeeded. {0} ", signedUrl.ToString());
    var metadata = client.GetObjectMetadata(bucketName, objectName);
    var etag = metadata.ETag;
    // Generate a signed URL used to download the object.
    // Set the bandwidth limit to 100 KB/s, which is equal to 819200 bit/s.
    var req = new GeneratePresignedUriRequest(bucketName, objectName, SignHttpMethod.Get);
    req.AddQueryParam("x-oss-traffic-limit", "819200");
    var uri = client.GeneratePresignedUri(req);
    // Use the signed URL to download the object.
    OssObject ossObject = client.GetObject(uri);
    using (var file = File.Open(downloadFilename, FileMode.OpenOrCreate))
    {
        using (Stream stream = ossObject.Content)
        {
            {
                int length = 4 * 1024;
                var buf = new byte[length];
                do
                {
                    length = stream.Read(buf, 0, length);
                }
            }
        }
    }
}
```

```
        file.Write(buf, 0, length);
    } while (length != 0);
}

Console.WriteLine("Get object by signature succeeded. {0} ", uri.ToString());
}
catch (Exception ex)
{
    Console.WriteLine("Put object failed, {0}", ex.Message);
}
```

7.12. Server-side encryption

OSS supports server-side encryption for uploaded data. This means that when user data is uploaded, OSS encrypts the data and permanently stores the data. Then, when the data is downloaded by a user, OSS automatically decrypts the data, returns the original data to the user, and declares in the header of the response that the data has been encrypted on the server.

The following server-side encryption methods are available for different application scenarios:

- Server-side encryption that uses CMKs managed by KMS for encryption and decryption (SSE-KMS)

When uploading an object, you can use a specified CMK ID or the default CMK managed by KMS to encrypt and decrypt a large amount of data. This method is cost-effective because you do not need to send user data to the KMS server through networks for encryption and decryption.

 **Notice** API calling fees are incurred if you use CMKs to encrypt or decrypt data.

- Server-side encryption fully managed by OSS (SSE-OSS)

When you upload an object, OSS encrypts the object on the server side by using AES256 fully managed by OSS. In this method, OSS uses AES256 to encrypt each object with an individual key. Furthermore, the individual keys are encrypted by a CMK that is updated periodically for higher security. This method applies to encrypt or decrypt bulk data.

 **Notice**

- Only one server-side encryption method can be used for an object at one time.
- If you configure server-side encryption for a bucket, you can still configure the encryption method for a single object when uploading or copying it. In this case, the encryption method configured for the object takes precedence. For more information, see [Put Object](#).
- For more information about server-side encryption, see [Server-side encryption](#).

Configure server-side encryption for buckets

You can run the following code to configure the default encryption method for a bucket. After the method is configured, all objects that are uploaded to the bucket without their encryption methods configured are encrypted in the configured default encryption method.

```
using Aliyun.OSS;
using Aliyun.OSS.Common;
var endpoint = "<yourEndpoint>";
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to lo
g on to OSS because the account has permissions on all API operations. We recommend that yo
u use your RAM user's credentials to call API operations or perform routine operations and
maintenance. To create your RAM user, log on to the RAM console.
var accessKeyId = "<yourAccessKeyId>";
var accessKeySecret = "<yourAccessKeySecret>";
var bucketName = "<yourBucketName>";
// Create an OSSClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
try
{
    // Configure server-side encryption for the bucket.
    var request = new SetBucketEncryptionRequest(bucketName, "KMS", null);
    client.SetBucketEncryption(request);
    Console.WriteLine("Set bucket:{0} Encryption succeeded ", bucketName);
}
catch (OssException ex)
{
    Console.WriteLine("Failed with error code: {0}; Error info: {1}. \nRequestId:{2}\tHostI
D:{3}",
        ex.ErrorCode, ex.Message, ex.RequestId, ex.HostId);
}
catch (Exception ex)
{
    Console.WriteLine("Failed with error info: {0}", ex.Message);
}
```

Query the server-side encryption configurations of buckets

You can run the following code to obtain the server-side encryption settings of a bucket:

```
using Aliyun.OSS;
using Aliyun.OSS.Common;
var endpoint = "<yourEndpoint>";
var accessKeyId = "<yourAccessKeyId>";
var accessKeySecret = "<yourAccessKeySecret>";
var bucketName = "<yourBucketName>";
// Create an OSSClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
try
{
    // Query the server-side encryption configurations of the bucket.
    var result = client.GetBucketEncryption(bucketName);
    Console.WriteLine("Get bucket:{0} Encryption succeeded ", bucketName);
    Console.WriteLine("SSEAlgorithm: {0}", rules.SSEAlgorithm);
    Console.WriteLine("KMSMasterKeyId: {0}", rules.KMSMasterKeyId);
}
catch (OssException ex)
{
    Console.WriteLine("Failed with error code: {0}; Error info: {1}. \nRequestId:{2}\tHostID:{3}",
        ex.ErrorCode, ex.Message, ex.RequestId, ex.HostId);
}
catch (Exception ex)
{
    Console.WriteLine("Failed with error info: {0}", ex.Message);
}
```

Delete the server-side encryption configurations of a bucket

You can run the following code to delete the server-side encryption configurations of a bucket:


```
using Aliyun.OSS;
using Aliyun.OSS.Common;
var endpoint = "<yourEndpoint>";
var accessKeyId = "<yourAccessKeyId>";
var accessKeySecret = "<yourAccessKeySecret>";
var bucketName = "<yourBucketName>";
// Create an OSSClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
try
{
    // Delete the server-side encryption configurations of the bucket.
    client.DeleteBucketEncryption(bucketName);
    Console.WriteLine("Delete bucket:{0} Encryption succeeded ", bucketName);
}
catch (OssException ex)
{
    Console.WriteLine("Failed with error code: {0}; Error info: {1}. \nRequestID:{2}\tHostID:{3}",
        ex.ErrorCode, ex.Message, ex.RequestId, ex.HostId);
}
catch (Exception ex)
{
    Console.WriteLine("Failed with error info: {0}", ex.Message);
}
```

7.13. Authorized access

This topic describes how to authorize temporary access to Object Storage Service (OSS) by using Security Token Service (STS) or a signed URL.

Use STS to authorize temporary access

You can use Alibaba Cloud STS to authorize temporary access to OSS. STS is a web service that provides temporary access tokens for users. You can use STS to grant a set of temporary access credentials that have a custom validity period and custom permissions to a third-party application or a RAM user managed by you. For more information about STS, see [What is STS?](#)

STS provides the following benefits:

- You need only to generate an access token and send the access token to a third-party application, instead of exposing your AccessKey pair to the third-party application. You can define the access permissions and validity period of this token.
- The token automatically expires after the validity period. Therefore, you do not need to manually revoke the access permissions of a token.

 **Note** For more information about how to use STS, see [Use a temporary credential provided by STS to access OSS](#) in OSS User Guide. You can call the [AssumeRole](#) operation or use STS SDKs for various programming languages to obtain temporary access credentials. For more information, see [STS SDK overview](#). The temporary access credentials consist of an AccessKey pair and a security token. The AccessKey pair contains an AccessKey ID and an AccessKey secret. The minimum validity period of temporary access credentials is 900 seconds. The maximum validity period of temporary access credentials is the maximum session duration specified for the current role. For more information, see [Specify the maximum session duration for a RAM role](#).

```
using Aliyun.OSS;
// Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com.
var endpoint = "<yourEndpoint>";
// Specify a temporary AccessKey pair obtained from STS. An AccessKey pair consists of an AccessKey ID and an AccessKey secret.
var accessKeyId = "yourAccessKeyId";
var accessKeySecret = "yourAccessKeySecret";
// Specify a security token obtained from STS.
var securityToken = "yourSecurityToken";
// After you obtain a temporary access credential from STS, you can use the security token and temporary AccessKey pair that are contained in the credential to create an OssClient instance.
// Create an OssClient instance.
var ossStsClient = new OssClient(endpoint, accessKeyId, accessKeySecret, securityToken);
// Perform corresponding operations.
```

Use a signed URL to authorize temporary access

You can generate a signed URL and provide the URL to a visitor for temporary access. When you generate a signed URL, you can specify the validity period of the URL to limit the period of access from visitors. By default, the validity period of a signed URL is 3,600 seconds. The maximum validity period of a signed URL is 32,400 seconds.

 **Notice** A validity period must be set for an STS temporary access credential and a signed URL. When you use an STS temporary account credential to generate a signed URL that is used to perform operations such as object upload and download, the minimum validity period takes precedence. For example, you can set the validity period of your STS temporary access credential to 1,200 seconds and the validity period of your signed URL generated by using this credential to 3,600 seconds. In this case, the signed URL cannot be used to upload objects after the STS temporary access credential expires, even if the signed URL is within its validity period.

For the complete code that is used to authorize temporary access by using a signed URL, visit [GitHub](#).

This section provides examples on how to use a signed URL to authorize temporary access.

- Use a signed URL to upload an object

The following code provides an example on how to use a signed URL to upload an object:

```

using Aliyun.OSS;
using Aliyun.OSS.Common;
// Specify the endpoint of the region in which the bucket is located. For example, if the
bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hang
zhou.aliyuncs.com.
var endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to
access Object Storage Service (OSS) because the account has permissions on all API operat
ions. We recommend that you use a RAM user to call API operations or perform routine O&M.
To create a RAM user, log on to the RAM console.
var accessKeyId = "yourAccessKeyId";
var accessKeySecret = "yourAccessKeySecret";
// Specify the bucket name. Example: examplebucket.
var bucketName = "examplebucket";
// Specify the full path of the object. The full path cannot contain the bucket name. Exa
mple: exampledir/exampleobject.txt.
var objectName = "exampledir/exampleobject.txt";
var objectContent = "More than just cloud.";
// Create an OssClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
try
{
    // Generate a signed URL.
    var generatePresignedUriRequest = new GeneratePresignedUriRequest(bucketName, objectN
ame, SignHttpMethod.Put)
    {
        // Specify the validity period of the signed URL. Default value: 3600. Unit: seco
nds.
        Expiration = DateTime.Now.AddHours(1),
    };
    var signedUrl = client.GeneratePresignedUri(generatePresignedUriRequest);
    // Use the signed URL to upload the object.
    var buffer = Encoding.UTF8.GetBytes(objectContent);
    using (var ms = new MemoryStream(buffer))
    {
        client.PutObject(signedUrl, ms);
    }
    Console.WriteLine("Put object by signatrue succeeded. {0} ", signedUrl.ToString());
}
catch (OssException ex)
{
    Console.WriteLine("Failed with error code: {0}; Error info: {1}. \nRequestID:{2}\tHos
tID:{3}",
        ex.ErrorCode, ex.Message, ex.RequestId, ex.HostId);
}
catch (Exception ex)
{
    Console.WriteLine("Failed with error info: {0}", ex.Message);
}
}

```

- Use a signed URL to download an object

The following code provides an example on how to use a signed URL to download an object:

```
using Aliyun.OSS;
```

```

using Aliyun.OSS.Common;
// Specify the endpoint of the region in which the bucket is located. For example, if the
bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hang
zhou.aliyuncs.com.
var endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to
access OSS because the account has permissions on all API operations. We recommend that y
ou use a RAM user to call API operations or perform routine O&M. To create a RAM user, lo
g on to the RAM console.
var accessKeyId = "yourAccessKeyId";
var accessKeySecret = "yourAccessKeySecret";
// Specify the bucket name. Example: examplebucket.
var bucketName = "examplebucket";
// Specify the full path of the object. The full path cannot contain the bucket name. Exa
mple: exampledir/exampleobject.txt.
var objectName = "exampledir/exampleobject.txt";
// Specify the full path of the local file to which you want to download the object. Exam
ple: D:\\localpath\\examplefile.txt. If the specified local file exists, the object to do
wnload replaces the file. If the specified local file does not exist, a file is created.
var downloadFilename = "D:\\localpath\\examplefile.txt";
// Create an OssClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
try
{
    var metadata = client.GetObjectMetadata(bucketName, objectName);
    var etag = metadata.ETag;
    // Generate a signed URL.
    var req = new GeneratePresignedUriRequest(bucketName, objectName, SignHttpMethod.Get)
    {
        // Specify the validity period of the signed URL. Default value: 3600. Unit: seco
nds.
        Expiration = DateTime.Now.AddHours(1),
    };
    var uri = client.GeneratePresignedUri(req);
    // Use the signed URL to download the object.
    OssObject ossObject = client.GetObject(uri);
    using (var file = File.Open(downloadFilename, FileMode.OpenOrCreate))
    {
        using (Stream stream = ossObject.Content)
        {
            int length = 4 * 1024;
            var buf = new byte[length];
            do
            {
                length = stream.Read(buf, 0, length);
                file.Write(buf, 0, length);
            } while (length != 0);
        }
    }
    Console.WriteLine("Get object by signatrue succeeded. {0} ", uri.ToString());
}
catch (OssException ex)
{
    Console.WriteLine("Failed with error code: {0}; Error info: {1}. \nRequestID:{2}\tHos

```

```
tID:{3}",
    ex.ErrorCode, ex.Message, ex.RequestId, ex.HostId);
}
catch (Exception ex)
{
    Console.WriteLine("Failed with error info: {0}", ex.Message);
}
```

7.14. IMG

Image Processing (IMG) provided by Object Storage Service (OSS) is a secure, cost-effective, and highly reliable image processing service that can process large amounts of data. After you upload source images to OSS, you can call RESTful APIs to process the images anytime, anywhere, and on all Internet devices.

For more information about IMG parameters, see [Parameters](#).

Use a single IMG parameter to process images

- The following code provides an example on how to use a single IMG parameter to process an image and save the image to your computer:

```
using System;
using System.IO;
using Aliyun.OSS;
using Aliyun.OSS.Common;
using Aliyun.OSS.Util;
namespace ImageProcess
{
    class Program
    {
        static void Main(string[] args)
        {
            Program.ImageProcess();
            Console.ReadKey();
        }
        public static void ImageProcess()
        {
            var endpoint = "<yourEndpoint>";
            var accessKeyId = "<yourAccessKeyId>";
            var accessKeySecret = "<yourAccessKeySecret>";
            var bucketName = "<yourBucketName>";
            var objectName = "<yourObjectName>";
            // Specify the name of the local image to upload the image.
            // var localImageFilename = "<yourLocalImageFilename>";
            // Specify the local directory to which the processed image is saved.
            var downloadDir = "<yourDownloadDir>";
            // Create an OSSClient instance.
            var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
            try
            {
                // If the image does not exist in the required bucket, you must upload the
                // image to the bucket.
                // client.PutObject(bucketName, objectName, localImageFilename);
            }
        }
    }
}
```

```

        // Resize the image to the height and width of 100 pixels.
        var process = "image/resize,m_fixed,w_100,h_100";
        var ossObject = client.GetObject(new GetObjectRequest(bucketName, objectName, process));

        // Specify the name of the processed image.
        WriteToFile(downloadDir + "/" + LocalFilename, ossObject.Content);
    }
    catch (OssException ex)
    {
        Console.WriteLine("Failed with error code: {0}; Error info: {1}. \nRequestID:{2}\tHostID:{3}",
            ex.ErrorCode, ex.Message, ex.RequestId, ex.HostId);
    }
    catch (Exception ex)
    {
        Console.WriteLine("Failed with error info: {0}", ex.Message);
    }
}

private static void WriteToFile(string filePath, Stream stream)
{
    using (var requestStream = stream)
    {
        using (var fs = File.Open(filePath, FileMode.OpenOrCreate))
        {
            IoUtils.WriteTo(stream, fs);
        }
    }
}
}
}

```

- Use different IMG parameters to process an image and separately save the image as a local image

```

using System;
using System.IO;
using Aliyun.OSS;
using Aliyun.OSS.Common;
using Aliyun.OSS.Util;
namespace ImageProcess
{
    class Program
    {
        static void Main(string[] args)
        {
            Program.ImageProcess();
            Console.ReadKey();
        }
        public static void ImageProcess()
        {
            var endpoint = "<yourEndpoint>";
            var accessKeyId = "<yourAccessKeyId>";
            var accessKeySecret = "<yourAccessKeySecret>";
            var bucketName = "<yourBucketName>";
            var objectName = "<yourObjectName>";
            // Specify the name of the local image to upload the image.

```

```

        // var localImageFilename = "<yourLocalImageFilename>";
        // Specify the local directory to which the processed image is saved.
        var downloadDir = "<yourDownloadDir>";
        // Create an OSSClient instance.
        var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
        try
        {
            // If the image does not exist in the required bucket, you must upload the image to the bucket.
            // client.PutObject(bucketName, objectName, localImageFilename);
            // Resize the image to the height and width of 100 pixels.
            var process = "image/resize,m_fixed,w_100,h_100";
            var ossObject = client.GetObject(new GetObjectRequest(bucketName, objectName, process));
            // Specify the name of the processed image.
            WriteToFile(downloadDir + "/" + LocalFilename, ossObject.Content);
            // Crop the image to the height and width of 100 pixels by setting the coordinate pair to (100, 100).
            process = "image/crop,w_100,h_100,x_100,y_100";
            ossObject = client.GetObject(new GetObjectRequest(bucketName, objectName, process));
            WriteToFile(downloadDir + "/" + LocalFilename, ossObject.Content);
            // Rotate the image 90 degrees.
            process = "image/rotate,90";
            ossObject = client.GetObject(new GetObjectRequest(bucketName, objectName, process));
            WriteToFile(downloadDir + "/" + LocalFilename, ossObject.Content);
        }
        catch (OssException ex)
        {
            Console.WriteLine("Failed with error code: {0}; Error info: {1}. \nRequestId:{2}\tHostID:{3}",
                ex.ErrorCode, ex.Message, ex.RequestId, ex.HostId);
        }
        catch (Exception ex)
        {
            Console.WriteLine("Failed with error info: {0}", ex.Message);
        }
    }
    private static void WriteToFile(string filePath, Stream stream)
    {
        using (var requestStream = stream)
        {
            using (var fs = File.Open(filePath, FileMode.OpenOrCreate))
            {
                IoUtils.WriteTo(stream, fs);
            }
        }
    }
}

```

- Use multiple IMG parameters to process an image and save the image to your computer

The following code provides an example on how to use multiple IMG parameters to process an image. Separate these parameters with forward slashes (/).

```
using System;
using System.IO;
using Aliyun.OSS;
using Aliyun.OSS.Common;
using Aliyun.OSS.Util;
namespace ImageProcessCascade
{
    class Program
    {
        static void Main(string[] args)
        {
            Program.ImageProcessCascade();
            Console.ReadKey();
        }
        public static void ImageProcessCascade()
        {
            // Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com.
            var endpoint = "yourEndpoint";
            // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a Resource Access Management (RAM) user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
            var accessKeyId = "yourAccessKeyId";
            var accessKeySecret = "yourAccessKeySecret";
            // Specify the name of the bucket in which the source image is stored. Example: examplebucket.
            var bucketName = "examplebucket";
            // Specify the name of the source image. If the image is not stored in the root directory of the bucket, you must add the full path of the image. Example: exampledir/example.jpg.
            var objectName = "exampledir/example.jpg";
            // Specify the local full path of the source image.
            // var localImageFilename = "D:\\localpath\\example.jpg";
            // Specify the local directory to which the processed image is saved.
            var downloadDir = "yourDownloadDir";
            // Create an OSSClient instance.
            var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
            try
            {
                // If the image does not exist in the required bucket, you must upload the image to the bucket.
                // client.PutObject(bucketName, objectName, localImageFilename);
                // After you resize the image to the height and width of 100 pixels, rotate the image 90 degrees.
                var process = "image/resize,m_fixed,w_100,h_100/rotate,90";
                var ossObject = client.GetObject(new GetObjectRequest(bucketName, objectName, process));
                // Specify the name of the processed image.
                WriteToFile(downloadDir + "/yourLocalFilename", ossObject.Content);
            }
            catch { }
        }
    }
}
```



```

        Console.WriteLine("Get Object:{0} with process:{1} succeeded ", objectName, process);
    }
    catch (OssException ex)
    {
        Console.WriteLine("Failed with error code: {0}; Error info: {1}. \nRequestId:{2}\tHostID:{3}",
            ex.ErrorCode, ex.Message, ex.RequestId, ex.HostId);
    }
    catch (Exception ex)
    {
        Console.WriteLine("Failed with error info: {0}", ex.Message);
    }
}

private static void WriteToFile(string filePath, Stream stream)
{
    using (var requestStream = stream)
    {
        using (var fs = File.Open(filePath, FileMode.OpenOrCreate))
        {
            IoUtils.WriteTo(stream, fs);
        }
    }
}
}
}

```

Use an image style to process images

You can encapsulate multiple IMG parameters in a style, and then use the style to concurrently process multiple images. For more information, see [Image styles](#). The following code provides an example on how to use an image style to process an image:

```

using System;
using System.IO;
using Aliyun.OSS;
using Aliyun.OSS.Common;
using Aliyun.OSS.Util;
namespace ImageProcessCustom
{
    class Program
    {
        static void Main(string[] args)
        {
            Program.ImageProcessCustomStyle();
            Console.ReadKey();
        }

        public static void ImageProcessCustomStyle()
        {
            // Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com.
            var endpoint = "yourEndpoint";
            // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We

```

operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.

```
var accessKeyId = "yourAccessKeyId";
var accessKeySecret = "yourAccessKeySecret";
// Specify the name of the bucket in which the source image is stored. Example:
examplebucket.
```

```
var bucketName = "examplebucket";
// Specify the name of the source image. If the image is not stored in the root
directory of the bucket, you must add the full path of the image. Example: exampledir/example.jpg.
```

```
var objectName = "exampledir/example.jpg";
// Specify the local full path of the source image.
// var localImageFilename = "D:\\localpath\\example.jpg";
// Specify the local directory to which the processed image is saved.
var downloadDir = "yourDownloadDir";
// Create an OSSClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
try
{
```

```
    // If the image does not exist in the required bucket, you must upload the
image to the bucket.
```

```
    // client.PutObject(bucketName, objectName, localImageFilename);
    // Use the image style to process the image. Set yourCustomStyleName to the
name of the image style you create in the OSS console.
```

```
var process = "style/yourCustomStyleName";
var ossObject = client.GetObject(new GetObjectRequest(bucketName, objectName, process));
```

```
    // Specify the name of the processed image.
    WriteToFile(downloadDir + "/yourLocalFilename", ossObject.Content);
    Console.WriteLine("Get Object:{0} with process:{1} succeeded ", objectName,
process);
```

```
    }
    catch (OssException ex)
    {
```

```
        Console.WriteLine("Failed with error code: {0}; Error info: {1}. \nRequestId:{2}\tHostID:{3}",
            ex.ErrorCode, ex.Message, ex.RequestId, ex.HostId);
```

```
    }
    catch (Exception ex)
    {
```

```
        Console.WriteLine("Failed with error info: {0}", ex.Message);
    }
}
```

```
private static void WriteToFile(string filePath, Stream stream)
{
```

```
    using (var requestStream = stream)
    {
```

```
        using (var fs = File.Open(filePath, FileMode.OpenOrCreate))
        {
```

```
            IoUtils.WriteTo(stream, fs);
        }
    }
}
```

```
    }
}
```

```
}
```

Generate a signed object URL that includes IMG parameters

URLs of private objects must be signed. IMG parameters cannot be added to the end of a signed URL. If you want to process a private object, add IMG parameters to the signature. The following code provides an example on how to add IMG parameters to the signature:

```
using Aliyun.OSS;
using Aliyun.OSS.Common;
// Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com.
var endpoint = "yourEndpoint";
// The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
var accessKeyId = "yourAccessKeyId";
var accessKeySecret = "yourAccessKeySecret";
// Specify the name of the bucket in which the source image is stored. Example: examplebucket.
var bucketName = "examplebucket";
// Specify the name of the source image. If the image is not stored in the root directory of the bucket, you must add the full path of the image. Example: exampledir/example.jpg.
var objectName = "exampledir/example.jpg";
// Create an OSSClient instance.
var client = new OssClient(endpoint, accessKeyId, accessKeySecret);
try
{
    // Resize the image to the height and width of 100 pixels.
    var process = "image/resize,m_fixed,w_100,h_100";
    var req = new GeneratePresignedUriRequest(bucketName, objectName, SignHttpMethod.Get)
    {
        Expiration = DateTime.Now.AddHours(1),
        Process = process
    };
    // Generate a signed object URL that includes IMG parameters.
    var uri = client.GeneratePresignedUri(req);
    Console.WriteLine("Generate Presigned Uri:{0} with process:{1} succeeded ", uri, process);
}
catch (OssException ex)
{
    Console.WriteLine("Failed with error code: {0}; Error info: {1}. \nRequestId:{2}\tHostID:{3}",
        ex.ErrorCode, ex.Message, ex.RequestId, ex.HostId);
}
catch (Exception ex)
{
    Console.WriteLine("Failed with error info: {0}", ex.Message);
}
```

7.15. Exception handling

OSS .NET SDK has two types of exceptions: `ClientException` and `OSSEException`. Both share the properties of `RuntimeException`.

ClientException

The `ClientException` exception occurs when the client attempts to send requests and when the data is transmitted to OSS. For example, when a request is sent, `ClientException` may occur due to unavailable network connections. `ClientException` may occur during file uploads due to I/O exceptions.

OSSEException

`OSSEException` indicates a server exception. The exception occurs because a server error message is parsed. `OSSEException` includes the error code and information returned from OSS. It is used to locate and handle problems.

The following table describes common `OSSEException` information.

Parameter	Description
Code	Specifies the error code returned by OSS.
Message	Specifies the detailed error information returned by OSS.
RequestId	Specifies the UUID. It is unique and used to identify a request. If a problem persists, send the RequestId to OSS developers for help.
HostId	Identifies an OSS cluster. Ensure that the value of HostId is consistent with the Host (endpoint to access a bucket) used in the request.

OSS error codes

Error code	Description
AccessDenied	Access denied
BucketAlreadyExists	The bucket already exists.
BucketNotEmpty	The bucket is not empty.
EntityTooLarge	The entity size exceeds the maximum limit.
EntityTooSmall	The entity size is below the minimum limit.
FileGroupTooLarge	The total file group size exceeds the maximum limit.
FilePartNotExist	No such part exists.
FilePartStale	The part has expired.

Error code	Description
InvalidArgument	The parameter format is invalid.
InvalidAccessKeyId	No such AccessKeyId exists.
InvalidBucketName	The bucket name is invalid.
InvalidDigest	The digest is invalid.
InvalidObjectName	The object name is invalid.
InvalidPart	The part is invalid.
InvalidPartOrder	The part order is invalid.
InvalidTargetBucketForLogging	The buckets that store log files are invalid.
InternalError	Internal OSS error
MalformedXML	The XML format is invalid.
MethodNotAllowed	The method is not allowed.
MissingArgument	Parameters are not configured.
MissingContentLength	The content length is not configured.
NoSuchBucket	No such bucket exists.
NoSuchKey	No such object exists.
NoSuchUpload	No such uploadId exists.
NotImplemented	The methods are not implemented.
PreconditionFailed	Precondition error
RequestTimeTooSkewed	The local time set for OSSClient is deviated from the time set for the OSS server by over 15 minutes.
RequestTimeout	Request timeout
SignatureDoesNotMatch	Signature mismatch
TooManyBuckets	The number of buckets exceeds the limit.

8.Android

8.1. Preface

This topic is written based on Object Storage Service (OSS) SDK for Android V2.9.5.

Compatibility

OSS SDK for Android is backward compatible.

Source code

For more information about the source code of OSS SDK for Android, visit [GitHub](#).

Sample code

OSS SDK for Android provides various sample code files for your reference or use.

The following table describes the sample code files provided by OSS SDK for Android.

Sample code file	Content
BaseTestCase.java	Initialization
OSSBucketTest.java	• Create buckets, Query bucket ACLs, Query bucket information, Delete buckets, and List buckets • Manage lifecycle rules, Hotlink protection, and Access logging
OSSPutObjectTest.java	Simple upload , Append upload , Upload callback , Delete multiple objects in a batch , and Progress bar
MultipartUploadTest.java	Multipart upload
ResumableUploadTest.java	Resumable upload
OSSGetObjectTest.java	Download objects
ManageObjectTest.java	Determine whether a specified object exists , Query the ACL of an object , Copy objects , List objects , Delete a single object , Set the Content-Type of an object , and Obtain the metadata of an object
RestoreObjectTest.java	Restore an archive object
SymlinkTest.java	Manage symbolic links
CRC64Test.java	Data security
OSSAuthenticationTest.java	Sign a URL and Authorized access
ImagePersistTest.java	Image Processing (IMG)

8.2. Installation

This topic describes how to install Object Storage Service (OSS) SDK for Android.

 **Note** We recommend that you specify the version of OSS SDK for Android that you want to install.

Prerequisites

- Android 2.3 or later is used.
- OSS is activated.

Download OSS SDK for Android

- Download by adding dependencies in Android Studio

Add the following dependencies in Android Studio:

```
dependencies { compile 'com.aliyun.dpa:oss-android-sdk:2.9.11' }
```

- [Download the installation package](#)
- [Download from GitHub](#)

Install OSS SDK for Android

You can use one of the following methods to install OSS SDK for Android:

- Method 1 (Recommended): Add dependencies to your Maven project

```
dependencies {  
    compile 'com.aliyun.dpa:oss-android-sdk:2.9.11'
```

- Method 2: Manually import a JAR package

Import the following three JAR packages to the libs directory: [aliyun-oss-sdk-android-2.9.11.jar](#), [okhttp-3.x.x.jar](#), and [okio-1.x.x.jar](#).

Related settings

This section describes the related settings that you must configure before you use OSS SDK for Android.

- Permission settings

The following permissions are required to use OSS SDK for Android.

 **Notice** Make sure that the preceding permissions are configured in the AndroidManifest.xml file. Otherwise, OSS SDK for Android cannot function properly.

```
<uses-permission android:name="android.permission.INTERNET"></uses-permission>  
<uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE"></uses-permission>  
</uses-permission>
```

- ProGuard settings

Add the following code to the ProGuard settings:

```
-keep class com.alibaba.sdk.android.oss.** { *; }  
-dontwarn okio.**  
-dontwarn org.apache.commons.codec.binary.**
```

8.3. Initialization

OSSClient is the client of OSS for Android systems. It provides the caller with a series of methods to operate and manage the buckets and objects. Before you use the OSS Android SDK to initiate a request to OSS, you must first initialize an OSSClient instance and complete necessary settings.

 **Note** Make sure that the life cycle of the OSSClient is consistent with that of the application. Create a global OSSClient instance when an application starts, and destroy it when the application ends.

Determine an endpoint

An endpoint is the URL of Alibaba Cloud OSS in a region. The following table describes the supported formats of OSS endpoints.

- OSS endpoints

When you use the endpoint of a bucket, you can query the endpoint in either of the following ways:

- For more information about how to query the relationship between the endpoint and region, see [Regions and endpoints](#).
- You can log on to the [OSS console](#). In the left-side navigation pane, locate the Buckets section, enter bucket-1 in the search box, and click the search icon. Click the bucket name to go to the Overview tab. In the Domain Names section, the suffix of `bucket-1.oss-cn-hangzhou.aliyuncs.com` is `oss-cn-hangzhou.aliyuncs.com`, which is the endpoint for Internet access.

- Custom domain name

Use CNAME to bind a custom domain name to a bucket. Use the custom domain name to access objects in the bucket.

For example, if you bind the `example.aliyundoc.com` domain name to the bucket named `examplebucket` in China (Hangzhou). Contact your DNS provider of `aliyundoc.com` to configure a new CNAME record. This record is used to resolve `https://example.aliyundoc.com` to `https://examplebucket.oss-cn-hangzhou.aliyuncs.com`. The record type is CNAME.

Configure the endpoint and credential

A mobile client is an untrusted environment. An extremely high risk occurs if you store the `AccessKeyId` and `AccessKeySecret` in a mobile client for signing requests. We recommend you use STS Authentication Mode or Self-signed Mode. For more information, see [Authorized access](#) and [Set up direct data transfer for mobile apps](#).

 **Note** If you are using STS authentication mode, we recommend you use `OSSAuthCredentialsProvider` to access the application server. Expired tokens are updated automatically.

For more information about sample projects, see [GitHub](#).

An example of EndPoint and CredentialProvider settings is shown as follows:

```
String endpoint = "http://oss-cn-hangzhou.aliyuncs.com";
String stsServer = "STS application server address, such as https://example.aliyundoc.com"
//We recommend you use OSSAuthCredentialsProvider because expired tokens can be updated quickly.
OSSCredentialProvider credentialProvider = new OSSAuthCredentialsProvider(stsServer);
//If this configuration class is not set, default configuration applies.
ClientConfiguration conf = new ClientConfiguration();
conf.setConnectionTimeout(15 * 1000); // Connection time-out. The default value is 15 seconds.
conf.setSocketTimeout(15 * 1000); // Socket time-out. The default value is 15 seconds.
conf.setMaxConcurrentRequest(5); // The maximum number of concurrent requests. The default value is 5.
conf.setMaxErrorRetry(2); // The maximum number of retries after a failure. The default value is 2.
OSS oss = new OSSClient(getApplicationContext(), endpoint, credentialProvider);
```

 **Note** For more authentication modes, see [Authorized access](#).

Configure the endpoint according to the CNAME

If you have bound the CNAME to the bucket, you can direct the endpoint to the CNAME directly. For example,

```
String endpoint = "https://examplebucket.oss-cn-hangzhou.aliyuncs.com";
...
OSS oss = new OSSClient(getApplicationContext(), endpoint, credentialProvider);
```

Enable logging

Mobile terminals have complicated environments during usage. The OSS SDK may fail to function in a region or during a period of time. To help developers locate issues, the OSS SDK records some logs to a local directory after the log feature is enabled. To enable the log feature, you must initialize the OSSClient before using it, by calling the following method.

```
//The log style.
//You can call the OSSLog.enableLog() method to enable displaying logs in the console.
//This setting also supports saving one copy of the log file to the SD card of the mobile phone at \OSSLog\logs.csv. This setting is not enabled by default.
//The log records the request data, returned data and exceptions of OSS operations.
//Such as requestId and response header. The following is a log record case.
//android_version: 5.1. The Android version
//mobile_model: XT1085 . The Android phone model
//network_state: connected. The network status
//network_type: WIFI. The network connection type
//Specific operation information:
//[2017-09-05 16:54:52] - Encounter local exception: //java.lang.IllegalArgumentException:
The bucket name is invalid.
//A bucket name must:
//1) be comprised of lower-case characters, numbers or dash(-);
//2) start with lower case or numbers;
//3) be between 3-63 characters long.
//----->end of log
OSSLog.enableLog(); //Call this method to enable logs.
```

Note

- Log files are stored on the SD card at `\OSSLog\logs.csv`.
- You can upload the file to the server to further trace the issue.
- You can also upload the log files after connecting to Alibaba Cloud Log Service. Go to [Log Service](#).

Set network parameters

You can set detailed ClientConfiguration during the initialization process as follows:

```
String endpoint = "http://oss-cn-hangzhou.aliyuncs.com";
// We recommend you initialize OSSClient using STS on the mobile client. For more information, see: [Resource Access Management].
OSSCredentialProvider credentialProvider = new OSSStsTokenCredentialProvider("<StsToken.AccessKeyId>", "<StsToken.SecretKeyId>", "<StsToken.SecurityToken>");
ClientConfiguration conf = new ClientConfiguration();
conf.setConnectionTimeout(15 * 1000); // Connection time-out. The default value is 15 seconds.
conf.setSocketTimeout(15 * 1000); // Socket time-out. The default value is 15 seconds.
conf.setMaxConcurrentTaskNum(5); // The maximum number of concurrent requests. The default value is 5.
conf.setMaxErrorRetry(2); // The maximum number of retries after a failure. The default value is 2.
OSS oss = new OSSClient(getApplicationContext(), endpoint, credentialProvider, conf);
```

Synchronous and asynchronous interfaces in the SDK

Network requests are not allowed to be processed in the UI thread in mobile development, so most SDK interfaces support synchronous and asynchronous calls to handle requests. After it has been called, the synchronous interface blocks other requests while waiting for the returned results. The asynchronous interface has to import a callback function that processes the result of requests.

Synchronous interfaces cannot be called on the UI thread. If a `ClientException` or `ServiceException` occurs, it is thrown immediately. `ClientException` refers to local exceptions, including network exceptions, invalid parameters, and so on. `ServiceException` refers to service exceptions returned by OSS, such as authentication failures and server errors, among others.

If an exception occurs while an asynchronous request is being processed, it is handled by a callback function.

When an asynchronous interface is called, the function returns a task. You can cancel a task, wait for it to finish, or get the results directly. For example:

```
OSSAsyncTask task = oss.asyncGetObject(...);
task.cancel(); // Cancel the task
task.waitForFinished(); // Wait until the task is finished
GetObjectResult result = task.getResult(); // Block other requests and wait for the returned results
```

 **Note** The interface supports both the synchronous and asynchronous call methods. For simplicity, this document only provides examples of how to call important interfaces in synchronous and asynchronous modes. For other interfaces, only asynchronous call examples are provided.

Set whether to enable the DNS settings

```
ClientConfiguration conf = new ClientConfiguration();
conf.setHttpDnsEnable(true); // The default value is "true", meaning enable. To disable the feature, set the value to "false".
```

Set the custom user-agent

```
ClientConfiguration conf = new ClientConfiguration();
// The set ua is added to the end of the default ua of the SDK by default. Use the last ua value.
conf.setUserAgentMark("customUserAgent");
```

8.4. Quick start

This topic describes how to use Object Storage Service (OSS) SDK for Android to perform common OSS operations, such as creating a bucket, uploading an object, and downloading an object.

You can also use the sample project in the following two methods:

- View the sample directory which includes sample projects of uploading local files, downloading objects, resumable upload, and setting callback. For more information, visit [.](#)

- Run the git clone command to clone the [project](#).

Before running the project, you must configure the Config parameter. The following code provides an example on how to configure this parameter:

```
public class Config {
    // To run the sample correctly, the following variables must have valid values.
    // The endpoint value below is just the example. Please use proper value according to y
    our region
    // Specify the endpoint that you want to access.
    public static final String OSS_ENDPOINT = "http://oss-cn-shanghai.aliyuncs.com";
    // Specify the URL used to test the callback function.
    public static final String OSS_CALLBACK_URL = "http://oss-demo.aliyuncs.com:23450";
    // Specify the address of the authentication server.
    // You can also start the local STS authentication server by running the local authenti
    cation script in the sts_local_server directory of the project.
    public static final String STS_SERVER_URL = "http://****/sts/getsts";// Specify the URL
    of the STS authentication server.
    public static final String BUCKET_NAME = "Input the bucket name";
    public static final String OSS_ACCESS_KEY_ID = "<yourAccessKeyId>";
    public static final String OSS_ACCESS_KEY_SECRET = "<yourAccessKeySecret>";
    public static final int DOWNLOAD_SUC = 1;
    public static final int DOWNLOAD_Fail = 2;
    public static final int UPLOAD_SUC = 3;
    public static final int UPLOAD_Fail = 4;
    public static final int UPLOAD_PROGRESS = 5;
    public static final int LIST_SUC = 6;
    public static final int HEAD_SUC = 7;
    public static final int RESUMABLE_SUC = 8;
    public static final int SIGN_SUC = 9;
    public static final int BUCKET_SUC = 10;
    public static final int GET_STS_SUC = 11;
    public static final int MULTIPART_SUC = 12;
    public static final int STS_TOKEN_SUC = 13;
    public static final int FAIL = 9999;
    public static final int REQUESTCODE_AUTH = 10111;
    public static final int REQUESTCODE_LOCALPHOTOS = 10112;
}
```

Note

- For more information about how to configure the URL of the STS authentication server, see [Set up direct data transfer for mobile apps](#).
- For more information about the script in the sts_local_server directory, visit [GitHub](#).

Create a bucket

```
CreateBucketRequest createBucketRequest = new CreateBucketRequest("<bucketName>");
// Set the access control list (ACL) of the bucket to public read. The default bucket ACL i
s private.
createBucketRequest.setBucketACL(CannedAccessControlList.PublicRead);
// Specify the region to which the bucket belongs.
createBucketRequest.setLocationConstraint("oss-cn-hangzhou");
OSSAsyncTask createTask = oss.asyncCreateBucket(createBucketRequest, new OSSCompletedCallba
ck<CreateBucketRequest, CreateBucketResult>() {
    @Override
    public void onSuccess(CreateBucketRequest request, CreateBucketResult result) {
        Log.d("locationConstraint", request.getLocationConstraint());
    }
    @Override
    public void onFailure(CreateBucketRequest request, ClientException clientException, Ser
viceException serviceException) {
        // Handle request exceptions.
        if (clientException != null) {
            // Handle client exceptions, such as network exceptions.
            clientException.printStackTrace();
        }
        if (serviceException != null) {
            // Handle service exceptions.
            Log.e("ErrorCode", serviceException.getErrorCode());
            Log.e("RequestId", serviceException.getRequestId());
            Log.e("HostId", serviceException.getHostId());
            Log.e("RawMessage", serviceException.getRawMessage());
        }
    }
});
```

Upload an object

The following code provides an example on how to upload a specified local file to OSS:

```
// Construct a request to upload the local file.
PutObjectRequest put = new PutObjectRequest("<bucketName>", "<objectName>", "<uploadFilePath>");
// When you upload the local file in asynchronous mode, you can set the progress callback.
put.setProgressCallback(new OSSProgressCallback<PutObjectRequest>() {
    @Override
    public void onProgress(PutObjectRequest request, long currentSize, long totalSize) {
        Log.d("PutObject", "currentSize: " + currentSize + " totalSize: " + totalSize);
    }
});
OSSAsyncTask task = oss.asyncPutObject(put, new OSSCompletedCallback<PutObjectRequest, PutObjectResult>() {
    @Override
    public void onSuccess(PutObjectRequest request, PutObjectResult result) {
        Log.d("PutObject", "UploadSuccess");
        Log.d("ETag", result.getETag());
        Log.d("RequestId", result.getRequestId());
    }
    @Override
    public void onFailure(PutObjectRequest request, ClientException clientException, ServiceException serviceException) {
        // Handle request exceptions.
        if (clientException != null) {
            // Handle client exceptions, such as network exceptions.
            clientException.printStackTrace();
        }
        if (serviceException != null) {
            // Handle service exceptions.
            Log.e("ErrorCode", serviceException.getErrorCode());
            Log.e("RequestId", serviceException.getRequestId());
            Log.e("HostId", serviceException.getHostId());
            Log.e("RawMessage", serviceException.getRawMessage());
        }
    }
});
// task.cancel(); // You can cancel the upload task.
// task.waitForCompletion(); // Wait until the file is uploaded.
```

Download a specified object

The following code provides an example on how to download a specified object to a local file:

```
// Construct an object download request.
GetObjectRequest get = new GetObjectRequest("<bucketName>", "<objectName>");
OSSAsyncTask task = oss.asyncGetObject(get, new OSSCompletedCallback<GetObjectRequest, GetObjectResult>() {
    @Override
    public void onSuccess(GetObjectRequest request, GetObjectResult result) {
        // The request is successful.
        Log.d("asyncGetObject", "DownloadSuccess");
        Log.d("Content-Length", "" + result.getContentLength());
        InputStream inputStream = result.getObjectContent();
        byte[] buffer = new byte[2048];
        int len;
        try {
            while ((len = inputStream.read(buffer)) != -1) {
                // You can add your own code here to process the downloaded data.
            }
        } catch (IOException e) {
            e.printStackTrace();
        }
    }
    @Override
    // If the GetObject request is successful, GetObjectResult that includes an instance of
    // the input stream is returned. Add your own code to process the input stream.
    public void onFailure(GetObjectRequest request, ClientException clientException, ServiceException serviceException) {
        // Handle request exceptions.
        if (clientException != null) {
            // Handle client exceptions, such as network exceptions.
            clientException.printStackTrace();
        }
        if (serviceException != null) {
            // Handle service exceptions.
            Log.e("ErrorCode", serviceException.getErrorCode());
            Log.e("RequestId", serviceException.getRequestId());
            Log.e("HostId", serviceException.getHostId());
            Log.e("RawMessage", serviceException.getRawMessage());
        }
    }
});
// task.cancel(); // You can cancel the upload task.
// task.waitForCompletion(); // Wait until the object is downloaded.
```

 **Note** You must process the returned input stream by running your own code.

8.5. Buckets

8.5.1. Create buckets

A bucket is a container used to store objects in Object Storage Service (OSS). Each object is contained in a bucket. This topic describes how to create a bucket.

Sample code

The following code shows how to create a bucket named `examplebucket`:

```
// Construct a request to create a bucket.
// Specify the name of the bucket.
CreateBucketRequest createBucketRequest = new CreateBucketRequest("examplebucket");
// Set the region in which the bucket is located to the China (Hangzhou) region.
CreateBucketRequest.setLocationConstraint("oss-cn-hangzhou");
// Specify the access control list (ACL) of the bucket.
// CreateBucketRequest.setBucketACL(CannedAccessControlList.Private);
// Specify the storage class of the bucket.
// CreateBucketRequest.setBucketStorageClass(StorageClass.Standard);
// Create the bucket asynchronously.
OSSAsyncTask createTask = oss.asyncCreateBucket(createBucketRequest, new OSSCompletedCallba
ck<CreateBucketRequest, CreateBucketResult>() {
    @Override
    public void onSuccess(CreateBucketRequest request, CreateBucketResult result) {
        Log.d("asyncCreateBucket", "Success");
    }
    @Override
    public void onFailure(CreateBucketRequest request, ClientException clientException, Ser
viceException serviceException) {
        // Handle request exceptions.
        if (clientException != null) {
            // Handle client exceptions, such as network exceptions.
            clientException.printStackTrace();
        }
        if (serviceException != null) {
            // Handle service exceptions.
            Log.e("ErrorCode", serviceException.getErrorCode());
            Log.e("RequestId", serviceException.getRequestId());
            Log.e("HostId", serviceException.getHostId());
            Log.e("RawMessage", serviceException.getRawMessage());
        }
    }
});
```

References

- For more information about the complete sample code that is used to create a bucket, visit [GitHub](#).
- For more information about the API operation that you can call to create a bucket, see [PutBucket](#).

8.5.2. List buckets

A bucket is a container that is used to store objects in Object Storage Service (OSS). Every object is contained in a bucket. You can list all buckets that belong to an Alibaba Cloud account or buckets that meet the specified conditions. The buckets are listed in alphabetical order of the bucket names.

List all buckets that belong to an Alibaba Cloud account


```
// Specify the address of the STS application server. Example: http://example.com.
String stsServer = "http://example.com";
// We recommend that you use OSSAuthCredentialsProvider to make sure that the token is auto
matically updated when the token expires.
OSSCredentialProvider credentialProvider = new OSSAuthCredentialsProvider(stsServer);
OSSClient ossClient = new OSSClient(getApplicationContext(), credentialProvider, null);
ListBucketsRequest request = new ListBucketsRequest();
ossClient.asyncListBuckets(request, new OSSCompletedCallback<ListBucketsRequest, ListBucket
sResult>() {
    @Override
    public void onSuccess(ListBucketsRequest request, ListBucketsResult result) {
        List<OSSBucketSummary> buckets = result.getBuckets();
        for (int i = 0; i < buckets.size(); i++) {
            OSSLog.logDebug("name: " + buckets.get(i).name + " "
                + "location: " + buckets.get(i).location);
        }
    }
    @Override
    public void onFailure(ListBucketsRequest request, ClientException clientException, Serv
iceException serviceException) {
        // Handle request exceptions.
        if (clientException != null) {
            // Handle client-side exceptions such as network errors.
            clientException.printStackTrace();
        }
        if (serviceException != null) {
            // Handle server-side exceptions.
            Log.e("ErrorCode", serviceException.getErrorCode());
            Log.e("RequestId", serviceException.getRequestId());
            Log.e("HostId", serviceException.getHostId());
            Log.e("RawMessage", serviceException.getRawMessage());
        }
    }
});
```

List buckets whose names contain a specified prefix

```
// Specify the address of the STS application server. Example: http://example.com.
String stsServer = "http://example.com";
// We recommend that you use OSSAuthCredentialsProvider to make sure that the token is auto
// matically updated when the token expires.
OSSCredentialProvider credentialProvider = new OSSAuthCredentialsProvider(stsServer);
OSSClient ossClient = new OSSClient(getApplicationContext(), credentialProvider, null);
ListBucketsRequest request = new ListBucketsRequest();
// Specify the prefix. Example: example.
request.setPrefix("example");
ossClient.asyncListBuckets(request, new OSSCompletedCallback<ListBucketsRequest, ListBucket
sResult>() {
    @Override
    public void onSuccess(ListBucketsRequest request, ListBucketsResult result) {
        List<OSSBucketSummary> buckets = result.getBuckets();
        for (int i = 0; i < buckets.size(); i++) {
            OSSLog.logDebug("name: " + buckets.get(i).name + " "
                + "location: " + buckets.get(i).location);
        }
    }
    @Override
    public void onFailure(ListBucketsRequest request, ClientException clientException, Serv
iceException serviceException) {
        // Handle request exceptions.
        if (clientException != null) {
            // Handle client-side exceptions such as network errors.
            clientException.printStackTrace();
        }
        if (serviceException != null) {
            // Handle server-side exceptions.
            Log.e("ErrorCode", serviceException.getErrorCode());
            Log.e("RequestId", serviceException.getRequestId());
            Log.e("HostId", serviceException.getHostId());
            Log.e("RawMessage", serviceException.getRawMessage());
        }
    }
});
```

List buckets whose names are alphabetically after the bucket specified by a marker

```
// Specify the address of the STS application server. Example: http://example.com.
String stsServer = "http://example.com";
// We recommend that you use OSSAuthCredentialsProvider to make sure that the token is auto-
// matically updated when the token expires.
OSSCredentialProvider credentialProvider = new OSSAuthCredentialsProvider(stsServer);
OSSClient ossClient = new OSSClient(getApplicationContext(), credentialProvider, null);
ListBucketsRequest request = new ListBucketsRequest();
// Specify the marker. Example: examplebucket. Buckets whose names are alphabetically after
// the bucket specified by the marker are returned.
request.setMarker("examplebucket");
ossClient.asyncListBuckets(request, new OSSCompletedCallback<ListBucketsRequest, ListBucket
sResult>() {
    @Override
    public void onSuccess(ListBucketsRequest request, ListBucketsResult result) {
        List<OSSBucketSummary> buckets = result.getBuckets();
        for (int i = 0; i < buckets.size(); i++) {
            OSSLog.logDebug("name: " + buckets.get(i).name + " "
                + "location: " + buckets.get(i).location);
        }
    }
    @Override
    public void onFailure(ListBucketsRequest request, ClientException clientException, Serv
iceException serviceException) {
        // Handle request exceptions.
        if (clientException != null) {
            // Handle client-side exceptions such as network errors.
            clientException.printStackTrace();
        }
        if (serviceException != null) {
            // Handle server-side exceptions.
            Log.e("ErrorCode", serviceException.getErrorCode());
            Log.e("RequestId", serviceException.getRequestId());
            Log.e("HostId", serviceException.getHostId());
            Log.e("RawMessage", serviceException.getRawMessage());
        }
    }
});
```

List a specified number of buckets

```
// Specify the address of the STS application server. Example: http://example.com.
String stsServer = "http://example.com";
// We recommend that you use OSSAuthCredentialsProvider to make sure that the token is auto
// matically updated when the token expires.
OSSCredentialProvider credentialProvider = new OSSAuthCredentialsProvider(stsServer);
OSSClient ossClient = new OSSClient(getApplicationContext(), credentialProvider, null);
ListBucketsRequest request = new ListBucketsRequest();
// Set the maximum number of buckets that can be listed each time to 500. The default value
// is 100, and the maximum value is 1000.
request.setMaxKeys(500);
ossClient.asyncListBuckets(request, new OSSCompletedCallback<ListBucketsRequest, ListBucket
sResult>() {
    @Override
    public void onSuccess(ListBucketsRequest request, ListBucketsResult result) {
        List<OSSBucketSummary> buckets = result.getBuckets();
        for (int i = 0; i < buckets.size(); i++) {
            OSSLog.logDebug("name: " + buckets.get(i).name + " "
                + "location: " + buckets.get(i).location);
        }
    }
    @Override
    public void onFailure(ListBucketsRequest request, ClientException clientException, Serv
iceException serviceException) {
        // Handle request exceptions.
        if (clientException != null) {
            // Handle client-side exceptions such as network errors.
            clientException.printStackTrace();
        }
        if (serviceException != null) {
            // Handle server-side exceptions.
            Log.e("ErrorCode", serviceException.getErrorCode());
            Log.e("RequestId", serviceException.getRequestId());
            Log.e("HostId", serviceException.getHostId());
            Log.e("RawMessage", serviceException.getRawMessage());
        }
    }
});
```

8.5.3. Query bucket ACLs

A bucket is a container for objects stored in OSS. Every object is contained in a bucket. This topic describes how to query the ACL of a bucket.

The following table describes the bucket ACLs.

ACL	Description	Value
-----	-------------	-------

ACL	Description	Value
Private	Only the owner or authorized users of this bucket can read and write objects in the bucket. Other users, including anonymous users cannot access the objects in the bucket without authorization.	private
Public read	Only the owner or authorized users of this bucket can write objects in the bucket. Other users, including anonymous users can only read objects in the bucket. We recommend that you exercise with caution when you set the bucket ACL to public read.	public-read
Public read/write	Any users, including anonymous users can read and write objects in the bucket. Exercise caution when set the bucket ACL to public read/write.	public-read-write

The following code provides an example on how to query the ACL of a bucket:

```
GetBucketACLRequest getBucketACLRequest = new GetBucketACLRequest("<bucketName>");
// Query the ACL of the bucket.
OSSAsyncTask getBucketAclTask = oss.asyncGetBucketACL(getBucketACLRequest, new OSSCompleted
Callback<GetBucketACLRequest, GetBucketACLResult>() {
    @Override
    public void onSuccess(GetBucketACLRequest request, GetBucketACLResult result) {
        Log.d("asyncGetBucketACL", "Success!");
        Log.d("BucketAcl", result.getBucketACL());
        Log.d("Owner", result.getBucketOwner());
        Log.d("ID", result.getBucketOwnerID());
    }
    @Override
    public void onFailure(GetBucketACLRequest request, ClientException clientException, Ser
viceException serviceException) {
        // Handle request exceptions.
        if (clientException != null) {
            // Handle client exceptions, such as network exceptions.
            clientException.printStackTrace();
        }
        if (serviceException != null) {
            // Handle service exceptions.
            Log.e("ErrorCode", serviceException.getErrorCode());
            Log.e("RequestId", serviceException.getRequestId());
            Log.e("HostId", serviceException.getHostId());
            Log.e("RawMessage", serviceException.getRawMessage());
        }
    }
});
```

 Note

- Three ACLs are available for a bucket: public read/write, public read, and private.
- Only the owner of a bucket can call the GetBucketAcl API operation on the bucket.
- The returned result contains the bucket ACL, ID of the bucket owner, and the name of the bucket owner (consistent with the ID).
- For more information about how to query the ACL of a bucket, see [GetBucketAcl](#).

8.5.4. Query bucket information

A bucket is a container for objects stored in Object Storage Service (OSS). Every object is contained in a bucket. This topic describes how to query the information about a bucket.

The following code provides an example on how to query the information about a bucket:

```
GetBucketInfoRequest request = new GetBucketInfoRequest("<bucketName>");
// Query the information about the bucket.
OSSAsyncTask task = oss.asyncGetBucketInfo(request, new OSSCompletedCallback<GetBucketInfoRequest, GetBucketInfoResult>() {
    @Override
    public void onSuccess(GetBucketInfoRequest request, GetBucketInfoResult result) {
        OSSLog.logInfo("code: " + result.getStatusCode());
    }
    @Override
    public void onFailure(GetBucketInfoRequest request, ClientException clientException, ServiceException serviceException) {
        // Handle request exceptions.
        if (clientException != null) {
            // Handle client exceptions, such as network exceptions.
            clientException.printStackTrace();
        }
        if (serviceException != null) {
            // Handle service exceptions.
            Log.e("ErrorCode", serviceException.getErrorCode());
            Log.e("RequestId", serviceException.getRequestId());
            Log.e("HostId", serviceException.getHostId());
            Log.e("RawMessage", serviceException.getRawMessage());
        }
    }
});
```

For more information about how to query bucket information, see [GetBucketInfo](#).

8.5.5. Delete buckets

 Note

-
- To delete the parts generated in multipart upload tasks, call [List Multipart Uploads](#) to list all the parts first, and call [Abort Multipart Upload](#) to delete the parts.

```
DeleteBucketRequest deleteBucketRequest = new DeleteBucketRequest("<bucketName>");
// Delete the bucket asynchronously.
OSSAsyncTask deleteBucketTask = oss.asyncDeleteBucket(deleteBucketRequest, new OSSCompleted
Callback<DeleteBucketRequest, DeleteBucketResult>() {
    @Override
    public void onSuccess(DeleteBucketRequest request, DeleteBucketResult result) {
        Log.d("asyncDeleteBucket", "Success!");
    }
    @Override
    public void onFailure(DeleteBucketRequest request, ClientException clientException, Ser
viceException serviceException) {
        // Handle request exceptions.
        if (clientException != null) {
            // Handle client exceptions, such as network exceptions.
            clientException.printStackTrace();
        }
        if (serviceException != null) {
            // Handle service exceptions.
            Log.e("ErrorCode", serviceException.getErrorCode());
            Log.e("RequestId", serviceException.getRequestId());
            Log.e("HostId", serviceException.getHostId());
            Log.e("RawMessage", serviceException.getRawMessage());
        }
    }
});
```

 Notice

- To prevent unexpected data loss, OSS does not allow users to delete non-empty buckets.
- You cannot recover deleted buckets. Exercise caution when you perform this operation.
- For more information about how to delete a bucket, see [DeleteBucket](#).

8.5.6. Manage lifecycle rules

Object Storage Service (OSS) allows you to configure lifecycle rules to delete expired objects and parts or convert the storage class of expired objects to Infrequent Access (IA) or Archive to reduce your storage costs. This topic describes how to manage lifecycle rules for buckets.

Background information

Each lifecycle rule contains the following information:

- Rule ID. It uniquely identifies a rule in a bucket.
- Policy. You can use one of the following methods to configure a policy. Only a single method can be

configured for each bucket.

- Match by prefix: You can create multiple rules to configure different prefixes. Each prefix must be unique.
- Match by bucket: If you select this method, only a single rule can be created.
- Expiration time. You can use one of the following configuration methods to configure the expiration time:
 - Specify a validity period for which to retain objects after they are last modified.
 - Specify an expiration date that objects that are last modified before this date expire.
- Status.

Configure lifecycle rules


```
PutBucketLifecycleRequest request = new PutBucketLifecycleRequest();
request.setBucketName("yourBucketName");
BucketLifecycleRule rule1 = new BucketLifecycleRule();
// Specify the rule ID and the object prefix to match the rule.
rule1.setIdentifier("1");
rule1.setPrefix("A");
//Specify whether to run the lifecycle rule. If this parameter is set to true, OSS runs this rule at specific intervals. If this parameter is set to false, OSS ignores this rule.
rule1.setStatus(true);
// Specify that objects expire 200 days after they are last modified.
rule1.setDays("200");
// Specify that the storage class of objects is converted to Archive 30 days after they are last modified.
rule1.setArchiveDays("30");
// Specify that parts expire three days after they fail to be uploaded.
rule1.setMultipartDays("3");
// Specify that the storage class of objects is converted to IA 15 days after they are last modified.
rule1.setIADays("15");
BucketLifecycleRule rule2 = new BucketLifecycleRule();
rule2.setIdentifier("2");
rule2.setPrefix("B");
rule2.setStatus(true);
rule2.setDays("300");
rule2.setArchiveDays("30");
rule2.setMultipartDays("3");
rule2.setIADays("15");
ArrayList<BucketLifecycleRule> lifecycleRules = new ArrayList<BucketLifecycleRule>();
lifecycleRules.add(rule1);
lifecycleRules.add(rule2);
request.setLifecycleRules(lifecycleRules);
OSSAsyncTask task = oss.asyncPutBucketLifecycle(request, new OSSCompletedCallback<PutBucketLifecycleRequest, PutBucketLifecycleResult>() {
    @Override
    public void onSuccess(PutBucketLifecycleRequest request, PutBucketLifecycleResult result) {
        OSSLog.logInfo("code: "+result.getStatusCode());
    }
    @Override
    public void onFailure(PutBucketLifecycleRequest request, ClientException clientException, ServiceException serviceException) {
        OSSLog.logError("error: "+serviceException.getRawMessage());
    }
});
task.waitForCompletion();
```

For more information, see [PutBucketLifecycle](#).

View lifecycle rules

The following code provides an example on how to query the lifecycle rules configured for the bucket named examplebucket:

```
GetBucketLifecycleRequest request = new GetBucketLifecycleRequest();
request.setBucketName("yourBucketName");
OSSAsyncTask task = oss.asyncGetBucketLifecycle(request, new OSSCompletedCallback<GetBucketLifecycleRequest, GetBucketLifecycleResult>() {
    @Override
    public void onSuccess(GetBucketLifecycleRequest request, GetBucketLifecycleResult result) {
        ArrayList<BucketLifecycleRule> list = result.getLifecycleRules();
        for (BucketLifecycleRule rule : list){
            OSSLog.logInfo("info: " + rule.getIdentifier());
        }
    }
    @Override
    public void onFailure(GetBucketLifecycleRequest request, ClientException clientException, ServiceException serviceException) {
        OSSLog.logError("error: "+serviceException.getRawMessage());
    }
});
task.waitForTaskToFinish();
```

For more information, see [GetBucketLifecycle](#).

Delete lifecycle rules

The following code provides an example on how to delete lifecycle rules configured for the bucket named examplebucket:

```
DeleteBucketLifecycleRequest request = new DeleteBucketLifecycleRequest();
request.setBucketName("yourBucketName");
OSSAsyncTask task = oss.asyncDeleteBucketLifecycle(request, "yourBucketName" new OSSCompletedCallback<DeleteBucketLifecycleRequest, DeleteBucketLifecycleResult>() {
    @Override
    public void onSuccess(DeleteBucketLifecycleRequest request, DeleteBucketLifecycleResult result) {
        OSSLog.logInfo("code : "+result.getStatusCode());
    }
    @Override
    public void onFailure(DeleteBucketLifecycleRequest request, ClientException clientException, ServiceException serviceException) {
        OSSLog.logError("error: "+serviceException.getRawMessage());
    }
});
task.waitForTaskToFinish();
```

For more information, see [DeleteBucketLifecycle](#).

8.5.7. Hotlink protection

This topic describes how to use hot link protection.

To configure hotlinking protection, you need to set the following parameters:

- Referrer Whitelist: the referer whitelist that specifies the domains that are allowed to access OSS

resources.

- **Allow Empty Referer:** indicates whether the Referer field can be left empty in a request. If the Referer field cannot be left empty, users must include the Referer field in the HTTP or HTTPS request header so as to access OSS resources.

For more information about hotlink protection, see [Configure hotlink protection](#).

Configure hotlink protection

The following code provides an example on how to configure a Referer whitelist for a bucket:

```
PutBucketRefererRequest request = new PutBucketRefererRequest();
request.setBucketName("yourBucketName");
// Add a Referer whitelist. You can use an asterisk (*) or a question mark (?) as a wildcard character to set the Referer parameter.
ArrayList<String> referers = new ArrayList<String>();
referers.add("http://example.com");
referers.add("http://www. *.com");
referers.add("http://www.?.com");
request.setReferers(referers);
OSSAsyncTask task = oss.asyncPutBucketReferer(request, new OSSCompletedCallback<PutBucketRefererRequest, PutBucketRefererResult>() {
    @Override
    public void onSuccess(PutBucketRefererRequest request, PutBucketRefererResult result) {
        OSSLog.logInfo("code: " + result.getStatusCode());
    }
    @Override
    public void onFailure(PutBucketRefererRequest request, ClientException clientException, ServiceException serviceException) {
        OSSLog.logError("error: "+serviceException.getRawMessage());
    }
});
task.waitForCompletion();
```

Obtain hotlink protection information

The following code provides an example on how to query a Referer whitelist of a bucket:

```
GetBucketRefererRequest request = new GetBucketRefererRequest();
request.setBucketName("yourBucketName");
OSSAsyncTask task = oss.asyncGetBucketReferer(request, new OSSCompletedCallback<GetBucketRefererRequest, GetBucketRefererResult>() {
    @Override
    public void onSuccess(GetBucketRefererRequest request, GetBucketRefererResult result) {
        // Obtain the Referer whitelist of the bucket.
        ArrayList<String> list = result.getReferers();
        for (String ref : list){
            OSSLog.logInfo("info: " + ref);
        }
    }
    @Override
    public void onFailure(GetBucketRefererRequest request, ClientException clientException,
        ServiceException serviceException) {
        OSSLog.logError("error: "+serviceException.getRawMessage());
    }
});
task.waitForCompletion();
```

Clear hotlink protection configurations

The following code provides an example on how to clear a Referer whitelist of a bucket:

```
PutBucketRefererRequest request = new PutBucketRefererRequest();
request.setBucketName("yourBucketName");
request.setAllowEmpty(true);
// You cannot clear hotlink protection configurations directly. To clear hotlink protection
// configurations, you must create a rule that allows an empty Referer field to replace the existing rule.
ArrayList<String> referers = new ArrayList<String>();
request.setReferers(referers);
OSSAsyncTask task = oss.asyncPutBucketReferer(request, new OSSCompletedCallback<PutBucketRefererRequest, PutBucketRefererResult>() {
    @Override
    public void onSuccess(PutBucketRefererRequest request, PutBucketRefererResult result) {
        OSSLog.logInfo("code: " + result.getStatusCode());
    }
    @Override
    public void onFailure(PutBucketRefererRequest request, ClientException clientException,
        ServiceException serviceException) {
        OSSLog.logError("error: "+serviceException.getRawMessage());
    }
});
task.waitForCompletion();
```

8.5.8. Access logging

You can enable access logging to record bucket access to log files, which are stored in a specified bucket.

The log file format is as follows:

```
<TargetPrefix><SourceBucket>YYYY-mm-DD-HH-MM-SS-UniqueString
```

For more information about access log files, see [Set access logging](#).

Enable access logging

Run the following code to enable bucket access logging:

```
PutBucketLoggingRequest request = new PutBucketLoggingRequest();
//Specify the source bucket for which to enable access logging.
request.setBucketName("<yourSourceBucketName>");
//Specify the destination bucket for storing access logs.
//You can specify the same bucket or different ones to function as the source and destination buckets. The source and destination bucket can both be the same or different buckets.
request.setTargetBucketName("<yourTargetBucketName>");
//Set the directory where access logs are stored.
request.setTargetPrefix("<yourTargetPrefix>");
OSSAsyncTask task = oss.asyncPutBucketLogging(request, new OSSCompletedCallback<PutBucketLoggingRequest, PutBucketLoggingResult>() {
    @Override
    public void onSuccess(PutBucketLoggingRequest request, PutBucketLoggingResult result) {
        OSSLog.logInfo("code::"+result.getStatusCode());
    }
    @Override
    public void onFailure(PutBucketLoggingRequest request, ClientException clientException, ServiceException serviceException) {
        OSSLog.logError("error: "+serviceException.getRawMessage());
    }
});
task.waitForCompletion();
```

For more information about how to enable access logging for a bucket, see [PutBucketLogging](#).

View access logging configurations

Run the following code to view the access logging configurations for a bucket:

```
GetBucketLoggingRequest request = new GetBucketLoggingRequest();
request.setBucketName("<yourSourceBucketName>");
OSSAsyncTask task = oss.asyncGetBucketLogging(request, new OSSCompletedCallback<GetBucketLoggingRequest, GetBucketLoggingResult>() {
    @Override
    public void onSuccess(GetBucketLoggingRequest request, GetBucketLoggingResult result) {
        OSSLog.logInfo("info: " + result.getTargetBucketName()+"*"+result.getTargetPrefix());
    }
    @Override
    public void onFailure(GetBucketLoggingRequest request, ClientException clientException, ServiceException serviceException) {
        OSSLog.logError("error: "+serviceException.getRawMessage());
    }
});
task.waitForCompletion();
```

For more information about how to view access logging configurations of a bucket, see [GetBucketLogging](#).

Disable access logging

Run the following code to disable access logging for a bucket:

```
DeleteBucketLoggingRequest request = new DeleteBucketLoggingRequest();
request.setBucketName("<yourSourceBucketName>");
OSSAsyncTask task = oss.asyncDeleteBucketLogging(request, new OSSCompletedCallback<DeleteBucketLoggingRequest, DeleteBucketLoggingResult>() {
    @Override
    public void onSuccess(DeleteBucketLoggingRequest request, DeleteBucketLoggingResult result) {
        OSSLog.logInfo("code::"+result.getStatusCode());
    }
    @Override
    public void onFailure(DeleteBucketLoggingRequest request, ClientException clientException, ServiceException serviceException) {
        OSSLog.logError("error: "+serviceException.getRawMessage());
    }
});
task.waitForTaskToFinish();
```

For more information about how to disable access logging for a bucket, see [DeleteBucketLogging](#).

8.6. Upload objects

8.6.1. Overview

This topic describes the methods provided by OSS Android SDK to upload objects.

In OSS, an object is the basic unit for data operations. OSS Android SDK provides the following object upload methods:

- **Simple upload**: Supports uploading local files and binary byte[] arrays that are smaller than or equal to 5 GB.
- **Append upload**: Supports uploading files that are smaller than or equal to 5 GB.
- **Resumable upload**: Supports concurrent resumable upload tasks and custom part size. We recommend you use this method when uploading files that are larger than 5 GB and smaller than or equal to 48.8 TB.

 **Note** For more information about the application scenarios for the preceding upload methods, see Upload objects in the OSS Developer Guide.

When uploading a file, you can view the upload progress through the **progress bar**. After a file is uploaded, you can perform **upload callback**.

8.6.2. Simple upload

To perform simple upload, you can use the `PutObject` method to upload a single object. You can use simple upload to upload local files and byte arrays in binary.

Upload a local file

You can upload a local file to Object Storage Service (OSS) in synchronous or asynchronous mode.

- Upload a local file in synchronous mode

The following code provides an example on how to upload a local file named `examplefile.txt` to the `exampleobject.txt` object in the `exampledir` directory of the `examplebucket` bucket in synchronous mode:

```
// Construct a request to upload the local file.
// Specify the name of the bucket such as examplebucket, the full path of the object such
as exampledir/exampleobject.txt, and the full path of the local file such as /storage/emu
lated/0/oss/examplefile.txt.
// The full path of the object cannot contain the bucket name.
PutObjectRequest put = new PutObjectRequest("examplebucket", "exampledir/exampleobject.tx
t", "/storage/emulated/0/oss/examplefile.txt");
// Configure object metadata.
// ObjectMetadata metadata = new ObjectMetadata();
// Specify Content-Type.
// metadata.setContentType("application/octet-stream");
// Specify the MD5 hash that is used for MD5 verification.
// metadata.setContentMD5(BinaryUtil.calculateBase64Md5(uploadFilePath));
// Specify the access control list (ACL) of the object. The default value is private.
// metadata.setHeader("x-oss-object-acl", "private");
// Specify the storage class of the object. By default, the storage class of the object i
s the same as the storage class of the bucket in which the object is stored. In this exam
ple, this parameter is set to Standard.
// metadata.setHeader("x-oss-storage-class", "Standard");
// put.setMetadata(metadata);
try {
    PutObjectResult putResult = oss.putObject(put);
    Log.d("PutObject", "UploadSuccess");
    Log.d("ETag", putResult.getETag());
    Log.d("RequestId", putResult.getRequestId());
} catch (ClientException e) {
    // Handle client-side exceptions such as network errors.
    e.printStackTrace();
} catch (ServiceException e) {
    // Handle server-side exceptions.
    Log.e("RequestId", e.getRequestId());
    Log.e("ErrorCode", e.getErrorCode());
    Log.e("HostId", e.getHostId());
    Log.e("RawMessage", e.getRawMessage());
}
```

For scoped storage of Android 10 and later versions, you can use the URI of a file to upload the file to OSS.

```
// Construct a request to upload the local file.
// Specify the bucket name such as examplebucket and the full path of the object such as
// examplebucket/exampleobject.txt.
// The full path of the object cannot contain the bucket name.
PutObjectRequest put = new PutObjectRequest("examplebucket", "examplebucket/exampleobject.txt",
fileUri);
// Configure object metadata.
// ObjectMetadata metadata = new ObjectMetadata();
// Specify Content-Type.
// metadata.setContentType("text/plain");
// Specify the MD5 hash that is used for MD5 verification.
// metadata.setContentMD5(BinaryUtil.calculateBase64Md5(uploadFilePath));
// put.setMetadata(metadata);
try {
    PutObjectResult putResult = oss.putObject(put);
    Log.d("PutObject", "UploadSuccess");
    Log.d("ETag", putResult.getETag());
    Log.d("RequestId", putResult.getRequestId());
} catch (ClientException e) {
    // Handle client-side exceptions such as network errors.
    e.printStackTrace();
} catch (ServiceException e) {
    // Handle server-side exceptions.
    Log.e("RequestId", e.getRequestId());
    Log.e("ErrorCode", e.getErrorCode());
    Log.e("HostId", e.getHostId());
    Log.e("RawMessage", e.getRawMessage());
}
```

- Upload a local file in asynchronous mode

 **Note** When you use OSS SDK for Android, you can use the synchronous mode to upload a local file to OSS only in the subthread rather than the UI thread. Otherwise, an exception occurs. To upload a local file in the UI thread, use the asynchronous mode to upload the file.

The following code provides an example on how to upload a local file named examplefile.txt to the exampleobject.txt object in the examplebucket directory of the examplebucket bucket in asynchronous mode:


```
// Construct a request to upload the local file.
// Specify the name of the bucket such as examplebucket, the full path of the object such
as exampledir/exampleobject.txt, and the full path of the local file such as /storage/emu
lated/0/oss/examplefile.txt.
// The full path of the object cannot contain the bucket name.
PutObjectRequest put = new PutObjectRequest("examplebucket", "exampledir/exampleobject.tx
t", "/storage/emulated/0/oss/examplefile.txt");
// When you upload the local file in asynchronous mode, you can configure the progress ca
llback.
put.setProgressCallback(new OSSProgressCallback<PutObjectRequest>() {
    @Override
    public void onProgress(PutObjectRequest request, long currentSize, long totalSize) {
        Log.d("PutObject", "currentSize: " + currentSize + " totalSize: " + totalSize);
    }
});
OSSAsyncTask task = oss.asyncPutObject(put, new OSSCompletedCallback<PutObjectRequest, Pu
tObjectResult>() {
    @Override
    public void onSuccess(PutObjectRequest request, PutObjectResult result) {
        Log.d("PutObject", "UploadSuccess");
        Log.d("ETag", result.getETag());
        Log.d("RequestId", result.getRequestId());
    }
    @Override
    public void onFailure(PutObjectRequest request, ClientException clientException, Servi
ceException serviceException) {
        // Handle request exceptions.
        if (clientException != null) {
            // Handle client-side exceptions such as network errors.
            clientException.printStackTrace();
        }
        if (serviceException != null) {
            // Handle server-side exceptions.
            Log.e("ErrorCode", serviceException.getErrorCode());
            Log.e("RequestId", serviceException.getRequestId());
            Log.e("HostId", serviceException.getHostId());
            Log.e("RawMessage", serviceException.getRawMessage());
        }
    }
});
// Cancel the upload task.
// task.cancel();
// Wait until the upload task is completed.
// task.waitForCompletion();
```

For scoped storage of Android 10 and later versions, you can use the URI of a file to upload the file to OSS.

```

// Construct a request to upload the local file.
// Specify the bucket name such as examplebucket and the full path of the object such as
// examplebucket/exampleobject.txt.
// The full path of the object cannot contain the bucket name.
PutObjectRequest put = new PutObjectRequest("examplebucket", "examplebucket/exampleobject.txt",
fileUri);
// When you upload the local file in asynchronous mode, you can configure the progress callback.
put.setProgressCallback(new OSSProgressCallback<PutObjectRequest>() {
    @Override
    public void onProgress(PutObjectRequest request, long currentSize, long totalSize) {
        Log.d("PutObject", "currentSize: " + currentSize + " totalSize: " + totalSize);
    }
});
OSSAsyncTask task = oss.asyncPutObject(put, new OSSCompletedCallback<PutObjectRequest, PutObjectResult>() {
    @Override
    public void onSuccess(PutObjectRequest request, PutObjectResult result) {
        Log.d("PutObject", "UploadSuccess");
        Log.d("ETag", result.getETag());
        Log.d("RequestId", result.getRequestId());
    }
    @Override
    public void onFailure(PutObjectRequest request, ClientException clientException, ServiceException serviceException) {
        // Handle request exceptions.
        if (clientException != null) {
            // Handle client-side exceptions such as network errors.
            clientException.printStackTrace();
        }
        if (serviceException != null) {
            // Handle server-side exceptions.
            Log.e("ErrorCode", serviceException.getErrorCode());
            Log.e("RequestId", serviceException.getRequestId());
            Log.e("HostId", serviceException.getHostId());
            Log.e("RawMessage", serviceException.getRawMessage());
        }
    }
});
// Cancel the upload task.
// task.cancel();
// Wait until the upload task is completed.
// task.waitForCompletion();

```

Upload a byte array in binary

The following code provides an example on how to upload a byte array in binary to the exampleobject.txt object in the examplebucket directory of the examplebucket bucket in asynchronous mode:

```
byte[] uploadData = new byte[100 * 1024];
new Random().nextBytes(uploadData);
// Construct a request to upload the byte array in binary.
// Specify the bucket name such as examplebucket and the full path of the object such as ex
ampledir/exampleobject.txt.
// The full path of the object cannot contain the bucket name.
PutObjectRequest put = new PutObjectRequest("examplebucket", "exampledir/exampleobject.txt"
, uploadData);
try {
    PutObjectResult putResult = oss.putObject(put);
    Log.d("PutObject", "UploadSuccess");
    Log.d("ETag", putResult.getETag());
    Log.d("RequestId", putResult.getRequestId());
} catch (ClientException e) {
    // Handle client-side exceptions such as network errors.
    e.printStackTrace();
} catch (ServiceException e) {
    // Handle server-side exceptions.
    Log.e("RequestId", e.getRequestId());
    Log.e("ErrorCode", e.getErrorCode());
    Log.e("HostId", e.getHostId());
    Log.e("RawMessage", e.getRawMessage());
}
```

References

- For more information about the complete sample code of simple upload, visit [GitHub](#).
- For more information about the API operation that you can call to perform simple upload, see [Put Object](#).

8.6.3. Append upload

You can call the AppendObject operation to append content to existing appendable objects.

 **Note** Objects uploaded by calling the AppendObject operation are append objects. Objects uploaded by calling the PutObject operation are normal objects.

When you upload objects by using append upload, you must configure the correct position from which the append operation starts.

- When you create an append object, set the position from which the append operation starts to 0.
- When you append content to an append object, set the position from which the append operation starts to the current length of the object.

You can obtain the length of the object from the returned message after the append upload operation or by calling HeadObject.

The following code provides an example on how to perform append upload:

```
// Specify the name of the bucket such as examplebucket, the full path of the object such as
// examplebucket/exampleobject.txt, and the full path of the local file such as /storage/emulated/0/oss/examplefile.txt.
// The full path of the object cannot contain the bucket name.
AppendObjectRequest append = new AppendObjectRequest("examplebucket", "examplebucket/exampleobject.txt", "/storage/emulated/0/oss/examplefile.txt");
ObjectMetadata metadata = new ObjectMetadata();
metadata.setContentType("text/plain");
append.setMetadata(metadata);
// Configure the position from which the append operation starts.
append.setPosition(0);
// Configure a callback function.
append.setProgressCallback(new OSSProgressCallback<AppendObjectRequest>() {
    @Override
    public void onProgress(AppendObjectRequest request, long currentSize, long totalSize) {
        Log.d("AppendObject", "currentSize: " + currentSize + " totalSize: " + totalSize);
    }
});
// Perform the append upload operation in the asynchronous mode.
OSSAsyncTask task = oss.asyncAppendObject(append, new OSSCompletedCallback<AppendObjectRequest, AppendObjectResult>() {
    @Override
    public void onSuccess(AppendObjectRequest request, AppendObjectResult result) {
        Log.d("AppendObject", "AppendSuccess");
        Log.d("NextPosition", "" + result.getNextPosition());
    }
    @Override
    public void onFailure(AppendObjectRequest request, ClientException clientException, ServiceException serviceException) {
        // Handle exceptions.
    }
});
```

For partition storage of Android 10 and later versions, you can use the Uniform Resource Identifier (URI) of an object to upload the object to Object Storage Service (OSS).

```
// Specify the bucket name such as examplebucket and the full path of the object such as ex
ampledir/exampleobject.txt.
// The full path of the object cannot contain the bucket name.
AppendObjectRequest append = new AppendObjectRequest("examplebucket", "exampledir/exampleob
ject.txt", fileUri);
ObjectMetadata metadata = new ObjectMetadata();
metadata.setContentType("text/plain");
append.setMetadata(metadata);
// Configure the position from which the append operation starts.
append.setPosition(0);
// Configure a callback function.
append.setProgressCallback(new OSSProgressCallback<AppendObjectRequest>() {
    @Override
    public void onProgress(AppendObjectRequest request, long currentSize, long totalSize) {
        Log.d("AppendObject", "currentSize: " + currentSize + " totalSize: " + totalSize);
    }
});
// Perform the append upload operation in the asynchronous mode.
OSSAsyncTask task = oss.asyncAppendObject(append, new OSSCompletedCallback<AppendObjectRequ
est, AppendObjectResult>() {
    @Override
    public void onSuccess(AppendObjectRequest request, AppendObjectResult result) {
        Log.d("AppendObject", "AppendSuccess");
        Log.d("NextPosition", "" + result.getNextPosition());
    }
    @Override
    public void onFailure(AppendObjectRequest request, ClientException clientException, Serv
iceException serviceException) {
        // Handle exceptions.
    }
});
```

8.6.4. Multipart upload

By using the multipart upload feature provided by OSS, you can split a large object into multiple parts and upload them separately. After all parts are uploaded, call the CompleteMultipartUpload operation to combine these parts into a single object to implement resumable upload.

Multipart upload process

To implement multipart upload, perform the following operations:

1. Initiate a multipart upload task.

You can call the `oss.initMultipartUpload` method to create a unique upload ID in Object Storage Service (OSS).

2. Upload the parts.

You can call the `oss.uploadPart` method to upload parts.

Note

- For parts that are uploaded by a multipart upload task with a specific upload ID, part numbers identify their relative positions in an object. If you upload a part with the same part number as an existing part, the existing part is overwritten by the uploaded part.
- OSS includes the MD5 hash of part data in the ETag header and returns the MD5 hash to the user.
- OSS calculates the MD5 hash of uploaded data and compares the MD5 hash with the MD5 hash calculated by the SDK. If the two hashes are different, the `InvalidDigest` error code is returned.

3. Complete the multipart upload.

After all parts are uploaded, you can call the `oss.CompleteMultipartUpload` method to combine the parts into a complete object.

Complete sample code

The following code provides a complete example that describes the process of multipart upload:

```
// Specify the name of the bucket to which you want to upload the object. Example: examplebucket.
String bucketName = "examplebucket";
// Specify the full path of the object. Example: exampledir/exampleobject.txt. The full path of the object cannot contain the bucket name.
String objectName = "exampledir/exampleobject.txt";
// Specify the full path of the local file that you want to upload to OSS by using multipart upload. Example: /storage/emulated/0/oss/examplefile.txt.
String localFilepath = "/storage/emulated/0/oss/examplefile.txt";
// Initiate a multipart upload task.
InitiateMultipartUploadRequest init = new InitiateMultipartUploadRequest(bucketName, objectName);
InitiateMultipartUploadResult initResult = oss.initMultipartUpload(init);
// Obtain the upload ID, which uniquely identifies the multipart upload task. You can use the upload ID to cancel or query the multipart upload task.
String uploadId = initResult.getUploadId();
// Set the size of a single part. Unit: bytes. Valid values: 100 KB to 5 GB.
int partCount = 100 * 1024;
// Implement the multipart upload task.
List<PartETag> partETags = new ArrayList<>();
for (int i = 1; i < 5; i++) {
    byte[] data = new byte[partCount];
    RandomAccessFile raf = new RandomAccessFile(localFilepath, "r");
    long skip = (i-1) * partCount;
    raf.seek(skip);
    raf.readFully(data, 0, partCount);
    UploadPartRequest uploadPart = new UploadPartRequest();
    uploadPart.setBucketName(bucketName);
    uploadPart.setObjectKey(objectName);
    uploadPart.setUploadId(uploadId);
    // Set the part number for each part. The number starts from 1. Each part to upload has a part number. Valid values: 1 to 10000.
    uploadPart.setPartNumber(i);
```

```

        uploadPart.setPartContent(data);
        try {
            UploadPartResult result = oss.uploadPart(uploadPart);
            PartETag partETag = new PartETag(uploadPart.getPartNumber(), result.getETag());
            partETags.add(partETag);
        } catch (ServiceException serviceException) {
            OSSLog.logError(serviceException.getErrorCode());
        }
    }
    Collections.sort(partETags, new Comparator<PartETag>() {
        @Override
        public int compare(PartETag lhs, PartETag rhs) {
            if (lhs.getPartNumber() < rhs.getPartNumber()) {
                return -1;
            } else if (lhs.getPartNumber() > rhs.getPartNumber()) {
                return 1;
            } else {
                return 0;
            }
        }
    });
    // Complete the multipart upload task.
    CompleteMultipartUploadRequest complete = new CompleteMultipartUploadRequest(bucketName, ob
jectName, uploadId, partETags);
    // Configure upload callback. You can set the CALLBACK_SERVER parameter when you complete t
he multipart upload request. A callback request is sent to the specified server address aft
er the multipart upload task is completed. You can view the servercallback result in comple
teResult.getServerCallbackReturnBody() of the response.
    complete.setCallbackParam(new HashMap<String, String>() {
        {
            put("callbackUrl", CALLBACK_SERVER); // Set the CALLBACK_SERVER parameter to your s
erver address.
            put("callbackBody", "test");
        }
    });
    CompleteMultipartUploadResult completeResult = oss.completeMultipartUpload(complete);
    OSSLog.logError("----- serverCallback: " + completeResult.getServerCallbackReturnB
ody());

```

The preceding code uses `uploadPart` to upload each part.

- You must specify the upload ID and part numbers for each multipart upload request. Valid values of `PartNumber` range from 1 to 10000. If a part number is not within this range, the `InvalidArgument` error code is returned.
- If you use `uploadPart`, each part except the last part must be larger than 100 KB in size. When you use `uploadPart`, the size of each part is verified only after all parts are uploaded.
- Each time you upload a part, make sure that the stream is directed to the start position of the part to upload.
- Each time a part is uploaded, OSS returns a response that contains the ETag value of the part. If the ETag value is the same as the MD5 hash, combine the ETag value with the part number into a `PartETag` and save the `PartETag` for subsequent uploads of parts.

Upload a local file by using multipart upload

You can upload local files to OSS by using multipart upload in synchronous mode.

Note The preceding complete sample code is used to implement a multipart upload task by following the multipart upload process. The complete sample code is encapsulated to upload a local file by using multipart upload. This way, you need only to use `MultipartUploadRequest` to implement a multipart upload task for a local file.

- Upload a local file by using multipart upload in synchronous mode

The following code provides an example on how to upload a local file named `examplefile.txt` to the `exampleobject.txt` object in the `exampledir` directory of the `examplebucket` bucket by using multipart upload in synchronous mode:

```
// Specify the name of the bucket to which you want to upload the object. Example: examplebucket.
String bucketName = "examplebucket";
// Specify the full path of the object. Example: exampledir/exampleobject.txt. The full path of the object cannot contain the bucket name.
String objectName = "exampledir/exampleobject.txt";
// Specify the full path of the local file that you want to upload to OSS by using multipart upload. Example: /storage/emulated/0/oss/examplefile.txt.
String localFilepath = "/storage/emulated/0/oss/examplefile.txt";
ObjectMetadata meta = new ObjectMetadata();
// Configure object metadata.
meta.setHeader("x-oss-object-acl", "public-read-write");
MultipartUploadRequest rq = new MultipartUploadRequest(bucketName, objectName, localFilepath, meta);
// Set PartSize. The default value of PartSize is 256 KB. The minimum value is 100 KB.
rq.setPartSize(1024 * 1024);
rq.setProgressCallback(new OSSProgressCallback<MultipartUploadRequest>() {
    @Override
    public void onProgress(MultipartUploadRequest request, long currentSize, long totalSize) {
        OSSLog.logDebug("[testMultipartUpload] - " + currentSize + " " + totalSize, false);
    }
});
CompleteMultipartUploadResult result = oss.multipartUpload(rq);
```

For partition storage of Android 10 and later, you can use the Uniform Resource Identifier (URI) of an object to upload the object to OSS.


```
// Specify the name of the bucket to which you want to upload the object. Example: examplebucket.
String bucketName = "examplebucket";
// Specify the full path of the object. Example: exampledir/exampleobject.txt. The full path of the object cannot contain the bucket name.
String objectName = "exampledir/exampleobject.txt";
ObjectMetadata meta = new ObjectMetadata();
// Configure object metadata.
meta.setHeader("x-oss-object-acl", "public-read-write");
MultipartUploadRequest rq = new MultipartUploadRequest(bucketName, objectName, fileUri, meta);
// Set PartSize. The default value of PartSize is 256 KB. The minimum value is 100 KB.
rq.setPartSize(1024 * 1024);
rq.setProgressCallback(new OSSProgressCallback<MultipartUploadRequest>() {
    @Override
    public void onProgress(MultipartUploadRequest request, long currentSize, long totalSize) {
        OSSLog.logDebug("[testMultipartUpload] - " + currentSize + " " + totalSize, false);
    }
});
CompleteMultipartUploadResult result = oss.multipartUpload(rq);
```

- Upload a local file by using multipart upload in asynchronous mode

The following code provides an example on how to upload a local file named `examplefile.txt` to the `exampleobject.txt` object in the `exampledir` directory of the `examplebucket` bucket by using multipart upload in asynchronous mode:

```
// Specify the name of the bucket to which you want to upload the object. Example: examplebucket.
String bucketName = "examplebucket";
// Specify the full path of the object. Example: exampledir/exampleobject.txt. The full path of the object cannot contain the bucket name.
String objectName = "exampledir/exampleobject.txt";
// Specify the full path of the local file that you want to upload to OSS by using multipart upload. Example: /storage/emulated/0/oss/examplefile.txt.
String localFilepath = "/storage/emulated/0/oss/examplefile.txt";
MultipartUploadRequest request = new MultipartUploadRequest(bucketName, objectName, localFilepath);
request.setProgressCallback(new OSSProgressCallback<MultipartUploadRequest>() {
    @Override
    public void onProgress(MultipartUploadRequest request, long currentSize, long totalSize) {
        OSSLog.logDebug("[testMultipartUpload] - " + currentSize + " " + totalSize, false);
    }
});
OSSAsyncTask task = oss.asyncMultipartUpload(request, new OSSCompletedCallback<MultipartUploadRequest, CompleteMultipartUploadResult>() {
    @Override
    public void onSuccess(MultipartUploadRequest request, CompleteMultipartUploadResult result) {
        OSSLog.logInfo(result.getServerCallbackReturnBody());
    }
    @Override
    public void onFailure(MultipartUploadRequest request, ClientException clientException, ServiceException serviceException) {
        OSSLog.logError(serviceException.getRawMessage());
    }
});
//Thread.sleep(100);
// Cancel the multipart upload task.
//task.cancel();
task.waitForTaskToFinish();
```

For partition storage of Android 10 and later, you can use the URI of an object to upload the object to OSS.

```
// Specify the name of the bucket to which you want to upload the object. Example: examplebucket.
String bucketName = "examplebucket";
// Specify the full path of the object. Example: exampledir/exampleobject.txt. The full path of the object cannot contain the bucket name.
String objectName = "exampledir/exampleobject.txt";
MultipartUploadRequest request = new MultipartUploadRequest(bucketName, objectName, fileUri);
request.setProgressCallback(new OSSProgressCallback<MultipartUploadRequest>() {
    @Override
    public void onProgress(MultipartUploadRequest request, long currentSize, long totalSize) {
        OSSLog.logDebug("[testMultipartUpload] - " + currentSize + " " + totalSize, false);
    }
});
OSSAsyncTask task = oss.asyncMultipartUpload(request, new OSSCompletedCallback<MultipartUploadRequest, CompleteMultipartUploadResult>() {
    @Override
    public void onSuccess(MultipartUploadRequest request, CompleteMultipartUploadResult result) {
        OSSLog.logInfo(result.getServerCallbackReturnBody());
    }
    @Override
    public void onFailure(MultipartUploadRequest request, ClientException clientException, ServiceException serviceException) {
        OSSLog.logError(serviceException.getRawMessage());
    }
});
//Thread.sleep(100);
// Cancel the multipart upload task.
//task.cancel();
task.waitForCompletion();
```

List uploaded parts

You can call the `oss.listParts` method to obtain all the parts that are uploaded in a multipart upload task.

The following code provides an example on how to list uploaded parts:

```
// Specify the name of the bucket to which the parts are uploaded. Example: examplebucket.
String bucketName = "examplebucket";
// Specify the full path of the object. Example: exampledir/exampleobject.txt. The full path of the object cannot contain the bucket name.
String objectName = "exampledir/exampleobject.txt";
// Specify the upload ID.
String uploadId = "0004B999EF518A1FE585B0C9360D";
// List uploaded parts.
ListPartsRequest listParts = new ListPartsRequest(bucketName, objectName, uploadId);
ListPartsResult result = oss.listParts(listParts);
List<PartETag> partETagList = new ArrayList<PartETag>();
for (PartSummary part : result.getParts()) {
    partETagList.add(new PartETag(part.getPartNumber(), part.getETag()));
}
```



Notice By default, if a bucket contains more than 1,000 parts that are uploaded by using multipart upload, OSS returns the first 1,000 parts. In the response, the `IsTruncated` field is set to false and `NextPartNumberMarker` is returned to indicate the next starting position to continue reading data.

Cancel a multipart upload task

You can call the `ossClient.abortMultipartUpload` method to cancel a multipart upload task. If you cancel a multipart upload task, you cannot use the upload ID to upload parts. The uploaded parts are deleted.

The following code provides an example on how to cancel a multipart upload task:

```
// Specify the name of the bucket to which the parts are uploaded. Example: examplebucket.
String bucketName = "examplebucket";
// Specify the full path of the object. Example: exampledir/exampleobject.txt. The full path of the object cannot contain the bucket name.
String objectName = "exampledir/exampleobject.txt";
// Specify the upload ID.
String uploadId = "0004B999EF518A1FE585B0C9360D";
// Cancel the multipart upload task.
AbortMultipartUploadRequest abort = new AbortMultipartUploadRequest(bucketName, objectName, uploadId);
AbortMultipartUploadResult abortResult = oss.abortMultipartUpload(abort);
```

8.6.5. Resumable upload

In a wireless network environment, it takes a long time to upload a large object and the upload may fail due to poor network connectivity or network switchover. If the upload fails, the entire object must be uploaded again. To resolve this issue, Object Storage Service (OSS) SDK for Android provides the resumable upload feature.

Background information

We recommend that you do not use resumable upload when you upload objects smaller than 5 GB in size from a mobile device. Resumable upload is implemented by using the multipart upload method. Resumable upload of a single object requires multiple network requests, which is inefficient. The following section provides the precautions you need to understand before and during resumable upload.

- Before resumable upload

Before you upload an object to OSS by using the resumable upload method, you can specify a folder for the checkpoint file that stores resumable upload records. The checkpoint file applies only to the current resumable upload task.

- If you do not specify the folder for the checkpoint file and part of a large object fails to be uploaded due to network issues, it takes a large amount of time and consumes a large amount of traffic to upload the entire large object again.
- If you specify the folder for the checkpoint file, a failed resumable upload task can be resumed from the position recorded in the checkpoint file.

- During resumable upload

When you use resumable upload, you must understand the following information:

- Resumable upload allows you to upload only local files. Resumable upload supports upload callback. You can use resumable upload in a similar way you use upload callback. For more information, see [Callback](#).
- Resumable upload is implemented by calling the following API operations: `InitMultipartUpload`, `UploadPart`, `ListParts`, `CompleteMultipartUpload`, and `AbortMultipartUpload`. If you want to use the STS authentication mode to perform resumable upload tasks, make sure that you are authorized to call the preceding API operations.
- By default, the MD5 hash of each part is verified in a resumable upload task. Therefore, you do not need to specify the `Content-Md5` header in the request.
- When a resumable upload task fails, the uploaded parts become unnecessary in OSS. To resolve this issue, you can configure lifecycle rules for the bucket that contains the parts to clear the parts on a regular basis. For more information, see [Configure lifecycle rules](#).

Examples

You can use resumable upload to upload a local file to OSS in synchronous or asynchronous mode.

- Call the synchronous API operation to upload a local file by using resumable upload

The following code provides an example on how to use resumable upload in synchronous mode to upload a local file named `examplefile.txt` to the `exampledir` directory in `examplebucket`. After the local file is uploaded, the object is named `exampleobject.txt`. The checkpoint file is stored in your computer.

```
// Specify the name of the bucket. Example: examplebucket.
String bucketName = "examplebucket";
// Specify the full path of the object. Example: exampledir/exampleobject.txt. The full path of the object cannot contain bucket names.
String objectName = "exampledir/exampleobject.txt";
// Specify the full path of the local file. Example: /storage/emulated/0/oss/examplefile.txt.
String localFilepath = "/storage/emulated/0/oss/examplefile.txt";
String recordDirectory = Environment.getExternalStorageDirectory().getAbsolutePath() + "/oss_record/";
File recordDir = new File(recordDirectory);
// Make sure that the folder for the checkpoint file exists. If the folder does not exist, create a folder.
if (!recordDir.exists()) {
    recordDir.mkdirs();
}
// Create a resumable upload request and specify the absolute path of the checkpoint file.
ResumableUploadRequest request = new ResumableUploadRequest(bucketName, objectName, localFilepath, recordDirectory);
// Set the DeleteUploadOnCancelling parameter when the OSSAsyncTask cancel() method is called. When this parameter is set to false, the checkpoint file is retained. If you do not specify this parameter, default value true is used, which indicates that the checkpoint file is deleted. The entire object is uploaded again during the next upload.
request.setDeleteUploadOnCancelling(false);
// Configure upload callback.
request.setProgressCallback(new OSSProgressCallback<ResumableUploadRequest>() {
    @Override
    public void onProgress(ResumableUploadRequest request, long currentSize, long totalSize) {
        Log.d("resumableUpload", "currentSize: " + currentSize + " totalSize: " + totalSize);
    }
});
ResumableUploadResult uploadResult = oss.resumableUpload(request);
```

If you use Android 10 or later for partitioned storage, you can use the Uniform Resource Identifier (URI) of the local file to upload the file to OSS.

```
// Specify the name of the bucket. Example: examplebucket.
String bucketName = "examplebucket";
// Specify the full path of the object. Example: exampledir/exampleobject.txt. The full p
ath of the object cannot contain bucket names.
String objectName = "exampledir/exampleobject.txt";
String recordDirectory = getApplication().getFilesDir().getAbsolutePath() + "/oss_record/
";
File recordDir = new File(recordDirectory);
// Make sure that the folder for the checkpoint file exists. If the folder does not exist
, create a folder.
if (!recordDir.exists()) {
    recordDir.mkdirs();
}
// Create a resumable upload request and specify the absolute path of the checkpoint file
.
ResumableUploadRequest request = new ResumableUploadRequest(bucketName, objectName, fileU
ri, recordDirectory);
// Set the DeleteUploadOnCancelling parameter when the OSSAsyncTask cancel() method is ca
lled. When this parameter is set to false, the checkpoint file is retained. If you do not
specify this parameter, default value true is used, which indicates that the checkpoint f
ile is deleted. The entire object is uploaded again during the next upload.
request.setDeleteUploadOnCancelling(false);
// Configure upload callback.
request.setProgressCallback(new OSSProgressCallback<ResumableUploadRequest>() {
    @Override
    public void onProgress(ResumableUploadRequest request, long currentSize, long totalSi
ze) {
        Log.d("resumableUpload", "currentSize: " + currentSize + " totalSize: " + totalSi
ze);
    }
});
ResumableUploadResult uploadResult = oss.resumableUpload(request);
```

The following code provides an example on how to implement resumable upload without storing the checkpoint file in your computer:

```
// Specify the name of the bucket. Example: examplebucket.
String bucketName = "examplebucket";
// Specify the full path of the object. Example: exampledir/exampleobject.txt. The full path of the object cannot contain bucket names.
String objectName = "exampledir/exampleobject.txt";
// Specify the full path of the local file. Example: /storage/emulated/0/oss/examplefile.txt.
String localFilepath = "/storage/emulated/0/oss/examplefile.txt";
// Create a resumable upload request.
ResumableUploadRequest request = new ResumableUploadRequest(bucketName, objectName, localFilepath);
// Set the DeleteUploadOnCancelling parameter when the OSSAsyncTask cancel() method is called. When this parameter is set to false, the checkpoint file is retained. If you do not specify this parameter, default value true is used, which indicates that the checkpoint file is deleted. The entire object is uploaded again during the next upload.
request.setDeleteUploadOnCancelling(false);
// Configure upload callback.
request.setProgressCallback(new OSSProgressCallback<ResumableUploadRequest>() {
    @Override
    public void onProgress(ResumableUploadRequest request, long currentSize, long totalSize) {
        Log.d("resumableUpload", "currentSize: " + currentSize + " totalSize: " + totalSize);
    }
});
ResumableUploadResult uploadResult = oss.resumableUpload(request);
```

- Call the asynchronous API operation to upload a local file by using resumable upload

The following code provides an example on how to use resumable upload in asynchronous mode to upload a local file named `examplefile.txt` to the `exampledir` directory in `examplebucket`. After the local file is uploaded, the object is named `exampleobject.txt`. The checkpoint file is stored in your computer.


```
// Specify the name of the bucket. Example: examplebucket.
String bucketName = "examplebucket";
// Specify the full path of the object. Example: exampledir/exampleobject.txt. The full path of the object cannot contain bucket names.
String objectName = "exampledir/exampleobject.txt";
// Specify the full path of the local file. Example: /storage/emulated/0/oss/examplefile.txt.
String localFilepath = "/storage/emulated/0/oss/examplefile.txt";
String recordDirectory = Environment.getExternalStorageDirectory().getAbsolutePath() + "/oss_record/";
File recordDir = new File(recordDirectory);
// Make sure that the folder for the checkpoint file exists. If the folder does not exist, create a folder.
if (!recordDir.exists()) {
    recordDir.mkdirs();
}
// Create a resumable upload request and specify the absolute path of the checkpoint file.
ResumableUploadRequest request = new ResumableUploadRequest(bucketName, objectName, localFilepath, recordDirectory);
// Configure upload callback.
request.setProgressCallback(new OSSProgressCallback<ResumableUploadRequest>() {
    @Override
    public void onProgress(ResumableUploadRequest request, long currentSize, long totalSize) {
        Log.d("resumableUpload", "currentSize: " + currentSize + " totalSize: " + totalSize);
    }
});
OSSAsyncTask resumableTask = oss.asyncResumableUpload(request, new OSSCompletedCallback<ResumableUploadRequest, ResumableUploadResult>() {
    @Override
    public void onSuccess(ResumableUploadRequest request, ResumableUploadResult result) {
        Log.d("resumableUpload", "success!");
    }
    @Override
    public void onFailure(ResumableUploadRequest request, ClientException clientException, ServiceException serviceException) {
        // Handle exceptions.
    }
});
// Wait until the resumable upload task is complete.
resumableTask.waitForCompletion();
```

If you use Android 10 or later for partitioned storage, you can use the URI of the local file to upload the file to OSS.

```

// Specify the name of the bucket. Example: examplebucket.
String bucketName = "examplebucket";
// Specify the full path of the object. Example: exampledir/exampleobject.txt. The full p
ath of the object cannot contain bucket names.
String objectName = "exampledir/exampleobject.txt";
String recordDirectory = getApplication().getFilesDir().getAbsolutePath() + "/oss_record/
";
File recordDir = new File(recordDirectory);
// Make sure that the folder for the checkpoint file exists. If the folder does not exist
, create a folder.
if (!recordDir.exists()) {
    recordDir.mkdirs();
}
// Create a resumable upload request and specify the absolute path of the checkpoint file
.
ResumableUploadRequest request = new ResumableUploadRequest(bucketName, objectName, fileU
ri, recordDirectory);
// Set the DeleteUploadOnCancelling parameter when the OSSAsyncTask cancel() method is ca
lled. When this parameter is set to false, the checkpoint file is retained. If you do not
specify this parameter, default value true is used, which indicates that the checkpoint f
ile is deleted. The entire object is uploaded again during the next upload.
request.setDeleteUploadOnCancelling(false);
// Configure upload callback.
request.setProgressCallback(new OSSProgressCallback<ResumableUploadRequest>() {
    @Override
    public void onProgress(ResumableUploadRequest request, long currentSize, long totalSi
ze) {
        Log.d("resumableUpload", "currentSize: " + currentSize + " totalSize: " + totalSi
ze);
    }
});
OSSAsyncTask resumableTask = oss.asyncResumableUpload(request, new OSSCompletedCallback<R
esumableUploadRequest, ResumableUploadResult>() {
    @Override
    public void onSuccess(ResumableUploadRequest request, ResumableUploadResult result) {
        Log.d("resumableUpload", "success!");
    }
    @Override
    public void onFailure(ResumableUploadRequest request, ClientException clientExcepion,
ServiceException serviceException) {
        // Handle exceptions.
    }
});
// Wait until the resumable upload task is complete.
resumableTask.waitForCompletion();

```

The following code provides an example on how to implement resumable upload without storing the checkpoint file in your computer:

```
// Specify the name of the bucket. Example: examplebucket.
String bucketName = "examplebucket";
// Specify the full path of the object. Example: exampledir/exampleobject.txt. The full path of the object cannot contain bucket names.
String objectName = "exampledir/exampleobject.txt";
// Specify the full path of the local file. Example: /storage/emulated/0/oss/examplefile.txt.
String localFilepath = "/storage/emulated/0/oss/examplefile.txt";
// Create a resumable upload request.
ResumableUploadRequest request = new ResumableUploadRequest(bucketName, objectName, localFilepath);
// Configure upload callback.
request.setProgressCallback(new OSSProgressCallback<ResumableUploadRequest>() {
    @Override
    public void onProgress(ResumableUploadRequest request, long currentSize, long totalSize) {
        Log.d("resumableUpload", "currentSize: " + currentSize + " totalSize: " + totalSize);
    }
});
// Call the asyncResumableUpload method to perform the resumable upload task.
OSSAsyncTask resumableTask = oss.asyncResumableUpload(request, new OSSCompletedCallback<ResumableUploadRequest, ResumableUploadResult>() {
    @Override
    public void onSuccess(ResumableUploadRequest request, ResumableUploadResult result) {
        Log.d("resumableUpload", "success!");
    }
    @Override
    public void onFailure(ResumableUploadRequest request, ClientException clientException, ServiceException serviceException) {
        // Handle exceptions.
    }
});
// Wait until the resumable upload task is complete.
resumableTask.waitUntilFinished();
```

8.6.6. Progress bar

This topic describes how to use the progress bar to indicate the upload progress.

The following code shows how to call the `asyncPutObject` method to use the progress bar.

```
// Construct an upload request.
PutObjectRequest put = new PutObjectRequest("<bucketName>", "<objectName>", "<uploadFilePath>");
// Configure the callback function to display the progress bar.
put.setProgressCallback(new OSSProgressCallback<PutObjectRequest>() {
    @Override
    public void onProgress(PutObjectRequest request, long currentSize, long totalSize) {
        Log.d("PutObject", "currentSize: " + currentSize + " totalSize: " + totalSize);
    }
});
// Upload the file asynchronously.
OSSAsyncTask task = oss.asyncPutObject(put, new OSSCompletedCallback<PutObjectRequest, PutObjectResult>() {
    @Override
    public void onSuccess(PutObjectRequest request, PutObjectResult result) {
        Log.d("PutObject", "UploadSuccess");
        Log.d("ETag", result.getETag());
        Log.d("RequestId", result.getRequestId());
    }
    @Override
    public void onFailure(PutObjectRequest request, ClientException clientException, ServiceException serviceException) {
        // A request exception occurs.
        if (clientException != null) {
            // A local exception (such as network exception) occurs.
            clientException.printStackTrace();
        }
        if (serviceException != null) {
            // A service exception occurs.
            Log.e("ErrorCode", serviceException.getErrorCode());
            Log.e("RequestId", serviceException.getRequestId());
            Log.e("HostId", serviceException.getHostId());
            Log.e("RawMessage", serviceException.getRawMessage());
        }
    }
});
// task.cancel(); // You can cancel the upload task.
// task.waitForCompletion(); // Wait until the upload task is complete.
```

8.6.7. Upload callback

After an object is uploaded, Object Storage Service (OSS) can send a callback request to the application server. To configure upload callback, you need to only add related callback parameters to the upload request that is sent to OSS. This topic describes how to configure upload callback.

 **Note** Compared with simple upload, the upload with callback configured requires the client to wait for a longer time to process a callback request and response.

```

PutObjectRequest put = new PutObjectRequest(testBucket, testObject, uploadFilePath);
put.setCallbackParam(new HashMap<String, String>() {
    {
        put("callbackUrl", "192.168.10.106/callback");
        put("callbackHost", "yourCallbackHost");
        put("callbackBodyType", "application/json");
        put("callbackBody", "{\"mimeType\":\"${mimeType}\",\"size\":\"${size}\"}");
    }
});
OSSAsyncTask task = oss.asyncPutObject(put, new OSSCompletedCallback<PutObjectRequest, PutObjectResult>() {
    @Override
    public void onSuccess(PutObjectRequest request, PutObjectResult result) {
        Log.d("PutObject", "UploadSuccess");
        // Obtain the returned callback information. The callback information can be obtained only if servercallback is configured in the request.
        String serverCallbackReturnJson = result.getServerCallbackReturnBody();
        Log.d("servercallback", serverCallbackReturnJson);
    }
    @Override
    public void onFailure(PutObjectRequest request, ClientException clientException, ServiceException serviceException) {
        // Handle exceptions.
    }
});

```

The following code provides an example on how to configure custom parameters:

```

put.setCallbackParam(new HashMap<String, String>() {
    {
        put("callbackUrl", "http://192.168.192.125/leibin/notify.php");
        put("callbackHost", "yourCallbackHost");
        put("callbackBodyType", "application/json");
        put("callbackBody", "{\"object\":\"${object}\",\"size\":\"${size}\",\"my_var1\":\"${x:var1}\",\"my_var2\":\"${x:var2}\"}");
    }
});
put.setCallbackVars(new HashMap<String, String>() {
    {
        put("x:var1", "value1");
        put("x:var2", "value2");
    }
});

```

For more information about upload callback, see [Callback](#).

8.7. Download objects

8.7.1. Overview

OSS Android SDK provides various methods for you to download objects, including:

- [Streaming download](#)
- [Range download](#)

8.7.2. Streaming download

If you have a large object that takes a long time to download in its entirety, you can use streaming download to download the object incrementally until the object is downloaded in full.

You can download an object in the streaming download method and then obtain the input stream of the object. You must have read permissions on the object to perform streaming download.

The following code provides an example on how to call the `GetObjectRequest` method synchronously to perform a streaming download task:

```

// Construct an object download request.
// objectKey is equivalent to objectName and indicates the complete path of the object that
// you want to download from OSS. The path must include the file extension of the object. For
// example, you can set objectKey to abc/efg/123.jpg.
GetObjectRequest get = new GetObjectRequest("<bucketName>", "<objectKey>");
// Configure download callback.
get.setProgressListener(new OSSProgressCallback<GetObjectRequest>() {
    @Override
    public void onProgress(GetObjectRequest request, long currentSize, long totalSize) {
        OSSLog.logDebug("getobj_progress: " + currentSize+" total_size: " + totalSize, false);
    }
});
try {
    // Process the download request synchronously and obtain the result.
    GetObjectResult getResult = oss.getObject(get);
    Log.d("Content-Length", "" + getResult.getContentLength());
    // Obtain the input stream of the object.
    InputStream inputStream = getResult.getObjectContent();
    byte[] buffer = new byte[2048];
    int len;
    while ((len = inputStream.read(buffer)) != -1) {
        // Process the downloaded data. For example, display the image or perform a write operation on the object.
    }
    // View the object metadata.
    ObjectMetadata metadata = getResult.getMetadata();
    Log.d("ContentType", metadata.getContentType());
} catch (ClientException e) {
    // Handle client exceptions, such as network exceptions.
    e.printStackTrace();
} catch (ServiceException ex) {
    // Handle service exceptions.
    Log.e("RequestId", e.getRequestId());
    Log.e("ErrorCode", e.getErrorCode());
    Log.e("HostId", e.getHostId());
    Log.e("RawMessage", e.getRawMessage());
} catch (IOException e) {
    e.printStackTrace();
}

```

The following code provides an example on how to call the `GetObjectRequest` method asynchronously to perform a streaming download task:

```

GetObjectRequest get = new GetObjectRequest("<bucketName>", "<objectKey>");
// Configure download callback.
get.setProgressListener(new OSSProgressCallback<GetObjectRequest>() {
    @Override
    public void onProgress(GetObjectRequest request, long currentSize, long totalSize) {
        OSSLog.logDebug("getobj_progress: " + currentSize+" total_size: " + totalSize, false);
    }
});
OSSAsyncTask task = oss.asyncGetObject(get, new OSSCompletedCallback<GetObjectRequest, GetObjectResult>() {
    @Override
    public void onSuccess(GetObjectRequest request, GetObjectResult result) {
        // The object is downloaded.
        InputStream inputStream = result.getObjectContent();
        byte[] buffer = new byte[2048];
        int len;
        try {
            while ((len = inputStream.read(buffer)) != -1) {
                // Process the downloaded data.
            }
        } catch (IOException e) {
            e.printStackTrace();
        }
    }
    @Override
    public void onFailure(GetObjectRequest request, ClientException clientException, ServiceException serviceException) {
        // Handle request exceptions.
        if (clientException != null) {
            // Handle client exceptions, such as network exceptions.
            clientException.printStackTrace();
        }
        if (serviceException != null) {
            // Handle service exceptions.
            Log.e("ErrorCode", serviceException.getErrorCode());
            Log.e("RequestId", serviceException.getRequestId());
            Log.e("HostId", serviceException.getHostId());
            Log.e("RawMessage", serviceException.getRawMessage());
        }
    }
});

```

8.7.3. Range download

If you need only part of the data in an object, you can use range download to download data within the specified range.

Run the following code to perform a range download task:


```

// objectKey is equivalent to objectName and indicates the complete path of the object that
// you want to download from OSS. The path must include the file extension of the object. For
// example, you can set objectKey to abc/efg/123.jpg.
GetObjectRequest get = new GetObjectRequest("<bucketName>", "<objectKey>");
// Set the download range.
get.setRange(new Range(0, 99)); // Download the 100 bytes of data within the range of Byte
0 to Byte 99. The valid range starts from Byte 0.
// get.setRange(new Range(100, Range.INFINITE)); // Download the data within the range of B
yte 100 to the end of the object.
OSSAsyncTask task = oss.asyncGetObject(get, new OSSCompletedCallback<GetObjectRequest, GetO
bjectResult>() {
    @Override
    public void onSuccess(GetObjectRequest request, GetObjectResult result) {
        // The object is downloaded.
        InputStream inputStream = result.getObjectContent();
        byte[] buffer = new byte[2048];
        int len;
        try {
            while ((len = inputStream.read(buffer)) != -1) {
                // Process the downloaded data
            }
        } catch (IOException e) {
            e.printStackTrace();
        }
    }
    @Override
    public void onFailure(GetObjectRequest request, ClientException clientException, Service
Exception serviceException) {
        // Handle request exceptions.
        if (clientException != null) {
            // Handle client exceptions, such as network exceptions.
            clientException.printStackTrace();
        }
        if (serviceException != null) {
            // Handle service exceptions.
            Log.e("ErrorCode", serviceException.getErrorCode());
            Log.e("RequestId", serviceException.getRequestId());
            Log.e("HostId", serviceException.getHostId());
            Log.e("RawMessage", serviceException.getRawMessage());
        }
    }
});

```

8.7.4. Download objects to local files

This topic describes how to download an object to a local file.

The following code provides an example on how to download a specified object to a local file:

```

// Set BucketName to the name of the bucket.
// Set the ObjectKey parameter. ObjectKey indicates the full path of the object that you want to download from Object Storage Service (OSS). The path must include the extension of the object. Example: exampledir/exampleobject.jpg.
GetObjectRequest get = new GetObjectRequest("BucketName", "ObjectKey");
oss.asyncGetObject(get, new OSSCompletedCallback<GetObjectRequest, GetObjectResult>() {
    @Override
    public void onSuccess(GetObjectRequest request, GetObjectResult result) {
// Read the object content.
        long length = result.getContentLength();
        if (length > 0) {
            byte[] buffer = new byte[(int) length];
            int readCount = 0;
            while (readCount < length) {
                try{
                    readCount += result.getObjectContent().read(buffer, readCount, (int) length - readCount);
                }catch (Exception e){
                    OSSLog.logInfo(e.toString());
                }
            }
            // Store the downloaded object as a local file in a specified path. Example: D:\\localpath\\exampleobject.jpg.
            try {
                FileOutputStream fout = new FileOutputStream("download_filePath");
                fout.write(buffer);
                fout.close();
            } catch (Exception e) {
                OSSLog.logInfo(e.toString());
            }
        }
    }
    @Override
    public void onFailure(GetObjectRequest request, ClientException clientException, ServiceException serviceException) {
    }
});

```

8.7.5. Conditional download

You can configure one or more conditions for downloads. If the specified conditions are met, the object can be downloaded. If the specified conditions are not met, an error code is returned and the object cannot be downloaded.

You can configure the following conditions:

Parameter	Description
If-Modified-Since	If the specified time is earlier than the time the object is modified, the object can be downloaded. Otherwise, an error (304 Not modified) is returned.

Parameter	Description
If-Unmodified-Since	If the specified time is later than or equal to the time the object is modified, the object can be downloaded. Otherwise, an error (412 Precondition failed) is returned.
If-Match	If the specified ETag matches that of an object, the object can be downloaded. Otherwise, an error (412 Precondition failed) is returned.
If-None-Match	If the specified ETag does not match that of an object, the object can be downloaded. Otherwise, an error (304 Not modified) is returned.

Run the following code for conditional download:

```
public void testGetObjectWithHeaders() throws Exception {
    GetObjectRequest request = new GetObjectRequest(bucketName, objectName);
    request.setRequestHeaders(...);
    OSSAsyncTask task = oss.asyncGetObject(request, getCallback);
    task.waitForCompletion();
    GetObjectRequest rq = getCallback.request;
    GetObjectResult result = getCallback.result;
}
```

8.7.6. Progress bars

You can use a progress bar to indicate the progress of a task initiated to upload or download an object. `GetObject` is used in the example to describe how to display the progress bar of an object download task.

The following code provides an example on how to display the progress bar of the task initiated to download the `exampleobject.txt` object from `examplebucket`:

```
// Construct a download request.
// Specify the name of the bucket and the full path of the object. The full path of the object cannot contain bucket names.
GetObjectRequest get = new GetObjectRequest("examplebucket", "exampleobject.txt");
// Set the callback function for the progress bar and display the progress bar.
get.setProgressListener(new OSSProgressCallback<GetObjectRequest>() {
    @Override
    public void onProgress(GetObjectRequest request, long currentSize, long totalSize) {
        Log.d("GetObject", "currentSize: " + currentSize + " totalSize: " + totalSize);
    }
});
// Asynchronously download the object.
OSSAsyncTask task = oss.asyncGetObject(get, new OSSCompletedCallback<GetObjectRequest, GetObjectResult>() {
    @Override
    public void onSuccess(GetObjectRequest request, GetObjectResult result) {
        Log.d("GetObject", "UploadSuccess");
    }
    @Override
    public void onFailure(GetObjectRequest request, ClientException clientException, ServiceException serviceException) {
        // Request exceptions.
        if (clientException != null) {
            // Client-side exceptions such as network errors.
            clientException.printStackTrace();
        }
        if (serviceException != null) {
            // Server-side exceptions.
            Log.e("ErrorCode", serviceException.getErrorCode());
            Log.e("RequestId", serviceException.getRequestId());
            Log.e("HostId", serviceException.getHostId());
            Log.e("RawMessage", serviceException.getRawMessage());
        }
    }
});
```

8.8. Manage objects

8.8.1. Overview

You can manage objects in a bucket by calling APIs to perform the following operations:

- **Determine whether an object exists**
- **Obtain Object Meta**
- **List objects**
- **Copy objects**
- **Delete objects**

8.8.2. Determine whether a specified object exists

OSS provides a synchronous interface for you to check whether a specified object exists in a bucket.

Run the following code to determine whether a specified object exists:

```
try {
    if (oss.doesObjectExist("<bucketName>", "<objectKey>")) {
        Log.d("doesObjectExist", "object exist.");
    } else {
        Log.d("doesObjectExist", "object does not exist.");
    }
} catch (ClientException e) {
    // Catch local exceptions, such as network exceptions.
    e.printStackTrace();
} catch (ServiceException e) {
    // Catch service exceptions.
    Log.e("ErrorCode", e.getErrorCode());
    Log.e("RequestId", e.getRequestId());
    Log.e("HostId", e.getHostId());
    Log.e("RawMessage", e.getRawMessage());
}
```

8.8.3. Query the ACL of an object

Object access control lists (ACLs) include private, public read, and public read/write. This topic describes how to query the ACL of an object.

Object ACLs

The following table describes the three types of object ACLs.

Object ACL	Description
Private	Only the owner or authorized users of the bucket can read and write the object.
Public read	Only the object owner or authorized users can write the object. Other users, including anonymous users can only read the object. Exercise caution when you specify this ACL.
Public read/write	All users, including anonymous users can read and write the object. Exercise caution when you specify this ACL.

For more information about object ACLs, see [Object ACL](#).

Examples

The following code provides an example on how to query the ACL of the exampleobject.txt object in the examplebucket bucket.

```
// Specify the name of the bucket and the full path of the object. The full path of the object cannot contain bucket names.
GetObjectACLRequest request = new GetObjectACLRequest("examplebucket", "exampleobject.txt");
;
// Query the ACL of the object.
oss.asyncGetObjectACL(request, new OSSCompletedCallback<GetObjectACLRequest, GetObjectACLResult>() {
    @Override
    public void onSuccess(GetObjectACLRequest request, GetObjectACLResult result) {
        Log.d("GetObjectACL", "Success!");
        Log.d("ObjectAcl", result.getObjectACL());
        Log.d("Owner", result.getObjectOwner());
        Log.d("ID", result.getObjectOwnerID());
    }
    @Override
    public void onFailure(GetObjectACLRequest request, ClientException clientException, ServiceException serviceException) {
        // Request exceptions.
        if (clientException != null) {
            // Client-side exceptions such as network errors.
            clientException.printStackTrace();
        }
        if (serviceException != null) {
            // Server-side exceptions.
            Log.e("ErrorCode", serviceException.getErrorCode());
            Log.e("RequestId", serviceException.getRequestId());
            Log.e("HostId", serviceException.getHostId());
            Log.e("RawMessage", serviceException.getRawMessage());
        }
    }
});
```

For more information about how to query the ACL of an object, see [GetObjectACL](#).

8.8.4. Copy objects

Usage notes

- To copy an object, you must have read permissions on the source object and read and write permissions on the destination bucket.
- The source bucket and destination bucket must be in the same region. For example, objects in a bucket located in the China (Hangzhou) region cannot be copied to another bucket located in the China (Qingdao) region.
- The size of the object that you want to copy cannot exceed 1 GB.

Sample code

The following code provides an example on how to copy an object:

```
// Specify the name of the source bucket.
String srcBucketName = "srcbucket";
// Specify the full path of the object in the source bucket.
String srcObjectKey = "dir1/srcobject.txt";
```

```

String srcObjectKey = "dir1/srcobject.txt";
// Specify the name of the destination bucket. The destination bucket must be in the same r
egion as the source bucket.
String destBucketName = "destbucket";
// Specify the full path of the object in the destination bucket.
String destObjectKey = "dir2/destobject.txt";
// Construct a request to copy the object.
CopyObjectRequest copyObjectRequest = new CopyObjectRequest(srcBucketName, srcObjectKey, de
stBucketName, destObjectKey);
// ObjectMetadata objectMetadata = new ObjectMetadata();
// Set the access control list (ACL) of the object. In this example, this parameter is set
to private.
// objectMetadata.setHeader("x-oss-object-acl", "private");
// Specify the storage class of the object. In this example, this parameter is set to Stand
ard.
// objectMetadata.setHeader("x-oss-storage-class", "Standard");
// Specify whether the CopyObject operation overwrites the object with the same name. If th
is parameter is set to true, the object with the same name cannot be overwritten.
// objectMetadata.setHeader("x-oss-forbid-overwrite", "true");
// If the ETag value of the source object is equal to the ETag value that is specified in t
he request, the copy operation is performed.
// objectMetadata.setHeader("x-oss-copy-source-if-match", "5B3C1A2E053D763E1B002CC607C5****
");
// Specify the path of the source object.
// objectMetadata.setHeader("x-oss-copy-source", "/examplebucket/recode-test.txt");
// If the ETag value of the source object is different from the ETag value that is specifie
d in the request, the copy operation is performed.
// objectMetadata.setHeader("x-oss-copy-source-if-none-match", "5B3C1A2E053D763E1B002CC607C
5*****");
// If the time that is specified in the request is equal to or later than the actual modifi
ed time of the source object, the copy operation is performed.
// objectMetadata.setHeader("x-oss-copy-source-if-unmodified-since", "2021-12-09T07:01:56.0
00Z");
// If the source object is modified after the time that is specified in the request, the co
py operation is performed.
// objectMetadata.setHeader("x-oss-copy-source-if-modified-since", "2021-12-09T07:01:56.000
Z");
// Specify the method that is used to configure the metadata of the destination object. If
you set the method to COPY, the metadata of the source object is copied to the destination
object.
// objectMetadata.setHeader("x-oss-metadata-directive", "COPY");
// Specify the server-side encryption algorithm that is used to encrypt the destination obj
ect when OSS creates the destination object.
// objectMetadata.setHeader("x-oss-server-side-encryption", "SSE-KMS");
// Specify the customer master key (CMK) that is managed by Key Management Service (KMS). T
his parameter takes effect only when x-oss-server-side-encryption is set to KMS.
// objectMetadata.setHeader("x-oss-server-side-encryption-key-id", "9468da86-3509-4f8d-a61e
-6eableac*****");
// Specify the tag for the destination object. You can specify multiple tags at the same ti
me.
// objectMetadata.setHeader("x-oss-tagging", "a:1");
// Specify the method that is used to configure the tags for the destination object. If you
set the method to COPY, the tags of the source object are copied to the destination object.

// objectMetadata.setHeader("x-oss-tagging-directive", "COPY");

```

```
// OSSAsyncTaskRunner (OSS logging disabled, OSS //  
// Copy the object in asynchronous mode.  
OSSAsyncTask copyTask = oss.asyncCopyObject(copyObjectRequest, new OSSCompletedCallback<CopyObjectRequest, CopyObjectResult>() {  
    @Override  
    public void onSuccess(CopyObjectRequest request, CopyObjectResult result) {  
        Log.d("copyObject", "copy success!");  
    }  
    @Override  
    public void onFailure(CopyObjectRequest request, ClientException clientException, ServiceException serviceException) {  
        // Handle request exceptions.  
        if (clientException != null) {  
            // Handle client-side exceptions such as network errors.  
            clientException.printStackTrace();  
        }  
        if (serviceException != null) {  
            // Handle server-side exceptions.  
            Log.e("ErrorCode", serviceException.getErrorCode());  
            Log.e("RequestId", serviceException.getRequestId());  
            Log.e("HostId", serviceException.getHostId());  
            Log.e("RawMessage", serviceException.getRawMessage());  
        }  
    }  
});
```

References

- For more information about the complete sample code that is used to copy an object, visit [GitHub](#).
- For more information about the API operation that you can call to copy an object, see [CopyObject](#).

8.8.5. List objects

This topic describes how to list objects in a bucket, including all objects, a specified number of objects, and objects whose names contain a specified prefix.

List a specified number of objects

The following code provides an example on how to list 20 objects in a bucket named examplebucket:


```
// Specify the bucket name.
ListObjectsRequest request = new ListObjectsRequest("examplebucket");
// Specify the maximum number of returned objects. By default, the value of this parameter
// is set to 100. The maximum value you can specify is 1000.
request.setMaxKeys(20);
oss.asyncListObjects(request, new OSSCompletedCallback<ListObjectsRequest, ListObjectsResult>() {
    @Override
    public void onSuccess(ListObjectsRequest request, ListObjectsResult result) {
        for (OSSObjectSummary objectSummary : result.getObjectSummaries()) {
            Log.i("ListObjects", objectSummary.getKey());
        }
    }
    @Override
    public void onFailure(ListObjectsRequest request, ClientException clientException, ServiceException serviceException) {
        // Handle request exceptions.
        if (clientException != null) {
            // Handle client-side exceptions such as network errors.
            clientException.printStackTrace();
        }
        if (serviceException != null) {
            // Handle server-side exceptions.
            Log.e("ErrorCode", serviceException.getErrorCode());
            Log.e("RequestId", serviceException.getRequestId());
            Log.e("HostId", serviceException.getHostId());
            Log.e("RawMessage", serviceException.getRawMessage());
        }
    }
});
```

List objects whose names contain a specified prefix

The following code provides an example on how to list objects whose names contain the "file" prefix in a bucket named examplebucket:

```
// Specify the bucket name.
ListObjectsRequest request = new ListObjectsRequest("examplebucket");
// Specify the prefix.
request.setPrefix("file");
oss.asyncListObjects(request, new OSSCompletedCallback<ListObjectsRequest, ListObjectsResult>() {
    @Override
    public void onSuccess(ListObjectsRequest request, ListObjectsResult result) {
        for (OSSObjectSummary objectSummary : result.getObjectSummaries()) {
            Log.i("ListObjects", objectSummary.getKey());
        }
    }
    @Override
    public void onFailure(ListObjectsRequest request, ClientException clientException, ServiceException serviceException) {
        // Handle request exceptions.
        if (clientException != null) {
            // Handle client-side exceptions such as network errors.
            clientException.printStackTrace();
        }
        if (serviceException != null) {
            // Handle server-side exceptions.
            Log.e("ErrorCode", serviceException.getErrorCode());
            Log.e("RequestId", serviceException.getRequestId());
            Log.e("HostId", serviceException.getHostId());
            Log.e("RawMessage", serviceException.getRawMessage());
        }
    }
});
```

List objects whose names are alphabetically after than the object specified by marker

The following code provides an example on how to list objects whose names are alphabetically after the object named `exampleobject.txt` in a bucket named `examplebucket`:

```
// Specify the bucket name.
ListObjectsRequest request = new ListObjectsRequest("examplebucket");
// Specify the value of marker. Objects whose names are alphabetically after the object specified by marker are returned.
// If the object specified by marker does not exist in the bucket, objects whose names are alphabetically after the value of marker are returned.
request.setMarker("exampleobject.txt");
oss.asyncListObjects(request, new OSSCompletedCallback<ListObjectsRequest, ListObjectsResult>() {
    @Override
    public void onSuccess(ListObjectsRequest request, ListObjectsResult result) {
        for (OSSObjectSummary objectSummary : result.getObjectSummaries()) {
            Log.i("ListObjects", objectSummary.getKey());
        }
    }
    @Override
    public void onFailure(ListObjectsRequest request, ClientException clientException, ServiceException serviceException) {
        // Handle request exceptions.
        if (clientException != null) {
            // Handle client-side exceptions such as network errors.
            clientException.printStackTrace();
        }
        if (serviceException != null) {
            // Handle server-side exceptions.
            Log.e("ErrorCode", serviceException.getErrorCode());
            Log.e("RequestId", serviceException.getRequestId());
            Log.e("HostId", serviceException.getHostId());
            Log.e("RawMessage", serviceException.getRawMessage());
        }
    }
});
```

List all objects by page

The following code provides an example on how to list all objects in a bucket named examplebucket by page. Up to 20 objects can be listed on each page.

```
private String marker = null;
private boolean isCompleted = false;
// List all objects in the bucket by page.
public void getAllObject() {
    do {
        OSSAsyncTask task = getObjectList();
        // Wait until NextMarker is returned for the preceding request. Set the marker parameter in the current request to the value of NextMarker returned in the response to the preceding request. You do not need to set marker in the first request.
        // In this example, a looping function is used to list objects by page. Therefore, a request can be sent only after the NextMarker value for the preceding request is returned. You can determine whether to wait for the NextMarker value returned for the preceding request.
        task.waitForCompletion();
    } while (!isCompleted);
}
```

```

    }
    // List objects in a page.
    public OSSAsyncTask getObjectList() {
        // Specify the bucket name.
        ListObjectsRequest request = new ListObjectsRequest("examplebucket");
        // Specify the maximum number of objects that can be listed on each page. By default, the
        // value of this parameter is set to 100. The maximum value you can specify is 1000.
        request.setMaxKeys(20);
        request.setMarker(marker);
        OSSAsyncTask task = oss.asyncListObjects(request, new OSSCompletedCallback<ListObjectsRequest, ListObjectsResult>() {
            @Override
            public void onSuccess(ListObjectsRequest request, ListObjectsResult result) {
                for (OSSObjectSummary objectSummary : result.getObjectSummaries()) {
                    Log.i("ListObjects", objectSummary.getKey());
                }
                // List objects in the last page.
                if (!result.isTruncated()) {
                    isCompleted = true;
                    return;
                }
                // Obtain the marker value for the next request.
                marker = result.getNextMarker();
            }
            @Override
            public void onFailure(ListObjectsRequest request, ClientException clientException, ServiceException serviceException) {
                isCompleted = true;
                // Handle request exceptions.
                if (clientException != null) {
                    // Handle client-side exceptions such as network errors.
                    clientException.printStackTrace();
                }
                if (serviceException != null) {
                    // Handle server-side exceptions.
                    Log.e("ErrorCode", serviceException.getErrorCode());
                    Log.e("RequestId", serviceException.getRequestId());
                    Log.e("HostId", serviceException.getHostId());
                    Log.e("RawMessage", serviceException.getRawMessage());
                }
            }
        });
        return task;
    }
}

```

List objects whose names contain a specified prefix by page

The following code provides an example on how to list objects whose names contain the "file" prefix in a bucket named examplebucket by page. Up to 20 objects can be listed on each page.

```

private String marker = null;
private boolean isCompleted = false;
// List all objects by page,
public void getAllObject() {

```

```

do {
    OSSAsyncTask task = getObjectList();
    // Wait until NextMarker is returned for the preceding request. Set the marker parameter in the current request to the value of NextMarker returned in the preceding request. You do not need to set marker in the first request.
    // In this example, a looping function is used to list objects by page. Therefore, a request can be sent only after the NextMarker value for the preceding request is returned. You can determine whether to wait for the NextMarker value returned for the preceding request.
    task.waitUntilFinished();
} while (!isCompleted);
}
// List objects in a page.
public OSSAsyncTask getObjectList() {
    // Specify the bucket name.
    ListObjectsRequest request = new ListObjectsRequest("examplebucket");
    // Specify the maximum number of objects that can be listed on each page. By default, the value of this parameter is set to 100. The maximum value you can specify is 1000.
    request.setMaxKeys(20);
    // Specify the prefix.
    request.setPrefix("file");
    request.setMarker(marker);
    OSSAsyncTask task = oss.asyncListObjects(request, new OSSCompletedCallback<ListObjectsRequest, ListObjectsResult>() {
        @Override
        public void onSuccess(ListObjectsRequest request, ListObjectsResult result) {
            for (OSSObjectSummary objectSummary : result.getObjectSummaries()) {
                Log.i("ListObjects", objectSummary.getKey());
            }
            // List objects in the last page.
            if (!result.isTruncated()) {
                isCompleted = true;
                return;
            }
            // Obtains the marker value for the next request.
            marker = result.getNextMarker();
        }
        @Override
        public void onFailure(ListObjectsRequest request, ClientException clientException, ServiceException serviceException) {
            isCompleted = true;
            // Handle request exceptions.
            if (clientException != null) {
                // Handle client-side exceptions such as network errors.
                clientException.printStackTrace();
            }
            if (serviceException != null) {
                // Handle server-side exceptions.
                Log.e("ErrorCode", serviceException.getErrorCode());
                Log.e("RequestId", serviceException.getRequestId());
                Log.e("HostId", serviceException.getHostId());
                Log.e("RawMessage", serviceException.getRawMessage());
            }
        }
    });
}

```

```
    }  
    return task;  
}
```

List objects whose names contain special characters

If the name of an object contains one of the following special characters, you must encode the object name before you transmit the object. Only URL encoding is supported in OSS.

- Single quotation marks (')
- Double quotations marks (")
- Ampersands (&)
- Angle brackets (<>)
- Chinese characters

The following code provides an example on how to list objects whose names contain special characters in a bucket named `examplebucket`:

```
// Specify the bucket name.  
ListObjectsRequest request = new ListObjectsRequest("examplebucket");  
// Specify the encoding type of the object names.  
request.setEncodingType("url");  
oss.asyncListObjects(request, new OSSCompletedCallback<ListObjectsRequest, ListObjectsResult>() {  
    @Override  
    public void onSuccess(ListObjectsRequest request, ListObjectsResult result) {  
        for (OSSObjectSummary objectSummary : result.getObjectSummaries()) {  
            Log.i("ListObjects", URLDecoder.decode(objectSummary.getKey(), "UTF-8"));  
        }  
    }  
    @Override  
    public void onFailure(ListObjectsRequest request, ClientException clientException, ServiceException serviceException) {  
        // Handle request exceptions.  
        if (clientException != null) {  
            // Handle client-side exceptions such as network errors.  
            clientException.printStackTrace();  
        }  
        if (serviceException != null) {  
            // Handle server-side exceptions.  
            Log.e("ErrorCode", serviceException.getErrorCode());  
            Log.e("RequestId", serviceException.getRequestId());  
            Log.e("HostId", serviceException.getHostId());  
            Log.e("RawMessage", serviceException.getRawMessage());  
        }  
    }  
});
```

For more information about how to list objects, see [GetBucket \(ListObjects\)](#).

8.8.6. Prevent objects from being overwritten by objects with the same names

By default, if you upload an object that has the same name as an existing object on which you have access permissions, the existing object is overwritten by the uploaded object. This topic describes how to configure the `x-oss-forbid-overwrite` request header to prevent objects from being overwritten by objects with the same names when you copy objects or perform simple upload or multipart upload.

Simple upload

The following code provides an example on how to prevent objects from being overwritten by objects with the same names in simple upload:

```

// Specify the name of the bucket. Example: examplebucket. For more information about the naming conventions for buckets, see Bucket naming conventions.
String bucketName = "examplebucket";
// Specify the full path of the object. The full path of the object cannot contain the bucket name. Example: exampledir/exampleobject.txt. For more information about the naming conventions for objects, see Object naming conventions.
String objectKey = "exampledir/exampleobject.txt";
// Specify the full path of the local file that you want to upload.
String localFile = "/storage/emulated/0/oss/examplefile.txt";
// Construct a request to upload the local file.
PutObjectRequest put = new PutObjectRequest(bucketName, objectKey, localFile);
ObjectMetadata metadata = new ObjectMetadata();
// Specify whether to overwrite the object with the same name.
// By default, if x-oss-forbid-overwrite is not specified, existing objects are overwritten by the objects with the same names.
// If x-oss-forbid-overwrite is set to false, existing objects are overwritten by the objects with the same names.
// If x-oss-forbid-overwrite is set to true, existing objects are not overwritten by the objects with the same names. If an existing object has the same name as the uploaded object, an error is reported.
metadata.setHeader("x-oss-forbid-overwrite", "true");
put.setMetadata(metadata);
OSSAsyncTask task = oss.asyncPutObject(put, new OSSCompletedCallback<PutObjectRequest, PutObjectResult>() {
    @Override
    public void onSuccess(PutObjectRequest request, PutObjectResult result) {
        Log.d("PutObject", "UploadSuccess");
        Log.d("ETag", result.getETag());
        Log.d("RequestId", result.getRequestId());
    }
    @Override
    public void onFailure(PutObjectRequest request, ClientException clientException, ServiceException serviceException) {
        // Handle request exceptions.
        if (clientException != null) {
            // Handle client-side exceptions such as network errors.
            clientException.printStackTrace();
        }
        if (serviceException != null) {
            // Handle server-side exceptions.
            Log.e("ErrorCode", serviceException.getErrorCode());
            Log.e("RequestId", serviceException.getRequestId());
            Log.e("HostId", serviceException.getHostId());
            Log.e("RawMessage", serviceException.getRawMessage());
        }
    }
});

```

Copy objects

The following code provides an example on how to prevent an existing object from being overwritten when you copy an object with the same name:


```

// Specify the name of the source bucket.
String srcBucketName = "srcbucket";
// Specify the full path of the object in the source bucket.
String srcObjectKey = "dir1/srcobject.txt";
// Specify the name of the destination bucket.
String destBucketName = "destbucket";
// Specify the full path of the object in the destination bucket.
String destObjectKey = "dir2/destobject.txt";
// Construct a request to copy the object.
CopyObjectRequest copyObjectRequest = new CopyObjectRequest(srcBucketName, srcObjectKey, destBucketName, destObjectKey);
ObjectMetadata metadata = new ObjectMetadata();
// Specify whether to overwrite the object with the same name.
// By default, if x-oss-forbid-overwrite is not specified, existing objects are overwritten by the objects with the same names.
// If x-oss-forbid-overwrite is set to false, existing objects are overwritten by the objects with the same names.
// If x-oss-forbid-overwrite is set to true, existing objects are not overwritten by the objects with the same names. If an existing object has the same name as the copied object, an error is reported.
metadata.setHeader("x-oss-forbid-overwrite", "true");
copyObjectRequest.setNewObjectMetadata(metadata);
// Copy the object in asynchronous mode.
OSSAsyncTask copyTask = oss.asyncCopyObject(copyObjectRequest, new OSSCompletedCallback<CopyObjectRequest, CopyObjectResult>() {
    @Override
    public void onSuccess(CopyObjectRequest request, CopyObjectResult result) {
        Log.d("copyObject", "copy success!");
    }
    @Override
    public void onFailure(CopyObjectRequest request, ClientException clientException, ServiceException serviceException) {
        // Handle request exceptions.
        if (clientException != null) {
            // Handle client-side exceptions such as network errors.
            clientException.printStackTrace();
        }
        if (serviceException != null) {
            // Handle service exceptions.
            Log.e("ErrorCode", serviceException.getErrorCode());
            Log.e("RequestId", serviceException.getRequestId());
            Log.e("HostId", serviceException.getHostId());
            Log.e("RawMessage", serviceException.getRawMessage());
        }
    }
});

```

Multipart upload

The following code provides an example on how to prevent objects from being overwritten by objects with the same names in multipart upload:

```

// Specify the name of the bucket. Example: examplebucket.
String bucketName = "examplebucket";

```

```

String bucketName = "examplebucket";
// Specify the full path of the object. The full path of the object cannot contain the bucket name. Example: exampledir/exampleobject.txt.
String objectKey = "exampledir/exampleobject.txt";
// Specify the full path of the local file that you want to upload.
String localFile = "/storage/emulated/0/oss/examplefile.txt";
// Initiate a multipart upload task.
InitiateMultipartUploadRequest init = new InitiateMultipartUploadRequest(bucketName, objectKey);
ObjectMetadata metadata = new ObjectMetadata();
// Specify whether to overwrite the object with the same name.
// By default, if x-oss-forbid-overwrite is not specified, existing objects are overwritten by the objects with the same names.
// If x-oss-forbid-overwrite is set to false, existing objects are overwritten by the objects with the same names.
// If x-oss-forbid-overwrite is set to true, existing objects are not overwritten by the objects with the same names. If an existing object has the same name as the uploaded object, an error is reported.
metadata.setHeader("x-oss-forbid-overwrite", "true");
init.setMetadata(metadata);
InitiateMultipartUploadResult initResult = oss.initMultipartUpload(init);
// Obtain the upload ID. The upload ID uniquely identifies the multipart upload task. You can use the upload ID to cancel or query the multipart upload task.
String uploadId = initResult.getUploadId();
// Specify the size of a single part. Unit: bytes. Valid values: 100 KB to 5 GB.
int partCount = 100 * 1024;
// Start the multipart upload task.
List<PartETag> partETags = new ArrayList<PartETag>();
for (int i = 1; i < 5; i++) {
    byte[] data = new byte[partCount];
    RandomAccessFile raf = new RandomAccessFile(localFile, "r");
    long skip = (i-1) * partCount;
    raf.seek(skip);
    raf.readFully(data, 0, partCount);
    UploadPartRequest uploadPart = new UploadPartRequest();
    uploadPart.setBucketName(bucketName);
    uploadPart.setObjectKey(objectKey);
    uploadPart.setUploadId(uploadId);
    // Specify the part number for each part. The number starts from 1. Each part to upload has a part number. Valid values: 1 to 10000.
    uploadPart.setPartNumber(i);
    uploadPart.setPartContent(data);
    try {
        UploadPartResult result = oss.uploadPart(uploadPart);
        PartETag partETag = new PartETag(uploadPart.getPartNumber(), result.getETag());
        partETags.add(partETag);
    } catch (ServiceException serviceException) {
        OSSLog.logError(serviceException.getErrorCode());
    }
}
Collections.sort(partETags, new Comparator<PartETag>() {
    @Override
    public int compare(PartETag lhs, PartETag rhs) {
        if (lhs.getPartNumber() < rhs.getPartNumber()) {
            return -1;

```

```
        } else if (lhs.getPartNumber() > rhs.getPartNumber()) {
            return 1;
        } else {
            return 0;
        }
    }
});
// Complete the multipart upload task.
CompleteMultipartUploadRequest complete = new CompleteMultipartUploadRequest(bucketName, objectKey, uploadId, partETags);
metadata = new ObjectMetadata();
// Specify whether to overwrite the object with the same name.
// By default, if x-oss-forbid-overwrite is not specified, existing objects are overwritten by the objects with the same names.
// If x-oss-forbid-overwrite is set to false, existing objects are overwritten by the objects with the same names.
// If x-oss-forbid-overwrite is set to true, existing objects are not overwritten by the objects with the same names. If an existing object has the same name as the uploaded object, an error is reported.
metadata.setHeader("x-oss-forbid-overwrite", "true");
complete.setMetadata(metadata);
CompleteMultipartUploadResult completeResult = oss.completeMultipartUpload(complete);
```

References

- For more information about the API operation that you can call to perform simple upload, see [Put Object](#).
- For more information about the API operation that you can call to copy an object, see [Copy Object](#).
- The multipart upload task involves the following three API operations:
 - The API operation that you can call to initiate a multipart upload task. For more information, see [Initiate Multipart Upload](#).
 - The API operation that you can call to upload data by part. For more information, see [Upload Part](#).
 - The API operation that you can call to complete the multipart upload task. For more information, see [Complete Multipart Upload](#).

8.8.7. Delete objects

This topic describes how to delete a single object or batch delete objects.



Warning Deleted objects cannot be recovered. Exercise caution when you delete objects.

Usage notes

To delete an object, you must have write permissions on the bucket in which the object is stored.

Delete a single object

The following code provides an example on how to delete an object named `exampleobject.txt` from a bucket named `examplebucket`:

```

// Create the delete request.
// Specify the bucket name and the full path of the object that you want to delete. The full path of the object cannot contain bucket names.
DeleteObjectRequest delete = new DeleteObjectRequest("examplebucket", "exampleobject.txt");// Delete the object asynchronously.
OSSAsyncTask deleteTask = oss.asyncDeleteObject(delete, new OSSCompletedCallback<DeleteObjectRequest, DeleteObjectResult>() {
    @Override
    public void onSuccess(DeleteObjectRequest request, DeleteObjectResult result) {
        Log.d("asyncDeleteObject", "success!");
    }
    @Override
    public void onFailure(DeleteObjectRequest request, ClientException clientException, ServiceException serviceException) {
        // Handle exceptions.
        if (clientException != null) {
            // Handle client exceptions, such as network exceptions.
            clientException.printStackTrace();
        }
        if (serviceException != null) {
            // Handle server exceptions.
            Log.e("ErrorCode", serviceException.getErrorCode());
            Log.e("RequestId", serviceException.getRequestId());
            Log.e("HostId", serviceException.getHostId());
            Log.e("RawMessage", serviceException.getRawMessage());
        }
    }
});

DeleteObjectRequest delete = new DeleteObjectRequest("examplebucket", "exampleobject.txt");
// Delete the object asynchronously.
OSSAsyncTask deleteTask = oss.asyncDeleteObject(delete, new OSSCompletedCallback<DeleteObjectRequest, DeleteObjectResult>() {
    @Override
    public void onSuccess(DeleteObjectRequest request, DeleteObjectResult result) {
        Log.d("asyncDeleteObject", "success!");
    }
    @Override
    public void onFailure(DeleteObjectRequest request, ClientException clientException, ServiceException serviceException) {
        // Handle exceptions.
        if (clientException != null) {
            // Handle client exceptions, such as network exceptions.
            clientException.printStackTrace();
        }
        if (serviceException != null) {
            // Handle server exceptions.
            Log.e("ErrorCode", serviceException.getErrorCode());
            Log.e("RequestId", serviceException.getRequestId());
            Log.e("HostId", serviceException.getHostId());
            Log.e("RawMessage", serviceException.getRawMessage());
        }
    }
});

```

Delete multiple objects in a batch

You can batch delete up to 1,000 objects each time.

The result can be returned in the following two modes. Select a return mode based on your requirements.

- **verbose:** If `isQuiet` is not specified or is set to `false`, a list of all deleted objects is returned. This is the default return mode.
- **quiet:** If `isQuiet` is set to `true`, a list of objects that failed to be deleted is returned.

The following code provides an example on how to delete multiple specified objects from a bucket named `examplebucket` and return the result in the quiet mode:

```
// Specify the full paths of the objects that you want to delete. The full paths of the objects cannot contain bucket names.
List<String> objectKeys = new ArrayList<String>();
objectKeys.add("exampleobject.txt");
objectKeys.add("testfolder/sampleobject.txt");
// Set the return mode to quiet to return a list of objects that failed to be deleted.
DeleteMultipleObjectRequest request = new DeleteMultipleObjectRequest("examplebucket", objectKeys, true);
mOss.asyncDeleteMultipleObject(request, new OSSCompletedCallback<DeleteMultipleObjectRequest, DeleteMultipleObjectResult>() {
    @Override
    public void onSuccess(DeleteMultipleObjectRequest request, DeleteMultipleObjectResult result) {
        Log.i("DeleteMultipleObject", "success");
    }
    @Override
    public void onFailure(DeleteMultipleObjectRequest request, ClientException clientException, ServiceException serviceException) {
        // Handle exceptions.
        if (clientException != null) {
            // Handle client exceptions, such as network exceptions.
            clientException.printStackTrace();
        }
        if (serviceException != null) {
            // Handle server exceptions.
            Log.e("ErrorCode", serviceException.getErrorCode());
            Log.e("RequestId", serviceException.getRequestId());
            Log.e("HostId", serviceException.getHostId());
            Log.e("RawMessage", serviceException.getRawMessage());
        }
    }
});
```

8.8.8. Set the Content-Type of an object

In Web services, the Content-Type of an object specifies the type of the object and determines the method and encoding used to read the object.

In some cases, you must specify the Content-Type of an object to be uploaded. Otherwise, the object cannot be read in the appropriate method. If you do not specify the Content-Type when using OSS Android SDK to upload an object, the SDK automatically specifies it according to the suffix of the object.

```
// Construct an upload request.
PutObjectRequest put = new PutObjectRequest("<bucketName>", "<objectKey>", "<uploadFilePath>");
ObjectMetadata metadata = new ObjectMetadata();
// Set the Content-Type of the object to be uploaded.
metadata.setContentType("application/octet-stream");
// Customize the user metadata.
metadata.addUserMetadata("x-oss-meta-name1", "value1");
put.setMetadata(metadata);
OSSAsyncTask task = oss.asyncPutObject(put, new OSSCompletedCallback<PutObjectRequest, PutObjectResult>() {
    ...
});
```

8.8.9. Obtain the metadata of an object

You can call the HeadObject method to obtain only the metadata but not the content of an object.

Run the following code to obtain the metadata of an object:

```
// Create a request to obtain the metadata of an object by calling the synchronous interface.
HeadObjectRequest head = new HeadObjectRequest("<bucketName>", "<objectKey>");
// Obtain the metadata of an object.
OSSAsyncTask task = oss.asyncHeadObject(head, new OSSCompletedCallback<HeadObjectRequest, HeadObjectResult>() {
    @Override
    public void onSuccess(HeadObjectRequest request, HeadObjectResult result) {
        // Obtain the size of the object.
        Log.d("headObject", "object Size: " + result.getMetadata().getContentLength());
        // Obtain the type of the object.
        Log.d("headObject", "object Content Type: " + result.getMetadata().getContentType());
    };
    @Override
    public void onFailure(HeadObjectRequest request, ClientException clientException, ServiceException serviceException) {
        // Handle the exceptions returned for the request.
        if (clientException != null) {
            // A local exception (such as network exception) occurs.
            clientException.printStackTrace();
        }
        if (serviceException != null) {
            // A service exception occurs.
            Log.e("ErrorCode", serviceException.getErrorCode());
            Log.e("RequestId", serviceException.getRequestId());
            Log.e("HostId", serviceException.getHostId());
            Log.e("RawMessage", serviceException.getRawMessage());
        }
    }
});
// task.waitForCompletion(); // Wait until the task is complete.
```

8.8.10. Restore an archive object

You must restore an archive object before you read it. Do not call `restoreObject` for non-archive objects.

Run the following code to restore an archive object:

```
//Restore the archive.
RestoreObjectRequest restore = new RestoreObjectRequest();
restore.setBucketName("yourBucketName");
restore.setObjectKey("yourObjectName");
OSSAsyncTask task = oss.asyncRestoreObject(restore, new OSSCompletedCallback<RestoreObjectRequest,
    RestoreObjectResult>() {
    @Override
    public void onSuccess(RestoreObjectRequest request, RestoreObjectResult result) {
        OSSLog.logInfo("code: "+result.getStatusCode());
    }
    @Override
    public void onFailure(RestoreObjectRequest request, ClientException clientException,
        ServiceException serviceException) {
        OSSLog.logError("error: "+serviceException.getRawMessage());
    }
});
task.waitUntilFinished();
```

8.8.11. Manage symbolic links

Symbolic links can be used to access objects that are frequently used in buckets. After you create a symbolic link for an object, you can use the symbolic link to access the object.

Create a symbolic link

The following code provides an example on how to create a symbolic link:


```

// Construct a request.
PutSymlinkRequest putSymlink = new PutSymlinkRequest();
// Specify the name of the bucket.
putSymlink.setBucketName("yourBucketName");
// Specify the name of the symbolic link.
putSymlink.setObjectKey("yourSymLink");
// Specify the name of the object to which the symbolic link points.
putSymlink.setTargetObjectName("yourTargetObjectName");
ObjectMetadata metadata = new ObjectMetadata();
// Specify whether to overwrite the object with the same name. In this example, this parameter is set to true, which specifies that the object with the same name cannot be overwritten.
//metadata.setHeader("x-oss-forbid-overwrite", "false");
// Specify the access control list (ACL) of the object. In this example, this parameter is set to private.
//metadata.setHeader("x-oss-object-acl", "private");
// Specify the storage class of the object. In this example, this parameter is set to Standard.
//metadata.setHeader("x-oss-storage-class", "Standard");
putSymlink.setMetadata(metadata);
OSSAsyncTask task = oss.asyncPutSymlink(putSymlink, new OSSCompletedCallback<PutSymlinkRequest, PutSymlinkResult>() {
    @Override
    public void onSuccess(PutSymlinkRequest request, PutSymlinkResult result) {
        Log.d("PutSymlink", "PutSymlink success");
    }
    @Override
    public void onFailure(PutSymlinkRequest request, ClientException clientException, ServiceException serviceException) {
        // Handle request exceptions.
        if (clientException != null) {
            // Handle client-side exceptions such as network errors.
            clientException.printStackTrace();
        }
        if (serviceException != null) {
            // Handle server-side exceptions.
            Log.e("ErrorCode", serviceException.getErrorCode());
            Log.e("RequestId", serviceException.getRequestId());
            Log.e("HostId", serviceException.getHostId());
            Log.e("RawMessage", serviceException.getRawMessage());
        }
    }
});
task.waitForTask();

```

Query the name of the object to which a symbolic link points

To query a symbolic link, you must have read permissions on the symbolic link. The following code provides an example on how to query the name of the object to which a symbolic link points:

```
// Construct a request.
GetSymlinkRequest getSymlink = new GetSymlinkRequest();
// Specify the name of the bucket.
getSymlink.setBucketName("yourBucketName");
// Specify the name of the symbolic link that you want to query.
getSymlink.setObjectKey("yourSymLink");
OSSAsyncTask task = oss.asyncGetSymlink(getSymlink, new OSSCompletedCallback<GetSymlinkRequest,
    GetSymlinkResult>() {
    @Override
    public void onSuccess(GetSymlinkRequest request, GetSymlinkResult result) {
        OSSLog.logInfo("target:" + result.getTargetObjectName());
    }
    @Override
    public void onFailure(GetSymlinkRequest request, ClientException clientException,
        ServiceException serviceException) {
        OSSLog.logError("error: " + serviceException.getRawMessage());
    }
});
task.waitForCompletion();
```

References

- For more information about the complete sample code of symbolic links, visit [GitHub](#).
- For more information about the API operation that you can call to create a symbolic link, see [Put Symlink](#).
- For more information about the API operation that you can call to query a symbolic link, see [Get Symlink](#).

8.9. Data security

OSS Android SDK provides methods to verify data integrity, which ensure data security when you uploading, downloading, and copying objects.

Due to the complex network environment for mobile devices, errors may occur when data is transmitted between the client and the OSS server. Therefore, OSS Android SDK provides two methods to verify data integrity: CRC end-to-end verification and MD5 verification.

- CRC verification

If you enable CRC verification when reading the download stream, data integrity is automatically verified after the stream is completely read.

Run the following code to enable CRC verification:

```
GetObjectRequest request = new GetObjectRequest(OSSTestConfig.ANDROID_TEST_BUCKET, testFile);
// Enable CRC verification.
request.setCRC64(OSSRequest.CRC64Config.YES);
//....
try{
    GetObjectResult result = oss.getObject(request);
    InputStream in = result.getObjectContent();
    ByteArrayOutputStream output = new ByteArrayOutputStream();
    byte[] buffer = new byte[BUFFER_SIZE];
    int len;
    while ((len = in.read(buffer)) > -1) {
        output.write(buffer, 0, len);
    }
    output.flush();
    in.close();
}catch(ClientException e){
    //...
}catch(InconsistentException e){
    //....
}
```

- MD5 verification

To check whether the object uploaded in the multipart upload method is the same as the local file, you can upload the parts with their Content-MD5 values. The OSS server checks the MD5 values of the uploaded parts and determines the upload task is successful only when the MD5 values are the same as the specified Content-MD5 values, ensuring that the uploaded object is the same as the local file.

Run the following code to enable MD5 verification:

```
UploadPartRequest uploadPart = new UploadPartRequest("<bucketName>", "<objectKey>", uploadId, currentIndex);
// Configure the part content.
uploadPart.setPartContent(partData);
// Configure the MD5 value.
uploadPart.setMd5Digest(BinaryUtil.calculateBase64Md5(data));
```

8.10. Sign a URL

SDK supports the signing of the URL with a validity period or a public resource URL so that the URL can be transferred to a third party for authorized access.

Sign a private resource URL with a validity period

For a bucket or an object not assigned the public-read permission, the following interface can be called to obtain a signed URL to access the bucket or object:

```
String url = oss.presignConstrainedObjectURL("<bucketName>", "<objectKey>", 30 * 60);
```

Sign a public resource URL

For a bucket or an object with the public-read permission, the following interface can be called to obtain a public URL to access the bucket or object:

```
String url = oss.presignPublicObjectURL("<bucketName>", "<objectKey>");
```

8.11. Authorized access

Object Storage Service (OSS) SDK for Android provides two authentication modes to ensure the data security of mobile devices: the Security Token Service (STS) authentication mode and the self-signed mode.

Overview

When you use the STS authentication mode or self-signed mode, ensure that the callback function that you implement returns results for Token and Signature. If you must obtain the token and signature from the app server by sending a request in the callback function, we recommend that you call the synchronous API operations included in the network library. The callback function is run in the child thread of the request generated by the SDK and does not block the main thread.

STS authentication mode

 **Note** To use the STS authentication mode, you must first activate Alibaba Cloud RAM.

You can use Alibaba Cloud STS to authorize temporary access to OSS. STS is a web service that provides temporary access tokens for users. You can use STS to grant a set of temporary access credentials that have a custom validity period and custom permissions to a third-party application or a RAM user managed by you. For more information about STS, see [What is STS?](#)

STS provides the following benefits:

- You need only to generate an access token and send the access token to a third-party application, instead of exposing your AccessKey pair to the third-party application. You can define the access permissions and validity period of this token.
- The token automatically expires after the validity period. Therefore, you do not need to manually revoke the access permissions of a token.

 **Note** For more information about how to set up STS, see [Use a temporary credential provided by STS to access OSS](#) in OSS Developer Guide. You can call the [AssumeRole](#) operation or use [STS SDKs for various programming languages](#) to obtain a temporary access credential. The temporary access credential contains a security token and a temporary AccessKey pair that consists of an AccessKey ID and an AccessKey secret. The minimum validity period of a temporary access credential is 900 seconds. The maximum validity period of a temporary access credential is the maximum session duration specified for the current role. For more information, see [Specify the maximum session duration for a RAM role](#).

- Configure an STS token

You can obtain an STS token in the app by a method, such as sending a request to the app server, and use the STS token to initialize the SDK. If you use this method, you must pay attention to the validity period of the STS token. When the STS token is about to expire, you must update the new STS token for the SDK.

The following code provides an example on how to use an STS token to initialize the SDK:

```
String endpoint = "http://oss-cn-hangzhou.aliyuncs.com";
OSSCredentialProvider credentialProvider = new OSSStsTokenCredentialProvider("<StsToken.AccessKeyId>", "<StsToken.SecretKeyId>", "<StsToken.SecurityToken>");
OSS oss = new OSSClient(getApplicationContext(), endpoint, credentialProvider);
```

When the STS token is about to expire, you can create a new OSSClient or update CredentialProvider by using the following method:

```
oss.updateCredentialProvider(new OSSStsTokenCredentialProvider("<StsToken.AccessKeyId>", "<StsToken.SecretKeyId>", "<StsToken.SecurityToken>"));
```

- Obtain the STS token by implementing callback

If you want the SDK to automatically update the STS token, you must implement callback in your app. The app uses the callback to obtain a federation token (STS token) and returns the token to the SDK. The SDK uses the STS token to generate signatures. When the STS token needs to be updated, the SDK calls the callback to obtain a new token.

```
String endpoint = "http://oss-cn-hangzhou.aliyuncs.com";
OSSCredentialProvider credentialProvider = new OSSEFederationCredentialProvider() {
    @Override
    public OSSEFederationToken getFederationToken() {
        // Obtain a federation token, construct the token, and then return the token as an OSSEFederationToken object. If you fail to obtain the federation token, null is returned.
        OSSEFederationToken token;
        // Obtain a federation token from your server.
        return token;
    }
};
OSS oss = new OSSClient(getApplicationContext(), endpoint, credentialProvider);
```

 **Note** Additionally, if you obtain all fields required to generate a token in other ways, you can also return these fields by using the callback. In this case, you must manually update the token, and then reconfigure OSSCredentialProvider of the OSSClient instance.

If the URL of the server from which you request a token is `http://localhost:8080/distribute-token.json`, the following data is returned:

```
{
  "StatusCode": 200,
  "AccessKeyId": "STS.iA645eTOXEqP3cg3****",
  "AccessKeySecret": "rV3VQrpFQ4BsyHSAvi5NVLpPIVffDJv4LojU****",
  "Expiration": "2015-11-03T09:52:59Z",
  "SecurityToken": "CAES7QIIARKAAZPlqaN9ILiQZPS+JDkS/GSZN45RLx4YS/p30gaUC+oJl3XS1bJ7StKpQ****"
}
```

The following code provides an example on how to implement OSSEFederationCredentialProvider:

```

OSSCredentialProvider credetialProvider = new OSSFederationCredentialProvider() {
    @Override
    public OSSFederationToken getFederationToken() {
        try {
            URL stsUrl = new URL("http://localhost:8080/distribute-token.json");
            HttpURLConnection conn = (HttpURLConnection) stsUrl.openConnection();
            InputStream input = conn.getInputStream();
            String jsonText = IOUtils.readStreamAsString(input, OSSConstants.DEFAULT_CHAR
SET_NAME);
            JSONObject jsonObj = new JSONObject(jsonText);
            String ak = jsonObj.getString("AccessKeyId");
            String sk = jsonObj.getString("AccessKeySecret");
            String token = jsonObj.getString("SecurityToken");
            String expiration = jsonObj.getString("Expiration");
            return new OSSFederationToken(ak, sk, token, expiration);
        } catch (Exception e) {
            e.printStackTrace();
        }
        return null;
    }
};

```

Self-signed mode

You can perform the following operations to save the AccessKey ID and AccessKey secret on your own server and then use them to sign the client information.

1. Obtain the string-to-sign from the client and send the string to your own server.
 - i. Use the `signContent` method of `OSSCustomSignerCredentialProvider` provided by OSS SDK for Android to obtain the string-to-sign when you create the request.
 - ii. Send the string-to-sign to your own server.
2. Sign the string on your own server and return the signed string to the client.
 - i. Use the specified signature algorithm to sign the string. For more information about the signature algorithm, see [Add signatures to headers](#).

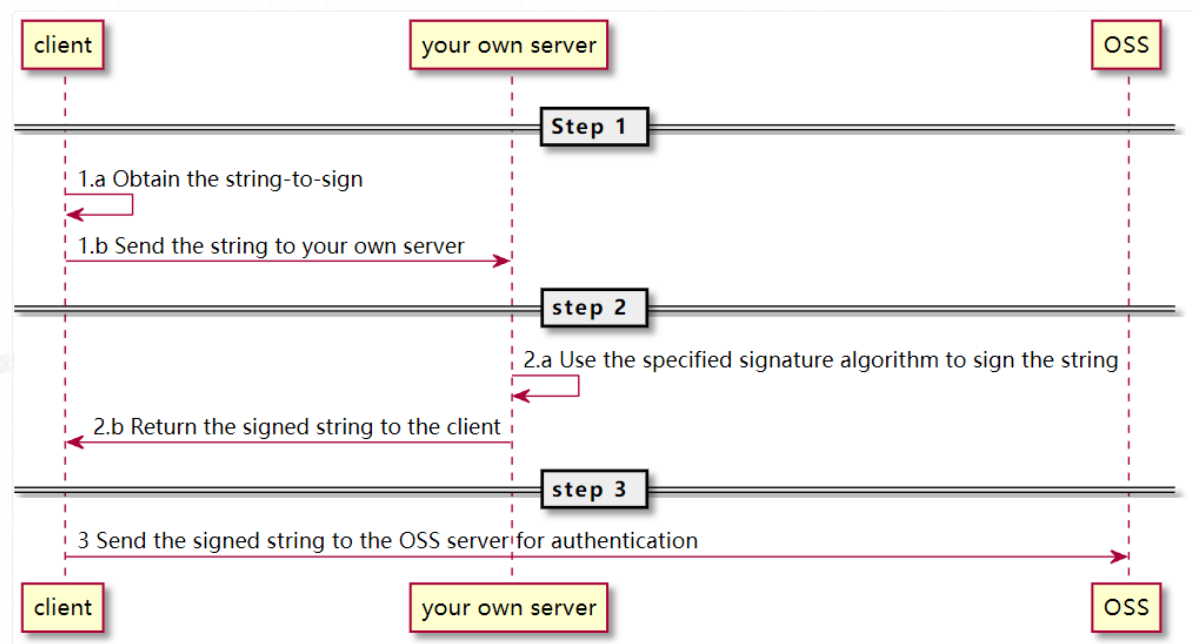
The signature is in the following format: `signature = "OSS " + AccessKeyId + ":" + base64(hmac-shal (AccessKeySecret, content))`, in which `content` is the string that is concatenated based on the request parameters.

- ii. Return the signed string to the client.

For example, the URL of the server is `http://localhost:8080/sign`. You can send the content to the server to generate a signature, and the server returns the signature to the client. The following code provides an example on how to sign the string based on the signature algorithm.

```
String endpoint = "http://oss-cn-hangzhou.aliyuncs.com";
OSSCredentialProvider credentialProvider = new OSSCustomSignerCredentialProvider()
{
    @Override
    public String signContent(String content) {
        URL stsUrl = new URL("http://localhost:8080/sign?content=" + content);
        HttpURLConnection conn = (HttpURLConnection) stsUrl.openConnection();
        InputStream input = conn.getInputStream();
        String jsonText = IOUtils.readStreamAsString(input, OSSConstants.DEFAULT_CHARSET_NAME);
        JSONObject jsonObj = new JSONObject(jsonText);
        String signature = jsonObj.getString("signature");
        return signature;
    }
};
OSS oss = new OSSClient(getApplicationContext(), endpoint, credentialProvider);
```

3. Send the signed string on the client to the OSS server for authentication.



8.12. IMG

OSS Image Processing (IMG) is a secure, cost-effective, and highly reliable image processing service that can process large amounts of data. After you upload source images to OSS, you can call RESTful API operations to process the images anytime, anywhere, and on any Internet device.

Usage

- Anonymous access

```
String url = oss.presignPublicObjectURL(testBucket, testObject);
OSSLog.logDebug("signPublicURL", "get url: " + url);
```

Add the `x-oss-process:operation` parameter to the generated URL, in which operation indicates the operation performed on the image.

- Authorized access

When you process images by using OSS SDKs, you can call the `request.setxOssProcess()` operation to configure IMG parameters in the following code. Examples:

```
// Construct an image download request.
GetObjectRequest get = new GetObjectRequest("<bucketName>", "example.jpg");
// Process the image.
request.setxOssProcess("image/resize,m_fixed,w_100,h_100");
OSSAsyncTask task = oss.asyncGetObject(get, new OSSCompletedCallback<GetObjectRequest, GetObjectResult>() {
    @Override
    public void onSuccess(GetObjectRequest request, GetObjectResult result) {
        // The request is successful.
        InputStream inputStream = result.getObjectContent();
        byte[] buffer = new byte[2048];
        int len;
        try {
            while ((len = inputStream.read(buffer)) != -1) {
                // Process the downloaded data.
            }
        } catch (IOException e) {
            e.printStackTrace();
        }
    }
    @Override
    public void onFailure(GetObjectRequest request, ClientException clientExcepcion, ServiceException serviceException) {
        // Handle the exceptions. You can add your own code.
    }
});
```

 **Note** To perform other operations on the image, replace the parameters of `request.setxOssProcess()` .

- Access by using SDKs

```
GetObjectRequest request = new GetObjectRequest("bucket-name", "image-name");
request.setxOssProcess("image/resize,m_lfit,w_100,h_100"); // Configure IMG parameters.
OSSAsyncTask task = ossClient.asyncGetObject(request, getCallback);
```

IMG persistence

The following code provides an example on how to save processed images:


```
// fromBucket indicates the name of the source bucket. toBucket indicates the name of the destination bucket.
// fromObjectKey indicates the names of the source object. toObjectkey indicates the destination object. Their names must be a complete path including the file suffix. Example: abc/efg/123.jpg.
// action indicates the IMG operation.
ImagePersistRequest request = new ImagePersistRequest(fromBucket, fromObjectKey, toBucket, toObjectkey, action);
OSSAsyncTask task = mOss.asyncImagePersist(request, new OSSCompletedCallback<ImagePersistRequest, ImagePersistResult>() {
    @Override
    public void onSuccess(ImagePersistRequest request, ImagePersistResult result) {
        // success callback
    }
    @Override
    public void onFailure(ImagePersistRequest request, ClientException clientException, ServiceException serviceException) {
        // error callback
    }
});
```

8.13. Exception handling

OSS SDK for Android has two types of exceptions: `ClientException` and `ServiceException`.

ClientException

A `ClientException` indicates an exception that occurs when a client sends a request or transmit data to OSS. For example, a `ClientException` is returned when a request is sent under poor network conditions. A `ClientException` is also returned when an I/O exception occurs during object upload.

ServiceException

A `ServiceException` indicates a server error that is resolved from a server error message. A `ServiceException` includes the error code and message returned by OSS so that you can identify and resolve the error.

The following table describes the information included in a `ServiceException`.

Parameter	Description
Code	The error code returned by OSS.
Message	The detailed error message returned by OSS.
RequestId	The UUID used to uniquely identify the request. If you need help from OSS development engineers to handle an exception, provide them with the RequestId value.
HostId	The ID of the host in the accessed OSS cluster, which is the same as the host ID specified in the request.
rawMessage	The raw text of the message body in the HTTP response.

Common error codes

For more information about the common error codes of OSS, see [Error responses](#).

9.Go

9.1. Preface

This topic describes the sample code of the OSS Go SDK in various use cases.

SDK source code and API documents

For the source code of the OSS Go SDK, visit [GitHub](#). For more information, see [OSS Go SDK API documents](#).

Sample programs

The OSS Go SDK provides a variety of sample programs for your reference or use. The following table describes the sample programs provided.

Sample file	Content
new_bucket.go	Initialization
put_object.go	Upload objects, including Simple upload and Resumable upload
append_object.go	Append upload
get_object.go	Download objects, including Streaming download , Conditional download , and Download a compressed object
delete_object.go	Delete objects
copy_object.go	Copy objects , Copy objects , and Copy objects
list_objects.go	List objects , including listing objects with a specified prefix and listing a specified number of objects
object_meta.go	Manage object metadata
object_acl.go	Manage the ACL of an object
sign_url.go	Authorized access
cname_sample.go	Initialization
create_bucket.go	Create buckets
list_buckets.go	Bucket based on default or specified parameters
bucket_acl.go	Create buckets
bucket_referer.go	Add, query, and delete the Hotlink protection configurations for a bucket

Sample file	Content
bucket_logging.go	Add, query, and delete the Logging configurations for a bucket
bucket_lifecycle.go	Add, query, and delete the Lifecycle for a bucket
bucket_cors.go	Add, query, and delete the CORS configurations for a bucket
object_tagging.go	Configure object tagging , Obtain the tags added to an object , Delete the tags added to an object , and Object tagging and lifecycle rules .
bucket_requestpayment.go	Enable the Pay-by-requester
bucket_encryption.go	Add, query, and delete the Server-side encryption configurations for a bucket

9.2. Installation

Preparation

- Environment requirements

You must use Golang 1.6 and later versions.

Refer to [Installing Go from source](#) to download and install the compiling and running environment of Go. After Go is installed, set the GOPATH to the path of your code. For more information about GOPATH, run `go help gopath`.

- Check the version of Go

Run `go version` to check the version of Go.

Download SDK

- [Download from GitHub](#)
- [Download previous versions](#)

Install SDK

Run the following command to install OSS Go SDK:

```
go get github.com/aliyun/aliyun-oss-go-sdk/oss
```

 **Note** No messages are printed to the interface during installation. Please wait. In case of installation time-out, run the command again.

Check SDK version

Run the following code to check the version of OSS Go SDK:

```
package main
import (
    "fmt"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    fmt.Println("OSS Go SDK Version: ", oss.Version)
}
```

9.3. Initialization

OSSClient serves as the OSS Java client to manage OSS resources such as buckets and objects.

To create an OSSClient instance, you need to specify an endpoint. For more information about endpoints, see [Regions and endpoints](#) and [Bind a custom domain name](#).

The following table describes the parameters that you can configure:

Parameter	Description	Default value
UseCname	Specifies whether the CNAME is used.	No
Timeout	Specifies the request timeout time in seconds, including the connection timeout and socket read-write timeout.	30 seconds for the connection timeout and 60 seconds for the read-write timeout
SecurityToken	Specifies the securitytoken for temporary users	Null
EnableMD5	Specifies whether the MD5 verification is enabled. CRC verification is recommended because it is more efficient than MD5 verification.	No
EnableCRC	Specifies whether CRC verification is enabled.	Yes
Proxy	Specifies the proxy server, such as <code>http: // 192.168.0.1: 3128</code>	Null
AuthProxy	Specifies a proxy server that requires an account and a password to login.	Null

Use an OSS domain to create an OSSClient instance

Run the following code to create an OSSClient instance with a domain assigned by OSS:

```
import "github.com/aliyun/aliyun-oss-go-sdk/oss"
client, err := oss.New("<yourEndpoint>", "<yourAccessKeyId>", "<yourAccessKeySecret>")
if err != nil {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
```

Use a custom domain (CNAME) to create an OSSClient instance

For the complete code of custome domain, see [Git Hub](#).

Run the following code to create an OSSClient instance with CNAME:

```
import "github.com/aliyun/aliyun-oss-go-sdk/oss"
// oss.UseCname (true) indicates thath the CNAME is enabled. CNAME indicates a custom domain bound to a bucket.
client, err := oss.New("<yourEndpoint>", "<yourAccessKeyId>", "<yourAccessKeySecret>", oss.UseCname(true))
if err != nil {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
```



Notice If the CNAME is enabled, you cannot list buckets because the CNAME is bound to a certain bucket.

Use STS to create an OSSClient instance

Run the following code to create an OSSClient instance with Security Token Service (STS):

```
import "github.com/aliyun/aliyun-oss-go-sdk/oss"
client, err := oss.New("<yourEndpoint>", "<yourAccessKeyId>", "<yourAccessKeySecret>", oss.SecurityToken("<yourSecurityToken>"))
if err != nil {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
```

For more information, see [Authorized access](#).

Set timeout

Run the following code to set connection timeout:

```
import "github.com/aliyun/aliyun-oss-go-sdk/oss"
// oss.Timeout(10, 120) indicates that the HTTP connection timeout is set to 10 seconds (the default value is 30 seconds) and the HTTP read-write timeout is set to 120 seconds (the default value is 60 seconds). The value 0 indicates that timeout never occurs, which is not recommended.
client, err := oss.New("<yourEndpoint>", "<yourAccessKeyId>", "<yourAccessKeySecret>", oss.Timeout(10, 120))
if err != nil {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
```

CRC verification

CRC verification is enabled by default during uploads and downloads to ensure data integrity. Run the following code to disable CRC verification:

 **Warning** We recommend that you do not disable CRC verification. If you disable CRC verification, the data integrity during uploads and downloads is not guaranteed.

```
import "github.com/aliyun/aliyun-oss-go-sdk/oss"
client, err := oss.New("<yourEndpoint>", "<yourAccessKeyId>", "<yourAccessKeySecret>", oss.EnableCRC(false))
if err != nil {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
```

9.4. Quick start

This topic describes how to use the Object Storage Service (OSS) SDK for Go to perform routine operations, such as creating buckets, uploading objects, and downloading objects.

Create a bucket

A bucket is a global namespace in OSS. It is similar to a data container that stores files. The following code provides an example on how to create a bucket:

```
package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func handleError(err error) {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
func main() {
    // The China (Hangzhou) region is used in this example as the endpoint. Specify the actual endpoint.
    endpoint := "http://oss-cn-hangzhou.aliyuncs.com"
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access Object Storage Service (OSS) because the account has permissions on all API operations. We recommend that you use a Resource Access Management (RAM) user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
    accessKeyId := "<yourAccessKeyId>"
    accessKeySecret := "<yourAccessKeySecret>"
    bucketName := "<yourBucketName>"
    // Create an OSSClient instance.
    client, err := oss.New(endpoint, accessKeyId, accessKeySecret)
    if err != nil {
        handleError(err)
    }
    // Create the bucket.
    err = client.CreateBucket(bucketName)
    if err != nil {
        handleError(err)
    }
}
```

For more information about bucket naming conventions, see [Basic concepts](#). For more information about how to create a bucket, see [Create a bucket](#).

For more information about how to obtain an endpoint, see [Regions and endpoints](#).

Upload an object

The following code provides an example on how to upload an object to OSS:


```
package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func handleError(err error) {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
func main() {
    // The China (Hangzhou) region is used in this example as the endpoint. Specify the actual endpoint.
    endpoint := "http://oss-cn-hangzhou.aliyuncs.com"
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
    accessKeyId := "<yourAccessKeyId>"
    accessKeySecret := "<yourAccessKeySecret>"
    bucketName := "<yourBucketName>"
    // <yourObjectName> indicates the complete path of the object you want to upload to OSS, and must include the file extension of the object. Example: abc/efg/123.jpg.
    objectName := "<yourObjectName>"
    // <yourLocalFileName> consists of a local file path and an object name with extension. Example: /users/local/myfile.txt.
    localFileName := "<yourLocalFileName>"
    // Create an OSSClient instance.
    client, err := oss.New(endpoint, accessKeyId, accessKeySecret)
    if err != nil {
        handleError(err)
    }
    // Obtain the bucket.
    bucket, err := client.Bucket(bucketName)
    if err != nil {
        handleError(err)
    }
    // Upload the object.
    err = bucket.PutObjectFromFile(objectName, localFileName)
    if err != nil {
        handleError(err)
    }
}
```

For more information about how to upload objects, see [Overview](#).

Download an object

The following code provides an example on how to download an object to your local device:

```
package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func handleError(err error) {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
func main() {
    // The China (Hangzhou) region is used in this example as the endpoint. Specify the actual endpoint.
    endpoint := "http://oss-cn-hangzhou.aliyuncs.com"
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
    accessKeyId := "<yourAccessKeyId>"
    accessKeySecret := "<yourAccessKeySecret>"
    bucketName := "<yourBucketName>"
    // <yourObjectName> indicates the complete path of the object you want to download from OSS. The path must include the file extension of the object. For example, you can set <yourObjectName> to abc/efg/123.jpg.
    objectName := "<yourObjectName>"
    downloadedFileName := "<yourDownloadedFileName>"
    // Create an OSSClient instance.
    client, err := oss.New(endpoint, accessKeyId, accessKeySecret)
    if err != nil {
        handleError(err)
    }
    // Obtain the bucket.
    bucket, err := client.Bucket(bucketName)
    if err != nil {
        handleError(err)
    }
    // Download the object.
    err = bucket.GetObjectToFile(objectName, downloadedFileName)
    if err != nil {
        handleError(err)
    }
}
```

For more information about how to download objects, see [Overview](#).

List objects

The following code provides an example on how to list objects in a specified bucket. You can list a maximum of 100 objects by default.

```
package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func HandleError(err error) {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
func main() {
    // The China (Hangzhou) region is used in this example as the endpoint. Specify the actual endpoint.
    endpoint := "http://oss-cn-hangzhou.aliyuncs.com"
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
    accessKeyId := "<yourAccessKeyId>"
    accessKeySecret := "<yourAccessKeySecret>"
    // Create an OSSClient instance.
    client, err := oss.New(endpoint, accessKeyId, accessKeySecret)
    if err != nil {
        HandleError(err)
    }
    // Obtain the bucket.
    bucketName := "<yourBucketName>"
    bucket, err := client.Bucket(bucketName)
    if err != nil {
        handleError(err)
    }
    // List the objects in the bucket.
    marker := ""
    for {
        lsRes, err := bucket.ListObjects(oss.Marker(marker))
        if err != nil {
            HandleError(err)
        }
        // Display the listed objects. By default, 100 objects are returned at a time.
        for _, object := range lsRes.Objects {
            fmt.Println("Bucket: ", object.Key)
        }
        if lsRes.IsTruncated {
            marker = lsRes.NextMarker
        } else {
            break
        }
    }
}
```

For more information about how to list objects, see [List objects](#).

Delete an object

The following code provides an example on how to delete an object:

```
package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func handleError(err error) {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
func main() {
    // The China (Hangzhou) region is used in this example as the endpoint. Specify the actual endpoint.
    endpoint := "http://oss-cn-hangzhou.aliyuncs.com"
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
    accessKeyId := "<yourAccessKeyId>"
    accessKeySecret := "<yourAccessKeySecret>"
    bucketName := "<yourBucketName>"
    // <yourObjectName> indicates the complete path of the object that you want to delete from OSS. The path must include the extension of the object name. Example: abc/efg/123.jpg.
    objectName := "<yourObjectName>"
    // Create an OSSClient instance.
    client, err := oss.New(endpoint, accessKeyId, accessKeySecret)
    if err != nil {
        handleError(err)
    }
    // Obtain the bucket.
    bucket, err := client.Bucket(bucketName)
    if err != nil {
        handleError(err)
    }
    // Delete the object.
    err = bucket.DeleteObject(objectName)
    if err != nil {
        handleError(err)
    }
}
```

For more information about deleting objects, see [Delete objects](#) in [Manage objects](#).

9.5. Buckets

9.5.1. Create buckets

A bucket is a container used to store objects in Object Storage Service (OSS). Each object is contained in a bucket. This topic describes how to create a bucket.

Sample code

The following code provides an example on how to create a bucket named `examplebucket`:

```
package main

import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)

func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com. Specify the endpoint based on your business requirements.
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }

    // Create a bucket named examplebucket of which the storage class is IA (oss.StorageIA), the access control list (ACL) is public-read (oss.ACLPublicRead), and the redundancy type is zone-redundant storage (oss.RedundantZRS).
    err = client.CreateBucket("examplebucket", oss.StorageClass(oss.StorageIA), oss.ACL(oss.ACLPublicRead), oss.RedundancyType(oss.RedundancyZRS))
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
}
```

References

- For more information about the complete sample code that is used to create a bucket, visit [Git Hub](#).
- For more information about the API operation that you can call to create a bucket, see [Put Bucket](#).

9.5.2. List buckets

A bucket is a container that is used to store objects in Object Storage Service (OSS). Every object is contained in a bucket. Bucket names are listed in alphabetical order. You can list all buckets that belong to the current Alibaba Cloud account, buckets whose names contain a specified prefix, and a specified number of buckets.

List all buckets that belong to an Alibaba Cloud account

The following code provides an example on how to list all buckets that belong to an Alibaba Cloud account:

```
package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com. Specify your actual endpoint.
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // List all buckets that belong to the current Alibaba Cloud account.
    marker := ""
    for {
        lsRes, err := client.ListBuckets(oss.Marker(marker))
        if err != nil {
            fmt.Println("Error:", err)
            os.Exit(-1)
        }
        // By default, a maximum of 100 buckets are listed each time.
        for _, bucket := range lsRes.Buckets {
            fmt.Println("Bucket: ", bucket.Name)
        }
        if lsRes.IsTruncated {
            marker = lsRes.NextMarker
        } else {
            break
        }
    }
}
```

List buckets whose names contain a specified prefix

The following code provides an example on how to list buckets whose names contain a specified prefix:

```
package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com. Specify your actual endpoint.
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Specify the prefix to list buckets whose names contain the prefix.
    lsRes, err = client.ListBuckets(oss.Prefix("yourBucketPrefix"))
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Display the list of buckets.
    fmt.Println("Buckets with prefix: ", lsRes.Buckets)
    for _, bucket := range lsRes.Buckets {
        fmt.Println("Bucket with prefix: ", bucket.Name)
    }
}
```

List buckets whose names are alphabetically after the bucket specified by the marker parameter

The following code provides an example on how to list buckets whose names are alphabetically after the bucket specified by the marker parameter:

```
package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com. Specify your actual endpoint.
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Specify marker that is used to list buckets whose names are alphabetically after the bucket specified by the marker parameter.
    lsRes, err = client.ListBuckets(oss.Marker("yourBucketMarker"))
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Display the list of buckets.
    fmt.Println("My buckets with marker :", lsRes.Buckets)
    for _, bucket := range lsRes.Buckets {
        fmt.Println("Bucket with marker: ", bucket.Name)
    }
}
```

List a specified number of buckets

The following code provides an example on how to list a specified number of objects:


```
package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com. Specify your actual endpoint.
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Set the maximum number of buckets to list each time to 500. The default value is 100, and the maximum value is 1000.
    lsRes, err = client.ListBuckets(oss.MaxKeys(500))
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Display the list of buckets.
    fmt.Println("My buckets max num:", lsRes.Buckets)
    for _, bucket := range lsRes.Buckets {
        fmt.Println("Bucket with maxKeys: ", bucket.Name)
    }
}
```

References

- For more information about the complete sample code that is used to list buckets, visit [GitHub](#).
-

9.5.3. Determine whether buckets exist

A bucket is a container used to store objects in Object Storage Service (OSS). Every object is contained in a bucket. This topic describes how to determine whether a bucket exists.

```
package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Create an OSSClient instance.
    client, err := oss.New("<yourEndpoint>", "<yourAccessKeyId>", "<yourAccessKeySecret>")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Determine whether the bucket exists.
    isExist, err := client.IsBucketExist("<yourBucketName>")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    fmt.Println("IsBucketExist result : ", isExist)
}
```

9.5.4. Query the regions of buckets

A bucket is a container for objects stored in Object Storage Service (OSS). Every object is contained in a bucket. This topic describes how to query the region of a bucket.

Sample code

The following sample code shows how to query the region or location of a bucket:

```
package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com. Specify the actual endpoint.
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS, because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Query the region of the bucket.
    // Set yourBucketName to the name of the bucket.
    loc, err := client.GetBucketLocation("yourBucketName")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    fmt.Println("Bucket Location:", loc)
}
```

References

For more information about the API operation that you can call to query the region of a bucket, see [GetBucketLocation](#).

9.5.5. Query bucket information

A bucket is a container for objects stored in Object Storage Service (OSS). Every object is contained in a bucket. This topic describes how to query the information about a bucket.

The following code provides an example on how to query the information about a bucket, including the region and creation date of the bucket:

```
package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Create an OSSClient instance.
    client, err := oss.New("<yourEndpoint>", "<yourAccessKeyId>", "<yourAccessKeySecret>")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Query the information about a bucket, including the region (Region or Location), cre
    ation date (CreationDate), permissions (ACL), owner (Owner), and storage class (StorageClas
    s) of the bucket.
    res, err := client.GetBucketInfo("<yourBucketName>")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    fmt.Println("BucketInfo.Location: ", res.BucketInfo.Location)
    fmt.Println("BucketInfo.CreationDate: ", res.BucketInfo.CreationDate)
    fmt.Println("BucketInfo.ACL: ", res.BucketInfo.ACL)
    fmt.Println("BucketInfo.Owner: ", res.BucketInfo.Owner)
    fmt.Println("BucketInfo.StorageClass: ", res.BucketInfo.StorageClass)
    fmt.Println("BucketInfo.RedundancyType: ", res.BucketInfo.RedundancyType)
    fmt.Println("BucketInfo.ExtranetEndpoint: ", res.BucketInfo.ExtranetEndpoint)
    fmt.Println("BucketInfo.IntranetEndpoint: ", res.BucketInfo.IntranetEndpoint)
}
```

For more information about how to query bucket information, see [GetBucketInfo](#).

9.5.6. Manage bucket ACLs

A bucket is a container for objects stored in Object Storage Service (OSS). Every object is contained in a bucket. This topic describes how to configure and query the access control list (ACL) of a bucket.

Configure the ACL of a bucket

The following table describes the bucket ACLs.

ACL	Description	Method
private	Only the bucket owner and authorized users can perform read and write operations on objects in the bucket. Other users cannot access the objects in the bucket.	oss.ACLPrivate

ACL	Description	Method
public read	Only the bucket owner and authorized users of this bucket can perform read and write operations on objects in the bucket. Other users, including anonymous users, can only read objects in the bucket. Exercise caution when you set the bucket ACL to this value.	oss.ACLPublicRead
public read/write	All users, including anonymous users, can perform read and write operations on objects in the bucket. Exercise caution when you set the bucket ACL to this value.	oss.ACLPublicReadWrite

The following code provides an example on how to configure the ACL of a bucket:

```
package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com. Specify the endpoint based on your business requirements.
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Specify yourBucketName as the name of the bucket.
    // Set the ACL of the bucket to public read.
    err = client.SetBucketACL("yourBucketName", oss.ACLPublicRead)
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
}
```

For more information about how to configure the ACL of a bucket, see [PutBucketAcl](#).

Query the ACL of a bucket

The following code provides an example on how to query the ACL of a bucket:

```
package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com. Specify the endpoint based on your business requirements.
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Specify yourBucketName as the name of the bucket.
    // Query the ACL of a bucket.
    aclRes, err := client.GetBucketACL("yourBucketName")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    fmt.Println("Bucket ACL:", aclRes.ACL)
}
```

For more information about how to query the ACL of a bucket, see [GetBucketAcl](#).

9.5.7. Delete a bucket

If you no longer use a bucket, delete the bucket to stop unnecessary charges.



Warning Deleted buckets cannot be recovered. Exercise caution when you perform this operation.

Prerequisites

- All objects in the bucket are deleted.
 - For more information about how to manually delete a small number of objects, see [删除文件](#).
 - If you want to delete a large number of objects, we recommend that you configure lifecycle rules to delete the objects in batches. For more information about how to delete a large number of objects, see [Configure lifecycle rules](#).

 Notice

If versioning is enabled for the bucket that you want to delete, make sure that all versions of objects in the bucket are deleted. For more information about how to delete all versions of objects in a bucket, see [Configure versioning](#).

- Parts generated by multipart upload or resumable upload tasks in the bucket are deleted. For more information about how to delete parts in a bucket, see [Manage parts](#).
- All LiveChannels in the bucket are deleted. For more information about how to delete LiveChannels, see [DeleteLiveChannel](#).

Sample code

The following sample code shows how to delete a bucket named examplebucket:

```
package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com. Specify the actual endpoint.
    // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to access OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine Q&M. To create a RAM user, log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Specify the name of the bucket. Example: examplebucket.
    // Delete the bucket.
    err = client.DeleteBucket("examplebucket")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
}
```

References

- For more information about the complete sample code that is used to delete a bucket, visit [GitHub](#).
- For more information about the API operation that you can call to delete a bucket, see [DeleteBucket](#).

9.5.8. Configure bucket tagging

You can use tags to identify buckets used for different purposes and manage these buckets by category.

Configure tags for a bucket

The following code provides an example on how to configure tags for a bucket:

```
package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com. Specify the actual endpoint.
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access Object Storage Service (OSS) because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine Q&M. To create a RAM user, log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Initialize tagging configurations.
    tag1 := oss.Tag{
        Key:    "key1",
        Value: "value1",
    }
    tag2 := oss.Tag{
        Key:    "key2",
        Value: "value2",
    }
    tagging := oss.Tagging{
        Tags: []oss.Tag{tag1, tag2},
    }
    // Specify the bucket name. Example: examplebucket.
    // Configure tags for the bucket.
    err = client.SetBucketTagging("examplebucket", tagging)
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
}
```

Query the tags of a bucket

The following code provides an example on how to query the tags configured for a bucket:


```
package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com. Specify the actual endpoint.
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine Q&M. To create a RAM user, log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Specify the bucket name. Example: examplebucket.
    // Query the tags of the bucket.
    ret, err := client.GetBucketTagging("examplebucket")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Display the number of returned tags.
    fmt.Println("Tag length: ", len(ret.Tags))
}
```

List buckets that have specified tags

The following code provides an example on how to list buckets that have specified tags:

```
package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com. Specify the actual endpoint.
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine Q&M. To create a RAM user, log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Query buckets by specifying the TagKey parameter.
    ret, err := client.ListBuckets(oss.TagKey("yourTaggingKey"))
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Display the information about the buckets.
    for _, bucket := range ret.Buckets{
        fmt.Println("bucket:", bucket)
    }
    // Query buckets by specifying the TagKey and TagValue parameters.
    // The TagValue parameter must be configured together with the TagKey parameter and can be unspecified. If you do not specify the TagValue parameter, buckets are listed based on only the TagKey parameter.
    res, err := client.ListBuckets(oss.TagKey("yourTaggingKey"), oss.TagValue("yourTaggingValue"))
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Display the information about the listed buckets.
    for _, b := range res.Buckets{
        fmt.Println("bucket:", b)
    }
}
```

Delete the tags of a bucket

The following code provides an example on how to delete the tags configured for a bucket:

```
package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com. Specify the actual endpoint.
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine Q&M. To create a RAM user, log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Specify the bucket name. Example: examplebucket.
    // Delete the tags of the bucket.
    err = client.DeleteBucketTagging("examplebucket")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
}
```

References

- For more information about how to configure bucket tags, see [PutBucketTags](#).
- For more information about how to query the tags configured for a bucket, see [GetBucketTags](#).
- For more information about how to delete the tags configured for a bucket, see [DeleteBucketTags](#).

9.5.9. Bucket inventory

This topic describes how to add an inventory for a bucket and how to query, list, and delete inventories configured for a bucket.

Create an inventory for a bucket

The following code provides an example on how to create an inventory for a bucket:

```
package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region in which the bucket is located. For e
```

example, if the bucket is located in the China (Hangzhou) region, set `yourEndpoint` to `https://oss-cn-hangzhou.aliyuncs.com`. Specify the endpoint based on your business requirements.

// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.

```
client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
if err != nil {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
IsEnabled := true
// The following code provides an example on how to encrypt the inventory lists by using CMKs hosted in KMS.
//var invEncryption oss.InvEncryption
//var invSseOss oss.InvSseOss
//var invSseKms oss.InvSseKms
//invSseKms.KmsId = "<yourKmsId>" // Specify your CMK ID.
//invEncryption.SseOss = &invSseOss // Use OSS-managed keys (SSE-OSS) to encrypt the inventory lists.
//invEncryption.SseKms = &invSseKms // Use CMKs stored in KMS (SSE-KMS) to encrypt the inventory lists.
invConfig := oss.InventoryConfiguration{
    // Specify the name of the inventory. The name must be globally unique in the current bucket.
    Id: "yourInventoryId2",
    // Enable an inventory configuration.
    IsEnabled: &isEnabled,
    // Specify the rule used to filter the objects to include in the inventory lists. The following code provides an example on how to filter the objects by prefix.
    Prefix: "yourFilterPrefix",
    OSSBucketDestination: oss.OSSBucketDestination{
        // Specify the format of the exported inventory list.
        Format: "CSV",
        // Specify the ID of the account granted by the bucket owner. Example: 109885487000****.
        AccountId: "yourGrantAccountId",
        // Specify the RAM role to which the bucket owner grants permissions. Example: acs:ram::109885487000****:role/ram-test.
        RoleArn: "yourRoleArn",
        // Specify the name of the bucket in which you want to store the generated inventory lists.
        Bucket: "acs:oss::" + "yourDestBucketName",
        // Specify the prefix of the path in which you want to store the generated inventory lists.
        Prefix: "yourDestPrefix",
        // To encrypt inventory lists, refer to the following code.
        //Encryption: &invEncryption,
    },
    // Specify the frequency at which inventory lists are exported.
    Frequency: "Daily",
    // Specify whether the information about all versions of the objects is included in the inventory list.
    IncludedObjectVersions: "All",
```

```
OptionalFields: oss.OptionalFields{
    // Specify the configuration fields included in the inventory lists.
    Field: []string{
        "Size", "LastModifiedDate", "ETag", "StorageClass", "IsMultipartUploaded",
        "EncryptionStatus",
    },
},
}
// Specify yourBucketName as the name of the bucket for which you want to configure the
inventory.
err = client.SetBucketInventory("yourBucketName", invConfig)
if err != nil {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
}
```

Query an inventory configured for a bucket

The following code provides an example on how to query inventories configured for a bucket:

```

package main
import (
    "encoding/xml"
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com. Specify the endpoint based on your business requirements.
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Specify yourBucketName as the name of the bucket whose inventories you want to query.
    // Specify yourInventoryId as the name of the inventory.
    result, err := client.GetBucketInventory("yourBucketName", "yourInventoryId")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Display the inventory.
    bs, err := xml.MarshalIndent(result, " ", " ")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    fmt.Println(string(bs))
}

```

List multiple inventories configured for a bucket

 **Note** You can query up to 100 inventories by sending a request. To query more than 100 inventory rules, you must send multiple requests and use the token returned for each request as the parameter for the next request.

The following code provides an example on how to list multiple inventories configured for a bucket at a time:

```
package main
import (
    "encoding/xml"
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com. Specify the endpoint based on your business requirements.
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    var sumResult oss.ListInventoryConfigurationsResult
    vmarker := ""
    for {
        // Specify yourBucketName as the name of the bucket from which you want to list the inventories.
        listResult, err := client.ListBucketInventory("yourBucketName", vmarker)
        if err != nil {
            fmt.Println("Error:", err)
            os.Exit(-1)
        }
        sumResult.InventoryConfiguration = append(sumResult.InventoryConfiguration, listResult.InventoryConfiguration...)
        if listResult.IsTruncated != nil && *listResult.IsTruncated {
            vmarker = listResult.NextContinuationToken
        } else {
            break
        }
    }
    // Display the inventories configured for the bucket.
    bs, err := xml.MarshalIndent(sumResult, " ", " ")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    fmt.Println(string(bs))
}
```

Delete an inventory configured for a bucket

The following code provides an example on how to delete an inventory configured for a bucket:

```
package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com. Specify the endpoint based on your business requirements.
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Delete the inventory.
    // Specify yourBucketName as the name of the bucket from which you want to delete the inventory.
    // Specify yourInventoryId as the name of the inventory.
    err = client.DeleteBucketInventory("yourBucketName", "yourInventoryId")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
}
```

References

- For more information about the complete sample code for operations related to bucket inventories, visit [Git Hub](#).
- For more information about the API operation that is called to create an inventory, see [PutBucketInventory](#).
- For more information about the API operation that is called to query inventories configured for a bucket, see [GetBucketInventory](#).
- For more information about the API operation that is called to list multiple inventories configured for a bucket, see [ListBucketInventory](#).
- For more information about the API operation that is called to delete inventories configured for a bucket, see [DeleteBucketInventory](#).

9.5.10. Transfer acceleration

The transfer acceleration feature allows users worldwide to access objects stored in Object Storage Service (OSS) more quickly. This feature is applicable to scenarios where data needs to be transferred over long geographical distances. This feature can also be used to download or upload large objects that are gigabytes or terabytes in size.

For more information about transfer acceleration, see [Transfer acceleration](#) in OSS Developer Guide.

Enable transfer acceleration

The following code provides an example on how to enable transfer acceleration for a bucket named `examplebucket`:

```
package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func HandleError(err error) {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
func main() {
    // Create an OSSClient instance.
    // Set Endpoint to the endpoint of the region in which the bucket is located. For example
    // , if your bucket is located in the China (Hangzhou) region, set Endpoint to https://oss-cn-
    // hangzhou.aliyuncs.com.
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to
    // log on to OSS because the account has permissions on all API operations. We recommend that
    // you use a RAM user to call API operations or perform routine operations and maintenance. To
    // create a RAM user, log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        HandleError(err)
    }
    // Specify the bucket name.
    bucketName := "examplebucket"
    // Enable transfer acceleration for the bucket.
    // The Enabled parameter specifies whether to enable transfer acceleration. A value of tr
    // ue specifies that transfer acceleration is enabled. A value of false specifies that transfe
    // r acceleration is disabled.
    accConfig := oss.TransferAccConfiguration{}
    accConfig.Enabled = true
    err = client.SetBucketTransferAcc(bucketName, accConfig)
    if err != nil {
        HandleError(err)
    }
    fmt.Printf("set bucket transfer accelerate success\n")
}
```

For more information about the operation used to enable transfer acceleration, see

[PutBucketTransferAcceleration](#).

Query the status of transfer acceleration

The following code provides an example on how to query the transfer acceleration status of a bucket named `examplebucket`:

```
package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func HandleError(err error) {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
func main() {
    // Create an OSSClient instance.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        HandleError(err)
    }
    // Specify the bucket name.
    bucketName := "examplebucket"
    // Query the transfer acceleration status of the bucket.
    accConfig, err := client.GetBucketTransferAcc(bucketName)
    if err != nil {
        HandleError(err)
    }
    fmt.Printf("accConfigRes.Enabled:%T\n", accConfig.Enabled)
}
```

For more information about the operation used to query the transfer acceleration status of a bucket, see [GetBucketTransferAcceleration](#).

9.5.11. Bucket policy

This topic describes how to configure, query, and delete policies for a bucket.

Background information

Bucket policies provide resource-based authorization for users. Bucket policies apply to the following scenarios:

- Authorize RAM users of other accounts to access your OSS resources.

You can authorize RAM users of other accounts to access your OSS resources.

- Authorize anonymous users to access your OSS resources using specific IP addresses or IP ranges.

In some cases, you must authorize anonymous users to access OSS resources using specific IP addresses or IP ranges. For example, confidential documents of an enterprise are only allowed to be accessed within the enterprise but not in other regions. Previously, configuring RAM policies for every user was a tedious and complex task because of the potential for a large number of internal users. To resolve this issue, you can configure access policies with IP restrictions based on bucket policies to authorize a large number of users easily and efficiently.

For more information about bucket policy configurations and use cases, see [Configure bucket policies to authorize other users to access OSS resources](#). For more information about bucket policy syntax, see [Policy structure and syntax](#).

Configure bucket policies

The following sample code shows how to configure a policy for a bucket:

```
package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Create an OSSClient instance.
    client, err := oss.New("<yourEndpoint>", "<yourAccessKeyId>", "<yourAccessKeySecret>")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Configure a bucket policy.
    policyConfig := `
{
    "Statement": [
        {
            "Action": [
                "oss:GetObject",
                "oss:ListObjects"
            ],
            "Effect" : "Allow",
            "Resource" : ["acs:oss:*:*:/user1/*"]
        }
    ],
    "Version": "1"
}`
    // Configure the policy for the bucket.
    err = client.SetBucketPolicy("<yourBucketName>", policyConfig)
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    fmt.Println("SetBucketPolicy success")
}
```

Query bucket policies

The following sample code shows how to query the policies configured for a bucket:

```
package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Create an OSSClient instance.
    client, err := oss.New("<yourEndpoint>", "<yourAccessKeyId>", "<yourAccessKeySecret>")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Obtain the bucket policy.
    strPolicy, err := client.GetBucketPolicy("<yourBucketName>")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    fmt.Println("Bucket policy:", strPolicy)
}
```

Delete bucket policies

The following sample code shows how to delete the policies configured for a bucket:

```
package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Create an OSSClient instance.
    client, err := oss.New("<yourEndpoint>", "<yourAccessKeyId>", "<yourAccessKeySecret>")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Delete the bucket policy.
    err = client.DeleteBucketPolicy("<yourBucketName>")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    fmt.Println("DeleteBucketPolicy success")
}
```

9.5.12. Lifecycle

Object Storage Service (OSS) allows you to configure lifecycle rules to delete expired objects and parts or convert the storage class of expired objects to Infrequent Access (IA) or Archive to reduce your storage costs. This topic describes how to manage lifecycle rules for buckets.

Configure lifecycle rules

The following sample code provides an example on how to configure a lifecycle rule that matches objects by prefix and tag for the bucket named `examplebucket`:

```
package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com. Specify the endpoint based on your business requirements.
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Create a lifecycle rule and set id to rule1, enable to true, prefix to foo/, and expiry to Days 3. This rule applies to objects whose names are prefixed with foo/. These objects expire three days after they are last modified.
    rule1 := oss.BuildLifecycleRuleByDays("rule1", "foo/", true, 3)
    // If an object in a versioned bucket is a delete marker and has no other versions, the delete marker is removed.
    deleteMark := true
    expiration := oss.LifecycleExpiration{
        ExpiredObjectDeleteMarker: &deleteMark,
    }
    // Specify that objects are deleted when they expire 30 days after they become previous versions.
    versionExpiration := oss.LifecycleVersionExpiration{
        NoncurrentDays: 30,
    }
    // Specify that the storage class of objects is converted to IA 10 days after they become previous versions.
    versionTransition := oss.LifecycleVersionTransition{
        NoncurrentDays: 10,
        StorageClass:   "IA",
    }
    // Create a lifecycle rule and set id to rule2.
    rule2 := oss.LifecycleRule{
        ID:             "rule2",
        Prefix:         "yourObjectPrefix",
        Status:         "Enabled",
        Expiration:     expiration,
        VersionExpiration: versionExpiration,
        VersionTransition: versionTransition,
    }
}
```

```

        Status:      Enabled ,
        Expiration:   &expiration,
        NonVersionExpiration: &versionExpiration,
        NonVersionTransition: &versionTransition,
    }

    // Create a lifecycle rule and set id to rule3. This rule applies to objects that have
    // the tag with the tag key of tagA and tag value of A. These objects expire three days after
    // they are last modified.
    rule3 := oss.LifecycleRule{
        ID:      "rule3",
        Prefix: "",
        Status: "Enabled",
        Tags: []oss.Tag{
            oss.Tag{
                Key:   "tagA",
                Value: "A",
            },
        },
        Expiration: &oss.LifecycleExpiration{Days: 3},
    }

    // Configure the lifecycle rules.
    rules := []oss.LifecycleRule{rule1, rule2, rule3}
    // Specify the name of the bucket. Example: examplebucket.
    bucketName := "examplebucket"
    err = client.SetBucketLifecycle(bucketName, rules)
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
}

```

Query lifecycle rules

The following code provides an example on how to query the lifecycle rules configured for the bucket named examplebucket:

```
package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com. Specify the endpoint based on your business requirements.
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Specify the name of the bucket. Example: examplebucket.
    bucketName := "examplebucket"
    // Query the lifecycle rules configured for the bucket.
    lcRes, err := client.GetBucketLifecycle(bucketName)
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    fmt.Println("Lifecycle Rules:", lcRes.Rules)
}
```

Delete lifecycle rules

The following code provides an example on how to delete lifecycle rules configured for the bucket named examplebucket:

```
package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region in which the bucket is located. For e
    xample, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https:
    //oss-cn-hangzhou.aliyuncs.com. Specify the endpoint based on your business requirements.
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account t
    o access OSS because the account has permissions on all API operations. We recommend that y
    ou use a RAM user to call API operations or perform routine O&M. To create a RAM user, log
    on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Specify the name of the bucket. Example: examplebucket.
    bucketName := "examplebucket"
    // Delete the lifecycle rules.
    err = client.DeleteBucketLifecycle(bucketName)
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
}
```

References

- For more information about the complete sample code for lifecycle rules, visit [Git Hub](#).
- For more information about the API operation that you can call to configure lifecycle rules, see [Put Bucket Lifecycle](#).
- For more information about the API operation that you can call to query lifecycle rules, see [Get Bucket Lifecycle](#).
- For more information about the API operation that you can call to delete lifecycle rules, see [Delete Bucket Lifecycle](#).

9.5.13. Pay-by-requester

When pay-by-requester is enabled for a bucket in Object Storage Service (OSS), the requester is charged the requests and traffic fees instead of the bucket owner. The bucket owner is charged only the storage fees. You can enable pay-by-requester for a bucket to share data in the bucket without paying for the request and traffic fees incurred by the access to your bucket.

Enable pay-by-requester

The following code provides an example on how to enable pay-by-requester for a bucket:


```
package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com. Specify the actual endpoint.
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS, because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Initialize the pay-by-requester mode.
    reqPayConf := oss.RequestPaymentConfiguration{
        Payer: "Requester",
    }
    // Enable pay-by-requester for the bucket.
    err = client.SetBucketRequestPayment("<yourBucketName>", reqPayConf)
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
}
```

Query pay-by-requester configurations

The following code provides an example on how to query the pay-by-requester configurations of a bucket:

```

package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com. Specify the actual endpoint.
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS, because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Query the pay-by-requester configurations of the bucket.
    ret, err := client.GetBucketRequestPayment("yourBucketName")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Specify that the pay-by-requester configurations of the bucket are displayed.
    fmt.Println("Bucket request payer:", ret.Payer)
}

```

Specify that third parties are charged for access to objects

If you specify that third parties are charged for access to objects in a bucket, requesters must include the `x-oss-request-payer:requester` header in the HTTP requests to perform operations on your objects. If this header is not included, an error is returned.

The following code provides an example on how to specify that third parties are charged when they access objects by calling the `PutObject`, `GetObject`, and `DeleteObject` operations. You can use the method to specify that third parties are charged when they perform read and write operations on objects by calling other API operations in the similar way.

```

package main
import (
    "fmt"
    "os"
    "io/ioutil"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
    "strings"
)
func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com. Specify the actual endpoint.

```

```

// Set yourAccessKeyId and yourAccessKeySecret to the AccessKey pair of the third-party.
payerClient, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
if err != nil {
    fmt.Println("New Error:", err)
    os.Exit(-1)
}
// Specify the bucket name.
payerBucket, err := payerClient.Bucket("examplebucket")
if err != nil {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
// If the bucket owner enables the pay-by-requester mode, requesters must include the oss
.RequestPayer(oss.Requester) parameter in the requests to access the authorized content.
// If the bucket owner does not enable the pay-by-requester mode, requesters do not need
to include the oss.RequestPayer(oss.Requester) parameter in the requests to access the auth
orized content.
// Upload an object.
// Specify the full path of the object. The path cannot contain the bucket name. Example:
exampledir/exampleobject.txt.
key := "exampledir/exampleobject.txt"
err = payerBucket.PutObject(key, strings.NewReader("objectValue"), oss.RequestPayer("requ
ester"))
if err != nil {
    fmt.Println("put Error:", err)
    os.Exit(-1)
}
// List all objects in the bucket.
lor, err := payerBucket.ListObjects(oss.RequestPayer(oss.Requester))
if err != nil {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
// Specify that the names of objects in the bucket are displayed.
for _, l := range lor.Objects {
    fmt.Println("the Key name is :", l.Key)
}
// Download the object.
body, err := payerBucket.GetObject(key, oss.RequestPayer(oss.Requester))
if err != nil {
    fmt.Println("Get Error:", err)
    os.Exit(-1)
}
// You must close the stream obtained by using the GetObject API operation after the obje
ct is read. Otherwise, connection leaks may occur. Consequently, no connections are availab
le and your application cannot work.
defer body.Close()
// Read and display the obtained content.
data, err := ioutil.ReadAll(body)
if err != nil {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
fmt.Println("data:", string(data))

```

```
// Delete the object.
err = payerBucket.DeleteObject(key, oss.RequestPayer(oss.Requester))
if err != nil {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
```

References

- For more information about the complete sample code that is used to enable pay-by-requester for a bucket, visit [GitHub](#).
- For more information about the API operation that you can call to enable pay-by-requester for a bucket, see [PutBucketRequestPayment](#).
- For more information about the API operation that you can call to query the pay-by-requester configurations of a bucket, see [GetBucketRequestPayment](#).

9.5.14. Retention policies

You can configure a time-based retention policy for a bucket. A retention policy has a protection period ranging from one day to 70 years. This topic describes how to create, query, and lock a retention policy.

Background information

OSS supports the Write Once Read Many (WORM) strategy that prevents an object from being deleted or overwritten for a specified period of time.

If a retention policy is not locked within 24 hours after it is created, the retention policy becomes invalid. After the retention policy configured for a bucket is locked, you can upload objects to or read objects from the bucket. However, you cannot delete objects in the bucket or the retention policy within the retention period of the policy. The retention period of the policy cannot be shortened but only be extended. For more information about retention policies, see [Retention policies](#).

Create a retention policy

 **Notice** A bucket cannot have versioning and retention policies configured at the same time. Therefore, before you create a retention policy for a bucket, make sure that versioning is disabled for the bucket.

The following code provides an example on how to create a retention policy:

```
package main
import (
    "fmt"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
    "os"
)
func HandleError(err error) {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com. Specify the endpoint based on your business requirements.
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS, because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
    client, err := oss.New("<yourEndpoint>", "<yourAccessKeyId>", "<yourAccessKeySecret>")
    if err != nil {
        HandleError(err)
    }
    // Specify the name of the bucket for which you want to create a retention policy.
    bucketname := "<yourBucketName>"
    // Set the retention period to 60 days.
    result, err := client.InitiateBucketWorm(bucketname, 60)
    if err != nil {
        HandleError(err)
    }
    fmt.Println(result)
}
```

Delete an unlocked retention policy

The following code provides an example on how to delete an unlocked retention policy:

```
package main
import (
    "fmt"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
    "os"
)
func HandleError(err error) {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com. Specify the endpoint based on your business requirements.
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS, because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
    client, err := oss.New("<yourEndpoint>", "<yourAccessKeyId>", "<yourAccessKeySecret>")
    if err != nil {
        HandleError(err)
    }
    // Specify the name of the bucket for which you want to delete the retention policy.
    bucketname := "<yourBucketName>"
    err = client.AbortBucketWorm(bucketname)
    if err != nil {
        HandleError(err)
    }
}
```

Lock a retention policy

The following code provides an example on how to lock a retention policy:

```
package main
import (
    "fmt"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
    "os"
)
func HandleError(err error) {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com. Specify the endpoint based on your business requirements.
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS, because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
    client, err := oss.New("<yourEndpoint>", "<yourAccessKeyId>", "<yourAccessKeySecret>")
    if err != nil {
        HandleError(err)
    }
    // Specify the name of the bucket for which you want to lock the retention policy.
    bucketname := "<yourBucketName>"
    wormConfig, err := client.GetBucketWorm(bucketname)
    err = client.CompleteBucketWorm(bucketname, wormConfig.WormId)
    if err != nil {
        HandleError(err)
    }
}
```

Query a retention policy

The following code provides an example on how to query a retention policy:

```

package main
import (
    "fmt"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
    "os"
)
func HandleError(err error) {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com. Specify the endpoint based on your business requirements.
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS, because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
    client, err := oss.New("<yourEndpoint>", "<yourAccessKeyId>", "<yourAccessKeySecret>")
    if err != nil {
        HandleError(err)
    }
    // Specify the name of the bucket for which you want to query the retention policy.
    bucketname := "<yourBucketName>"
    wormConfig, err := client.GetBucketWorm(bucketname)
    if err != nil {
        HandleError(err)
    }
    // Query the ID of the retention policy.
    fmt.Println(wormConfig.WormId)
    // Query the created time of the retention policy.
    fmt.Println(wormConfig.CreationDate)
    // Query the state of the retention policy. InProgress indicates that the retention policy is not locked. Locked indicates that the retention policy is locked.
    fmt.Println(wormConfig.State)
    // Query the retention period of objects in the bucket whose retention policy is locked
    .
    fmt.Println(wormConfig.RetentionPeriodInDays)
}

```

Extend the retention period of objects in a bucket whose retention policy is locked.

The following code provides an example on how to extend the retention period of objects in a bucket whose retention policy is locked:


```
package main
import (
    "fmt"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
    "os"
)
func HandleError(err error) {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com. Specify the endpoint based on your business requirements.
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS, because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
    client, err := oss.New("<yourEndpoint>", "<yourAccessKeyId>", "<yourAccessKeySecret>")
    if err != nil {
        HandleError(err)
    }
    // Specify the name of the bucket for which you want to extend the retention period.
    bucketname := "<yourBucketName>"
    wormConfig, err := client.GetBucketWorm(bucketname)
    // Extend the retention period of objects in the bucket whose retention policy is locked to 30 days.
    err = client.ExtendBucketWorm(bucketName, 30, wormConfig.WormId)
    if err != nil {
        HandleError(err)
    }
}
```

References

- For more information about the complete sample code for a retention policy, visit [GitHub](#).
- For more information about how to create a retention policy, see [InitiateBucketWorm](#).
- For more information about how to delete an unlocked retention policy, see [AbortBucketWorm](#).
- For more information about how to lock a retention policy, see [CompleteBucketWorm](#).
- For more information about how to query a retention policy, see [GetBucketWorm](#).
- For more information about how to extend the retention period of objects in a bucket whose retention policy is locked, see [ExtendBucketWorm](#).

9.5.15. Hotlink protection

This topic describes how to use hot link protection.

To prevent your data on OSS from being leached, OSS supports hot link protection through the referer field settings in the HTTP header, including the following parameters:

- **Referer whitelist**: Used to allow access only for specified domains to OSS data.
- **Empty referer**: Determines whether the referer can be empty. If it is not allowed, only requests with the referer filed in their HTTP or HTTPS headers can access OSS data.

For more information about hotlink protection, see [Hotlink protection](#) in OSS Developer Guide. For more information about the complete code that is used to configure hotlink protection, visit [GitHub](#).

Configure hotlink protection

The following code provides an example on how to configure a Referer whitelist for a bucket:

```
package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Create an OSSClient instance.
    client, err := oss.New("<yourEndpoint>", "<yourAccessKeyId>", "<yourAccessKeySecret>")
    if err!=nil{
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    bucketName := "<yourBucketName>"
    // Add Referers to the Referer whitelist and specify that empty Referer fields are not
    // allowed. You can use an asterisk (*) or a question mark (?) as a wildcard to set the Referer
    // parameter.
    referers := []string{"http://www.aliyun.com",
                        "http://www.???.aliyuncs.com",
                        "http://www.*.com"}
    err = client.SetBucketReferer(bucketName, referers, false)
    if err!=nil{
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
}
```

Query hotlink protection configurations

The following code provides an example on how to query a Referer whitelist of a bucket:

```
package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Create an OSSClient instance.
    client, err := oss.New("<yourEndpoint>", "<yourAccessKeyId>", "<yourAccessKeySecret>")
    if err!=nil{
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    bucketName := "<yourBucketName>"
    // Query hotlink protection configurations.
    refRes, err := client.GetBucketReferer(bucketName)
    if err!=nil{
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    fmt.Println("Referers: ", refRes.RefererList)
    fmt.Println("AllowEmptyReferer: ", refRes.AllowEmptyReferer)
}
```

Delete hotlink protection configurations

The following code provides an example on how to clear a Referer whitelist of a bucket:

```
package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Create an OSSClient instance.
    client, err := oss.New("<yourEndpoint>", "<yourAccessKeyId>", "<yourAccessKeySecret>")
    if err!=nil{
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    bucketName := "<yourBucketName>"
    // Delete hotlink protection configurations. // You cannot directly delete hotlink protection configurations. To delete hotlink protection configurations, you must create a rule that allows empty Referer fields to replace the existing rule.
    err = client.SetBucketReferer(bucketName, []string{}, true)
    if err!=nil{
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
}
```

9.5.16. Logging

A large number of logs are generated when Object Storage Service (OSS) resources are accessed. After you enable and configure logging for a bucket, OSS generates logs every hour in accordance with predefined naming conventions and then stores the logs as objects in a specified bucket.

Enable logging for a bucket

The following code provides an example on how to enable logging for a bucket:

```
package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com. Specify the actual endpoint.
    // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in Object Storage Service (OSS) is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Specify the name of the bucket for which you want to enable logging. Example: examplebucket.
    bucketName := "examplebucket"
    // Specify the name of the destination bucket used to store the log objects. The source bucket and the destination bucket can be the same bucket or different buckets.
    targetBucketName := "destbucket"
    // Specify that the log objects are stored in the log/ directory. If you specify this parameter, log objects are stored in a specified directory of the destination bucket. If you do not specify this parameter, log objects are stored in the root directory of the destination bucket.
    targetPrefix := "log/"
    // Enable logging for the bucket.
    err = client.SetBucketLogging(bucketName, targetBucketName, targetPrefix, true)
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
}
```

View the logging configurations of a bucket

The following code provides an example on how to view the logging configurations of a bucket:

```
package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com. Specify the actual endpoint.
    // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Specify the bucket name. Example: examplebucket.
    bucketName := "examplebucket"
    // View the logging configurations of the bucket.
    logRes, err := client.GetBucketLogging(bucketName)
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    fmt.Println("Target Bucket: ", logRes.LoggingEnabled.TargetBucket)
    fmt.Println("Target Prefix: ", logRes.LoggingEnabled.TargetPrefix)
}
```

Disable logging for a bucket

The following code provides an example on how to disable logging for a bucket:

```
package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com. Specify the actual endpoint.
    // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Specify the bucket name. Example: examplebucket.
    bucketName := "examplebucket"
    // Disable logging for the bucket.
    err = client.DeleteBucketLogging(bucketName)
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
}
```

References

- For the complete sample code that is used to configure logging for a bucket, visit [GitHub](#).
- For more information about the API operation that you can call to enable logging for a bucket, see [PutBucketLogging](#).
- For more information about the API operation that you can call to view the logging configurations of a bucket, see [GetBucketLogging](#).
- For more information about the API operation that you can call to disable logging for a bucket, see [DeleteBucketLogging](#).

9.5.17. Static website hosting (mirroring-based back-to-origin)

You can enable static website hosting for buckets and configure mirroring-based back-to-origin rules. After you host a static website on a bucket, you can access the bucket to visit the website. You are automatically redirected to the specified index page or error page. After mirroring-based back-to-origin rules are configured and take effect, you can use mirroring-based back-to-origin to seamlessly migrate data to Object Storage Service (OSS).

Static website hosting

Static websites are websites in which all web pages consist only of static content, including scripts such as JavaScript code that can be run on the client. You can use the static website hosting feature to host your static website in an OSS bucket and use the domain name of the bucket to access the website.

- The following code provides an example on how to configure static website hosting:

```
package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region in which the bucket is located. For
    // example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to ht
    // ps://oss-cn-hangzhou.aliyuncs.com. Specify the endpoint based on your business requiremen
    // ts.
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account
    // to access OSS because the account has permissions on all API operations. We recommend tha
    // t you use a RAM user to call API operations or perform routine operations and maintenance
    // . To create a RAM user, log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Specify the name of the bucket. Example: examplebucket.
    bucketName := "examplebucket"
    // Set the default homepage to index.html and the default 404 page to error.html for
    // the static website that is hosted in the bucket.
    err = client.SetBucketWebsite(bucketName, "index.html", "error.html")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
}
```

Configure static website hosting

- The following code provides an example on how to query static website hosting configurations:

```
package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region in which the bucket is located. For
    // example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to ht
    // ps://oss-cn-hangzhou.aliyuncs.com. Specify the endpoint based on your business requiremen
    // ts.
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account
    // to access OSS because the account has permissions on all API operations. We recommend tha
    // t you use a RAM user to call API operations or perform routine operations and maintenance
    // . To create a RAM user, log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Specify the name of the bucket. Example: examplebucket.
    bucketName := "examplebucket"
    // Query the static website hosting configurations.
    wsRes, err := client.GetBucketWebsite(bucketName)
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    fmt.Println("indexWebsite: ", wsRes.IndexDocument.Suffix)
    fmt.Println("errorWebsite: ", wsRes.ErrorDocument.Key)
}
```

Query static website hosting configurations

- The following code provides an example on how to delete static website hosting configurations:


```

package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region in which the bucket is located. For
    // example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to htt
    // ps://oss-cn-hangzhou.aliyuncs.com. Specify the endpoint based on your business requiremen
    // ts.
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account
    // to access OSS because the account has permissions on all API operations. We recommend tha
    // t you use a RAM user to call API operations or perform routine operations and maintenance
    // . To create a RAM user, log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Specify the name of the bucket. Example: examplebucket.
    bucketName := "examplebucket"
    // Delete the static website hosting configurations.
    err = client.DeleteBucketWebsite(bucketName)
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
}

```

Delete static website hosting configurations

Mirroring-based back-to-origin

Mirroring-based back-to-origin is used to seamlessly migrate data to OSS. For example, you can migrate services from a self-managed origin or from another cloud service to OSS without service interruption. You can use mirroring-based back-to-origin rules during migration to obtain the data that is not migrated to OSS. This ensures service continuity.

- If a requester attempts to access an object in the specified bucket and the object does not exist, you can specify the URL of the object in the origin and back-to-origin conditions to allow the requester to obtain the object. For example, a bucket named `examplebucket` is located in the China (Hangzhou) region. When a requester attempts to access an object in the `examplefolder` directory of the root directory of the bucket and the object does not exist, the requester is redirected to `www.example.com` to access the required object that is stored in the `examplefolder` directory of the origin.

The following code provides an example on how to configure mirroring-based back-to-origin rules in the preceding scenario:

```

package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"

```

```

)
func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region in which the bucket is located. For
    // example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to ht
    // ps://oss-cn-hangzhou.aliyuncs.com. Specify the endpoint based on your business requiremen
    // ts.
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account
    // to access OSS because the account has permissions on all API operations. We recommend tha
    // t you use a RAM user to call API operations or perform routine operations and maintenance
    // . To create a RAM user, log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Specify the name of the bucket. Example: examplebucket.
    bucketName := "examplebucket"
    putXml := `
        <IndexDocument>
            <Suffix>index.html</Suffix>
            <SupportSubDir>true</SupportSubDir>
            <Type>0</Type>
        </IndexDocument>
        <ErrorDocument>
            <Key>error.html</Key>
            <HttpStatus>404</HttpStatus>
        </ErrorDocument>
        <RoutingRules>
            <RoutingRule>
                <RuleNumber>1</RuleNumber>
                <Condition>
                    <!-- Specify the prefix of object names. Only the objects whose names contain the specified prefix match the rule. -->
                    <KeyPrefixEquals>examplefolder/</KeyPrefixEquals>
                    <!-- Specify the HTTP status code. The rule is matched only when the specified object is accessed and HTTP status code 404 is returned. -->
                    <HttpErrorCodeReturnedEquals>404</HttpErrorCodeReturnedEquals>
                </Condition>
                <Redirect>
                    <!-- Specify the redirect type. -->
                    <RedirectType>Mirror</RedirectType>
                    <PassQueryString>true</PassQueryString>
                    <!-- Specify the origin URL for the mirroring-based back-to-origin rule. In this example, the origin URL is https://www.example.com/. -->
                    <MirrorURL>http://example.com/</MirrorURL>-->
                    <!-- Specify the status codes such as 4xx and 5xx that OSS must pass through with the response body to the client after OSS receives the status codes from the origin. -->
                    <TransparentMirrorResponseCodes>400,404</TransparentMirrorResponseCodes>
                    <!-- Specify whether to include parameters of the original request in the redirect request when the system runs the redirection rule or the mirroring-based back-to-origin rule. -->
                    <PassQueryString>true</PassQueryString>
                </Redirect>
            </RoutingRule>
        </RoutingRules>
    </WebsiteConfiguration>
```

```

        <!-- This parameter plays the same role as PassQueryString and has a higher
        priority than PassQueryString. This parameter takes effect only when RedirectType is set
        to Mirror. -->
        <MirrorPassQueryString>true</MirrorPassQueryString>
        <!-- Specify the HTTP status code in the response. This parameter takes effect
        only when RedirectType is set to External or AliCDN. -->
        <HttpRedirectCode>302</HttpRedirectCode>
        <!-- Specify the domain name that is used for redirection, which must comply
        with the naming conventions for domains. -->
        <HostName>oss.aliyuncs.com</HostName>
        <!-- Specify the protocol that is used for redirection. This parameter takes
        effect only when RedirectType is set to External or AliCDN. -->
        <Protocol>https</Protocol>
        <!-- Specify the string that is used to replace the object name when the request
        is redirected. This parameter can be set to a variable. -->
        <ReplaceKeyWith>key.jpg</ReplaceKeyWith>
        <!-- If this parameter is set to true, the prefix of the object name is replaced
        with the value specified by ReplaceKeyPrefixWith. -->
        <EnableReplacePrefix>true</EnableReplacePrefix>
        <!-- Specify the string that is used to replace the prefix of the object name
        when the request is redirected. -->
        <ReplaceKeyPrefixWith>examplebucket</ReplaceKeyPrefixWith>
        <!-- Specify whether to check the MD5 hash of the body of the response that
        is returned by the origin. This parameter takes effect only when RedirectType is set to Mirror. -->
        <MirrorCheckMd5>false</MirrorCheckMd5>
        <!-- Specify whether to redirect the access to the address that is specified
        by Location if the origin returns HTTP status code 3xx. This parameter takes effect only
        when RedirectType is set to Mirror. -->
        <MirrorFollowRedirect>true</MirrorFollowRedirect>
        <MirrorHeaders>
            <!-- Specify whether to pass through all request headers to the origin. This
            parameter takes effect only when RedirectType is set to Mirror. -->
            <PassAll>true</PassAll>
            <!-- Specify the headers to pass through to the origin. This parameter takes
            effect only when RedirectType is set to Mirror. -->
            <Pass>myheader-key1</Pass>
            <Pass>myheader-key2</Pass>
            <!-- Specify the headers that are not allowed to pass through to the origin.
            This parameter takes effect only when RedirectType is set to Mirror. -->
            <Remove>myheader-key3</Remove>
            <Remove>myheader-key4</Remove>
            <!-- Specify the headers that are passed to the origin. The specified headers
            are passed to the origin regardless of whether they are included in the request. -->
            <Set>
                <Key>myheader-key5</Key>
                <Value>myheader-value5</Value>
            </Set>
        </MirrorHeaders>
    </Redirect>
</RoutingRule>
</RoutingRules>
</WebsiteConfiguration>
err = client.SetBucketWebsiteXml(bucketName,putXml)

```

```
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
}
```

Configure mirroring-based back-to-origin

- The following code provides an example on how to query mirroring-based back-to-origin configurations:

```
package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region in which the bucket is located. For
    // example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to ht
    // ps://oss-cn-hangzhou.aliyuncs.com. Specify the endpoint based on your business requiremen
    // ts.
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account
    // to access OSS because the account has permissions on all API operations. We recommend tha
    // t you use a RAM user to call API operations or perform routine operations and maintenance
    // . To create a RAM user, log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Specify the name of the bucket. Example: examplebucket.
    bucketName := "examplebucket"
    data, err := client.GetBucketWebsiteXml(bucketName)
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    fmt.Println(data)
}
```

Query mirroring-based back-to-origin configurations

- The following code provides an example on how to delete mirroring-based back-to-origin configurations:

```
package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region in which the bucket is located. For
    // example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to htt
    // ps://oss-cn-hangzhou.aliyuncs.com. Specify the endpoint based on your business requiremen
    // ts.
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account
    // to access OSS because the account has permissions on all API operations. We recommend tha
    // t you use a RAM user to call API operations or perform routine operations and maintenance
    // . To create a RAM user, log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Specify the name of the bucket. Example: examplebucket.
    bucketName := "examplebucket"
    // Delete mirroring-based back-to-origin configurations.
    err = client.DeleteBucketWebsite(bucketName)
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
}
```

Delete mirroring-based back-to-origin configurations

References

- For more information about the complete sample code of static website hosting and mirroring-based back-to-origin, visit [GitHub](#).
- For more information about the API operation that you can call to configure static website hosting or mirroring-based back-to-origin, see [PutBucketWebsite](#).
- For more information about the API operation that you can call to query static website hosting or mirroring-based back-to-origin configurations, see [GetBucketWebsite](#).
- For more information about the API operation that you can call to delete static website hosting or mirroring-based back-to-origin configurations, see [DeleteBucketWebsite](#).

9.5.18. CRR

Enable CRR

The following code provides an example on how to enable CRR for a bucket:

```
package main
import (
    "fmt"
```

```

    "github.com/aliyun/aliyun-oss-go-sdk/oss"
    "os"
)
func HandleError(err error) {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
// Enable CRR for a source bucket.
func main() {
    // Set yourEndpoint to the endpoint of the region in which the source bucket is located
    . For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to
    https://oss-cn-hangzhou.aliyuncs.com. Specify the endpoint based on your business requireme
    nts.

    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account t
    o access OSS because the account has permissions on all API operations. We recommend that y
    ou use a RAM user to call API operations or perform routine operations and maintenance. To
    create a RAM user, log on to the RAM console.

    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        HandleError(err)
    }
    // Specify the name of the source bucket.
    srcbucketName := "srcexamplebucket"
    // Specify that only data that is written to the source bucket after the rule is create
    d and operations performed on objects in the bucket are synchronized to the destination buc
    ket. Historical data in the source bucket is not synchronized to the destination bucket.
    putXml := `<?xml version="1.0" encoding="UTF-8"?>
    <ReplicationConfiguration>
        <Rule>
            <PrefixSet>
                <!-- Specify that objects whose names contain one of the following prefixes ar
                e replicated to the destination bucket: prefix_1 and prefix_2. After you configure the Pref
                ix parameter, only objects whose names contain the specified prefix are replicated to the d
                estination bucket. -->
                <!-- To replicate all objects from the source bucket to the destination bucket
                , do not configure the Prefix parameter.
                <Prefix>prefix_1</Prefix>
                <Prefix>prefix_2</Prefix>
            </PrefixSet>
            <!-- Specify the operations that can be synchronized to the destination bucket. Th
            e default value is ALL, which indicates that all operations performed on objects in the sou
            rce bucket are replicated to the destination bucket. -->
            <Action>ALL</Action>
            <Destination>
                <!-- Specify the destination bucket to which the data is replicated. -->
                <Bucket>destexamplebucket</Bucket>
                <!-- Specify the region in which the destination bucket is located. -->
                <Location>oss-cn-beijing</Location>
                <!-- Specify the link that is used to transfer data during data replication. I
                n this example, this parameter is set to oss_acc, which indicates that the link used for da
                ta transmission is accelerated. -->
                <TransferType>oss_acc</TransferType>
            </Destination>
            <!-- By default, historical data is synchronized. In this example, this parameter

```

```

    <!-- Specify whether historical data is synchronized. In this example, this parameter
is set to disabled, which indicates that historical data is not synchronized. -->
    <HistoricalObjectReplication>disabled</HistoricalObjectReplication>
    <!-- Specify the name of the role that you authorize OSS to use to replicate data.
If you want to use SSE-KMS to encrypt the objects that are replicated to the destination bu
cket, you must configure this parameter. -->
    <SyncRole>aliyunramrole</SyncRole>
    <SourceSelectionCriteria>
        <SseKmsEncryptedObjects>
            <!-- Specify whether to replicate objects that are encrypted by using SSE-KMS.
-->
            <Status>Enabled</Status>
        </SseKmsEncryptedObjects>
    </SourceSelectionCriteria>
    <EncryptionConfiguration>
        <!-- Specify the CMK ID used in SSE-KMS. If Status is set to Enabled, you must
configure this parameter. -->
        <ReplicaKmsKeyID>c4d49f85-ee30-426b-a5ed-95e9139d****</ReplicaKmsKeyID>
    </EncryptionConfiguration>
</Rule>
</ReplicationConfiguration>`
err = client.PutBucketReplication(srcbucketName, putXml)
if err != nil {
    HandleError(err)
}
}

```

Query the CRR rules configured for a bucket

The following code provides an example on how to query the CRR rules configured for a bucket:

```

package main
import (
    "fmt"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
    "os"
)
func HandleError(err error) {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
// Query the CRR rules configured for the bucket.
func main() {
    // Set yourEndpoint to the endpoint of the region in which the source bucket is located
    . For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to
    https://oss-cn-hangzhou.aliyuncs.com. Specify the endpoint based on your business requireme
    nts.
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account t
    o access OSS because the account has permissions on all API operations. We recommend that y
    ou use a RAM user to call API operations or perform routine operations and maintenance. To
    create a RAM user, log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        HandleError(err)
        os.Exit(-1)
    }
    // Specify the name of the source bucket.
    srcbucketName := "yourSourceBucket"
    result, err := client.GetBucketReplication(srcbucketName)
    if err != nil {
        HandleError(err)
        os.Exit(-1)
    }
    fmt.Println(result)
}

```

Query the progress of a CRR task that is performed on a bucket

Cross-region replication progress includes the progress of historical and real-time data synchronization.

- Percent signs (%) indicate the progress of historical data synchronization. This function only works for buckets for which historical data synchronization has been enabled.
- Time points (the point of latest time the data is written) indicate the progress of real-time data synchronization. If a time point is displayed, the data created before this time point has been synchronized.

The following code provides an example on how to query the progress of a CRR task that is performed on a source bucket:


```
package main
import (
    "fmt"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
    "os"
)
func HandleError(err error) {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
// Query the progress of the CRR task that is performed on the source bucket.
func main() {
    // Set yourEndpoint to the endpoint of the region in which the source bucket is located
    . For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to
    https://oss-cn-hangzhou.aliyuncs.com. Specify the endpoint based on your business requireme
    nts.
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account t
    o access OSS because the account has permissions on all API operations. We recommend that y
    ou use a RAM user to call API operations or perform routine operations and maintenance. To
    create a RAM user, log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        HandleError(err)
    }
    // Specify the name of the source bucket.
    srcbucketName := "yourSourceBucket"
    // Specify the CRR rule ID that is returned by the GetBucketReplication operation.
    result, err := client.GetBucketReplicationProgress(srcbucketName, "e047ce28-6806-4131-b1d
a-30142116****")
    if err != nil {
        HandleError(err)
    }
    fmt.Println(result)
}
```

Query the regions in which CRR can be performed

The following code provides an example on how to query the regions in which CRR can be performed:

```
package main
import (
    "fmt"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
    "os"
)
func HandleError(err error) {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
// Query the regions to which objects in a source bucket can be replicated.
func main() {
    // Set yourEndpoint to the endpoint of the region in which the source bucket is located
    . For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to
    https://oss-cn-hangzhou.aliyuncs.com. Specify the endpoint based on your business requireme
    nts.
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account t
    o access OSS because the account has permissions on all API operations. We recommend that y
    ou use a RAM user to call API operations or perform routine operations and maintenance. To
    create a RAM user, log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        HandleError(err)
    }
    // Specify the name of the source bucket.
    srcbucketName := "yourSourceBucket"
    result, err := client.GetBucketReplicationLocation(srcbucketName)
    if err != nil {
        HandleError(err)
    }
    fmt.Println(result)
}
```

Disable CRR for a bucket

The following code provides an example on how to disable CRR for a bucket:

```
package main
import (
    "fmt"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
    "os"
)
func HandleError(err error) {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
// Disable CRR for a bucket.
func main() {
    // Set yourEndpoint to the endpoint of the region in which the source bucket is located
    . For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to
    https://oss-cn-hangzhou.aliyuncs.com. Specify the endpoint based on your business requireme
    nts.
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account t
    o access OSS because the account has permissions on all API operations. We recommend that y
    ou use a RAM user to call API operations or perform routine operations and maintenance. To
    create a RAM user, log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        HandleError(err)
    }
    // Specify the name of the source bucket.
    srcbucketName := "yourSourceBucket"
    // Specify the CRR rule ID that is returned by the GetBucketReplication operation.
    err = client.DeleteBucketReplication(srcbucketName, "e047ce28-6806-4131-b1da-30142116***
    *")
    if err != nil {
        HandleError(err)
    }
}
```

References

- For more information about the API operation that you can call to enable CRR, see [PutBucketReplication](#).
- For more information about the API operation that you can call to query the CRR rules configured for a bucket, see [GetBucketReplication](#).
- For more information about the API operation that you can call to query the regions in which CRR can be performed, see [GetBucketReplicationLocation](#).
- For more information about the API operation that you can call to query the progress of a CRR task that is performed on a bucket, see [GetBucketReplicationProgress](#).
- For more information about the API operation that you can call to disable CRR for a bucket, see [DeleteBucketReplication](#).

9.5.19. CORS

Cross-origin resource sharing (CORS) is a standard cross-origin solution provided by HTML5 to allow web application servers to control cross-origin access and secure cross-origin data transmission.

Configure CORS rules

The following code provides an example on how to configure CORS rules for a specific bucket:

```
package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com. Specify the actual endpoint.
    // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to access OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Specify the name of the bucket. Example: examplebucket.
    bucketName := "examplebucket"
    rule1 := oss.CORSRule{
        AllowedOrigin: []string{"*"},
        AllowedMethod: []string{"PUT", "GET"},
        AllowedHeader: []string{},
        ExposeHeader:  []string{},
        MaxAgeSeconds: 200,
    }
    rule2 := oss.CORSRule{
        AllowedOrigin: []string{"http://example.com", "http://example.net"},
        AllowedMethod: []string{"POST"},
        AllowedHeader: []string{"Authorization"},
        ExposeHeader:  []string{"x-oss-test", "x-oss-test1"},
        MaxAgeSeconds: 100,
    }
    // Configure the CORS rules.
    err = client.SetBucketCORS(bucketName, []oss.CORSRule{rule1, rule2})
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
}
```

Query CORS rules

The following code provides an example on how to query CORS rules that are configured for a specific bucket:

```
package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com. Specify the actual endpoint.
    // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to access OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Specify the name of the bucket. Example: examplebucket.
    bucketName := "examplebucket"
    // Query the CORS rules.
    corsRes, err := client.GetBucketCORS(bucketName)
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    fmt.Println("Bucket CORS:", corsRes.CORSRules)
}
```

Delete CORS rules

The following code provides an example on how to delete the CORS rules that are configured for a specific bucket:

```
package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com. Specify the actual endpoint.
    // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to access OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Specify the name of the bucket. Example: examplebucket.
    bucketName := "examplebucket"
    // Delete the CORS rules.
    err = client.DeleteBucketCORS(bucketName)
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
}
```

References

- For more information about the complete sample code of CORS, visit [GitHub](#).
- For more information about the API operation that you can call to configure CORS rules, see [PutBucketCors](#).
- For more information about the API operation that you can call to query CORS rules, see [GetBucketCors](#).
- For more information about the API operation that you can call to delete CORS rules, see [DeleteBucketCors](#).

9.6. Upload objects

9.6.1. Overview

In OSS, an object is the basic unit for data operations. OSS Go SDK provides the following file upload methods:

- **Simple upload**: supports the upload of a file up to 5 GB in size.
- **Append upload**: supports the upload of a file up to 5 GB in size.
- **Resumable upload**: supports the upload of a file up to 48.8 TB in size. It applies to the upload of

large files. It supports concurrent upload and you can define the size of each part.

- **Multipart upload**: supports the upload of a file up to 48.8 TB in size. It applies to the upload of large files.

 **Note** For more information about the usage scenarios of each upload method, see “Upload files” in OSS Developer Guide.

During the file upload, you can set **Object Meta**, and view the upload progress in the **upload progress bar**.

9.6.2. Simple upload

To perform simple upload, you can call `PutObject` to upload a single object. Simple upload includes streaming upload and object upload. Streaming upload uses a file stream or network stream as the data source of an object uploaded to Object Storage Service (OSS). Object upload uses a local file as the data source of an object uploaded to OSS. This topic describes how to use streaming upload and object upload to upload an object.

Upload a local file

The following code provides an example on how to upload a local file to the `exampleobject.txt` object in the `exampledir` directory of the `examplebucket` bucket:

```
package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com. Specify the endpoint based on your business requirements.
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Specify the name of the bucket. Example: examplebucket.
    bucket, err := client.Bucket("examplebucket")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Specify the full path of the object. Example: exampledir/exampleobject.txt. Then, specify the full path of the local file. Example: D:\\localpath\\examplefile.txt.
    err = bucket.PutObjectFromFile("exampledir/exampleobject.txt", "D:\\localpath\\examplefile.txt")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
}
```

Upload a string

The following code provides an example on how to upload a string to the exampleobject.txt object in the exampledir directory of the examplebucket bucket:


```

package main
import (
    "fmt"
    "os"
    "strings"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region in which the bucket is located. For e
    // xample, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https:
    // //oss-cn-hangzhou.aliyuncs.com. Specify the endpoint based on your business requirements.
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account t
    // o access OSS because the account has permissions on all API operations. We recommend that y
    // ou use a RAM user to call API operations or perform routine operations and maintenance. To
    // create a RAM user, log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Specify the name of the bucket. Example: examplebucket.
    bucket, err := client.Bucket("examplebucket")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Set the storage class of the object to Infrequent Access (IA).
    storageType := oss.ObjectStorageClass(oss.StorageIA)
    // Set the access control list (ACL) of the object to private.
    objectAcl := oss.ObjectACL(oss.ACLPrivate)
    // Upload the "Hello OSS" string to the exampleobject.txt object in the exampleobject dire
    // ctory.
    err = bucket.PutObject("exampleobject.txt", strings.NewReader("Hello OSS"),
    storageType, objectAcl)
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
}

```

Upload a byte array

The following code provides an example on how to upload a byte array to the exampleobject.txt object in the exampleobject directory of the examplebucket bucket:

```
package main
import (
    "fmt"
    "os"
    "bytes"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region in which the bucket is located. For e
    xample, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https:
    //oss-cn-hangzhou.aliyuncs.com. Specify the endpoint based on your business requirements.
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account t
    o access OSS because the account has permissions on all API operations. We recommend that y
    ou use a RAM user to call API operations or perform routine operations and maintenance. To
    create a RAM user, log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Specify the name of the bucket. Example: examplebucket.
    bucket, err := client.Bucket("examplebucket")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Upload the byte array to the exampleobject.txt object in the exampledir directory.
    err = bucket.PutObject("exampledir/exampleobject.txt", bytes.NewReader([]byte("yourObjec
    ctValueByteArray")))
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
}
```

Upload a file stream

The following code provides an example on how to upload a file stream to the exampleobject.txt object in the exampledir directory of the examplebucket bucket:

```
package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com. Specify the endpoint based on your business requirements.
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Specify the name of the bucket. Example: examplebucket.
    bucket, err := client.Bucket("examplebucket")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Specify the full path of the local file. Example: D:\\localpath\\examplefile.txt.
    fd, err := os.Open("D:\\localpath\\examplefile.txt")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    defer fd.Close()
    // Upload the file stream to the exampleobject.txt object in the examplebucket directory.
    err = bucket.PutObject("examplebucket/exampleobject.txt", fd)
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
}
```

Upload a network stream

The following code provides an example on how to upload a network stream to the exampleobject.txt object in the examplebucket directory of the examplebucket bucket:

```
package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func HandleError(err error) {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region in which the bucket is located. For e
    xample, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https:
    //oss-cn-hangzhou.aliyuncs.com. Specify the endpoint based on your business requirements.
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account t
    o access OSS because the account has permissions on all API operations. We recommend that y
    ou use a RAM user to call API operations or perform routine operations and maintenance. To
    create a RAM user, log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Specify the name of the bucket. Example: examplebucket.
    bucket, err := client.Bucket("examplebucket")
    // Specify the full path of the object. Example: exampledir/exampleobject.txt.
    objectname := "exampledir/exampleobject.txt"
    // Specify the network stream that you want to upload.
    res, _ := http.Get("https://www.aliyun.com/")
    if err != nil {
        HandleError(err)
    }
    err = bucket.PutObject(objectname, io.Reader(res.Body))
    if err != nil {
        HandleError(err)
    }
}
```

References

- For more information about the complete sample code of simple upload, visit [GitHub](#).
- For more information about the API operation that you can call to perform simple upload, see [Put Object](#).

9.6.3. Append upload

You can call the AppendObject operation to append content to existing appendable objects.

Usage notes

- If the object to which you want to append content does not exist, an appendable object is created

when you call the AppendObject operation.

- When you call the AppendObject operation to append content to an object, you may encounter a few situations: If the object is an appendable object, and the specified position from which the append operation starts is equal to the current object length, the content is appended to the end of the object. If the specified position from which the append operation starts is not equal to the current object length, a `PositionNotEqualToLength` exception is thrown. If the object is not an appendable object, an `ObjectNotAppendable` exception is thrown.
- The CopyObject operation cannot be performed on appendable objects.

Sample code

The following code provides an example on how to perform append upload:

```
package main
import (
    "fmt"
    "os"
    "strings"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com. Specify the endpoint based on your business requirements.
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access Object Storage Service (OSS) because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Specify the name of the bucket. Example: examplebucket.
    bucket, err := client.Bucket("examplebucket")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Specify the full path of the object. The full path of the object cannot contain the bucket name. Example: exampledir/exampleobject.txt.
    objectName := "exampledir/exampleobject.txt"
    var nextPos int64 = 0
    // If the object is appended for the first time, the append position is 0. The position for the next append operation is included in the response. The position from which the next append operation starts is the current length of the object.
    // Specify the expiration time.
    expires := time.Date(2021, time.December, 10, 23, 0, 0, 0, time.UTC)
    option := []oss.Option{
        oss.Expires(expires),
        // Specify the caching behavior of the web page when the object is downloaded.
        //oss.CacheControl("no-cache"),
        // Specify the name of the object when the object is downloaded.
    }
```

```

// Specify the name of the object when the object is downloaded.
//oss.ContentDisposition("attachment;filename=FileName.txt"),
// Specify the encoding format for the content of the object.
//oss.ContentEncoding("gzip"),
// Specify the storage class of the object.
//oss.ObjectStorageClass(oss.StorageStandard),
// Specify the access control list (ACL) of the object.
//oss.ObjectACL(oss.ACLPrivate),
// Specify the server-side encryption method.
//oss.ServerSideEncryption("AES256"),
// You can add parameters whose names are prefixed with x-oss-meta- when you call t
he AppendObject operation to create an append object. These parameters cannot be included i
n the requests when you append content to an existing append object. Parameters whose names
are prefixed with x-oss-meta- are considered the metadata of the object.
//oss.Meta("x-oss-meta-author", "Alice"),
}
nextPos, err = bucket.AppendObject(objectName, strings.NewReader("YourObjectAppendValue
1"), nextPos, option...)
if err != nil {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
// If you have appended content to the object, you can obtain the position from which t
he current append operation starts from the X-Oss-Next-Append-Position field in the respons
e returned by the last operation or by using the bucket.GetObjectDetailedMeta method.
//props, err := bucket.GetObjectDetailedMeta("objectName")
//if err != nil {
//    fmt.Println("Error:", err)
//    os.Exit(-1)
//}
//nextPos, err = strconv.ParseInt(props.Get("X-Oss-Next-Append-Position"), 10, 64)
//if err != nil {
//    fmt.Println("Error:", err)
//    os.Exit(-1)
//}
// Perform the second append operation.
nextPos, err = bucket.AppendObject(objectName, strings.NewReader("YourObjectAppendValue
2"), nextPos)
if err != nil {
    fmt.Println("Error:", err, "aaa")
    os.Exit(-1)
}
// You can perform append operations on an object for multiple times.
}

```

References

- For more information about the complete sample code of append upload, visit [GitHub](#).
- For more information about the API operation that you can call to perform append upload, see [AppendObject](#).

9.6.4. Resumable upload

Before you upload an object to Object Storage Service (OSS) by using resumable upload, you can specify a directory for the checkpoint file that stores resumable upload records. If an object fails to upload due to a network exception or program error, the upload task is resumed from the position recorded in the checkpoint file to upload the remaining data.

Usage notes

- During resumable upload, the upload progress is recorded in a checkpoint file. If a part fails to be uploaded, the next upload starts from the position recorded in the checkpoint file. After the object is uploaded, the checkpoint file is deleted.
- The upload progress is recorded in the checkpoint file. Make sure that you have write permissions on the checkpoint file.
- The checkpoint file contains a checksum. This checksum cannot be edited. If the checkpoint file is damaged, you must upload all parts of the object again.
- If the local file is modified during the upload, you must upload all parts of the object again.

Implementation method

You can use the `Bucket.UploadFile` method to perform resumable upload. The following table describes the parameters that you can configure.

Parameter	Description
<code>objectKey</code>	The name of the object uploaded to Object Storage Service (OSS). This parameter plays the same role as <code>objectName</code> .
<code>filePath</code>	The path of the local file.
<code>partSize</code>	The size of each part. Valid values: 100 KB to 5 GB. Default value: 100 KB.
<code>options</code>	Valid values: • <code>Routines</code>: specifies the number of concurrent upload threads. The default value is 1, which indicates that only one thread is used to upload the parts. • <code>Checkpoint</code>: specifies whether resumable upload is enabled and the path of the checkpoint file. By default, resumable upload is disabled. For example, <code>oss.Checkpoint(true, "")</code> indicates that resumable upload is enabled and the checkpoint file is the <code>file.cp</code> file contained in the same directory as the source local file. <code>file</code> is the name of the local file. You can also use <code>oss.Checkpoint(true, "your-cp-file.cp")</code> to specify the checkpoint file.  Note For more information about other metadata, see Manage object metadata.

Sample code

The following code provides an example on how to perform resumable upload by using multipart upload:

```
package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Create an OSSClient instance.
    // Specify the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com. Specify your actual endpoint.
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Specify the bucket name. Example: examplebucket.
    bucket, err := client.Bucket("examplebucket")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Set the part size to 100 KB (100 × 1024), set the number of concurrent upload threads to 3, and enable resumable upload.
    // Set yourObjectName to the full path of the object. The full path cannot contain the bucket name. Example: exampledir/exampleobject.txt.
    // Set yourLocalFile to the full path of the local file. Example: D:\\localpath\\examplefile.txt. By default, if you do not specify the full path of the local file, the local file is uploaded from the path of the project to which the sample program belongs.
    err = bucket.UploadFile("exampledir/exampleobject.txt", "D:\\localpath\\examplefile.txt", 100*1024, oss.Routines(3), oss.Checkpoint(true, ""))
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
}
```

References

For the complete sample code that is used to perform resumable upload, visit [GitHub](#).

9.6.5. Multipart upload

By using the multipart upload feature provided by OSS, you can split a large object into multiple parts and upload them separately. After all parts are uploaded, call the `CompleteMultipartUpload` operation to combine these parts into a single object to implement resumable upload.

Multipart upload process

To implement multipart upload, perform the following operations:

1. Initiate a multipart upload task.

Call the `Bucket.InitiateMultipartUpload` method to obtain a unique upload ID in OSS.

2. Upload parts.

Call the `Bucket.UploadPart` method to upload the parts.

Note

- For parts that are uploaded by a multipart upload task with a specific upload ID, part numbers identify their relative positions in an object. If you upload a part and use its part number again to upload another part, the latter part overwrites the former part.
- OSS includes the MD5 hash of part data in the ETag header and returns the MD5 hash to the user.
- OSS calculates the MD5 hash of uploaded data and compares the MD5 hash with the MD5 hash calculated by the SDK. If the two hashes are different, the `InvalidDigest` error code is returned.

3. Complete the multipart upload task.

After all parts are uploaded, call the `Bucket.CompleteMultipartUpload` method to combine the parts into a complete object.

Complete sample code

The following code provides a complete example that describes the process of multipart upload:

```
package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com.
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
}
```

```

// Specify the name of the bucket.
bucketName := "examplebucket"
// Specify the full path of the object. The full path of the object cannot contain the
bucket name.
objectName := "exampleobject.txt"
// Specify the full path of the local file that you want to upload. By default, if you
do not specify the full path of the local file, the local file is uploaded from the path of
the project to which the sample program belongs.
localFilename := "D:\\localpath\\examplefile.txt"
bucket, err := client.Bucket(bucketName)
if err != nil {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
// Split the local file into three parts.
chunks, err := oss.SplitFileByPartNum(localFilename, 3)
fd, err := os.Open(localFilename)
defer fd.Close()
// Specify the expiration time.
expires := time.Date(2049, time.January, 10, 23, 0, 0, 0, time.UTC)
// The following code provides an example on how to specify the request headers when yo
u initiate a multipart upload task.
options := []oss.Option{
    oss.MetadataDirective(oss.MetaReplace),
    oss.Expires(expires),
    // Specify the caching behavior of the web page when the object is downloaded.
    // oss.CacheControl("no-cache"),
    // Specify the name of the object when the object is downloaded.
    // oss.ContentDisposition("attachment;filename=FileName.txt"),
    // Specify the encoding format for the content of the object.
    // oss.ContentEncoding("gzip"),
    // Specify the method that is used to encode the object name in the response. Only
URL encoding is supported.
    // oss.EncodingType("url"),
    // Specify the storage class of the object.
    // oss.ObjectStorageClass(oss.StorageStandard),
}
// Step 1: Initiate a multipart upload task and set the storage class to Standard.
imur, err := bucket.InitiateMultipartUpload(objectName, options...)
// Step 2: Upload parts.
var parts []oss.UploadPart
for _, chunk := range chunks {
    fd.Seek(chunk.Offset, os.SEEK_SET)
    // Call the UploadPart method to upload each part.
    part, err := bucket.UploadPart(imur, fd, chunk.Size, chunk.Number)
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    parts = append(parts, part)
}
// Set the access control list (ACL) of the object to public-read. By default, the obje
ct inherits the ACL of the bucket.
objectAcl := oss.ObjectACL(oss.ACLPublicRead)

```

```
// Step 3: Complete the multipart upload task and set the ACL of the object to public-read.
cmur, err := bucket.CompleteMultipartUpload(imur, parts, objectAcl)
if err != nil {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
fmt.Println("cmur:", cmur)
}
```

Cancel a multipart upload task

You can call the `bucket.AbortMultipartUpload` method to cancel a multipart upload task. If you cancel a multipart upload task, you cannot use the upload ID to manage parts. The uploaded parts are deleted.

The following code provides an example on how to cancel a multipart upload task:

```

package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region in which the bucket is located. For e
    xample, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https:
    //oss-cn-hangzhou.aliyuncs.com.
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account t
    o access OSS because the account has permissions on all API operations. We recommend that y
    ou use a RAM user to call API operations or perform routine operations and maintenance. To
    create a RAM user, log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Specify the bucket name.
    bucket, err := client.Bucket("examplebucket")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Initiate a multipart upload task.
    // Specify the full path of the object. The full path of the object cannot contain the
    bucket name.
    imur, err := bucket.InitiateMultipartUpload("exampleobject.txt")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Cancel the multipart upload task.
    err = bucket.AbortMultipartUpload(imur)
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
}

```

List the uploaded parts

The following code provides an example on how to list the parts that are uploaded in the specified multipart upload task:

```

package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {

```

```
func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region in which the bucket is located. For e
    // xample, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https:
    // //oss-cn-hangzhou.aliyuncs.com.
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account t
    // o access OSS because the account has permissions on all API operations. We recommend that y
    // ou use a RAM user to call API operations or perform routine operations and maintenance. To
    // create a RAM user, log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Specify the bucket name.
    bucketName := "examplebucket"
    // Specify the full path of the object. The full path of the object cannot contain the
    bucket name.
    objectName := "exampleobject.txt"
    // Specify the full path of the local file that you want to upload. By default, if you
    do not specify the local file, the local file is uploaded from the path of the project to w
    hich the sample program belongs.
    locaFilename := "D:\\localpath\\examplefile.txt"
    bucket, err := client.Bucket(bucketName)
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Split the local file into three parts.
    chunks, err := oss.SplitFileByPartNum(locaFilename, 3)
    fd, err := os.Open(locaFilename)
    defer fd.Close()
    // Initiate a multipart upload task.
    imur, err := bucket.InitiateMultipartUpload(objectName)
    uploadID := imur.UploadID
    fmt.Println("InitiateMultipartUpload UploadID: ", uploadID)
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Upload parts.
    var parts []oss.UploadPart
    for _, chunk := range chunks {
        fd.Seek(chunk.Offset, os.SEEK_SET)
        // Call the UploadPart method to upload each part.
        part, err := bucket.UploadPart(imur, fd, chunk.Size, chunk.Number)
        if err != nil {
            fmt.Println("Error:", err)
            os.Exit(-1)
        }
        fmt.Println("UploadPartNumber: ", part.PartNumber, ", ETag: ", part.ETag)
        parts = append(parts, part)
    }
    // List the uploaded parts based on InitiateMultipartUploadResult.
    lsRes, err := bucket.ListUnloadedParts(imur)
```

```

    }, err := bucket.ListUploadedParts(uploadID,
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Display the uploaded parts.
    fmt.Println("\nParts:", lsRes.UploadedParts)
    for _, upload := range lsRes.UploadedParts {
        fmt.Println("List PartNumber: ", upload.PartNumber, ", ETag: " ,upload.ETag, ", La
stModified: ", upload.LastModified)
    }
    // Specify objectName and uploadID to obtain InitiateMultipartUploadResult. List all up
loaded parts based on the result.
    var imur_with_uploadid oss.InitiateMultipartUploadResult
    imur_with_uploadid.Key = objectName
    imur_with_uploadid.UploadID = uploadID
    // List the uploaded parts.
    lsRes, err = bucket.ListUploadedParts(imur_with_uploadid)
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Display the uploaded parts.
    fmt.Println("\nListUploadedParts by UploadID: ", uploadID)
    for _, upload := range lsRes.UploadedParts {
        fmt.Println("List PartNumber: ", upload.PartNumber, ", ETag: " ,upload.ETag, ", La
stModified: ", upload.LastModified)
    }
    // Complete the multipart upload task.
    cmur, err := bucket.CompleteMultipartUpload(imur, parts)
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    fmt.Println("cmur:", cmur)
}

```

List multipart upload tasks

You can call the `Bucket.ListMultipartUploads` method to list all ongoing multipart upload tasks, which include tasks that have been initiated but are not completed and tasks that are canceled. The following table describes the parameters that you can configure to list these tasks.

Parameter	Description
Delimiter	The character that is used to group objects by name. Objects whose names contain the same string that ranges from the specified prefix to the delimiter that appears for the first time are grouped as an element.
MaxUploads	The maximum number of multipart upload tasks to return this time. Default value: 1000. Maximum value: 1000.

Parameter	Description
KeyMarker	Specifies that all multipart upload tasks with objects whose names are alphabetically after the value of the KeyMarker parameter are included in the list. You can use this parameter with the UploadIDMarker parameter to specify the start position to list the returned results.
Prefix	The prefix that the returned object names must contain. If you use a prefix for a query, the returned object name contains the prefix.
UploadIDMarker	The start position from which to list the returned results. This parameter is used together with the KeyMarker parameter. • If the KeyMarker parameter is not configured, this parameter is ignored. • If the KeyMarker parameter is configured, the response includes the following tasks: ◦ Multipart upload tasks in which the object names are alphabetically after the value of the KeyMarker parameter. ◦ Multipart upload tasks in which the objects names are the same as the value of the KeyMarker parameter but whose upload IDs are greater than the value of the UploadIDMarker parameter.

- List multipart upload tasks by using default parameter values

```

package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region in which the bucket is located. For
    // example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to ht
    // ps://oss-cn-hangzhou.aliyuncs.com.
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account
    // to access OSS because the account has permissions on all API operations. We recommend tha
    // t you use a RAM user to call API operations or perform routine operations and maintenance
    // . To create a RAM user, log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Specify the bucket name.
    bucketName := "examplebucket"
    bucket, err := client.Bucket(bucketName)
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // List all multipart upload tasks.
    keyMarker := ""
    uploadIdMarker := ""
    for {
        // By default, a maximum of 1,000 records can be returned at a time.
        lsRes, err := bucket.ListMultipartUploads(oss.KeyMarker(keyMarker), oss.UploadIDM
        arker(uploadIdMarker))
        if err != nil {
            fmt.Println("Error:", err)
            os.Exit(-1)
        }
        // Display the multipart upload tasks.
        for _, upload := range lsRes.Uploads {
            fmt.Println("Upload: ", upload.Key, ", UploadID: ",upload.UploadID)
        }
        if lsRes.IsTruncated {
            keyMarker = lsRes.NextKeyMarker
            uploadIdMarker = lsRes.NextUploadIDMarker
        } else {
            break
        }
    }
}

```

- Specify file as the prefix


```
lsRes, err := bucket.ListMultipartUploads(oss.Prefix("file"))
if err != nil {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
fmt.Println("Uploads:", lsRes.Uploads)
```

- Specify that a maximum of 100 multipart upload tasks are returned

```
lsRes, err := bucket.ListMultipartUploads(oss.MaxUploads(100))
if err != nil {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
fmt.Println("Uploads:", lsRes.Uploads)
```

- Specify file as the prefix and specify that a maximum of 100 multipart upload tasks are returned

```
lsRes, err := bucket.ListMultipartUploads(oss.Prefix("file"), oss.MaxUploads(100))
if err != nil {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
fmt.Println("Uploads:", lsRes.Uploads)
```

References

- For more information about the complete sample code of multipart upload, visit [GitHub](#).
- The following API operations are required to perform multipart upload:
 - The API operation that you can call to initiate a multipart upload task. For more information, see [InitiateMultipartUpload](#).
 - The API operation that you can call to upload data by part. For more information, see [UploadPart](#).
 - The API operation that you can call to complete a multipart upload task. For more information, see [CompleteMultipartUpload](#).
- For more information about the API operation that you can call to cancel a multipart upload task, see [AbortMultipartUpload](#).
- For more information about the API operation that you can call to list the uploaded parts, see [ListParts](#).
- For more information about the API operation that you can call to list all ongoing multipart upload tasks, see [ListMultipartUploads](#). Ongoing multipart upload tasks include tasks that have been initiated but are not completed and tasks that are canceled.

9.6.6. Progress bars

You can use progress bars to indicate the upload or download progress.

The following code is used as an example to describe how to view progress information with `Bucket.PutObjectFromFile`:

```

package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
// Define a progress bar listener.
type OssProgressListener struct {
}
// Define the progress event handler.
func (listener *OssProgressListener) ProgressChanged(event *oss.ProgressEvent) {
    switch event.EventType {
    case oss.TransferStartedEvent:
        fmt.Printf("Transfer Started, ConsumedBytes: %d, TotalBytes %d.\n",
            event.ConsumedBytes, event.TotalBytes)
    case oss.TransferDataEvent:
        fmt.Printf("\rTransfer Data, ConsumedBytes: %d, TotalBytes %d, %d%%.",
            event.ConsumedBytes, event.TotalBytes, event.ConsumedBytes*100/event.TotalBytes)
    )
    case oss.TransferCompletedEvent:
        fmt.Printf("\nTransfer Completed, ConsumedBytes: %d, TotalBytes %d.\n",
            event.ConsumedBytes, event.TotalBytes)
    case oss.TransferFailedEvent:
        fmt.Printf("\nTransfer Failed, ConsumedBytes: %d, TotalBytes %d.\n",
            event.ConsumedBytes, event.TotalBytes)
    default:
    }
}
func main() {
    // Create an OSSClient instance.
    client, err := oss.New("<yourEndpoint>", "<yourAccessKeyId>", "<yourAccessKeySecret>")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    bucketName := "<yourBucketName>"
    objectName := "<yourObjectName>"
    localFile := "<yourLocalFile>"
    // Obtain a bucket.
    bucket, err := client.Bucket(bucketName)
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Upload with a progress bar displayed.
    err = bucket.PutObjectFromFile(objectName, localFile, oss.Progress(&OssProgressListener
{}))
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
}

```

9.7. Download objects

9.7.1. Overview

For the complete code of object download, see [GitHub](#).

OSS Go SDK provides the following download methods:

- [Streaming download](#)
- [Download to your local file](#)
- [Resumable download](#)
- [Conditional download](#)
- [Download a compressed object](#)

You can view the download progress shown in the [download progress bar](#).

9.7.2. Streaming download

If you want to download a large object or it takes a long time to download an object at a time, you can use streaming download to download the object incrementally until the entire object is downloaded.

Download an object to a stream.

Run the following code to download a specified object to a stream:

```
package main
import (
    "fmt"
    "os"
    "io/ioutil"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Create an OSSClient instance.
    client, err := oss.New("<yourEndpoint>", "<yourAccessKeyId>", "<yourAccessKeySecret>")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Obtain the bucket.
    bucket, err := client.Bucket("<yourBucketName>")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Download an object to a stream.
    body, err := bucket.GetObject("<yourObjectName>")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // If you do not close the reader after the data is read, connection leaks may occur. C
    onsequently, no connection is available and the program cannot run properly.
    defer body.Close()
    data, err := ioutil.ReadAll(body)
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    fmt.Println("data:", string(data))
}
```

Download an object to the local cache

Run the following code to download a specified object to the local cache:

```
package main
import (
    "fmt"
    "os"
    "io"
    "bytes"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Create an OSSClient instance.
    client, err := oss.New("<yourEndpoint>", "<yourAccessKeyId>", "<yourAccessKeySecret>")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Obtain the bucket.
    bucket, err := client.Bucket("<yourBucketName>")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Download the object to the local cache.
    body, err := bucket.GetObject("<yourObjectName>")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    defer body.Close()
    buf := new(bytes.Buffer)
    io.Copy(buf, body)
    fmt.Println("buf:", buf)
}
```

Download an object to a local file stream

Run the following code to download a specified object to a local file stream:

```
package main
import (
    "fmt"
    "os"
    "io"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Create an OSSClient instance.
    client, err := oss.New("<yourEndpoint>", "<yourAccessKeyId>", "<yourAccessKeySecret>")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Obtain the bucket.
    bucket, err := client.Bucket("<yourBucketName>")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Download an object to a local file stream.
    body, err := bucket.GetObject("<yourObjectName>")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    defer body.Close()
    fd, err := os.OpenFile("LocalFile", os.O_WRONLY|os.O_CREATE, 0660)
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    defer fd.Close()
    io.Copy(fd, body)
}
```

9.7.3. Range download

If you need only part of the data in an object, you can use range download to download data within a specified range.

Specify a valid range

The following code provides an example on how to specify a valid range to download data:

```
package main
import (
    "fmt"
    "os"
    "io/ioutil"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Create an OSSClient instance.
    client, err := oss.New("<yourEndpoint>", "<yourAccessKeyId>", "<yourAccessKeySecret>")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Obtain the bucket.
    bucket, err := client.Bucket("<yourBucketName>")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Query the data that is within the range from byte 15 to byte 35, which includes a total of 21 bytes.
    // If the specified value range is invalid, for example, the specified range includes a negative number, or the specified value is larger than the object size, the entire object is downloaded.
    body, err := bucket.GetObject("<yourObjectName>", oss.Range(15, 35))
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // You must close the stream obtained by using ossClient.GetObject after the object is read. Otherwise, connection leaks may occur. Consequently, no connections are available and an exception occurs.
    defer body.Close()
    data, err := ioutil.ReadAll(body)
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    fmt.Println("data:", string(data))
}
```

An invalid range is specified

For an object whose size is 1,000 bytes, the valid range is from byte 0 to byte 999. If the specified range is not within the range from byte 0 to byte 999, the range does not take effect. Then, OSS returns HTTP status code 200 and the data of the entire object. The following list provides examples of invalid requests and the returned results:

- If you set Range: bytes to 500-2000, the value at the end of the range is invalid. Then, OSS returns HTTP status code 200 and the data of the entire object.
- If you set Range: bytes to 1000-2000, the value at the start of the range is invalid. Then, OSS returns HTTP status code 200 and the data of the entire object.

Compatible download

If you add `x-oss-range-behavior:standard` to the request header, the download behavior is modified when the specified range is not within the valid range. Example: Use range download to download an object whose size is 1,000 bytes.

- If you set `Range: bytes to 500-2000`, the value at the end of the range is invalid. Then, OSS returns HTTP status code 206 and the data that is within the range from byte 500 to byte 999.
- If you set `Range: bytes to 1000-2000`, the value at the start of the range is invalid. Then, OSS returns HTTP status code 416 and error code `InvalidRange`.

The following code provides an example on how to specify compatible behaviors to download data by range:

```
package main
import (
    "fmt"
    "io/ioutil"
    "os"
    "strings"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Create an OSSClient instance.
    client, err := oss.New("<yourEndpoint>", "<yourAccessKeyId>", "<yourAccessKeySecret>")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Obtain the bucket.
    bucket, err := client.Bucket("<yourBucketName>")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Upload data of 1,000 bytes.
    strContent := ""
    for i := 0; i < 100; i++ {
        strContent += "abcdefghij"
    }
    fmt.Printf("content len:%d\n", len(strContent))
    // Upload a string.
    err = bucket.PutObject("<yourObjectName>", strings.NewReader(strContent))
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // If you specify Range: bytes to 500-2000, the value at the end of the range is invalid. Then, OSS returns the HTTP status code 206 and the data that is within the range from byte 500 to byte 999.
    // If you specify Range: bytes to 1000-2000, the value at the start of the range is invalid. Then, OSS returns the HTTP status code 416 and the error code InvalidRange.
    body, err := bucket.GetObject("<yourObjectName>", oss.Range(500, 2000), oss.RangeBehavior("standard"))
    if err != nil {
```



```
    fmt.Println("Error:", err)
    os.Exit(-1)
}
// You must close the stream obtained by using ossClient.GetObject after the object is
read. Otherwise, connection leaks may occur. Consequently, no connections are available and
an exception occurs.
defer body.Close()
data, err := ioutil.ReadAll(body)
if err != nil {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
if len(data) != 500 {
    fmt.Println("read data error, len:%d", len(data))
    os.Exit(-1)
}
fmt.Println("data:", string(data))
}
```

9.7.4. Download objects

This topic describes how to download an object from a bucket to your local computer.

The following code provides an example on how to download `exampleobject.txt` from the `exampledir` directory in `examplebucket` to the `D:\localpath` path. After the object is downloaded, the local file is named `examplefile.txt`.

```

package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region where the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com.
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access Object Storage Service (OSS) because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Specify the name of the bucket. Example: examplebucket.
    bucket, err := client.Bucket("examplebucket")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Download the object to the specified local path. If the specified local file exists, the downloaded object replaces the local file. Otherwise, the local file is created.
    // If you do not specify the path of the local file, the downloaded object is saved to the path of the project to which the sample program belongs.
    // Specify the full path of the object. Example: exampledir/exampleobject.txt. Then, specify the full path of the local file. Example: D:\\localpath\\examplefile.txt. The full path of the object cannot contain bucket names.
    err = bucket.GetObjectToFile("exampledir/exampleobject.txt", "D:\\localpath\\examplefile.txt")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
}

```

9.7.5. Resumable download

You may fail to download a large object if the network is unstable or other exceptions occur. In some cases, you may still fail to download the object even after multiple attempts. To resolve this issue, Object Storage Service (OSS) SDK for Go provides the resumable download feature. In resumable download, the object that you want to download is split into multiple parts and each part is separately downloaded. After all parts are downloaded, these parts are combined into a complete object.

Usage notes

- During resumable download, the download progress is recorded in a checkpoint file. If the download

of a part fails, the next download starts from the position recorded in the checkpoint file. The checkpoint file is deleted after resumable download is completed.

- The SDK records the download progress in the checkpoint file. Make sure that you have write permissions on the checkpoint file.
- The checkpoint file contains a checksum. Do not modify the checksum. If the checkpoint file is damaged, you must download all parts again.
- If the ETag of the object is not consistent or parts are lost or modified during the download, the object is downloaded again.

Sample code

```

package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com. Specify your actual endpoint.
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Specify the name of the bucket. Example: examplebucket.
    bucket, err := client.Bucket("examplebucket")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Set yourObjectName to the full path of the object. The full path cannot contain the bucket name. Example: exampledir/exampleobject.txt.
    // Set yourLocalFile to the full path of the local file. Example: D:\\localpath\\examplefile.txt. If the path for the object is not specified, the downloaded object is saved to the path of the project to which the sample program belongs.
    // Set the size of each part to download to 100 KB (100 x 1024) and set the number of concurrent download parts to 3.
    // oss.Checkpoint(true, "") indicates that resumable download is enabled and the checkpoint file is the file.cp file contained in the same directory as the destination local file. file is a variable that indicates the local file name. You can also use oss.Checkpoint(true, "your-cp-file.cp") to specify the checkpoint file.
    err = bucket.DownloadFile("exampledir/exampleobject.txt", "D:\\localpath\\examplefile.txt", 100*1024, oss.Routines(3), oss.Checkpoint(true, ""))
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
}

```

References

- For more information about the complete sample code that is used to perform resumable download, visit [GitHub](#).
- For more information about the API operation that you can call to perform resumable download, see [GetObject](#).

9.7.6. Conditional download

You can configure one or more conditions for downloads. If the specified conditions are met, the object can be downloaded. If the specified conditions are not met, an error code is returned and the object cannot be downloaded.

You can configure the following conditions:

Parameter	Description	Configuration method
IfModifiedSince	If the specified time is earlier than the time the object is modified, the object can be downloaded. Otherwise, an error (304 Not modified) is returned.	oss.IfModifiedSince
IfUnmodifiedSince	If the specified time is later than or equal to the time the object is modified, the object can be downloaded. Otherwise, an error (412 Precondition failed) is returned.	oss.IfUnmodifiedSince
IfMatch	If the specified ETag matches that of an object, the object can be downloaded. Otherwise, an error (412 Precondition failed) is returned.	oss.IfMatch
IfNoneMatch	If the specified ETag does not match that of an object, the object can be downloaded. Otherwise, an error (304 Not modified) is returned.	oss.IfNoneMatch

Both If-Modified-Since and If-Unmodified-Since can exist at the same time as object download conditions. Both If-Match and If-None-Match can exist at the same time as object download conditions.

You can use `Bucket.GetObjectDetailedMeta` to Obtain the ETag.

```
package main
import (
    "fmt"
    "os"
    "time"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Create an OSSClient instance.
    client, err := oss.New("<yourEndpoint>", "<yourAccessKeyId>", "<yourAccessKeySecret>")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Obtain the bucket.
    bucket, err := client.Bucket("<yourBucketName>")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Configure time conditions.
    date := time.Date(2015, time.November, 10, 23, 0, 0, 0, time.UTC)
    // The condition is not met, and the object will not be downloaded.
    err = bucket.GetObjectToFile("<yourObjectName>", "LocalFile", oss.IfModifiedSince(date))
    if err == nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // The conditions are met, and the object will be downloaded.
    err = bucket.GetObjectToFile("<yourObjectName>", "LocalFile", oss.IfUnmodifiedSince(date))
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
}
```

9.7.7. Download a compressed object

This topic describes how to download a compressed object.

You can download a compressed object. Currently, GZIP is supported for object compression. `Bucket.GetObject` and `Bucket.GetObjectToFile` support compression in downloads.

```

package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Create an OSSClient instance.
    client, err := oss.New("<yourEndpoint>", "<yourAccessKeyId>", "<yourAccessKeySecret>")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Obtain the bucket.
    bucket, err := client.Bucket("<yourBucketName>")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Download an compressed object.
    err = bucket.GetObjectToFile("<yourObjectName>", "LocalFile.gzip", oss.AcceptEncoding("gzip"))
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
}

```

9.7.8. Use progress bars

OSS allows you to use progress bars to indicate the progress of an upload or download task. The `Bucket.GetObjectToFile` method is used in the example to describe how to display the progress bar of the task initiated to download an object.

The following code provides an example on how to display the progress bar of the task initiated to download the `exampleobject.txt` object from `examplebucket`:

```

package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
// Define the progress bar listener.
type OssProgressListener struct {
}
// Define the function used to process progress change events.
func (listener *OssProgressListener) ProgressChanged(event *oss.ProgressEvent) {
    switch event.EventType {
    case oss.TransferStartedEvent:
        fmt.Printf("Transfer Started, ConsumedBytes: %d, TotalBytes %d.\n",
            event.ConsumedBytes, event.TotalBytes)
    }
}

```

```

    case oss.TransferDataEvent:
        if event.TotalBytes != 0 {
            fmt.Printf("\rTransfer Data, ConsumedBytes: %d, TotalBytes %d, %d%%.",
                event.ConsumedBytes, event.TotalBytes, event.ConsumedBytes*100/event.TotalB
ytes)
        }
    case oss.TransferCompletedEvent:
        fmt.Printf("\nTransfer Completed, ConsumedBytes: %d, TotalBytes %d.\n",
            event.ConsumedBytes, event.TotalBytes)
    case oss.TransferFailedEvent:
        fmt.Printf("\nTransfer Failed, ConsumedBytes: %d, TotalBytes %d.\n",
            event.ConsumedBytes, event.TotalBytes)
    default:
    }
}

func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region where the bucket is located. For exam
ple, if your bucket is located in the China (Hangzhou) region, set yourEndpoint to https://
oss-cn-hangzhou.aliyuncs.com.
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account t
o log on to OSS because the account has permissions on all API operations. We recommend tha
t you use your RAM user's credentials to call API operations or perform routine operations
and maintenance. To create a RAM user, log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Specify the bucket name.
    bucketName := "examplebucket"
    // Specify the full path of the object. The full path of the object cannot contain buck
et names.
    objectName := "exampleobject.txt"
    // Specify the full path of the local file. If the specified local file exists, the dow
nloaded object replaces the local file. Otherwise, the local file is created.
    // If the path of the local file is not specified, the downloaded object is saved to th
e path of the project to which the sample program belongs.
    localFile := "D:\\localpath\\examplefile.txt"
    // Obtain the bucket.
    bucket, err := client.Bucket(bucketName)
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Display the download progress while you download the object.
    err = bucket.GetObjectToFile(objectName, localFile, oss.Progress(&OssProgressListener{}
))
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    fmt.Println("Transfer Completed.")
}

```


9.8. Manage objects

9.8.1. Overview

You can manage objects stored in buckets with APIs, including the following operations:

- [Determine whether an object exists](#)
- [Manage ACL for an object](#)
- [Manage Object Meta](#)
- [List objects](#)
- [Delete objects](#)
- [Copy objects](#)
- [Restore an archive object](#)
- [Manage a symbolic link](#)

9.8.2. Determine whether a specified object exists

This topic describes how to determine whether a specified object exists.

Run the following code to determine whether a specified object exists:

```
package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Create an OSSClient instance.
    client, err := oss.New("<yourEndpoint>", "<yourAccessKeyId>", "<yourAccessKeySecret>")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Obtain the bucket.
    bucket, err := client.Bucket("<yourBucketName>")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Determine whether the specified object exists.
    isExist, err := bucket.IsObjectExist("<yourObjectName>")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    fmt.Println("Exist:", isExist)
}
```

9.8.3. Manage the ACL of an object

This topic describes how to manage the access control list (ACL) of an object.

For the complete sample code that is used to manage the ACL of an object, visit [GitHub](#).

The following table describes the ACLs that you can configure for an object.

Permission	Description	Value
Inherited from the bucket	The ACL of the object is the same as the ACL of the bucket in which the object is stored.	oss.ACLDefault
Private	Only the object owner and authorized users are granted the read and write permissions on the object.	oss.ACLPrivate

Permission	Description	Value
Public read	Only the object owner and authorized users are granted the read and write permissions on the object. Other users are granted only the read permissions on the object. Exercise caution when you set the ACL of the object to this value.	oss.ACLPublicRead
Public read/write	All users are granted the read and write permissions on the object. Exercise caution when you set the object ACL to this value.	oss.PublicReadWrite

The ACL of the object takes precedence over the ACL of the bucket. For example, if the ACL of a bucket is private and the ACL of an object that is stored in the bucket is public, all users are granted the read and write permissions on the object. If the ACL of an object is not configured, the ACL of the object is the same as the ACL of the bucket in which the object is stored.

You can use the following complete code to configure and query the ACL of an object:

```
package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Create an OSSClient instance.
    client, err := oss.New("<yourEndpoint>", "<yourAccessKeyId>", "<yourAccessKeySecret>")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Query the bucket.
    bucket, err := client.Bucket("<yourBucketName>")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Configure the ACL of the object.
    err = bucket.SetObjectACL("<yourObjectName>", oss.ACLPublicReadWrite)
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Query the ACL of the object.
    aclRes, err := bucket.GetObjectACL("<yourObjectName>")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    fmt.Println("Object ACL:", aclRes.ACL)
}
```

9.8.4. Manage object metadata

Object metadata includes HTTP headers and user metadata. This topic describes how to configure, modify, and query object metadata.

Configure object metadata

The following code provides an example on how to configure object metadata in simple upload:

```
package main
import (
    "fmt"
    "os"
    "time"
    "strings"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
```

```

// Create an OSSClient instance.
// Specify the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com. Specify your actual endpoint.
// Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access Object Storage Service (OSS) because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
if err != nil {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
// Specify the bucket name. Example: examplebucket.
bucket, err := client.Bucket("examplebucket")
if err != nil {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
// Configure object metadata: Set the expiration time to 23:00:00 January 10, 2049 in GMT, access control list (ACL) to public read, and MyProp to MyPropVal as user metadata.
expires := time.Date(2049, time.January, 10, 23, 0, 0, 0, time.UTC)
options := []oss.Option{
    oss.Expires(expires),
    oss.ObjectACL(oss.ACLPublicRead),
    oss.Meta("MyProp", "MyPropVal"),
}
// Use a data stream to upload the object.
// Specify the full path of the object. The full path cannot contain the bucket name. Example: exampledir/exampleobject.txt.
err = bucket.PutObject("exampledir/exampleobject.txt", strings.NewReader("MyObjectValue"), options...)
if err != nil {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
// Query object metadata.
props, err := bucket.GetObjectDetailedMeta("exampledir/exampleobject.txt")
if err != nil {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
fmt.Println("Object Meta:", props)
}

```

Modify object metadata

The following code provides an example on how to modify one or multiple object metadata parameters at a time:

```

package main
import (
    "fmt"

```

```

    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Create an OSSClient instance.
    // Specify the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com. Specify your actual endpoint.
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Specify the bucket name. Example: examplebucket.
    bucketName := "examplebucket"
    // Specify the full path of the object.
    objectName := "exampledir/exampleobject.txt"
    // Query the bucket.
    bucket, err := client.Bucket(bucketName)
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Modify one object metadata parameter at a time.
    err = bucket.SetObjectMeta(objectName, oss.Meta("MyMeta", "MyMetaValue1"))
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Modify multiple object metadata parameters at a time.
    options := []oss.Option{
        oss.Meta("MyMeta", "MyMetaValue2"),
        oss.Meta("MyObjectLocation", "HangZhou"),
    }
    err = bucket.SetObjectMeta(objectName, options...)
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Query object metadata.
    props, err := bucket.GetObjectDetailedMeta(objectName)
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    fmt.Println("Object Meta:", props)
}

```

Query object metadata

The following code provides an example on how to query object metadata:

```
package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Create an OSSClient instance.
    // Specify the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com. Specify your actual endpoint.
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Specify the bucket name. Example: examplebucket.
    bucketName := "examplebucket"
    // Specify the full path of the object.
    objectName := "exampledir/exampleobject.txt"
    // Query the bucket.
    bucket, err := client.Bucket(bucketName)
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Query object metadata.
    props, err := bucket.GetObjectDetailedMeta(objectName)
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    fmt.Println("Object Meta:", props)
}
```

References

- For the complete sample code that is used to manage object metadata, visit [GitHub](#).
- For more information about the API operation that you can call to configure object metadata in simple upload, see [Put Object](#).
- For more information about the API operation that you can call to modify object metadata in a copy task, see [CopyObject](#).
- For more information about the API operation that you can call to query object metadata, see [Get Object Meta](#).

9.8.5. Convert the storage class of an object

OSS provides the following storage classes to cover various data storage scenarios from hot data to cold data: Standard, Infrequent Access (IA), Archive and Cold Archive. This topic describes how to convert the storage class of an object.

For more information about these storage classes, see [Overview](#) and [Convert storage classes](#) in the OSS Developer Guide.

The following code provides an example of how to convert the storage class of an object.

- The following code provides an example of how to convert the storage class of an object from Standard or IA to Archive:

```
package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Create an OSSClient instance.
    client, err := oss.New("<yourEndpoint>", "<yourAccessKeyId>", "<yourAccessKeySecret>")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    bucketName := "<yourBucketName>"
    objectName := "<yourObjectName>"
    // Obtain information about the bucket based on the bucket name.
    bucket, err := client.Bucket(bucketName)
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Modify the storage class of the object. Set the storage class to Archive.
    _, err = bucket.CopyObject(objectName, objectName, oss.ObjectStorageClass(oss.Storage
Archive))
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
}
```

- The following code provides an example of how to convert the storage class from Archive to Standard:

```
package main
import (
    "fmt"
    "os"
    "time"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
}
```



```
)  
func main() {  
    // Create an OSSClient instance.  
    client, err := oss.New("<yourEndpoint>", "<yourAccessKeyId>", "<yourAccessKeySecret>"  
)  
    if err != nil {  
        fmt.Println("Error:", err)  
        os.Exit(-1)  
    }  
    bucketName := "<yourBucketName>"  
    objectName := "<yourObjectName>"  
    // Obtain information about the bucket based on the bucket name.  
    bucket, err := client.Bucket(bucketName)  
    if err != nil {  
        fmt.Println("Error:", err)  
        os.Exit(-1)  
    }  
    // Check whether the storage class of the source object is Archive. If the storage class of the source object is Archive, you must restore the object before you can modify the storage class.  
    meta, err := bucket.GetObjectDetailedMeta(objectName)  
    if err != nil {  
        fmt.Println("Error:", err)  
        os.Exit(-1)  
    }  
    fmt.Println("X-Oss-Storage-Class : ", meta.Get(oss.HTTPHeaderOssStorageClass))  
    if meta.Get(oss.HTTPHeaderOssStorageClass) == string(oss.StorageArchive) {  
        // Restore the object.  
        err = bucket.RestoreObject(objectName)  
        if err != nil {  
            fmt.Println("Error:", err)  
            os.Exit(-1)  
        }  
        // Wait for about one minute until the object is restored.  
        meta, err = bucket.GetObjectDetailedMeta(objectName)  
        for meta.Get("X-Oss-Restore") == "ongoing-request=true" {  
            fmt.Println("X-Oss-Restore:" + meta.Get("X-Oss-Restore"))  
            time.Sleep(1 * time.Second)  
            meta, err = bucket.GetObjectDetailedMeta(objectName)  
        }  
    }  
    //Modify the storage class of the restored object. Set the storage class to Standard.  
    _, err = bucket.CopyObject(objectName, objectName, oss.ObjectStorageClass(oss.StorageStandard))  
    if err != nil {  
        fmt.Println("Error:", err)  
        os.Exit(-1)  
    }  
}
```

For more information about the storage classes and storage fees, see the [Billing overview](#) section in [Billing items](#).

9.8.6. List objects

This topic describes how to list all objects, objects whose names contain the specified prefix, and objects and subdirectories in the specified directory in a bucket.

Background information

You can call the `GetBucket (ListObjects)` or `GetBucketV2 (ListObjectsV2)` operation to list up to 1,000 objects in a bucket at a time. You can configure the parameters to use different methods to list objects. For example, you can configure the parameters to list objects from the specified start position, list objects and subdirectories in the specified directory, and list objects by page. The `GetBucket (ListObjects)` and `GetBucketV2 (ListObjectsV2)` operations are different in the following aspects:

- By default, if you call the `GetBucket (ListObjects)` operation to list objects, the owner information is included in the response.
- If you call the `GetBucketV2 (ListObjectsV2)` operation to list objects, you can configure the `fetchOwner` parameter to specify whether to include the owner information in the response.

 **Note** To list objects in a versioned bucket, we recommend that you call the `GetBucketV2 (ListObjectsV2)` operation.

The following tables describe the parameters that you can configure when you call the `GetBucket (ListObjects)` or `GetBucketV2 (ListObjectsV2)` operation to list objects.

- Call the `GetBucket (ListObjects)` operation to list objects

The following table describes the parameters that you can configure when you call the `GetBucket (ListObjects)` operation to list objects.

Parameter	Description
delimiter	The character that is used to group objects by name. Objects whose names contain the same string that ranges from the specified prefix to the delimiter that appears for the first time are grouped as a <code>commonPrefixes</code> element.
prefix	The prefix that the names of returned objects must contain.
maxKeys	The maximum number of objects that can be listed at a time. The default value is 100, and the maximum value is 1000.
marker	The position from which the list operation starts.

For more information, see [GetBucket \(ListObjects\)](#).

- Call the `GetBucketV2 (ListObjectsV2)` operation to list objects

The following table describes the parameters that you can configure when you call the `GetBucketV2 (ListObjectsV2)` operation to list objects.

Parameter	Description
prefix	The prefix that the names of returned objects must contain.

Parameter	Description
delimiter	The character that is used to group objects by name. Objects whose names contain the same string that ranges from the specified prefix to the delimiter that appears for the first time are grouped as a <code>commonPrefixes</code> element.
startAfter	The position from which the list operation starts.
fetchOwner	Specifies whether to include the owner information in the response. ◦ <code>true</code>: The response includes the owner information. ◦ <code>false</code>: The response does not include the owner information.

For more information, see [GetBucketV2 \(ListObjectsV2\)](#).

Simple list

- Call the `GetBucket (ListObjects)` operation to list objects

The following code provides an example on how to call the `GetBucket (ListObjects)` operation to list 100 objects in the specified bucket:

```

package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func HandleError(err error) {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region in which the bucket is located. For
    // example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com. Specify the endpoint based on your business requirements.
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account
    // to access Object Storage Service (OSS) because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        HandleError(err)
    }
    // Specify the bucket name.
    bucketName := "yourBucketName"
    bucket, err := client.Bucket(bucketName)
    if err != nil {
        HandleError(err)
    }
    // List objects.
    marker := ""
    for {
        lsRes, err := bucket.ListObjects(oss.Marker(marker))
        if err != nil {
            HandleError(err)
        }
        // Display the listed objects. By default, a maximum of 100 objects are returned at a time.
        for _, object := range lsRes.Objects {
            fmt.Println("Bucket: ", object.Key)
        }
        if lsRes.IsTruncated {
            marker = lsRes.NextMarker
        } else {
            break
        }
    }
}

```

- Call the GetBucketV2 (ListObjectsV2) operation to list objects

The following code provides an example on how to call the GetBucketV2 (ListObjectsV2) operation to list 100 objects in the specified bucket:

```
package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func HandleError(err error) {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region in which the bucket is located. For
    // example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to ht
    // ps://oss-cn-hangzhou.aliyuncs.com. Specify the endpoint based on your business requiremen
    // ts.
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account
    // to access OSS because the account has permissions on all API operations. We recommend tha
    // t you use a RAM user to call API operations or perform routine operations and maintenance
    // . To create a RAM user, log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        HandleError(err)
    }
    // Specify the bucket name.
    bucketName := "yourBucketName"
    bucket, err := client.Bucket(bucketName)
    if err != nil {
        HandleError(err)
    }
    continueToken := ""
    for {
        lsRes, err := bucket.ListObjectsV2(oss.ContinuationToken(continueToken))
        if err != nil {
            HandleError(err)
        }
        // Display the listed objects. By default, a maximum of 100 objects are returned
        // at a time.
        for _, object := range lsRes.Objects {
            fmt.Println(object.Key, object.Type, object.Size, object.ETag, object.LastM
            odified, object.StorageClass)
        }
        if lsRes.IsTruncated {
            continueToken = lsRes.NextContinuationToken
        } else {
            break
        }
    }
}
```

List the specified number of objects

- Call the GetBucket (ListObjects) operation to list objects

The following code provides an example on how to call the GetBucket (ListObjects) operation to list the specified number of objects:

```
package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region in which the bucket is located. For
    // example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to ht
    // ps://oss-cn-hangzhou.aliyuncs.com. Specify the endpoint based on your business requiremen
    // ts.
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account
    // to access OSS because the account has permissions on all API operations. We recommend tha
    // t you use a RAM user to call API operations or perform routine operations and maintenance
    // . To create a RAM user, log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Specify the bucket name.
    bucketName := "yourBucketName"
    bucket, err := client.Bucket(bucketName)
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Set the maximum number of objects that can be listed at a time and list the obje
    // cts.
    lsRes, err := bucket.ListObjects(oss.MaxKeys(200))
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Display the listed objects. By default, a maximum of 100 objects are returned at
    // a time.
    fmt.Println("Objects:", lsRes.Objects)
    for _, object := range lsRes.Objects {
        fmt.Println("Object:", object.Key)
    }
}
```

- Call the GetBucketV2 (ListObjectsV2) operation to list objects

The following code provides an example on how to call the GetBucketV2 (ListObjectsV2) operation to list the specified number of objects:

```

package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func HandleError(err error) {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com. Specify the endpoint based on your business requirements.
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        HandleError(err)
    }
    // Specify the bucket name.
    bucketName := "yourBucketName"
    bucket, err := client.Bucket(bucketName)
    if err != nil {
        HandleError(err)
    }
    // Configure the MaxKeys parameter to specify the maximum number of objects that can be listed at a time and list the objects.
    lsRes, err := bucket.ListObjectsV2(oss.MaxKeys(200))
    if err != nil {
        HandleError(err)
    }
    // Display the listed objects. By default, a maximum of 100 objects are returned at a time.
    for _, object := range lsRes.Objects {
        fmt.Println(object.Key, object.Type, object.Size, object.ETag, object.LastModified, object.StorageClass, object.Owner.ID, object.Owner.DisplayName)
    }
}

```

List objects whose names contain the specified prefix

- Call the GetBucket (ListObjects) operation to list objects

The following code provides an example on how to call the GetBucket (ListObjects) operation to list objects whose names contain the specified prefix:

```

package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region in which the bucket is located. For
    // example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to ht
    // ps://oss-cn-hangzhou.aliyuncs.com. Specify the endpoint based on your business requiremen
    // ts.
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account
    // to access OSS because the account has permissions on all API operations. We recommend tha
    // t you use a RAM user to call API operations or perform routine operations and maintenance
    // . To create a RAM user, log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Specify the bucket name.
    bucketName := "yourBucketName"
    bucket, err := client.Bucket(bucketName)
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // List objects whose names contain the specified prefix. By default, a maximum of 10
    // 0 objects are listed.
    lsRes, err := bucket.ListObjects(oss.Prefix("my-object-"))
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Display the listed objects. By default, a maximum of 100 objects are returned at a
    // time.
    fmt.Println("Objects:", lsRes.Objects)
    for _, object := range lsRes.Objects {
        fmt.Println("Object:", object.Key)
    }
}

```

- Call the GetBucketV2 (ListObjectsV2) operation to list objects

The following code provides an example on how to call the GetBucketV2 (ListObjectsV2) operation to list objects whose names contain the specified prefix:


```

package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func HandleError(err error) {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region in which the bucket is located. For
    example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to ht
    ps://oss-cn-hangzhou.aliyuncs.com. Specify the endpoint based on your business requiremen
    ts.
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account
    to access OSS because the account has permissions on all API operations. We recommend tha
    t you use a RAM user to call API operations or perform routine operations and maintenance
    . To create a RAM user, log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        HandleError(err)
    }
    // Specify the bucket name.
    bucketName := "yourBucketName"
    bucket, err := client.Bucket(bucketName)
    if err != nil {
        HandleError(err)
    }
    // Set the Prefix parameter to my-object-.
    lsRes, err := bucket.ListObjectsV2(oss.Prefix("my-object-"))
    if err != nil {
        HandleError(err)
    }
    // Display the listed objects. By default, a maximum of 100 objects are returned at a
    time.
    for _, object := range lsRes.Objects {
        fmt.Println(object.Key, object.Type, object.Size, object.ETag, object.LastModifie
        d, object.StorageClass, object.Owner.ID, object.Owner.DisplayName)
    }
}

```

List all objects from the specified start position

- Call the GetBucket (ListObjects) operation to list objects

You can configure the Marker parameter to specify the start position from which the list operation starts. All objects whose names are alphabetically after the value of Marker are returned.

```

package main
import (
    "fmt"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
    "os"
)
func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region in which the bucket is located. For
    // example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to ht
    // ps://oss-cn-hangzhou.aliyuncs.com. Specify the endpoint based on your business requiremen
    // ts.
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account
    // to access OSS because the account has permissions on all API operations. We recommend tha
    // t you use a RAM user to call API operations or perform routine operations and maintenance
    // . To create a RAM user, log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Specify the bucket name.
    bucketName := "yourBucketName"
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    marker := oss.Marker("")
    for {
        lsRes, err := bucket.ListObjects(marker)
        if err != nil {
            fmt.Println("Error:", err)
            os.Exit(-1)
        }
        // Display the listed objects. By default, a maximum of 100 objects are returne
        // d at a time.
        fmt.Println("Objects:", lsRes.Objects)
        for _, object := range lsRes.Objects {
            fmt.Println("Object:", object.Key)
        }
        if lsRes.IsTruncated {
            marker = oss.Marker(lsRes.NextMarker)
        } else {
            break
        }
    }
}

```

- Call the GetBucketV2 (ListObjectsV2) operation to list objects

You can configure the StartAfter parameter to specify the start position from which the list operation starts. All objects whose names are alphabetically after the value of StartAfter are returned.

```
package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func HandleError(err error) {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region in which the bucket is located. For
    // example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to ht
    // ps://oss-cn-hangzhou.aliyuncs.com. Specify the endpoint based on your business requiremen
    // ts.
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account
    // to access OSS because the account has permissions on all API operations. We recommend tha
    // t you use a RAM user to call API operations or perform routine operations and maintenance
    // . To create a RAM user, log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        HandleError(err)
    }
    // Specify the bucket name.
    bucketName := "yourBucketName"
    bucket, err := client.Bucket(bucketName)
    if err != nil {
        HandleError(err)
    }
    // Configure the StartAfter parameter to list objects whose names are alphabetically
    // after the value of StartAfter. By default, a maximum of 100 objects are listed.
    lsRes, err := bucket.ListObjectsV2(oss.StartAfter("my-object-"))
    if err != nil {
        HandleError(err)
    }
    // Display the listed objects. By default, a maximum of 100 objects are returned at a
    // time.
    for _, object := range lsRes.Objects {
        fmt.Println(object.Key, object.Type, object.Size, object.ETag, object.LastModifie
        d, object.StorageClass, object.Owner.ID, object.Owner.DisplayName)
    }
}
```

List all objects by page

- Call the GetBucket (ListObjects) operation to list objects

The following code provides an example on how to call the GetBucket (ListObjects) operation to list all objects in the specified bucket by page: You can configure the MaxKeys parameter to specify the maximum number of objects that can be listed on each page.

```

package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region in which the bucket is located. For
    // example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to ht
    // ps://oss-cn-hangzhou.aliyuncs.com. Specify the endpoint based on your business requiremen
    // ts.
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account
    // to access OSS because the account has permissions on all API operations. We recommend tha
    // t you use a RAM user to call API operations or perform routine operations and maintenance
    // . To create a RAM user, log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Specify the bucket name.
    bucketName := "yourBucketName"
    bucket, err := client.Bucket(bucketName)
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // List all objects by page, with up to 100 objects on each page.
    marker := oss.Marker("")
    for {
        lsRes, err := bucket.ListObjects(oss.MaxKeys(100), marker)
        if err != nil {
            fmt.Println("Error:", err)
            os.Exit(-1)
        }
        // Display the listed objects. By default, a maximum of 100 objects are returned
        // at a time.
        fmt.Println("Objects:", lsRes.Objects)
        if lsRes.IsTruncated {
            marker = oss.Marker(lsRes.NextMarker)
        } else {
            break
        }
    }
}

```

- Call the GetBucketV2 (ListObjectsV2) operation to list objects

The following code provides an example on how to call the GetBucketV2 (ListObjectsV2) operation to list all objects in the specified bucket by page: You can configure the MaxKeys parameter to specify the maximum number of objects that can be listed on each page.

```

package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func HandleError(err error) {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region in which the bucket is located. For
    // example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to ht
    // ps://oss-cn-hangzhou.aliyuncs.com. Specify the endpoint based on your business requiremen
    // ts.
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account
    // to access OSS because the account has permissions on all API operations. We recommend tha
    // t you use a RAM user to call API operations or perform routine operations and maintenance
    // . To create a RAM user, log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        HandleError(err)
    }
    // Specify the bucket name.
    bucketName := "yourBucketName"
    bucket, err := client.Bucket(bucketName)
    if err != nil {
        HandleError(err)
    }
    // List all objects by page, with up to 100 objects on each page.
    continueToken := ""
    for {
        lsRes, err := bucket.ListObjectsV2(oss.MaxKeys(100), oss.ContinuationToken(contin
        ueToken))
        if err != nil {
            HandleError(err)
        }
        // Display the listed objects. By default, a maximum of 100 objects are returned
        // at a time.
        for _, object := range lsRes.Objects {
            fmt.Println(object.Key, object.Type, object.Size, object.ETag, object.LastMod
            ified, object.StorageClass)
        }
        if lsRes.IsTruncated {
            continueToken = lsRes.NextContinuationToken
        } else {
            break
        }
    }
}

```

List objects whose names contain the specified prefix by page

- Call the GetBucket (ListObjects) operation to list objects

The following code provides an example on how to call the GetBucket (ListObjects) operation to list objects whose names contain the specified prefix by page. You can configure the MaxKeys parameter to specify the maximum number of objects that can be listed on each page.

```

package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region in which the bucket is located. For
    // example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com. Specify the endpoint based on your business requirements.
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account
    // to access OSS because the account has permissions on all API operations. We recommend that
    // you use a RAM user to call API operations or perform routine operations and maintenance
    // . To create a RAM user, log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Specify the bucket name.
    bucketName := "yourBucketName"
    bucket, err := client.Bucket(bucketName)
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // List the objects whose names contain the specified prefix by page, with up to 80 objects
    // on each page.
    prefix := oss.Prefix("my-object-")
    marker := oss.Marker("")
    for {
        lsRes, err := bucket.ListObjects(oss.MaxKeys(80), marker, prefix)
        if err != nil {
            fmt.Println("Error:", err)
            os.Exit(-1)
        }
        // Display the listed objects. By default, a maximum of 100 objects are returned
        // at a time.
        fmt.Println("Objects:", lsRes.Objects)
        if lsRes.IsTruncated {
            prefix = oss.Prefix(lsRes.Prefix)
            marker = oss.Marker(lsRes.NextMarker)
        } else {
            break
        }
    }
}

```

- Call the GetBucketV2 (ListObjectsV2) operation to list objects

The following code provides an example on how to call the GetBucketV2 (ListObjectsV2) operation to list objects whose names contain the specified prefix by page. You can configure the MaxKeys parameter to specify the maximum number of objects that can be listed on each page.

```
package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func HandleError(err error) {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region in which the bucket is located. For
    // example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to ht
    // ps://oss-cn-hangzhou.aliyuncs.com. Specify the endpoint based on your business requiremen
    // ts.
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account
    // to access OSS because the account has permissions on all API operations. We recommend tha
    // t you use a RAM user to call API operations or perform routine operations and maintenance
    // . To create a RAM user, log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        HandleError(err)
    }
    // Specify the bucket name.
    bucketName := "yourBucketName"
    bucket, err := client.Bucket(bucketName)
    if err != nil {
        HandleError(err)
    }
    // List the objects whose names contain the specified prefix by page, with up to 80 o
    // bjects on each page.
    prefix := oss.Prefix("my-object-")
    continueToken := ""
    for {
        lsRes, err := bucket.ListObjectsV2(prefix, oss.MaxKeys(80), oss.ContinuationToken
        (continueToken))
        if err != nil {
            HandleError(err)
        }
        // Display the listed objects. By default, a maximum of 100 objects are returned
        // at a time.
        for _, object := range lsRes.Objects {
            fmt.Println(object.Key, object.Type, object.Size, object.ETag, object.LastMod
            ified, object.StorageClass)
        }
        if lsRes.IsTruncated {
            continueToken = lsRes.NextContinuationToken
            prefix = oss.Prefix(lsRes.Prefix)
        } else {

```



```
        break
    }
}
```

List the information about all objects in the specified directory

- Call the GetBucket (ListObjects) operation to list the information about all objects in the specified directory

The following code provides an example on how to call the GetBucket (ListObjects) operation to list the information about all objects in the specified directory, including the sizes, last modified time, and names of the objects:

```

package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region in which the bucket is located. For
    // example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to ht
    // ps://oss-cn-hangzhou.aliyuncs.com. Specify the endpoint based on your business requiremen
    // ts.
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account
    // to access OSS because the account has permissions on all API operations. We recommend tha
    // t you use a RAM user to call API operations or perform routine operations and maintenance
    // . To create a RAM user, log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Specify the bucket name.
    bucketName := "yourBucketName"
    bucket, err := client.Bucket(bucketName)
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Traverse the objects.
    marker := oss.Marker("")
    prefix := oss.Prefix("test")
    for {
        lor, err := bucket.ListObjects(marker, prefix)
        if err != nil {
            fmt.Println("Error:", err)
            os.Exit(-1)
        }
        for _, object := range lor.Objects {
            fmt.Println(object.Key, object.Type, object.Size, object.ETag, object.LastModi
            fied, object.StorageClass)
        }
        if lor.IsTruncated {
            prefix = oss.Prefix(lor.Prefix)
            marker = oss.Marker(lor.NextMarker)
        } else {
            break
        }
    }
}

```

- Call the GetBucketV2 (ListObjectsV2) operation to list the information about all objects in the specified directory

The following code provides an example on how to call the `GetBucketV2` (`ListObjectsV2`) operation to list the information about all objects in the specified directory, including the sizes, last modified time, and names of the objects:

```
package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func HandleError(err error) {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region in which the bucket is located. For
    // example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to ht
    // ps://oss-cn-hangzhou.aliyuncs.com. Specify the endpoint based on your business requiremen
    // ts.
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account
    // to access OSS because the account has permissions on all API operations. We recommend tha
    // t you use a RAM user to call API operations or perform routine operations and maintenance
    // . To create a RAM user, log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        HandleError(err)
    }
    // Specify the bucket name.
    bucketName := "yourBucketName"
    bucket, err := client.Bucket(bucketName)
    if err != nil {
        HandleError(err)
    }
    continueToken := ""
    prefix := oss.Prefix("fun")
    for {
        lsRes, err := bucket.ListObjectsV2(prefix, oss.ContinuationToken(continueToken))
        if err != nil {
            HandleError(err)
        }
        // Display the listed objects. By default, a maximum of 100 objects are returned
        // at a time.
        for _, object := range lsRes.Objects {
            fmt.Println(object.Key, object.Type, object.Size, object.ETag, object.LastModi
            fied, object.StorageClass)
        }
        if lsRes.IsTruncated {
            continueToken = lsRes.NextContinuationToken
            prefix = oss.Prefix(lsRes.Prefix)
        } else {
            break
        }
    }
}
```

List the information about all subdirectories in the specified directory

- Call the `GetBucket (ListObjects)` operation to list the information about all subdirectories in the specified directory

The following code provides an example on how to call the `GetBucket (ListObjects)` operation to list the information about all subdirectories in the specified directory:

```

package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com. Specify the endpoint based on your business requirements.
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Specify the bucket name.
    bucketName := "yourBucketName"
    bucket, err := client.Bucket(bucketName)
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    marker := oss.Marker("")
    prefix := oss.Prefix("yourDirPrefix")
    for {
        lor, err := bucket.ListObjects(marker, prefix, oss.Delimiter("/"))
        if err != nil {
            fmt.Println("Error:", err)
            os.Exit(-1)
        }
        for _, dirName := range lor.CommonPrefixes {
            fmt.Println(dirName)
        }
        if lor.IsTruncated {
            prefix = oss.Prefix(lor.Prefix)
            marker = oss.Marker(lor.NextMarker)
        } else {
            break
        }
    }
}

```

- Call the GetBucketV2 (ListObjectsV2) operation to list the information about all subdirectories in the specified directory

The following code provides an example on how to call the GetBucketV2 (ListObjectsV2) operation to list the information about all subdirectories in the specified directory:

```

package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func HandleError(err error) {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region in which the bucket is located. For
    // example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to ht
    // ps://oss-cn-hangzhou.aliyuncs.com. Specify the endpoint based on your business requiremen
    // ts.
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account
    // to access OSS because the account has permissions on all API operations. We recommend tha
    // t you use a RAM user to call API operations or perform routine operations and maintenance
    // . To create a RAM user, log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        HandleError(err)
    }
    // Specify the bucket name.
    bucketName := "yourBucketName"
    bucket, err := client.Bucket(bucketName)
    if err != nil {
        HandleError(err)
    }
    continueToken := ""
    prefix := oss.Prefix("")
    for {
        lsRes, err := bucket.ListObjectsV2(prefix, oss.ContinuationToken(continueToken), os
        s.Delimiter("/"))
        if err != nil {
            HandleError(err)
        }
        for _, dirName := range lsRes.CommonPrefixes {
            fmt.Println(dirName)
        }
        if lsRes.IsTruncated {
            prefix = oss.Prefix(lsRes.Prefix)
            continueToken = lsRes.NextContinuationToken
        } else {
            break
        }
    }
}

```

List objects and their owner information

The following code provides an example on how to call the GetBucketV2 (ListObjectsV2) operation to list objects and their owner information:

```
package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func HandleError(err error) {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com. Specify the endpoint based on your business requirements.
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        HandleError(err)
    }
    // Specify the bucket name.
    bucketName := "yourBucketName"
    bucket, err := client.Bucket(bucketName)
    if err != nil {
        HandleError(err)
    }
    // Obtain the owner information.
    lsRes, err := bucket.ListObjectsV2(oss.FetchOwner(true))
    if err != nil {
        HandleError(err)
    }
    // Display the listed objects. By default, a maximum of 100 objects are returned at a time.
    for _, object := range lsRes.Objects {
        fmt.Println(object.Key, object.Owner.ID, object.Owner.DisplayName)
    }
}
```

Directory

OSS uses a flat structure for objects instead of a hierarchical structure to store objects. A directory is an object whose size is 0 and whose name ends with a forward slash (/). You can upload and download this object. By default, objects whose names end with a forward slash (/) are displayed as directories in the OSS console. For the complete sample code that is used to create directories, visit [GitHub](#).

You can specify the delimiter and prefix parameters to list objects by directory.

- If you set prefix to a directory name in the request, objects and subdirectories whose names contain

the prefix are listed.

- If you also set `delimiter` to a forward slash (/) in the request, the objects and subdirectories whose names start with the specified prefix in the directory are listed. Each subdirectory is listed as a single result element in `commonPrefixes`. The objects and directories in these subdirectories are not listed.

Example: The following four objects are stored in a bucket: `oss.jpg` , `fun/test.jpg` , `fun/movie/001.avi` , and `fun/movie/007.avi` . The forward slash (/) is specified as the directory delimiter. The following code provides examples on how to list objects by directory.

- List all objects in a bucket
 - The following code provides an example on how to call the `GetBucket (ListObjects)` operation to list all objects in the specified bucket:

```

package main
import (
    "fmt"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
    "os"
)
func HandleError(err error) {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region in which the bucket is located. For
    // example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to
    // https://oss-cn-hangzhou.aliyuncs.com. Specify the endpoint based on your business req
    // uirements.
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud accou
    // nt to access OSS because the account has permissions on all API operations. We recommen
    // d that you use a RAM user to call API operations or perform routine operations and main
    // tenance. To create a RAM user, log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        HandleError(err)
        os.Exit(-1)
    }
    // Specify the bucket name.
    bucketName := "yourBucketName"
    bucket, err := client.Bucket(bucketName)
    if err != nil {
        HandleError(err)
        os.Exit(-1)
    }
    // List all objects in the specified bucket.
    marker := oss.Marker("")
    for {
        lsRes, err := bucket.ListObjects(marker)
        if err != nil {
            HandleError(err)
            os.Exit(-1)
        }
        for _, object := range lsRes.Objects {
            fmt.Println("Bucket", object.Key)
        }
        if lsRes.IsTruncated {
            marker = oss.Marker(lsRes.NextMarker)
        } else {
            break
        }
    }
}

```

- The following code provides an example on how to call the GetBucketV2 (ListObjectsV2) operation to list all objects in the specified bucket:

```

package main
import (
    "fmt"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
    "os"
)
func HandleError(err error) {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region in which the bucket is located. For
    // example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to
    // https://oss-cn-hangzhou.aliyuncs.com. Specify the endpoint based on your business req
    // uirements.
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud accou
    // nt to access OSS because the account has permissions on all API operations. We recommen
    // d that you use a RAM user to call API operations or perform routine operations and main
    // tenance. To create a RAM user, log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        HandleError(err)
        os.Exit(-1)
    }
    // Specify the bucket name.
    bucketName := "yourBucketName"
    bucket, err := client.Bucket(bucketName)
    if err != nil {
        HandleError(err)
        os.Exit(-1)
    }
    // List all objects in the specified bucket.
    startAfter := ""
    continueToken := ""
    for {
        lsRes, err := bucket.ListObjectsV2(oss.StartAfter(startAfter), oss.ContinuationT
        oken(continueToken))
        if err != nil {
            HandleError(err)
            os.Exit(-1)
        }
        for _, object := range lsRes.Objects {
            fmt.Println("Bucket", object.Key)
        }
        if lsRes.IsTruncated {
            startAfter = lsRes.StartAfter
            continueToken = lsRes.NextContinuationToken
        } else {
            break
        }
    }
}

```

- The following results are returned when you use the preceding two methods to list all objects in the bucket:

```
Objects:
fun/movie/001.avi
fun/movie/007.avi
fun/test.jpg
oss.jpg
CommonPrefixes:
```

- List all objects in the specified directory
 - The following code provides an example on how to call the GetBucket (ListObjects) operation to list all objects in the specified directory:

```

package main
import (
    "fmt"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
    "os"
)
func HandleError(err error) {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region in which the bucket is located. For
    // example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to
    // https://oss-cn-hangzhou.aliyuncs.com. Specify the endpoint based on your business req
    // uirements.
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud accou
    // nt to access OSS because the account has permissions on all API operations. We recommen
    // d that you use a RAM user to call API operations or perform routine operations and main
    // tenance. To create a RAM user, log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Specify the bucket name.
    bucket, err := client.Bucket("yourBucketName")
    if err != nil {
        HandleError(err)
        os.Exit(-1)
    }
    // List all objects in the specified directory.
    prefix := oss.Prefix("aaa/db-init")
    marker := oss.Marker("")
    for {
        lsRes, err := bucket.ListObjects(prefix, marker)
        if err != nil {
            HandleError(err)
            os.Exit(-1)
        }
        for _, object := range lsRes.Objects {
            fmt.Println("Bucket", object.Key)
        }
        if lsRes.IsTruncated {
            prefix = oss.Prefix(lsRes.Prefix)
            marker = oss.Marker(lsRes.NextMarker)
        } else {
            break
        }
    }
}

```

- The following code provides an example on how to call the GetBucketV2 (ListObjectsV2) operation

to list all objects in the specified directory:

```
package main
import (
    "fmt"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
    "os"
)
func HandleError(err error) {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com. Specify the endpoint based on your business requirements.
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Specify the bucket name.
    bucket, err := client.Bucket("yourBucketName")
    if err != nil {
        HandleError(err)
        os.Exit(-1)
    }
    // List all objects in the specified directory.
    prefix := oss.Prefix("aaa/db-init")
    continueToken := ""
    for {
        lsRes, err := bucket.ListObjectsV2(prefix, oss.ContinuationToken(continueToken))
        if err != nil {
            HandleError(err)
            os.Exit(-1)
        }
        for _, object := range lsRes.Objects {
            fmt.Println("Bucket", object.Key)
        }
        if lsRes.IsTruncated {
            prefix = oss.Prefix(lsRes.Prefix)
            continueToken = lsRes.NextContinuationToken
        } else {
            break
        }
    }
}
```

- The following results are returned when you use the preceding two methods to list all objects in the specified directory:

```
Objects:
fun/movie/001.avi
fun/movie/007.avi
fun/test.jpg
CommonPrefixes:
```

- List objects and subdirectories in the specified directory
 - The following code provides an example on how to call the GetBucket (ListObjects) operation to list objects and subdirectories in the specified directory:

```
package main
import (
    "fmt"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
    "os"
)
func HandleError(err error) {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com. Specify the endpoint based on your business requirements.
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Specify the bucket name.
    bucket, err := client.Bucket("yourBucketName")
    if err != nil {
        HandleError(err)
        os.Exit(-1)
    }
    // List objects and subdirectories in the specified directory.
    prefix := oss.Prefix("aaa/db-init/")
    marker := oss.Marker("")
    delimiter := "/"
    for {
        lsRes, err := bucket.ListObjects(prefix, marker, oss.Delimiter(delimiter))
        if err != nil {
            HandleError(err)
            os.Exit(-1)
        }
    }
```

```

    for _, object := range lsRes.Objects {
        fmt.Println("Objects", object.Key)
    }
    for _, object := range lsRes.CommonPrefixes {
        fmt.Println("CommonPrefixes", object)
    }
    if lsRes.IsTruncated {
        prefix = oss.Prefix(lsRes.Prefix)
        marker = oss.Marker(lsRes.NextMarker)
    } else {
        break
    }
}
}

```

- The following code provides an example on how to call the GetBucketV2 (ListObjectsV2) operation to list objects and subdirectories in the specified directory:

```

package main
import (
    "fmt"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
    "os"
)
func HandleError(err error) {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region in which the bucket is located. For
    // example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to
    // https://oss-cn-hangzhou.aliyuncs.com. Specify the endpoint based on your business re-
    // quirements.
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account
    // to access OSS because the account has permissions on all API operations. We recom-
    // mend that you use a RAM user to call API operations or perform routine operations and
    // maintenance. To create a RAM user, log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Specify the bucket name.
    bucket, err := client.Bucket("yourBucketName")
    if err != nil {
        HandleError(err)
        os.Exit(-1)
    }
    // List objects and subdirectories in the specified directory.
    prefix := oss.Prefix("")
    continueToken := ""
    delimiter := "/"
    for {
        lsRes, err := bucket.ListObjectsV2(prefix, oss.Delimiter(delimiter), oss.Continua

```



```

tionToken(continueToken))
    if err != nil {
        HandleError(err)
        os.Exit(-1)
    }
    for _, object := range lsRes.Objects {
        fmt.Println("Objects", object.Key)
    }
    for _, object := range lsRes.CommonPrefixes {
        fmt.Println("CommonPrefixes", object)
    }
    if lsRes.IsTruncated {
        prefix = oss.Prefix(lsRes.Prefix)
        continueToken = lsRes.NextContinuationToken
    } else {
        break
    }
}
}

```

- The following results are returned when you use the preceding two methods to list objects and subdirectories in the specified directory:

```

Objects:
fun/test.jpg
CommonPrefixes:
fun/movie/

```

- List the sizes of objects in the specified directory
 - The following code provides an example on how to call the GetBucket (ListObjects) operation to list the sizes of objects in the specified directory:

```

package main
import (
    "fmt"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
    "os"
)
func HandleError(err error) {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region in which the bucket is located. For
    // example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to
    // https://oss-cn-hangzhou.aliyuncs.com. Specify the endpoint based on your business req
    // uirements.
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud accou
    // nt to access OSS because the account has permissions on all API operations. We recommen
    // d that you use a RAM user to call API operations or perform routine operations and main
    // tenance. To create a RAM user, log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Specify the bucket name.
    bucket, err := client.Bucket("yourBucketName")
    if err != nil {
        HandleError(err)
        os.Exit(-1)
    }
    // List the sizes of objects in the specified directory.
    prefix := "test/"
    marker := ""
    for {
        lsRes, err := bucket.ListObjects(oss.Prefix(prefix), oss.Marker(marker))
        if err != nil {
            HandleError(err)
            os.Exit(-1)
        }
        for _, object := range lsRes.Objects {
            fmt.Println("Objects", object.Key, "length", object.Size, "Byte.")
        }
        if lsRes.IsTruncated {
            marker = lsRes.NextMarker
            prefix = lsRes.Prefix
        } else {
            break
        }
    }
}

```

- The following code provides an example on how to call the GetBucketV2 (ListObjectsV2) operation

to list the sizes of objects in the specified directory:

```
package main
import (
    "fmt"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
    "os"
)
func HandleError(err error) {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com. Specify the endpoint based on your business requirements.
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Obtain the bucket.
    bucket, err := client.Bucket("yourBucketName")
    if err != nil {
        HandleError(err)
        os.Exit(-1)
    }
    // List the sizes of objects in the specified directory.
    prefix := oss.Prefix("/fun")
    continueToken := ""
    for {
        lsRes, err := bucket.ListObjectsV2(prefix, oss.ContinuationToken(continueToken))
        if err != nil {
            HandleError(err)
            os.Exit(-1)
        }
        for _, object := range lsRes.Objects {
            fmt.Println("Objects", object.Key, "length", object.Size, "Byte.")
        }
        if lsRes.IsTruncated {
            prefix = oss.Prefix(lsRes.Prefix)
            continueToken = lsRes.NextContinuationToken
        } else {
            break
        }
    }
}
```

References

- For more information about the complete sample code that is used to list objects, visit [GitHub](#).
- For more information about the API operations that you can call to list objects, see [GetBucket \(List Objects\)](#) and [GetBucketV2 \(List ObjectsV2\)](#).

9.8.7. Delete objects

You can delete a single object, multiple specified objects, objects whose names contain a specified prefix, or a specified directory and all objects in the directory.



Warning Deleted objects cannot be recovered. Exercise caution when you delete objects.

Delete a single object

The following code provides an example on how to delete an object named *exampleobject.txt* from a bucket named *examplebucket*:

```
package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Create an OSSClient instance.
    // Specify the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com. Specify your actual endpoint.
    // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in Object Storage Service (OSS) is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Specify the bucket name. Example: examplebucket.
    bucketName := "examplebucket"
    // Set objectName to the full path of the object that you want to delete. The full path must contain the extension of the object name and cannot contain the bucket name. Example: exampledir/exampleobject.txt.
    // To delete a directory, set objectName to the directory name. If the directory contains objects, you must delete all objects from the directory before you can delete the directory.
    objectName := "exampledir/exampleobject.txt"
    bucket, err := client.Bucket(bucketName)
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Delete a single object.
    err = bucket.DeleteObject(objectName)
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
}
```

Delete multiple objects

You can delete up to 1,000 objects at a time. You can delete multiple specified objects, objects whose names contain a specified prefix, or a specified directory and all objects in the directory.

You can also configure lifecycle rules to automatically delete objects. For more information, see [Lifecycle rules based on the last modified time](#).

Delete multiple specified objects

The following code provides an example on how to delete multiple specified objects:

```

package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Create an OSSClient instance.
    // Specify the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com. Specify your actual endpoint.
    // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Specify the bucket name. Example: examplebucket.
    bucketName := "examplebucket"
    bucket, err := client.Bucket(bucketName)
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Specify the full paths of multiple objects that you want to delete. The full paths cannot contain the bucket name.
    delRes, err := bucket.DeleteObjects([]string{"my-object-1", "my-object-2"})
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    fmt.Println("Deleted Objects:", delRes.DeletedObjects)
    // Set oss.DeleteObjectsQuiet to true, which indicates that the list of deleted objects is not returned.
    _, err = bucket.DeleteObjects([]string{"my-object-3", "my-object-4"},
    oss.DeleteObjectsQuiet(true))
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
}

```

Delete multiple objects with a specified prefix or in a specified directory

The following code provides an example on how to delete multiple objects with a specified prefix and how to delete a specified directory and all objects in the directory.

 **Warning** If the prefix is not specified or is set to NULL in the following code, all objects in the bucket are deleted. Exercise caution when you configure the prefix.

```
package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Create an OSSClient instance.
    // Specify the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com. Specify your actual endpoint.
    // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Specify the bucket name. Example: examplebucket.
    bucket, err := client.Bucket("examplebucket")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // List all objects whose names contain the specified prefix and then delete the objects.
    marker := oss.Marker("")
    // Specify the prefix of the objects that you want to delete. If you want to delete all objects whose names contain the src prefix, set the prefix to src. After you set the prefix to src, all non-directory objects whose names contain the src prefix, the src directory, and all objects in the directory are deleted.
    prefix := oss.Prefix("src")
    // If you want to delete only the src directory and all objects in the directory, set the prefix to src/.
    // prefix := oss.Prefix("src/")
    count := 0
    for {
        lor, err := bucket.ListObjects(marker, prefix)
        if err != nil {
            fmt.Println("Error:", err)
            os.Exit(-1)
        }
        objects := []string{}
        for _, object := range lor.Objects {
            objects = append(objects, object.Key)
        }
        // Set oss.DeleteObjectsQuiet to true, which indicates that the list of deleted objects is not returned.
```

```

    delRes, err := bucket.DeleteObjects(objects, oss.DeleteObjectsQuiet(true))
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    if len(delRes.DeletedObjects) > 0 {
        fmt.Println("these objects deleted failure,", delRes.DeletedObjects)
        os.Exit(-1)
    }
    count += len(objects)
    prefix = oss.Prefix(lor.Prefix)
    marker = oss.Marker(lor.NextMarker)
    if !lor.IsTruncated {
        break
    }
}
fmt.Printf("success,total delete object count:%d\n", count)
}

```

References

- For the complete sample code that is used to delete objects, visit [GitHub](#).
- For more information about the API operation that you can call to delete a single object, see [DeleteObject](#).
- For more information about the API operation that you can call to delete multiple objects, see [DeleteMultipleObjects](#).

9.8.8. Copy objects

Copy a small object

You can call `CopyObject` to copy an object within a bucket or across buckets in the same region. We recommend that you call `CopyObject` to copy only objects that are smaller than 1 GB in size.

- Copy an object within a bucket

The following code provides an example on how to copy an object within a bucket:

```

package main
import (
    "fmt"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
    "os"
    "time"
)
func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region in which the bucket is located. For
    // example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to htt
    // ps://oss-cn-hangzhou.aliyuncs.com. Specify the endpoint based on your business requiremen
    // ts.
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account
    // to access Object Storage Service (OSS) because the account has permissions on all API ope

```


rations. We recommend that you use a RAM user to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.

```

client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
if err != nil {
    fmt.Println("Error:", err)
    os.Exit(-1)
}

// Specify the name of the bucket. Example: examplebucket.
bucketName := "examplebucket"
// Specify the full path of the source object. Example: srcdir/srcobject.jpg.
objectName := "srcdir/srcobject.jpg"
// Specify the full path of the destination object. Example: destdir/destobject.jpg.
destObjectName := "destdir/destobject.jpg"
// Obtain the bucket.
bucket, err := client.Bucket(bucketName)
if err != nil {
    fmt.Println("Error:", err)
    os.Exit(-1)
}

// Specify metadata for the destination object.
expires := time.Date(2049, time.January, 10, 23, 0, 0, 0, time.UTC)
tag1 := oss.Tag{
    Key:    "a",
    Value:  "1",
}
taggingInfo := oss.Tagging{
    Tags: []oss.Tag{tag1},
}
options := []oss.Option{
    oss.MetadataDirective(oss.MetaReplace),
    oss.Expires(expires),
    oss.SetTagging(taggingInfo),
    // Specify that the tags of the source object are copied to the destination object.
    // oss.TaggingDirective(oss.TaggingCopy),
    // Set the access control list (ACL) of the destination object to private when OSS creates the destination object.
    // oss.ObjectACL(oss.ACLPrivate),
    // Specify the customer master key (CMK) that is managed by Key Management Service (KMS). This parameter takes effect only when x-oss-server-side-encryption is set to KMS.
    // oss.ServerSideEncryptionKeyID("9468da86-3509-4f8d-a61e-6eableac****"),
    // Specify the server-side encryption algorithm that is used to encrypt the destination object when OSS creates the destination object.
    // oss.ServerSideEncryption("AES256"),
    // Specify that the metadata of the source object is copied to the destination object.
    // oss.MetadataDirective(oss.MetaCopy),
    // Specify whether the CopyObject operation overwrites the object with the same name. In this example, this parameter is set to true, which indicates that the object with the same name cannot be overwritten.
    // oss.ForbidOverWrite(true),
    // If the ETag value of the source object is the same as the ETag value that is specified in the request, OSS copies the object and returns 200 OK.

```

```

    //oss.CopySourceIfMatch("5B3C1A2E053D763E1B002CC607C5****"),
    // If the ETag value of the source object is different from the ETag value that is
    // specified in the request, OSS copies the object and returns 200 OK.
    //oss.CopySourceIfNoneMatch("5B3C1A2E053D763E1B002CC607C5****"),
    // If the time that is specified in the request is earlier than the modified time
    // of the object, OSS copies the object and returns 200 OK.
    //oss.CopySourceIfModifiedSince(2021-12-09T07:01:56.000Z),
    // If the time that is specified in the request is the same as or later than the
    // modified time of the object, OSS copies the object and returns 200 OK.
    //oss.CopySourceIfUnmodifiedSince(2021-12-09T07:01:56.000Z),
    // Specify the storage class of the destination object. In this example, this parameter
    // is set to Standard.
    //oss.StorageClass("Standard"),
    }
    // Specify that the metadata of the source object is overwritten by the metadata that
    // is specified in the request.
    _, err = bucket.CopyObject(objectName, destObjectName, options...)
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
}

```

- Copy an object across buckets in the same region

The following code provides an example on how to copy an object across buckets in the same region:

```

package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to http://oss-cn-hangzhou.aliyuncs.com. Specify the endpoint based on your business requirements.
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Specify the name of the source bucket. Example: srcbucket.
    srcBucketName := "srcbucket"
    // Specify the full path of the source object. Example: srcobject.jpg.
    srcObjectName := "srcobject.jpg"
    // Specify the full path of the destination object. Example: destobject.jpg.
    dstObjectName := "destobject.jpg"
    // Specify the name of the destination bucket. Example: destbucket.
    destBucketName := "destbucket"
    // Obtain the bucket.
    bucket, err := client.Bucket(destBucketName)
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Copy an object named srcobject.jpg from the srcbucket bucket to an object named destobject.jpg in the destbucket bucket.
    bucket.CopyObjectFrom(srcBucketName, srcObjectName, dstObjectName)
    if err != nil {
        fmt.Println("CopyObjectFrom Error:", err)
        os.Exit(-1)
    }
}

```

Copy a large object

The following code provides an example on how to copy a large object:

```

package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)

```

```

func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region in which the bucket is located. For e
    // xample, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https:
    // //oss-cn-hangzhou.aliyuncs.com. Specify the endpoint based on your business requirements.
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account t
    // o access OSS because the account has permissions on all API operations. We recommend that y
    // ou use a RAM user to call API operations or perform routine operations and maintenance. To
    // create a RAM user, log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Specify the name of the bucket. Example: examplebucket.
    bucketName := "examplebucket"
    // Obtain the bucket.
    bucket, err := client.Bucket(bucketName)
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Specify the full path of the source object. Example: srcobject.txt.
    objectSrc := "srcobject.txt"
    // Specify the full path of the destination object. Example: destobject.txt.
    objectDest := "destobject.txt"
    var fileName = "/Users/liuzekun/Desktop/gol.11.8.darwin-amd64.pkg"
    chunks, err := oss.SplitFileByPartNum(fileName, 3)
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    err = bucket.PutObjectFromFile(objectSrc, fileName)
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    imur, err := bucket.InitiateMultipartUpload(objectDest)
    var parts []oss.UploadPart
    for _, chunk := range chunks {
        // Specify that the metadata of the source object is overwritten by the metadata th
        // at is specified in the request.
        options := []oss.Option{
            // Specify the range of data that you want to copy. For example, if you specify
            // 0 to 1023 as the range of data that you want to copy, the first 1024 bytes of the source ob
            // ject are copied.
            // oss.CopySourceRange(0,1023),
            // If the ETag value of the source object is the same as the ETag value that is
            // specified in the request, OSS copies the object and returns 200 OK.
            // oss.CopySourceIfMatch("5B3C1A2E053D763E1B002CC607C5****"),
            // If the ETag value of the source object is different from the ETag value that
            // is specified in the request, OSS copies the object and returns 200 OK.
            // oss.CopySourceIfNoneMatch("5B3C1A2E053D763E1B002CC607C5****"),
            // If the time that is specified in the request is the same as or later than th

```

```
e modified time of the object, OSS copies the object and returns 200 OK.
    // oss.CopySourceIfModifiedSince(2021-12-09T07:01:56.000Z),
    // If the time specified in the request is earlier than the modified time of th
e object, OSS copies the object and returns 200 OK.
    // oss.CopySourceIfUnmodifiedSince(2021-12-09T07:01:56.000Z),
}
part, err := bucket.UploadPartCopy(imur, bucketName, objectSrc, chunk.Offset, chunk
.Size, chunk.Number ,options...)
if err != nil {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
parts = append(parts, part)
}
cmur, err := bucket.CompleteMultipartUpload(imur, parts)
fmt.Println("cmur:", cmur)
if err != nil {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
}
```

References

- For more information about the complete sample code that is used to copy an object, visit [GitHub](#).
- For more information about the API operation that you can call to copy a small object, see [CopyObject](#).
- For more information about the API operation that you can call to copy a large object, see [UploadPart Copy](#).

9.8.9. Disable overwrite for objects with the same name

By default, if you upload an object that has the same name of an existing object, the existing object is overwritten by the uploaded object. This topic describes how to set the x-oss-forbid-overwrite request header to disable overwrite for objects with the same name when you copy objects or use the simple upload and multipart upload methods.

Simple upload

The following code provides an example on how to prevent objects from being overwritten by objects that have the same names in simple upload:

```
package main
import (
    "fmt"
    "os"
    "strings"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Create an OSSClient instance.
    client, err := oss.New("<yourEndpoint>", "<yourAccessKeyId>", "<yourAccessKeySecret>")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Specify the bucket name.
    bucket, err := client.Bucket("<yourBucketName>")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Specify whether to overwrite the object with the same name.
    // By default, if oss.ForbidOverWrite is not specified, the object with the same name i
s overwritten.
    // If oss.ForbidOverWrite is set to false, the object with the same name is overwritten
.
    // If oss.ForbidOverWrite is set to true, the object with the same name is not overwrit
ten. If an object with the same name already exists, an error is reported.
    forbidWrite := oss.ForbidOverWrite(true)
    // Upload a string.
    err = bucket.PutObject("<yourObjectName>", strings.NewReader("yourObjectValue"), forbid
Write)
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
}
```

Object copy

The following code provides an example on how to disable overwrite for the object with the same name when you copy an object:

```
func main() {
    // Create an OSSClient instance.
    client, err := oss.New("<yourEndpoint>", "<yourAccessKeyId>", "<yourAccessKeySecret>")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    bucketName := "<yourBucketName>"
    objectName := "<yourObjectName>"
    objectName := "<yourObjectName>"
    // Specify the bucket name.
    bucket, err := client.Bucket(bucketName)
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Specify whether to overwrite the target object with the same name.
    // By default, if oss.ForbidOverWrite is not specified, the object with the same name i
s overwritten.
    // If oss.ForbidOverWrite is set to false, the object with the same name is overwritten
.
    // If oss.ForbidOverWrite is set to true, the object with the same name is not overwrit
ten. If an object with the same name already exists, an error is reported.
    forbidWrite := oss.ForbidOverWrite(true)
    // Copy the object to another object in the same bucket.
    _, err = bucket.CopyObject(objectName, destObjectName, forbidWrite)
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
}
```

Multipart upload

The following code provides an example on how to prevent objects from being overwritten by newly uploaded objects that have the same names in multipart upload:

```

func main() {
    // Create an OSSClient instance.
    client, err := oss.New("<yourEndpoint>", "<yourAccessKeyId>", "<yourAccessKeySecret>")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    bucketName := "<yourBucketName>"
    objectName := "<yourObjectName>"
    String localFileName = "<yourLocalFileName>"
    // Specify the bucket name.
    bucket, err := client.Bucket(bucketName)
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    chunks, err := oss.SplitFileByPartNum("localFile", 3)
    fd1, err := os.Open(localFileOne)
    defer fd.Close()
    // Specify whether to overwrite the object with the same name.
    // By default, if oss.ForbidOverWrite is not specified, the object with the same name i
s overwritten.
    // If oss.ForbidOverWrite is set to false, the object with the same name is overwritten
.
    // If oss.ForbidOverWrite is set to true, the object with the same name is not overwrit
ten. If an object with the same name already exists, an error is reported.
    forbidWrite := oss.ForbidOverWrite(true)
    // Step 1: Initiate a multipart part upload task.
    imur, err := bucket.InitiateMultipartUpload(objectName, forbidWrite)
    // Step 2: Upload the parts.
    var parts []oss.UploadPart
    for _, chunk := range chunks {
        fd.Seek(chunk.Offset, os.SEEK_SET)
        // Call the UploadPart method to upload each part.
        part, err := bucket.UploadPart(imur, fd, chunk.Size, chunk.Number)
        if err != nil {
            fmt.Println("Error:", err)
            os.Exit(-1)
        }
        parts = append(parts, part)
    }
    // Step 3: Complete the multipart upload task. Disable overwrite for the object with th
e same name.
    cmur, err := bucket.CompleteMultipartUpload(imur, parts, forbidWrite)
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    fmt.Println("cmur:", cmur)
}

```


9.8.10. Restore objects

Notice

Restoration

If you want to read an Archive object or a Cold Archive object, you must restore the object in advance. It takes several minutes to restore an Archive object and takes several hours to restore a Cold Archive object.

The following section describes the status of an Archive object or a Cold Archive object throughout the restoration process:

1. By default, an Archive or a Cold Archive object is in the frozen state before restoration.
2. After you submit a restore request, the object is in the restoring state.
3. After the server completes the restore task, the object enters the restored state and you can read the object.
 - Archive objects

For Archive objects, the restored state lasts 24 hours by default. During the 24 hours, if you call `RestoreObject` again, the restored state is extended by 24 hours. You can extend the restored state to seven days by calling `RestoreObject` six times during the first 24 hours of restored state. You can also configure the duration of the restored state in days by calling `RestoreObject` once. You can specify a duration of at most seven days.
 - Cold Archive objects

For Cold Archive objects, you can specify the duration of restored state and restoration priority. The duration of the restored state must be at least one day and at most 365 days. The time required to restore a Cold Archive object to the readable state is determined based on the restoration priority of the object:

 - Expedited: The object is restored within one hour.
 - Standard: The object is restored within two to five hours. If the `JobParameters` element is not passed in, the default restoration priority is Standard.
 - Bulk: The object is restored within five to twelve hours.
4. After the restored state expires, the object returns to the frozen state.

Restore Archive objects

The following code provides an example on how to restore an Archive object:

```
package main
import (
    "fmt"
    "os"
    "strings"
    "time"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func HandleError(err error) {
    fmt.Println("Error:", err)
```

```

    os.Exit(-1)
}

func main() {
    // Create an OSSClient instance.
    // Specify the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com. Specify your actual endpoint.
    // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        HandleError(err)
    }
    // Specify the bucket name. Example: examplebucket.
    bucketName := "examplebucket"
    // Specify the full path of the object. The full path of the object cannot contain the bucket name. Example: exampledir/exampleobject.txt.
    objectName := "exampledir/exampleobject.txt"
    // Specify the full path of the local file that you want to upload. Example: D:\\localpath\\examplefile.txt.
    localFilename := "D:\\localpath\\examplefile.txt"
    // Create the bucket.
    err = client.CreateBucket(bucketName, oss.StorageClass(oss.StorageArchive))
    if err != nil {
        HandleError(err)
    }
    archiveBucket, err := client.Bucket(bucketName)
    if err != nil {
        HandleError(err)
    }
    // Upload the object.
    var val = "More than just cloud."
    err = archiveBucket.PutObject(objectName, strings.NewReader(val))
    if err != nil {
        HandleError(err)
    }
    // Query the storage class of the object.
    meta, err := archiveBucket.GetObjectDetailedMeta(objectName)
    if err != nil {
        HandleError(err)
    }
    fmt.Println("X-Oss-Storage-Class : ", meta.Get("X-Oss-Storage-Class"))
    // Restore the Archive object.
    err = archiveBucket.RestoreObject(objectName)
    if err != nil {
        HandleError(err)
    }
    // Wait until the Archive object is restored.
    meta, err = archiveBucket.GetObjectDetailedMeta(objectName)
    if err != nil {
        HandleError(err)
    }
}

```

```
for meta.Get("X-Oss-Restore") == "ongoing-request=\"true\"" {
    fmt.Println("x-oss-restore:" + meta.Get("X-Oss-Restore"))
    time.Sleep(10 * time.Second)
    meta, err = archiveBucket.GetObjectDetailedMeta(objectName)
    if err != nil {
        HandleError(err)
    }
}
fmt.Println("x-oss-restore:" + meta.Get("X-Oss-Restore"))
// Download the restored object.
err = archiveBucket.GetObjectToFile(objectName, localFilename)
if err != nil {
    HandleError(err)
}
// Restore the object again. The duration for which the object can remain in the restored state is extended by 24 hours.
err = archiveBucket.RestoreObject(objectName)
if err != nil {
    HandleError(err)
}
fmt.Println("ArchiveSample completed")
}
```

Restore a Cold Archive object

The following code provides an example on how to restore a Cold Archive object:

```
package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Create an OSSClient instance.
    // Specify the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com. Specify your actual endpoint.
    // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Specify the bucket name. Example: examplebucket.
    bucket, err := client.Bucket("examplebucket")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    var restoreConfig oss.RestoreConfiguration
    // The object is restored within 1 hour.
    // restoreConfig.Tier = string(oss.RestoreExpedited)
    // The object is restored within 2 to 5 hours.
    // restoreConfig.Tier = string(oss.RestoreStandard)
    // The object is restored within 5 to 12 hours.
    // restoreConfig.Tier = string(oss.RestoreBulk)
    // Specify the duration for which the object can remain in the restored state. Unit: days. Valid values: 1 to 365.
    restoreConfig.Days = 1
    // Specify the full path of the object. The full path of the object cannot contain the bucket name. Example: exampledir/exampleobject.txt.
    err = bucket.RestoreObjectDetail("exampledir/exampleobject.txt", restoreConfig)
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
}
```

References

For more information about the API operation that you can call to restore Archive and Cold Archive objects, see [RestoreObject](#).

9.8.11. Manage symbolic links

Symbolic links can be used to access objects that are frequently used in buckets. After you create a symbolic link for an object, you can use the symbolic link to access the object.

Create a symbolic link

The following code provides an example on how to create a symbolic link:

```

package main
import (
    "fmt"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
    "os"
    "strings"
)
func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region in which the bucket is located. For e
    // xample, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https:
    // //oss-cn-hangzhou.aliyuncs.com. Specify the endpoint based on your business requirements.
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account t
    // o access Object Storage Service (OSS) because the account has permissions on all API operat
    // ions. We recommend that you use a RAM user to call API operations or perform routine operat
    // ions and maintenance. To create a RAM user, log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Specify the name of the bucket. Example: examplebucket.
    bucketName := "examplebucket"
    // Specify the name of the symbolic link. Example: examplesymlink.
    symObjectKey := "examplesymlink.txt"
    // Specify the name of the object to which the symbolic link points. Example: exampleob
    // ject.txt.
    objectName := "exampleobject.txt"
    bucket, err := client.Bucket(bucketName)
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Create the symbolic link.
    option := [] oss.Option{
        // Specify whether to overwrite the object with the same name when the symbolic lin
        // k is created.
        oss.ForbidOverWrite(true),
        // Specify the access control list (ACL) of the object. In this example, this param
        // eter is set to private.
        oss.ObjectACL(oss.ACLPrivate),
        // Specify the storage class of the object. In this example, this parameter is set
        // to Standard.
        oss.StorageClass(oss.StorageStandard),
    }
    err = bucket.PutSymlink(symObjectKey, objectName, option...)
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
}

```

Query the name of the object to which a symbolic link points

To query a symbolic link, you must have read permissions on the symbolic link. The following code provides an example on how to query a symbolic link and the name of the object to which the symbolic link points:

```
package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com. Specify the endpoint based on your business requirements.
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Specify the name of the bucket. Example: examplebucket.
    bucketName := "examplebucket"
    // Specify the name of the symbolic link.
    symlinkName := "examplesymlink.txt"
    bucket, err := client.Bucket(bucketName)
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Query the name of the object to which the symbolic link points.
    meta, err := bucket.GetSymlink(symlinkName)
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    fmt.Println(meta.Get(oss.HTTPHeaderOssSymlinkTarget))
}
```

References

- For more information about the API operation that you can call to create a symbolic link, see [PutSymlink](#).
- For more information about the API operation that you can call to query a symbolic link, see [GetSymlink](#).

9.9. Versioning

9.9.1. Versioning

Versioning applies to all objects in a bucket.

By versioning, you can restore an object in the bucket to any of its historical versions if it is accidentally overwritten or deleted. A bucket can be in the following three versioning states: not versioned (default), enabled, and suspended.

For more information about versioning, see [Overview](#) in the Developer Guide.

Set the versioning state of a bucket

You can run the following code to set the versioning state of a bucket to enabled or suspended:

```
package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Creates an OSSClient instance.
    client, err := oss.New("<yourEndpoint>", "<yourAccessKeyId>", "<yourAccessKeySecret>")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Creates a bucket.
    err = client.CreateBucket("<yourBucketName>")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Sets the versioning state of the bucket to Enabled or Suspended. In this example, the
    versioning states is set to Enabled.
    config := oss.VersioningConfig{Status: "Enabled"}
    err = client.SetBucketVersioning("<yourBucketName>", config)
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
}
```

For more information about setting the versioning state of a bucket, see [PutBucketVersioning](#).

Obtain the versioning state of a bucket

You can run the following code to obtain the versioning state of a bucket:


```
package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Creates an OSSClient instance.
    client, err := oss.New("<yourEndpoint>", "<yourAccessKeyId>", "<yourAccessKeySecret>")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Obtains the versioning state of the bucket.
    ret, err := client.GetBucketVersioning("<yourBucketName>")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Prints the versioning state of the bucket.
    fmt.Println("The bucket Versioning status:", ret.Status)
}
```

For more information about obtaining the versioning state of a bucket, see [GetBucketVersioning](#).

List the versions of all objects in a bucket

Run the following code to list the versions of all objects (including delete markers) in a specified bucket:

```
package main
import (
    "fmt"
    "net/http"
    "os"
    "strings"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Creates an OSSClient instance.
    client, err := oss.New("<yourEndpoint>", "<yourAccessKeyId>", "<yourAccessKeySecret>")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Obtains the bucket.
    bucket, err := client.Bucket("<yourBucketName>")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Lists objects with the specified prefix.
    ret, err := bucket.ListObjectVersions(oss.Prefix("yourObjectPrefix"))
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Prints the object information.
    for _, object := range ret.ObjectVersions {
        fmt.Println("Object:", object)
    }
    // Prints the delete markers.
    for _, marker := range ret.ObjectDeleteMarkers {
        fmt.Println("Delete Markers:", marker)
    }
}
```

For more information about listing the versions of all objects (including delete markers) in a bucket, see [GetBucketVersions\(ListObjectVersions\)](#).

9.9.2. Upload objects

This topic describes how to upload objects to a bucket with versioning enabled or suspended.

Simple upload

When you upload an object to a bucket with versioning enabled, OSS automatically adds a unique version ID for the uploaded object, and returns the version ID as the `x-oss-version-id` field in the response header. In a bucket with versioning suspended, the version ID added to an upload object is null. If you upload an object with the same name as an existing object to a bucket with versioning suspended, the existing object is overwritten to ensure that each object only has one version of which the ID is null.

You can run the following code to simply upload an object :

```
package main
import (
    "fmt"
    "net/http"
    "os"
    "strings"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Creates an OSSClient instance.
    client, err := oss.New("<yourEndpoint>", "<yourAccessKeyId>", "<yourAccessKeySecret>")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Obtains the bucket.
    bucket, err := client.Bucket("<yourBucketName>")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    var retHeader http.Header
    // Uploads a string. The oss.GetResponseHeader method is used to obtain the returned header.
    err = bucket.PutObject("<yourObjectName>", strings.NewReader("yourObjectValue"), oss.GetResponseHeader(&retHeader))
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Prints x-oss-version-id.
    fmt.Println("x-oss-version-id:", oss.GetVersionId(retHeader))
}
```

For more information about simple upload, see [PutObject](#).

Append upload

In a bucket with versioning enabled or suspended, the AppendObject operation can be performed only on objects of which the current version is an appendable object.

 Note

- When you perform the AppendObject operation on an object of which the current version is an appendable object, OSS does not generate a historical version for the appendable object.
- When you perform the PutObject or DeleteObject operation on an object of which the current version is an appendable object, OSS stores the appendable object as a historical version and prevents the object from being further appended.
- You cannot perform the AppendObject operation on objects of which the current version is not an appendable object (such as a normal object or a delete marker).

You can run the following code to append files to an existing object:

```
package main
import (
    "fmt"
    "os"
    "strings"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Creates an OSSClient.
    client, err := oss.New("<yourEndpoint>", "<yourAccessKeyId>", "<yourAccessKeySecret>")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Obtains the bucket.
    bucket, err := client.Bucket("<yourBucketName>")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // For the first append operation, the value position is 0 and the returned value is the
    position for the next append operation. The position for the next operation is the size of
    the object appended in the last operation.
    var retHeader http.Header
    var nextPos int64 = 0
    nextPos, err = bucket.AppendObject("<yourObjectName>", strings.NewReader("YourObjectAppen
dValue1"), nextPos, oss.GetResponseHeader(&retHeader))
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Prints x-oss-version-id.
    fmt.Println("x-oss-version-id:", retHeader.Get("x-oss-version-id"))
    // Performs the second append operations. The oss.GetResponseHeader method is used to obt
    ain the returned header.
    nextPos, err = bucket.AppendObject("<yourObjectName>", strings.NewReader("YourObjectAppen
dValue2"), nextPos, oss.GetResponseHeader(&retHeader))
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Prints x-oss-version-id.
    fmt.Println("x-oss-version-id:", oss.GetVersionId(retHeader))
    // You can append data to an object for multiple times.
}
```

For more information about append upload, see [AppendObject](#).

Multipart upload

If you call `CompleteMultipartUpload` to complete the MultipartUpload task for an object in a bucket with versioning enabled or suspended, OSS generates a unique version ID for the object and returns it as the `x-oss-version-id` field in the response header

You can run the following code to upload an object in the multipart upload method:

```

package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Creates an OSSClient instance.
    client, err := oss.New("<yourEndpoint>", "<yourAccessKeyId>", "<yourAccessKeySecret>")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    bucketName := "<yourBucketName>"
    objectName := "<yourObjectName>"
    locaFilename := "<yourLocalFilename>"
    // Uses the oss.GetResponseHeader method to obtain the returned header.
    var retHeader http.Header
    // Obtains the bucket.
    bucket, err := client.Bucket(bucketName)
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    chunks, err := oss.SplitFileByPartNum(locaFilename, 3)
    fd, err := os.Open(locaFilename)
    defer fd.Close()
    // Step 1: Initializes a multipart upload event.
    imur, err := bucket.InitiateMultipartUpload(objectName)
    // Step 2: Uploads the parts.
    var parts []oss.UploadPart
    for _, chunk := range chunks {
        fd.Seek(chunk.Offset, os.SEEK_SET)
        // Calls UploadPart to upload each part.
        part, err := bucket.UploadPart(imur, fd, chunk.Size, chunk.Number)
        if err != nil {
            fmt.Println("Error:", err)
            os.Exit(-1)
        }
        parts = append(parts, part)
    }
    // Step 3: Completes the multipart upload task.
    cmur, err := bucket.CompleteMultipartUpload(imur, parts, oss.GetResponseHeader(&retHeader))
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    fmt.Println("cmur:", cmur)
    // Prints x-oss-version-id.
    fmt.Println("x-oss-version-id:", oss.GetVersionId(retHeader))
}

```

9.9.3. Download objects

If you call `GetObject` to download an object in a bucket with versioning enabled or suspended, only the current version of the object is returned by default.

When you perform the `GetObject` operation on an object in a bucket:

- If the current version of the object is a delete marker, the 4040 Not Found error is returned.
- If you specify the `versionId` of the target object in the request, the specified version of the object is returned. For example, if the `versionId` is specified as null in the request, the version of which the `versionId` is null is returned.
- If the version specified in the request is a delete marker, the 405 Method Not Allowed error is returned.

You can run the following code to download an object:

```
package main
import (
    "fmt"
    "net/http"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Creates an OSSClient instance.
    client, err := oss.New("<yourEndpoint>", "<yourAccessKeyId>", "<yourAccessKeySecret>")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Obtains the bucket.
    bucket, err := client.Bucket("<yourBucketName>")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    var retHeader http.Header
    // Downloads the object with the version ID: yourObjectVersionId to the cache.
    _, err = bucket.GetObject("yourObjectName", oss.VersionId("yourObjectVersionId"), oss.GetResponseHeader(&retHeader))
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Prints x-oss-version-id.
    fmt.Println("x-oss-version-id:", oss.GetVersionId(retHeader))
}
```

9.9.4. Copy objects

This topic describes how to copy objects in a versioned bucket. You can use `CopyObject` to copy an object that is up to 1 GB in size and use `UploadPartCopy` to copy an object larger than 1 GB.

Copy a small object

You can use `CopyObject` to copy an object that is up to 1 GB in size from a source bucket to a destination bucket within the same region.

- By default, the `x-oss-copy-source` header in a `CopyObject` request specifies the current version of the object to copy. If the current version of the object to copy is a delete marker, 404 Not Found is returned to indicate that the object does not exist. You can add a version ID in the `x-oss-copy-source` header to copy a specified object version. Delete markers cannot be copied.
- You can copy a previous version of an object to the same bucket. The copied previous version becomes the current version of the object. This way, you can restore an object to a specified previous version.
- If versioning is enabled for the destination bucket, OSS generates a unique version ID for the destination object. The version ID is returned in the response as the `x-oss-version-id` header value. If versioning is disabled or suspended for the destination bucket, OSS generates a version whose version ID is null for the destination object and overwrites the original version whose version ID is null.
- Append objects in a bucket with versioning enabled or suspended cannot be copied.

The following code provides an example on how to copy a small object:

```
package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Create an OSSClient instance.
    client, err := oss.New("<yourEndpoint>", "<yourAccessKeyId>", "<yourAccessKeySecret>")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    bucketName := "<yourBucketName>"
    objectName := "<yourObjectName>"
    destObjectName := "<yourDestObjectName>"
    // Specify the name of the bucket.
    bucket, err := client.Bucket(bucketName)
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Copy the specified version of the object to another object in the same bucket and query the version ID of the destination object.
    var retHeader http.Header
    _, err = bucket.CopyObject(objectName, destObjectName, oss.VersionId("yourObjectVersionId"), oss.GetResponseHeader(&retHeader))
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Display the version ID of the destination object.
    fmt.Println("x-oss-version-id:", oss.GetVersionId(retHeader))
    fmt.Println("x-oss-copy-source-version-id:", oss.GetCopySrcVersionId(retHeader))
}
```

For more information about how to copy small objects, see [CopyObject](#).

Copy a large object

To copy an object larger than 1 GB, you must split the object into parts and copy them sequentially by using UploadPartCopy.

By default, the UploadPartCopy operation uploads a part by copying data from the current version of an existing object. You can add versionId in the request header x-oss-copy-source as the condition to upload a part by copying data from a specified version of an existing object. Example: x-oss-copy-source : /SourceBucketName/SourceObjectName?versionId=111111.

 **Note** The value of SourceObjectName must be URL-encoded. The version ID of the copied object version is returned as the x-oss-copy-source-version-id header value in the response.

If `versionId` is not specified in the request and the current version of the source object is a delete marker, OSS returns 404 Not Found. If `versionId` is specified in the request and the specified version of the source object is a delete marker, OSS returns 400 Bad Request.

The following code provides an example on how to use multipart copy to copy large objects:

```
package main
import (
    "fmt"
    "net/http"
    "os"
    "time"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Create an OSSClient instance.
    client, err := oss.New("<yourEndpoint>", "<yourAccessKeyId>", "<yourAccessKeySecret>")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Specify the name of the bucket.
    bucket, err := client.Bucket("yourBucketName")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    bucketName := "yourBucketName"
    objectName := "yourObjectName"
    fileName := "yourFileName"
    futureDate := time.Date(2049, time.January, 10, 23, 0, 0, 0, time.UTC)
    var retHeader http.Header
    chunks, err := oss.SplitFileByPartNum(fileName, 3)
    // Upload a large object as the source object.
    err = bucket.PutObjectFromFile(objectName, fileName, oss.GetResponseHeader(&retHeader))
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Query the version ID of the uploaded source object.
    versionId := oss.GetVersionId(retHeader)
    options := []oss.Option{
        oss.Expires(futureDate), oss.Meta("my", "myprop"),
    }
    // Specify the name of the destination object.
    objectDest := objectName + "dest"
    var parts []oss.UploadPart
    // Set parameters for object copy.
    copyOptions := []oss.Option{
        oss.VersionId(versionId), oss.GetResponseHeader(&retHeader),
    }
    // Copy the large object.
    imu, err := bucket.InitiateMultipartUpload(objectDest, options...)
    for _, chunk := range chunks {
        part, err := bucket.UploadPartCopy(imu, bucketName, objectName, chunk.Offset, chunk.Siz
```

```

e, (int)(chunk.Number), copyOptions...)
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    parts = append(parts, part)
    // Display the value of the x-oss-copy-source-version-id header.
    fmt.Println("x-oss-copy-source-version-id:", oss.GetCopySrcVersionId(retHeader))
}
_, err = bucket.CompleteMultipartUpload(imu, parts, oss.GetResponseHeader(&retHeader))
if err != nil {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
// Display the value of x-oss-version-id.
fmt.Println("x-oss-version-id:", oss.GetVersionId(retHeader))
}

```

For more information about multipart copy, see [UploadPartCopy](#).

9.9.5. List objects

You can list objects in a versioned bucket, including all objects, objects whose names contain a specified prefix, and objects and subfolders within a specified folder.

Scenarios

The following shows the structure of the objects and folders in a bucket named `examplebucket`.

```

examplebucket
├── fun
│   ├── exampleobject.jpg
│   ├── examplefile.txt
│   └── destfolder
│       ├── image1.jpg
│       └── image2.png
├── srcfile.txt
└── oss.jpg

```

The following sections describe how to list different objects by specifying conditions.

For more information about the parameters that you can configure when you list objects, see [GetBucketVersions\(ListObjectVersions\)](#).

List the versions of all objects in a bucket

The following code provides an example on how to list the versions of all objects including delete markers within a specified bucket:

```

package main
import (
    "fmt"
    "os"
    "github.com/alivun/alivun-oss-go-sdk/oss"

```

```

)
func HandleError(err error) {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region where the bucket is located. For example, if your bucket is in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com.
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log on to OSS because the account has permissions on all API operations. We recommend that you use your RAM user's credentials to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        HandleError(err)
    }
    // Specify the bucket name.
    bucketName := "examplebucket"
    bucket, err := client.Bucket(bucketName)
    if err != nil {
        HandleError(err)
    }
    // List the versions of all objects including delete markers within the bucket.
    keyMarker := oss.KeyMarker("")
    // The VersionIdMarker parameter is configured together with the KeyMarker parameter to specify the position from which the list operation starts.
    versionIdMarker := oss.VersionIdMarker("")
    for {
        lor, err := bucket.ListObjectVersions(keyMarker, versionIdMarker)
        if err != nil {
            HandleError(err)
        }
        // Display the information about the versions of returned objects.
        for _, dirName := range lor.ObjectVersions {
            fmt.Println("Versionid:", dirName.VersionId)
            fmt.Println("Key:", dirName.Key)
            fmt.Println("Is Latest", dirName.IsLatest)
        }
        // Display the version information of the listed delete markers.
        for _, marker := range lor.ObjectDeleteMarkers {
            fmt.Println(marker.VersionId)
            fmt.Println(marker.Key)
        }
        // Check whether the versions of all objects and delete markers are listed. If the versions of only part of objects are listed, the list operation continues. If the versions of all objects are completely listed, the list operation ends.
        keyMarker = oss.KeyMarker(lor.NextKeyMarker)
        versionIdMarker = oss.VersionIdMarker(lor.NextVersionIdMarker)
        if !lor.IsTruncated {
            break
        }
    }
}

```

```
}

```

For more information about endpoints, see [Regions and endpoints](#). For more information about the naming conventions for buckets, see [bucket](#).

Returned results

```
Versionid: CAEQChiBgIDHzNPETHciIDYzZGQ5M2VjOTU2ODRmMzA4ZWM4ODk0NjVmMWEx****
Key: fun/
Is Latest true
Versionid: CAEQChiBgID6ltnEthciIDNmOGM2MTQ3ZjU1NTQ2MGFhYWJjNjQxMTQxZWZh****
Key: fun/destfolder/
Is Latest true
Versionid: CAEQChiBgMDczt3EthciIDEwYjliZmQ4MDNiMDQ2Njk4YWMxM2NhM2E5NzQ3****
Key: fun/destfolder/image1.jpg
Is Latest true
Versionid: CAEQChiBgIDSzt3EthciIGU5OWU0ZTllMGY3NTRmMmU5NzVjZmJkYmE3ZWYy****
Key: fun/destfolder/image2.png
Is Latest true
Versionid: CAEQChiBgICditzEthciIDBiNTglZTZkMDRlMzRjNDdiMjRjMTBlOGUzMTM0****
Key: fun/examplefile.txt
Is Latest true
Versionid: CAEQChiBgMC.itzEthciIDEzNWVlYjNhYzNmMjQ4NWM5Nzc2NzllY2FiYmQ3****
Key: fun/exampleobject.jpg
Is Latest true
Versionid: CAEQChiBgIDMgdbEthciIDUyMGIOmZlNThkODQwY2ZhNmZhNTQ1Njk4ZTd****
Key: oss.jpg
Is Latest true
Versionid: CAEQChiBgICIgdbEthciIDdlM2Q1YjYxZDIyZDQyMzI4MTRkNzVmYzdiMTBh****
Key: srcfile.txt
Is Latest true
```

List the versions of objects whose names contain a specified prefix

The following code provides an example on how to list all versions of objects whose names contain a specified prefix:

```
package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func HandleError(err error) {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
func main() {
    // Create an OSSClient instance.
    client, err := oss.New("yourEndpoint","yourAccessKeyId","yourAccessKeySecret")
    if err != nil {
        HandleError(err)
    }
    // Specify the bucket name.
    bucketName := "examplebucket"
    bucket,err := client.Bucket(bucketName)
    if err != nil {
        HandleError(err)
    }
    // Specify the Prefix parameter to list the versions of objects whose names contain the
    prefix fun.
    prefix := oss.Prefix("fun")
    keyMarker := oss.KeyMarker("")
    versionIdMarker := oss.VersionIdMarker("")
    for {
        lor, err := bucket.ListObjectVersions(prefix,keyMarker,versionIdMarker)
        if err != nil {
            HandleError(err)
        }
        // Display the information about the versions of returned objects.
        for _, dirName := range lor.ObjectVersions{
            fmt.Println("Versionid:",dirName.VersionId)
            fmt.Println("Key:",dirName.Key)
            fmt.Println("Is Latest",dirName.IsLatest)
        }
        // Check whether the versions of objects whose names contain the specified prefix a
        re returned. If the versions of only part of objects whose names contain the specified pref
        ix are listed, the list operation continues. If the versions of objects whose names contain
        the specified prefix are completely listed, the list operation ends.
        keyMarker = oss.KeyMarker(lor.NextKeyMarker)
        versionIdMarker = oss.VersionIdMarker(lor.NextVersionIdMarker)
        if ! lor.IsTruncated {
            break
        }
    }
}
```

Returned results

```
Versionid: CAEQChiBgIDHzNPethciIDYzZGQ5M2VjOTU2ODRmMzA4ZWM4ODk0NjVmMWEx****
Key: fun/
Is Latest true
Versionid: CAEQChiBgID61tnEthciIDNmOGM2MTQ3ZjU1NTQ2MGFhYWJjNjQxMTQxZWZh****
Key: fun/destfolder/
Is Latest true
Versionid: CAEQChiBgMDczt3EthciIDewYjliZmQ4MDNiMDQ2Njk4YWMxM2NhM2E5NzQ3****
Key: fun/destfolder/image1.jpg
Is Latest true
Versionid: CAEQChiBgIDszt3EthciIGU5OWU0ZTllMGY3NTRmMmU5NzVjZmJkYmE3ZWYy****
Key: fun/destfolder/image2.png
Is Latest true
Versionid: CAEQChiBgICditzEthciIDBiNTglZTZkMDRlMzRjNDdiMjRjMTBLOGUzMTM0****
Key: fun/examplefile.txt
Is Latest true
Versionid: CAEQChiBgMC.itzEthciIDezNWVlYjNhYzNmMjQ4NWM5Nzc2NzllY2FiYmQ3****
Key: fun/exampleobject.jpg
Is Latest true
```

List a specified number of object versions

The following code provides an example on how to list a specified number of object versions:


```

package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func HandleError(err error) {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
func main() {
    // Create an OSSClient instance.
    client, err := oss.New("yourEndpoint","yourAccessKeyId","yourAccessKeySecret")
    if err != nil {
        HandleError(err)
    }
    // Specify the bucket name.
    bucketName := "examplebucket"
    bucket,err := client.Bucket(bucketName)
    if err != nil {
        HandleError(err)
    }
    // Specify the MaxKeys parameter to list up to four object versions in alphabetical order of the object names.
    maxkey := oss.MaxKeys(4)
    keyMarker := oss.KeyMarker("")
    versionIdMarker := oss.VersionIdMarker("")
    for {
        lor, err := bucket.ListObjectVersions(maxkey,keyMarker,versionIdMarker)
        if err != nil {
            fmt.Println("Error:", err)
            os.Exit(-1)
        }
        // Display the information about the versions of returned objects.
        for _, dirName := range lor.ObjectVersions{
            fmt.Println("Versionid:",dirName.VersionId)
            fmt.Println("Key:",dirName.Key)
            fmt.Println("Is Latest",dirName.IsLatest)
        }
        // Check whether the specified number of object versions are listed. If the specified number of object versions are not completely listed, the list operation continues. If the specified number of object versions are completely listed, the list operation ends.
        keyMarker = oss.KeyMarker(lor.NextKeyMarker)
        versionIdMarker = oss.VersionIdMarker(lor.NextVersionIdMarker)
        if ! lor.IsTruncated {
            break
        }
    }
}

```

Returned results

```
Versionid: CAEQChiBgIDHzNPethciIDYzZGQ5M2VjOTU2ODRmMzA4ZW40ODk0NjVmMWEx****
Key: fun/
Is Latest true
Versionid: CAEQChiBgID61tnEthciIDNmOGM2MTQ3ZjU1NTQ2MGFhYWJjNjQxMTQxZWZh****
Key: fun/destfolder/
Is Latest true
Versionid: CAEQChiBgMDczt3EthciIDewYjliZmQ4MDNiMDQ2Njk4YWMxM2NhM2E5NzQ3****
Key: fun/destfolder/image1.jpg
Is Latest true
Versionid: CAEQChiBgIDszt3EthciIGU5OWU0ZTllMGY3NTRmMmU5NzVjZmJkYmE3ZWYy****
Key: fun/destfolder/image2.png
Is Latest true
```

List the versions of all objects by page

The following code provides an example on how to list the versions of all objects by page:

```

package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func HandleError(err error) {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
func main() {
    // Create an OSSClient instance.
    client, err := oss.New("yourEndpoint","yourAccessKeyId","yourAccessKeySecret")
    if err != nil {
        HandleError(err)
    }
    // Specify the bucket name.
    bucketName := "examplebucket"
    bucket, err := client.Bucket(bucketName)
    if err != nil {
        HandleError(err)
    }
    // List the versions of all objects in the bucket by page.
    keyMarker := oss.KeyMarker("")
    // Specify the MaxKeys parameter to list up to four object versions in alphabetical order of the object names.
    maxkey := oss.MaxKeys(4)
    versionIdMarker := oss.VersionIdMarker("")
    for {
        lor, err := bucket.ListObjectVersions(maxkey, keyMarker, versionIdMarker)
        if err != nil {
            fmt.Println("Error:", err)
            os.Exit(-1)
        }
        // Display the information about the versions of returned objects.
        for _, dirName := range lor.ObjectVersions{
            fmt.Println("Versionid:", dirName.VersionId)
            fmt.Println("Key:", dirName.Key)
            fmt.Println("Is Latest", dirName.IsLatest)
        }
        fmt.Println("-----")
        // Check whether the specified number of object versions are listed. If the version
        // of only part of objects are listed, the list operation continues. If the versions of all
        // objects are completely listed, the list operation ends.
        if lor.IsTruncated {
            keyMarker = oss.KeyMarker(lor.NextKeyMarker)
            versionIdMarker = oss.VersionIdMarker(lor.NextVersionIdMarker)
        } else {
            break
        }
    }
}

```

Returned results

```

Versionid: CAEQChiBgIDHzNPethciIDYzZGQ5M2VjOTU2ODRmMzA4ZW40ODk0NjVmMWEx****
Key: fun/
Is Latest true
Versionid: CAEQChiBgID61tnEthciIDNmOGM2MTQ3ZjU1NTQ2MGFhYWJjNjQxMTQxZWZh****
Key: fun/destfolder/
Is Latest true
Versionid: CAEQChiBgMDczt3EthciIDewYjliZmQ4MDNiMDQ2Njk4YWMxM2NhM2E5NzQ3****
Key: fun/destfolder/image1.jpg
Is Latest true
Versionid: CAEQChiBgIDsz3EthciIGU5OWU0ZTllMGY3NTRmMmU5NzVjZmJkYmE3ZWYy****
Key: fun/destfolder/image2.png
Is Latest true
-----
Versionid: CAEQChiBgICditzEthciIDBiNTglZTZkMDRlMzRjNDdiMjRjMTBlOGUzMTM0****
Key: fun/examplefile.txt
Is Latest true
Versionid: CAEQChiBgMC.itzEthciIDezNWVlYjNhYzNmMjQ4NWM5Nzc2NzllY2FiYmQ3****
Key: fun/exampleobject.jpg
Is Latest true
Versionid: CAEQChiBgIDMgdbEthciIDUyMGI0NmZlNThkODQwY2ZhNmZhNTQ1Njk4ZTd****
Key: oss.jpg
Is Latest true
Versionid: CAEQChiBgICIgdbEthciIDdlM2Q1YjYxZDIyZDQyMzI4MTRkNzVmYzdiMTBh****
Key: srcfile.txt
Is Latest true
-----

```

List objects by folder

OSS does not use a hierarchical structure for objects, but instead uses a flat structure. All elements are stored as objects in buckets. A folder is an object 0 KB in size whose name ends with a forward slash (/). You can upload and download this object. By default, objects whose names end with a forward slash (/) are displayed as folders in the OSS console.

You can use the delimiter and prefix parameters to list objects by folder.

- If you set prefix to a folder name in the request, objects and subfolders whose names contain the prefix are listed.
- If you also set delimiter to a forward slash (/) in the request, the objects and subfolders whose names start with the specified prefix in the folder are listed. Each subfolder is listed as a single result element in commonPrefixes. The objects and folders in these subfolders are not listed.

The following examples describe how to list objects by simulating folders.

- List the versions of objects within the root folder of a bucket

The following code provides an example on how to list the versions of objects within the root folder of a bucket:

```

package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"

```

```
)
func HandleError(err error) {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
func main() {
    // Create an OSSClient instance.
    client, err := oss.New("yourEndpoint","yourAccessKeyId","yourAccessKeySecret")
    if err != nil {
        HandleError(err)
    }
    // Specify the bucket name.
    bucketName := "examplebucket"
    bucket, err := client.Bucket(bucketName)
    if err != nil {
        HandleError(err)
    }
    // Set the delimiter parameter to a forward slash (/) to list the versions of objects
    and the names of subfolders within the root folder.
    delimiter := oss.Delimiter("/")
    keyMarker := oss.KeyMarker("")
    versionIdMarker := oss.VersionIdMarker("")
    for {
        lor, err := bucket.ListObjectVersions(keyMarker,delimiter,versionIdMarker)
        if err != nil {
            HandleError(err)
        }
        // Display the information about the versions of returned objects.
        for _, dirName := range lor.ObjectVersions{
            fmt.Println("Versionid:",dirName.VersionId)
            fmt.Println("Key:",dirName.Key)
            fmt.Println("Is Latest",dirName.IsLatest)
        }
        // Display the folders whose names end with a forward slash (/).
        for _,common_prefix := range lor.CommonPrefixes{
            fmt.Println("common_prefix:",common_prefix)
        }
        // Check whether the versions of objects within the root folder of the bucket are
        completely listed. If the versions of only part of objects within the root folder of the
        bucket are listed, the list operation continues. If the versions of objects within the ro
        ot folder of the bucket are completely listed, the list operation ends.
        if lor.IsTruncated {
            keyMarker = oss.KeyMarker(lor.NextKeyMarker)
            versionIdMarker = oss.VersionIdMarker(lor.NextVersionIdMarker)
        }else{
            break
        }
    }
}
```

Returned results

```

Versionid: CAEQChiBgIDMgdbEthciIDUyMGI0NmZlNThkODQwY2ZhNmZhNTQ1Njk4ZTdj****
Key: oss.jpg
Is Latest true
Versionid: CAEQChiBgICIgdbEthciIDdlM2Q1YjYxZDIyZDQyMzI4MTRkNzVmYzdiMTBh****
Key: srcfile.txt
Is Latest true
common_prefix: fun/

```

- List objects and subfolders within a specified folder

The following code provides an example on how to list the objects and subfolders within a specified folder:

```

package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func HandleError(err error) {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
func main() {
    // Create an OSSClient instance.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        HandleError(err)
    }
    // Specify the bucket name.
    bucketName := "examplebucket"
    bucket, err := client.Bucket(bucketName)
    if err != nil {
        HandleError(err)
    }
    // Configure the Prefix parameter to list all objects and subfolders within the fun f
    older. Set the delimiter parameter to a forward slash (/).
    prefix := oss.Prefix("fun/")
    delimiter := oss.Delimiter("/")
    keyMarker := oss.KeyMarker("")
    versionIdMarker := oss.VersionIdMarker("")
    for {
        lor, err := bucket.ListObjectVersions(prefix, delimiter, keyMarker, versionIdMarker)
        if err != nil {
            HandleError(err)
        }
        // Display the information about the versions of returned objects.
        for _, dirName := range lor.ObjectVersions{
            fmt.Println("Versionid:", dirName.VersionId)
            fmt.Println("Key:", dirName.Key)
            fmt.Println("Is Latest:", dirName.IsLatest)
        }
        // Display the folders whose names end with a forward slash (/).
        for _, common_prefix := range lor.CommonPrefixes{
            fmt.Println("common_prefix:", common_prefix)
        }
    }
}

```

```

        }
        // Check whether all objects and subfolders within the specified folder are listed. If the objects and subfolders within the specified folder are not completely listed, the list operation continues. If the objects and subfolders within the specified folder are completely listed, the list operation ends.
        if lor.IsTruncated {
            keyMarker = oss.KeyMarker(lor.NextKeyMarker)
            versionIdMarker = oss.VersionIdMarker(lor.NextVersionIdMarker)
        } else {
            break
        }
    }
}

```

Returned results

```

Versionid: CAEQChiBgIDHzNPEthciIDYzZGQ5M2VjOTU2ODRmMzA4ZW40ODk0NjVmMWE****
Key: fun/
Is Latest: true
Versionid: CAEQChiBgICditzEthciIDBiNTg1ZTZkMDRlMzRjNDdiMjRjMTBLOGUzMtM0****
Key: fun/examplefile.txt
Is Latest: true
Versionid: CAEQChiBgMC.itzEthciIDEzNWVlYjNhYzNmMjQ4NWM5Nzc2NzllY2FiYmQ3****
Key: fun/exampleobject.jpg
Is Latest: true
common_prefix: fun/destfolder/

```

9.9.6. Delete objects

This topic describes how to delete a single object, multiple objects, or objects whose names contain a specified prefix from a versioned bucket.

Delete operations in a versioned bucket

When you delete an object from a versioned bucket, you must determine whether to specify a version ID in the request.

- Delete an object without specifying a version ID (temporary deletion)

By default, if you do not specify the version ID of the object that you want to delete in the request, Object Storage Service (OSS) does not delete the current version of the object but adds a delete marker to the object as the new current version. If you perform the `GetObject` operation on the object, OSS identifies the current version of the object as a delete marker and returns `404 Not Found`. Additionally, the `x-oss-delete-marker` header and the `x-oss-version-id` header that indicates the version ID of the created delete marker are included in the response to the `DeleteObject` request.

The value of the `x-oss-delete-marker` header in the response is `true`, which indicates that the `x-oss-version-id` value is the version ID of a delete marker.

- Delete an object by specifying a version ID (permanent deletion)

If you specify the version ID of the object that you want to delete in the request, OSS permanently deletes the version specified by the `versionId` field in the `params` parameter. To delete a version whose ID is null, add `params['versionId'] = "null"` to `params` in the request. OSS identifies the string "null" as the ID of the version to delete and deletes the version whose ID is null.

Delete a single object

The following examples describe how to permanently or temporarily delete a single object from a versioned bucket.

- Permanently delete an object by specifying its version ID

The following code provides an example on how to permanently delete an object with a specified version ID from a versioned bucket:

```
package main
import (
    "fmt"
    "net/http"
    "os"
    "strings"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Create an OSSClient instance.
    client, err := oss.New("<yourEndpoint>", "<yourAccessKeyId>", "<yourAccessKeySecret>")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Obtain the bucket.
    bucket, err := client.Bucket("<yourBucketName>")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Permanently delete an object by specifying the version ID of the object.
    err = bucket.DeleteObject(key, oss.VersionId("yourObjectVersionId"))
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
}
```

- Temporarily delete an object without specifying its version ID

The following code provides an example on how to temporarily delete an object from a versioned bucket:


```
package main
import (
    "fmt"
    "net/http"
    "os"
    "strings"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Create an OSSClient instance.
    client, err := oss.New("<yourEndpoint>", "<yourAccessKeyId>", "<yourAccessKeySecret>")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Obtain the bucket.
    bucket, err := client.Bucket("<yourBucketName>")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Temporarily delete an object without specifying its version ID. A delete marker is added to the object.
    err = bucket.DeleteObject("youObjectName", oss.GetResponseHeader(&retHeader))
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Display the information about the delete marker added to the object.
    fmt.Println("x-oss-version-id:", oss.GetVersionId(retHeader))
    fmt.Println("x-oss-delete-marker:", oss.GetDeleteMark(retHeader))
}
```

For more information about how to delete a single object, see [DeleteObject](#).

Delete multiple objects

The following examples describe how to permanently or temporarily delete multiple objects or delete markers from a versioned bucket.

- Permanently delete multiple objects by specifying their version IDs

The following code provides an example on how to permanently delete multiple objects with specified version IDs from a versioned bucket:

```
package main
import (
    "fmt"
    "net/http"
    "os"
    "strings"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Create an OSSClient instance.
    client, err := oss.New("<yourEndpoint>", "<yourAccessKeyId>", "<yourAccessKeySecret>")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Obtain the bucket.
    bucket, err := client.Bucket("yourBucketName")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Specify the version IDs of the objects you want to delete.
    keyArray := []oss.DeleteObject{
        oss.DeleteObject{Key: "objectName1", VersionId:"objectVersionId1"},
        oss.DeleteObject{Key: "objectName2", VersionId:"objectVersionId2"},
    }
    // Delete the object with the specified version ID.
    ret, err := bucket.DeleteObjectVersions(keyArray)
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Display the information about the deleted objects.
    for _, object := range ret.DeletedObjectsDetail {
        fmt.Println("Key:", object.Key, "\n VersionId:", object.VersionId)
    }
}
```

- Permanently delete multiple delete markers by specifying their version IDs

The following code provides an example on how to permanently delete multiple delete markers with specified version IDs from a versioned bucket:

```
package main
import (
    "fmt"
    "net/http"
    "os"
    "strings"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Create an OSSClient instance.
    client, err := oss.New("<yourEndpoint>", "<yourAccessKeyId>", "<yourAccessKeySecret>")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Obtain the bucket.
    bucket, err := client.Bucket("yourBucketName")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Specify the version IDs of the objects that you want to delete.
    keyArray := []oss.DeleteObject{
        oss.DeleteObject{Key: "objectName1", VersionId:"objectVersionId1"},
        oss.DeleteObject{Key: "objectName2", VersionId:"objectVersionId2"},
    }
    // Permanently delete the delete markers with the specified version IDs.
    ret, err := bucket.DeleteObjectVersions(keyArray)
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Display the deleted objects and delete markers.
    for _, object := range ret.DeletedObjectsDetail {
        fmt.Println("key:", object.Key, "\n versionId:", object.VersionId, "\n DeleteMarker:"
        , object.DeleteMarker, "\nDeleteMarkerVersionId", object.DeleteMarkerVersionId)
    }
}
```

- Temporarily delete multiple objects without specifying their version IDs

The following code provides an example on how to temporarily delete multiple objects from a versioned bucket:

```

package main
import (
    "fmt"
    "net/http"
    "os"
    "strings"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Create an OSSClient instance.
    client, err := oss.New("<yourEndpoint>", "<yourAccessKeyId>", "<yourAccessKeySecret>")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Obtain the bucket.
    bucket, err := client.Bucket("yourBucketName")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Temporarily delete multiple object without specifying their version IDs. Delete mark
    // ers are added to the objects.
    keyArray := []oss.DeleteObject{
        oss.DeleteObject{Key: "objectName1"},
        oss.DeleteObject{Key: "objectName2"},
    }
    res, err := bucket.DeleteObjectVersions(keyArray)
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Display the delete markers that are added to the objects.
    for _, object := range res.DeletedObjectsDetail {
        fmt.Println("key:", object.Key, "\n DeleteMarker:", object.DeleteMarker, "\nDeleteMar
        kerVersionId", object.DeleteMarkerVersionId)
    }
}

```

For more information about how to delete multiple objects, see [DeleteMultipleObjects](#).

Delete objects whose names contain a specified prefix

The following code provides an example on how to delete objects whose names contain a specified prefix:

```

package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Create an OSSClient instance.
    client, err := oss.New("<yourEndpoint>", "<yourAccessKeyId>", "<yourAccessKeySecret>")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Obtain the bucket.
    bucket, err := client.Bucket("yourBucketName")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Temporarily delete multiple object without specifying their version IDs. Delete mark
    // ers are added to the objects.
    keyArray := []oss.DeleteObject{
        oss.DeleteObject{Key: "objectName1"},
        oss.DeleteObject{Key: "objectName2"},
    }
    res, err := bucket.DeleteObjectVersions(keyArray)
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Display the delete markers that are added to the objects.
    for _, object := range res.DeletedObjectsDetail {
        fmt.Println("key:", object.Key, "\n DeleteMarker:", object.DeleteMarker, "\nDeleteMar
        kerVersionId", object.DeleteMarkerVersionId)
    }
}

```

```

client, err := oss.New("<yourEndpoint>", "<yourAccessKeyId>", "<yourAccessKeySecret>")
if err != nil {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
// Obtain the bucket.
bucket, err := client.Bucket("<yourBucketName>")
if err != nil {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
// List the objects whose name contain the specified prefix and delete these objects.
prefix := oss.Prefix("<yourObjectPrefix>")
marker := oss.KeyMarker("")
version := oss.VersionIdMarker("")
for {
    lor, err := bucket.ListObjectVersions(marker, prefix, version)
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    objects := []oss.DeleteObject{}
    for _, object := range lor.ObjectDeleteMarkers {
        objects = append(objects, oss.DeleteObject{
            Key:      object.Key,
            VersionId: object.VersionId,
        })
    }
    for _, object := range lor.ObjectVersions {
        objects = append(objects, oss.DeleteObject{
            Key:      object.Key,
            VersionId: object.VersionId,
        })
    }
    delRes, err := bucket.DeleteObjectVersions(objects, oss.DeleteObjectsQuiet(true))
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    if len(delRes.DeletedObjectsDetail) > 0 {
        fmt.Println("these objects deleted failure", delRes.DeletedObjectsDetail)
        os.Exit(-1)
    }
    prefix = oss.Prefix(lor.Prefix)
    marker = oss.KeyMarker(lor.NextKeyMarker)
    version = oss.VersionIdMarker(lor.NextVersionIdMarker)
    if ! lor.IsTruncated {
        break
    }
}
fmt.Printf("delete success\n")
}

```

9.9.7. Restore objects

In a bucket with versioning enabled or suspended, the storage class of each version of an object can be different. `RestoreObject` restores the current version of the target object by default. You can specify the `versionId` in the request to restore the specified version of the object.

You can run the following code to restore an object:

```
package main
import (
    "fmt"
    "os"
    "time"
    "strings"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func HandleError(err error) {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
func main() {
    // Creates an OSSClient instance.
    client, err := oss.New("<yourEndpoint>", "<yourAccessKeyId>", "<yourAccessKeySecret>")
    if err != nil {
        HandleError(err)
    }
    bucketName := "<yourBucketName>"
    objectName := "<yourObjectName>"
    localFilename := "<yourLocalFilename>"
    // Creates a bucket of the Archive storage class.
    err = client.CreateBucket(bucketName, oss.StorageClass(oss.StorageArchive))
    if err != nil {
        HandleError(err)
    }
    // Enables the versioning function of the bucket.
    config := oss.VersioningConfig{Status: "Enabled"}
    err = client.SetBucketVersioning(bucketName, config)
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Obtains the bucket.
    archiveBucket, err := client.Bucket(bucketName)
    if err != nil {
        HandleError(err)
    }
    var retHeader http.Header
    // Uploads a string. The oss.GetResponseHeader method is used to obtain the returned header.
    err = archiveBucket.PutObject("<yourObjectName>", strings.NewReader("yourObjectValue"), oss.GetResponseHeader(&retHeader))
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
}
```

```
}
// Obtains the versionId of the uploaded object to be restored.
versionId := oss.GetVersionId(retHeader)
// Checks whether the storage class of the object is Archive.
meta, err := archiveBucket.GetObjectDetailedMeta(objectName, oss.VersionId(versionId))
if err != nil {
    HandleError(err)
}
if meta.Get("X-Oss-Storage-Class") == string(oss.StorageArchive) {
    err = archiveBucket.RestoreObject(objectName, oss.VersionId(versionId))
    if err != nil {
        HandleError(err)
    }
    // Waits until the object is restored.
    meta, err = archiveBucket.GetObjectDetailedMeta(objectName, oss.VersionId(versionId))
    for meta.Get("X-Oss-Restore") == "ongoing-request=true" {
        fmt.Println("x-oss-restore:" + meta.Get("X-Oss-Restore"))
        time.Sleep(1000 * time.Second)
        meta, err = archiveBucket.GetObjectDetailedMeta(objectName, oss.VersionId(versionId))
    }
}
// Downloads the restored object.
err = archiveBucket.GetObjectToFile(objectName, localFilename, oss.VersionId(versionId))
if err != nil {
    HandleError(err)
}
}
```

For more information about restoring an object, see [RestoreObject](#).

9.9.8. Obtain object metadata

By default, when you call the `HeadObject` operation on an object in a versioned bucket, only the metadata of the current version of the object is returned. To return the metadata of a specified version of an object, you can specify the version ID in request parameters.

Sample code

The following code provides an example on how to obtain the metadata of a specified version of an object:

```
package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com. Specify your actual endpoint.
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS, because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Specify the full path of the object. The full path of the object cannot contain the bucket name. Example: exampleobject.txt.
    objectName := "exampleobject.txt"
    // Specify the name of the bucket in which the object is stored. Example: examplebucket.
    bucket, err := client.Bucket("examplebucket")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Query part of metadata of the specified version of the object.
    // Set youObjectVersionId to the version ID of the object. For more information about how to obtain the version ID of the object, see List the versions of all objects in a bucket.
    props, err := bucket.GetObjectMeta(objectName, oss.VersionId("youObjectVersionId"))
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    fmt.Println("Object Meta:", props)
    // Query all metadata of the specified version of the object.
    // Set youObjectVersionId to the version ID of the object.
    props, err = bucket.GetObjectDetailedMeta(objectName, oss.VersionId("youObjectVersionId"))
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    fmt.Println("Object Meta:", props)
}
```

References

- For more information about the complete sample code that is used to obtain object metadata, visit [GitHub](#).

- For more information about the API operation that is called to obtain object metadata, see [GetObjectMeta](#).

9.9.9. Manage object ACL

This topic describes how to manage the ACLs for objects in a bucket with versioning enabled or suspended.

Set the ACL for an object

PutObjectACL sets the ACL for the current version of the target object by default. If the current version of the target object is a delete marker, the 404 Not Found error is returned. You can set the versionId in the request to set the ACL for a specified version of the target object.

You can run the following code to set the ACL for an object:

```
package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Creates an OSSClient instance.
    client, err := oss.New("<yourEndpoint>", "<yourAccessKeyId>", "<yourAccessKeySecret>")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Obtains the bucket.
    bucket, err := client.Bucket("<yourBucketName>")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Sets the ACL for a specified version of the object.
    err = bucket.SetObjectACL("<yourObjectName>", oss.ACLPublicReadWrite, oss.VersionId("youObje
ctVersionId"))
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
}
```

For more information about setting the ACL for an object, see [PutObjectACL](#).

Obtain the ACL for an object

GetObjectACL obtains the ACL for the current version of the target object by default. If the current version of the object is a delete marker, the 404 Not Found error is returned. You can specify the versionId in the request to obtain the ACL for a specified version of the target object.

You can run the following code to obtain the ACL for an object:

```
package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Creates an OSSClient instance.
    client, err := oss.New("<yourEndpoint>", "<yourAccessKeyId>", "<yourAccessKeySecret>")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Obtains the bucket.
    bucket, err := client.Bucket("<yourBucketName>")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Obtains the ACL for the specified version of the object.
    aclRes, err := bucket.GetObjectACL("<yourObjectName>", oss.VersionId("youObjectVersionId"))
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    fmt.Println("Object ACL:", aclRes.ACL)
}
```

For more information about obtaining the ACL for an object, see [GetObjectACL](#).

9.9.10. Manage symbol links

This topic describes how to manage symbol links in a bucket with versioning enabled or suspended.

Create a symbol link

You can create a symbol link that directs to the current version of the target object.

 **Note** In a bucket with versioning enabled or suspended, you cannot create a symbol link for a delete marker.

A symbol link can has multiple versions that direct to different target objects. The version ID of the symbol link is automatically generated and is returned as the x-oss-version-id field in the response header.

You can run the following code to create a symbol link:

```
package main
import (
    "fmt"
    "os"
    "strings"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Creates an OSSClient instance.
    client, err := oss.New("<yourEndpoint>", "<yourAccessKeyId>", "<yourAccessKeySecret>")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    bucketName := "<yourBucketName>"
    // Specifies the target object.
    objectName := "<yourSymlinkName>"
    // Specifies the symbol link that directs to the target object.
    targetObjectName := "<yourSymlinkTargetName>"
    // Obtains the bucket.
    bucket, err := client.Bucket(bucketName)
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Uploads the object to the target object.
    err = bucket.PutObject(targetObjectName, strings.NewReader("target"))
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Creates a symbol link and obtains the version ID of the symbol link.
    var retHeader http.Header
    err = bucket.PutSymlink(objectName, targetObjectName, oss.GetResponseHeader(&retHeader))
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Prints x-oss-version-id.
    fmt.Println("x-oss-version-id:", oss.GetVersionId(retHeader))
}
```

For more information about creating a symbol link, see [PutSymlink](#).

Obtain a symbol link

GetSymlink obtains the current version of the target symbol link by default. You can specify the versionId in the request to obtain the specified version of a symbol link. If the current version of the target symbol link is a delete marker, OSS returns the 404 Not Found error and includes `x-oss-delete-marker = true` and `x-oss-version-id` in the response header.

 **Note** You must have the read permission on a symbol link to obtain it.

You can run the following code to obtain a symbol link:

```
package main
import (
    "fmt"
    "os"
    "strings"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Creates an OSSClient instance.
    client, err := oss.New("<yourEndpoint>", "<yourAccessKeyId>", "<yourAccessKeySecret>")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    bucketName := "<yourBucketName>"
    // Specifies the target object.
    objectName := "<yourSymlinkName>"
    // Obtains the bucket.
    bucket, err := client.Bucket(bucketName)
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Obtains the content of the object that the specified version of the symbol link direct
    s to.
    meta, err := bucket.GetSymlink(objectName, oss.VersionId("youObjectVersionId"))
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Prints the symbol link.
    fmt.Println(meta.Get(oss.HTTPHeaderOssSymlinkTarget))
}
```

For more information about obtaining a symbol link, see [GetSymlink](#).

9.10. Object tagging

9.10.1. Configure object tagging

Object Storage Service (OSS) allows you to configure object tags to classify objects. Tags allow you to configure lifecycle rules and access control lists (ACLs) for objects that have the same tag.

Background information

When you configure object tagging, take note of the following items:

- You can add tags to an object when and after you upload the object. If the object already has tags, the existing tags are overwritten. For more information about object tags, see [Put Object Tagging](#).
- To add tags to an object, you must have permissions to call the Put Object Tagging operation.

You can create a custom policy by using scripts to grant the permissions to RAM users. For more information, see [Attach a custom policy to a RAM user](#).

- The Last-Modified value of an object is not updated when the tags are changed.
- You can add up to 10 tags to an object. The tags added to an object must have unique keys.
- A tag key can be up to 128 characters in length. A tag value can be up to 256 characters in length.
- Tag keys and tag values are case-sensitive.
- The key and the value of a tag can contain letters, digits, spaces, and the following special characters:

+ - = . _ : /

 **Note** If you configure the tags in the HTTP header and the tags contain special characters, you can encode the keys and the values of the tags by using URL encoding.

Object tagging uses a key-value pair to identify objects. For more information about object tags, see [Object tagging](#).

Add tags to an object when you upload the object

- Add tags to an object when you upload the object by using simple upload

The following code provides an example on how to add tags to an object when you upload the object by using simple upload:

```
package main
import (
    "fmt"
    "os"
    "strings"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region in which the bucket is located. For
    // example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to htt
    ps://oss-cn-hangzhou.aliyuncs.com.
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account
    // to access OSS because the account has permissions on all API operations. We recommend tha
    t you use a RAM user to call API operations or perform routine O&M. To create a RAM user,
    log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Specify the name of the bucket. Example: examplebucket.
    bucketName := "examplebucket"
    // Specify the full path of the object. The full path of the object cannot contain th
    e bucket name. Example: exampledir/exampleobject.txt.
    objectName := "exampledir/exampleobject.txt"
    // Obtain the bucket.
    bucket, err := client.Bucket(bucketName)
    if err != nil {
```

```

        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Specify the key and the value of the object tag. For example, set the key to owner
    and the value to John.
    tag1 := oss.Tag{
        Key:    "owner",
        Value: "John",
    }
    tag2 := oss.Tag{
        Key:    "type",
        Value: "document",
    }
    tagging := oss.Tagging{
        Tags: []oss.Tag{tag1, tag2},
    }
    // Add tags to the object.
    err = bucket.PutObject(objectName, strings.NewReader("Hello OSS"), oss.SetTagging(tagging))
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    fmt.Println(bucket.GetObjectTagging(objectName))
}

```

- Add tags to an object when you upload the object by using multipart upload

The following code provides an example on how to add tags to an object when you upload the object by using multipart upload:

```

package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region in which the bucket is located. For
    example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to ht
    ps://oss-cn-hangzhou.aliyuncs.com.
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account
    to access OSS because the account has permissions on all API operations. We recommend tha
    t you use a RAM user to call API operations or perform routine O&M. To create a RAM user,
    log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Specify the name of the bucket. Example: examplebucket.
    bucketName := "examplebucket"
    // Specify the full path of the object. The full path of the object cannot contain th
    e bucket name. Example: exampledir/exampleobject.txt.

```

```
objectName := "exampledir/exampleobject.txt"
// Specify the full path of the local file to upload. Example: D:\\localpath\\example
file.txt.
// By default, if you specify only the name of the local file such as examplefile.txt
without specifying the local path, the local file is uploaded from the path of the projec
t to which the sample program belongs.
fileName := "D:\\localpath\\examplefile.txt"
// Obtain the bucket.
bucket, err := client.Bucket(bucketName)
if err != nil {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
// Specify the key and the value of the object tag. Example: the key is owner, and the
value is John.
tag1 := oss.Tag{
    Key:   "owner",
    Value: "John",
}
tag2 := oss.Tag{
    Key:   "type",
    Value: "document",
}
tagging := oss.Tagging{
    Tags: []oss.Tag{tag1, tag2},
}
// You can split an object into multiple parts for upload based on the object size. I
n this example, the object is split into three parts.
chunks, err := oss.SplitFileByPartNum(fileName, 3)
if err != nil {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
// Open the object.
fd, err := os.Open(fileName)
if err != nil {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
defer fd.Close()
// Initialize the multipart upload and configure object tagging.
imur, err := bucket.InitiateMultipartUpload(objectName, oss.SetTagging(tagging))
if err != nil {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
// Implement the multipart upload task.
var parts []oss.UploadPart
for _, chunk := range chunks {
    fd.Seek(chunk.Offset, os.SEEK_SET)
    part, err := bucket.UploadPart(imur, fd, chunk.Size, chunk.Number)
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
}
```

```

    }
    parts = append(parts, part)
}
_, err = bucket.CompleteMultipartUpload(imur, parts)
if err != nil {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
fmt.Println(bucket.GetObjectTagging(objectName))
}

```

- Add tags to an object when you upload the object by using append upload

The following code provides an example on how to add tags to an object when you upload the object by using append upload:

```

package main
import (
    "fmt"
    "os"
    "strings"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region in which the bucket is located. For
    // example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to htt
    ps://oss-cn-hangzhou.aliyuncs.com.
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account
    // to access OSS because the account has permissions on all API operations. We recommend tha
    t you use a RAM user to call API operations or perform routine O&M. To create a RAM user,
    log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Specify the name of the bucket. Example: examplebucket.
    bucketName := "examplebucket"
    // Specify the full path of the object. The full path of the object cannot contain th
    e bucket name. Example: exampledir/exampleobject.txt.
    objectName := "exampledir/exampleobject.txt"
    // Obtain the bucket.
    bucket, err := client.Bucket(bucketName)
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Specify the key and the value of the object tag. For example, set the key to owner
    and the value to John.
    tag1 := oss.Tag{
        Key:    "owner",
        Value:  "John",
    }
    tag2 := oss.Tag{

```



```

        Key:  "type",
        Value: "document",
    }
    tagging := oss.Tagging{
        Tags: []oss.Tag{tag1, tag2},
    }
    var nextPos int64
    // Upload the object by using append upload for the first time. When you call AppendObject to add tags to an object when you upload the object, the tags can be added to the object only when the object is uploaded for the first time.
    nextPos, err = bucket.AppendObject(objectName, strings.NewReader("Hello OSS A \n"), nextPos, oss.SetTagging(tagging))
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Upload the object by using append upload for the second time.
    nextPos, err = bucket.AppendObject(objectName, strings.NewReader("Hello OSS B \n"), nextPos)
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    fmt.Println(bucket.GetObjectTagging(objectName))
}

```

- Add tags to an object when you upload the object by using resumable upload

The following code provides an example on how to add tags to an object when you upload the object by using resumable upload:

```

package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com.
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Specify the name of the bucket. Example: examplebucket.
    bucketName := "examplebucket"
    // Specify the full path of the object. The full path of the object cannot contain the bucket name. Example: exampledir/exampleobject.txt.

```

```
objectName := "exampledir/exampleobject.txt"
// Specify the full path of the local file to upload. Example: D:\\localpath\\example
file.txt.
// By default, if you specify only the name of the local file such as examplefile.txt
without specifying the local path, the local file is uploaded from the path of the projec
t to which the sample program belongs.
fileName := "D:\\localpath\\examplefile.txt"
// Obtain the bucket.
bucket, err := client.Bucket(bucketName)
if err != nil {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
// Specify the key and the value of the object tag. For example, set the key to owner
and the value to John.
tag1 := oss.Tag{
    Key:   "owner",
    Value: "John",
}
tag2 := oss.Tag{
    Key:   "type",
    Value: "document",
}
tagging := oss.Tagging{
    Tags: []oss.Tag{tag1, tag2},
}
// Split the object into multiple parts, each of which is 100 KB in size. Then, use 3
coroutines to concurrently upload the parts, and add tags to the object when you upload t
he parts.
err = bucket.UploadFile(objectName, fileName, 100*1024, oss.Routines(3), oss.SetTaggi
ng(tagging))
if err != nil {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
fmt.Println(bucket.GetObjectTagging(objectName))
}
```

Add tags to an existing object or modify the tags of the object

If an existing object has no tags or the added tags of the object do not meet your requirements, you can add tags to or modify the tags of the existing object.

The following code provides an example on how to add tags to or modify the tags of an existing object:

```
package main
import (
    "fmt"
    "os"
    "strings"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com.
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Specify the name of the bucket. Example: examplebucket.
    bucketName := "examplebucket"
    // Specify the full path of the object. The full path of the object cannot contain the bucket name. Example: exampledir/exampleobject.txt.
    objectName := "exampledir/exampleobject.txt"
    // Obtain the bucket.
    bucket, err := client.Bucket(bucketName)
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Specify the key and the value of the object tag. For example, set the key to owner and the value to John.
    tag1 := oss.Tag{
        Key:   "owner",
        Value: "John",
    }
    tag2 := oss.Tag{
        Key:   "type",
        Value: "document",
    }
    tagging := oss.Tagging{
        Tags: []oss.Tag{tag1, tag2},
    }
    // Add tags to the object.
    err = bucket.PutObjectTagging(objectName, tagging)
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    fmt.Println(bucket.GetObjectTagging(objectName))
}
```

Add tags to or modify the tags of a specified version of an object

If versioning is enabled for the bucket, you can add tags to or modify the tags of a specific version of the object by specifying the version ID of the object.

The following code provides an example on how to add tags to or modify the tags of a specified version of the object.

 **Note** For more information about how to obtain the version IDs of an object, see [List objects](#).

```
package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func HandleError(err error) {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com.
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        HandleError(err)
    }
    // Specify the name of the bucket. Example: examplebucket.
    bucketName := "examplebucket"
    // Specify the full path of the object. The full path of the object cannot contain the bucket name. Example: exampledir/exampleobject.txt.
    objectName := "exampledir/exampleobject.txt"
    // Specify the version ID of the object.
    versionId := "CAEQMxiBgICAof2D0BYiIDJhMGE3N2MlYTI1NDQzOGY5NTkyNTI3MGYyMzJm****"
    // Obtain the bucket.
    bucket, err := client.Bucket(bucketName)
    if err != nil {
        HandleError(err)
    }
    // Specify the key and the value of the object tag. For example, set the key to owner and the value to John.
    tag1 := oss.Tag{
        Key:    "owner",
        Value:  "John",
    }
    tag2 := oss.Tag{
        Key:    "type",
        Value:  "document",
    }
```

```

    }
    tagging := oss.Tagging{
        Tags: []oss.Tag{tag1, tag2},
    }
    // Add tags to the specified version of the object.
    err = bucket.PutObjectTagging(objectName, tagging, oss.VersionId(versionId))
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    fmt.Println(bucket.GetObjectTagging(objectName))
}

```

Add tags to an object when you copy the object

- Add tags to an object when you copy the object by calling CopyObject

The following code provides an example on how to add tags to an object smaller than 1 GB when you copy it by calling CopyObject:

```

package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region in which the bucket is located. For
    // example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to htt
    ps://oss-cn-hangzhou.aliyuncs.com.
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account
    // to access OSS because the account has permissions on all API operations. We recommend tha
    t you use a RAM user to call API operations or perform routine O&M. To create a RAM user,
    log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Specify the name of the bucket. Example: examplebucket.
    bucketName := "examplebucket"
    // Specify the full path of the source object. The full path of the object cannot con
    tain the bucket name. Example: srcexampledir/exampleobject.txt.
    srcObjectName := "srcexampledir/exampleobject.txt"
    // Specify the full path of the destination object. The full path of the object cannot
    contain the bucket name. Example: destexampledir1/exampleobject.txt.
    destObjectName1 := "destexampledir1/exampleobject.txt"
    // Specify the full path of the destination object. The full path of the object cannot
    contain the bucket name. Example: destexampledir2/exampleobject.txt.
    destObjectName2 := "destexampledir2/exampleobject.txt"
    // Obtain the bucket.
    bucket, err := client.Bucket(bucketName)
    if err != nil {

```

```

        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Specify the key and the value of the object tag. For example, set the key to owner
    and the value to John.
    tag1 := oss.Tag{
        Key:   "owner",
        Value: "John",
    }
    tag2 := oss.Tag{
        Key:   "type",
        Value: "document",
    }
    tagging := oss.Tagging{
        Tags: []oss.Tag{tag1, tag2},
    }
    // If you configure only the tagging parameter, tags cannot be added to the destination
    object.
    _, err = bucket.CopyObject(srcObjectName, destObjectName1, oss.SetTagging(tagging))
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Tags can be added to the destination object only when you configure both the Tagging
    Replace and tagging parameters.
    _, err = bucket.CopyObject(srcObjectName, destObjectName2, oss.SetTagging(tagging), oss.TaggingDirective(oss.TaggingReplace))
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    fmt.Println(bucket.GetObjectTagging(objectName))
}

```

- Add tags to an object when you copy the object by calling UploadPartCopy

The following code provides an example on how to add tags to an object larger than 1 GB when you copy it by calling UploadPartCopy:

```

package main
import (
    "fmt"
    "os"
    "strings"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region in which the bucket is located. For
    example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to
    https://oss-cn-hangzhou.aliyuncs.com.
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account
    to access OSS because the account has permissions on all API operations. We recommend
    that you use a RAM user to call API operations or perform routine O&M. To create a
    RAM user, log on to the RAM console.

```

```

client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
if err != nil {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
// Specify the name of the bucket. Example: examplebucket.
bucketName := "examplebucket"
// Specify the full path of the object. The full path of the object cannot contain the
// bucket name. Example: srcexampledir/exampleobject.txt.
srcObjectName := "srcexampledir/exampleobject.txt"
// Specify the full path of the object. The full path of the object cannot contain the
// bucket name. Example: destexampledir/exampleobject.txt.
destObjectName := "destexampledir/exampleobject.txt"
// Obtain the bucket.
bucket, err := client.Bucket(bucketName)
if err != nil {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
// Specify the key and the value of the object tag. For example, set the key to owner
// and the value to John.
tag1 := oss.Tag{
    Key:    "owner",
    Value:  "John",
}
tag2 := oss.Tag{
    Key:    "type",
    Value:  "document",
}
tagging := oss.Tagging{
    Tags: []oss.Tag{tag1, tag2},
}
// Upload an object for calling UploadPartCopy.
content := "this your object value"
err = bucket.PutObject(objectName, strings.NewReader(content))
if err != nil {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
// Initialize the uploaded object and set tagging for the object.
imur, err := bucket.InitiateMultipartUpload(destObjectName, oss.SetTagging(tagging))
if err != nil {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
// Use the UploadPartCopy method to upload the object as one part.
part, err := bucket.UploadPartCopy(imur, bucketName, srcObjectName, 0, int64(len(content)), 1)
if err != nil {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
parts := []oss.UploadPart{part}
_, err = bucket.CompleteMultipartUpload(imur, parts)

```

```

    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    fmt.Println(bucket.GetObjectTagging(objectName))
}

```

Add tags to a symbolic link

The following code provides an example on how to add tags to a symbolic link:

```

package main
import (
    "fmt"
    "os"
    "strings"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com.
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Specify the name of the bucket. Example: examplebucket.
    bucketName := "examplebucket"
    // Specify the full path of the object. The full path of the object cannot contain the bucket name. Example: exampledir/exampleobject.txt.
    objectName := "exampledir/exampleobject.txt"
    // Specify the full path of the symbolic link object. Example: shortcut/myobject.txt.
    symlinkName := "shortcut/myobject.txt"
    // Obtain the bucket.
    bucket, err := client.Bucket(bucketName)
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Specify the key and the value of the object tag. For example, set the key to owner and the value to John.
    tag1 := oss.Tag{
        Key:   "owner",
        Value: "John",
    }
    tag2 := oss.Tag{
        Key:   "type",
        Value: "document",
    }
}

```



```
    },
    tagging := oss.Tagging{
        Tags: []oss.Tag{tag1, tag2},
    }
    err = bucket.PutObject(objectName, strings.NewReader("Hello OSS"))
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    err = bucket.PutSymlink(objectName, symlinkName, oss.SetTagging(tagging))
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    fmt.Println(bucket.GetObjectTagging(objectName))
}
```

9.10.2. Obtain the tags added to an object

This topic describes how to obtain the tags added to an object.

The object tagging function uses a key-value pair to tag an object. For more information about the object tagging function, see [Object tagging](#).

Run the following code to obtain the tags added to an object:

```
package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Creates an OSSClient instance.
    client, err := oss.New("<yourEndpoint>", "<yourAccessKeyId>", "<yourAccessKeySecret>")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Obtains the bucket that stores the object.
    bucket, err := client.Bucket("<yourBucketName>")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Obtains the tags added to the object.
    taggingResult, err := bucket.GetObjectTagging("<yourObjectName>")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    fmt.Printf("Object Tagging: %v\n", taggingResult)
}
```

For more information about how to obtain the tags added to an object, see [GetObjectTagging](#).

9.10.3. Delete the tags added to an object

This topic describes how to delete the tags added to an object.

The object tagging function uses a key-value pair to tag an object. For more information about the object tagging function, see [Object tagging](#).

Run the following code to delete the tags added to an object:

```
package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Creates an OSSClient instance.
    client, err := oss.New("<yourEndpoint>", "<yourAccessKeyId>", "<yourAccessKeySecret>")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Obtains the bucket that stores the object.
    bucket, err := client.Bucket("<yourBucketName>")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Deletes the tags added to the object.
    err = bucket.DeleteObjectTagging("<yourObjectName>")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
}
```

For more information about how to delete the tags added to an object, see [DeleteObjectTagging](#).

9.10.4. Object tagging and lifecycle rules

Lifecycle rules can take effect on objects with specified prefixes or tags. You can also specify prefixes and tags as the condition of a lifecycle rule at the same time.

 **Note** If you set a specified tag as the condition of a lifecycle rule, the rule applies to an object only when the key and value in the tag of the object both match the specified tag. If you specify a prefix and multiple tags as the condition of a lifecycle rule, the rule applies to an object only when the prefix and tags of the object match the prefix and all tags specified in the rule.

Set tags as the matching condition of a lifecycle rule

Run the following code to set tags as the matching condition of a lifecycle rule:

```
package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Creates an OSSClient instance.
    client, err := oss.New("<yourEndpoint>", "<yourAccessKeyId>", "<yourAccessKeySecret>")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Sets the lifecycle rule ID to "rule2" and applies it to objects prefixed with "one".
    // The storage class of Objects prefixed with "one" is converted to IA after 3 days and i
s converted to Archive after 30 days.
    Days: 3,
    StorageClass: oss.StorageIA,
}
transitionArch := oss.LifecycleTransition{
    Days: 30,
    StorageClass: oss.StorageArchive,
}
// Configures the tags used as the matching condition.
tag1 := oss.Tag{
    Key:    "key1",
    Value: "value1",
}
tag2 := oss.Tag{
    Key:    "key2",
    Value: "value2",
}
// Sets the bucket lifecycle rule, which includes the tags set as the matching condition.
rule1 := oss.LifecycleRule{
    ID: "rule1",
    Prefix: "one",
    Status: "Enabled",
    Transitions: []oss.LifecycleTransition{transitionIA, transitionArch},
    Tags: []oss.Tag{tag1, tag2},
}
rules := []oss.LifecycleRule{rule1}
err = client.SetBucketLifecycle("<yourBucketName>", rules)
if err != nil {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
}
```

View the tags set as the matching condition of a lifecycle rule

Run the following code to view the tags set as the matching condition of a lifecycle rule:

```
package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Creates an OSSClient instance.
    client, err := oss.New("<yourEndpoint>", "<yourAccessKeyId>", "<yourAccessKeySecret>")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Obtains lifecycle rule for the bucket that stores the object.
    lc, err := client.GetBucketLifecycle("<yourBucketName>")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Prints the lifecycle rule that includes the tags set as the matching condition.
    for i, v := range lc.Rules {
        fmt.Printf("Bucket %d Lifecycle: %v", i, v)
        if v.Expiration != nil {
            fmt.Printf(", Expiration:%v", *v.Expiration)
        }
        fmt.Printf("\n")
    }
}
```

9.11. Common operations on LiveChannels

This topic describes common operations on LiveChannels, such as create, list, and delete LiveChannels.

Create a LiveChannel

Before you can upload audio and video data by using the Real-Time Messaging Protocol (RTMP), you must call the PutLiveChannel operation to create a LiveChannel. The response to the PutLiveChannel request includes the URL that is used to ingest streams to the LiveChannel and the URL that is used to play the ingested streams.

 **Notice** You can use the returned URLs to ingest and play streams. You can also perform operations based on the returned LiveChannel name, such as query the stream ingestion status, query stream ingestion records, and disable stream ingestion.

The following code provides an example on how to create a LiveChannel:

```
package main
import (
    "fmt"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
```

```
    "os"
)
func HandleError(err error) {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com. Specify the endpoint based on your business requirements.
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access Object Storage Service (OSS) because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        HandleError(err)
    }
    // Specify the name of the bucket that is used to store the LiveChannel.
    bucketName := "srcexamplebucket"
    bucket, err := client.Bucket(bucketName)
    if err != nil {
        HandleError(err)
    }
    // Specify the name of the LiveChannel that you want to create. The name cannot contain forward slashes (/).
    channelName := "mychannel"
    // Specify the name of the generated m3u8 file if you set Type to HLS. The name must be 6 to 128 bytes in length and end with .m3u8.
    playlistName := "playlist.m3u8"
    // Specify the name of the bucket that stores the results of high-frequency snapshot operations.
    destBucketName := "destexamplebucket"
    // Specify the name of the MNS topic that is used to notify users of the results of high-frequency screenshot operations.
    notify := "exampletopic"
    target := oss.LiveChannelTarget{
        PlaylistName: playlistName,
        // Specify the format in which the LiveChannel stores the uploaded data. Only the HLS format is supported.
        Type:         "HLS",
        // Specify the number of TS files that are included in the m3u8 file if you set Type to HLS.
        FragCount:    3,
        // Specify the duration of each TS file. Unit: seconds.
        FragDuration: 5,
    }
    snapshot := oss.LiveChannelSnapshot{
        // Specify the name of the role that is used to perform high-frequency snapshot operations. The role must have write permissions on DestBucket and the permission to send messages to NotifyTopic.
        RoleName:     "examplerole",
        // Specify the interval of high-frequency snapshot operations. Unit: seconds. Valid
```

```
values: 1 to 100.
    Interval:    10,
    DestBucket:  destBucketName,
    NotifyTopic: notify,
}
config := oss.LiveChannelConfiguration{
    // Specify the description of the LiveChannel. The description can be up to 128 bytes in length.
    Description : "this is my channel",
    // Specify the status of the LiveChannel. In this example, the Status parameter is set to enabled, which indicates that the LiveChannel is enabled. If you want to disable the LiveChannel, set the Status parameter to disabled.
    Status:      "enabled",
    Target:      target,
    Snapshot: &snapshot,
}
// Create the LiveChannel.
data, err := bucket.CreateLiveChannel(channelName, config)
if err != nil {
    HandleError(err)
}
fmt.Println(data)
}
```

List LiveChannels

The following code provides an example on how to list the specified LiveChannels:

```
package main
import (
    "fmt"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
    "os"
)
func HandleError(err error) {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com. Specify the endpoint based on your business requirements.
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        HandleError(err)
    }
    // Specify the name of the bucket.
    bucketName := "yourBucketName"
    bucket, err := client.Bucket(bucketName)
    // Specify that the LiveChannels whose names contain the test prefix are listed.
    prefix := "test"
    if err != nil {
        HandleError(err)
    }
    // List the LiveChannels.
    result, err := bucket.ListLiveChannel(oss.Prefix(prefix))
    if err != nil {
        HandleError(err)
    }
    fmt.Println(result)
}
```

Configure the LiveChannel status

The following code provides an example on how to configure the status of a LiveChannel:

```

package main
import (
    "fmt"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
    "os"
)
func HandleError(err error) {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com. Specify the endpoint based on your business requirements.
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        HandleError(err)
    }
    // Specify the name of the bucket.
    bucketName := "yourBucketName"
    bucket, err := client.Bucket(bucketName)
    // Specify the name of the LiveChannel.
    channelName := "mychannel"
    if err != nil {
        HandleError(err)
    }
    // A LiveChannel can be in one of the following states: enabled or disabled.
    // If a LiveChannel is in the disabled state, OSS prohibits you from ingesting streams to the LiveChannel. If you are ingesting a stream to the LiveChannel that is in the disabled state, your client is disconnected from the LiveChannel after approximately 10 seconds.
    err = bucket.PutLiveChannelStatus(channelName, "disabled")
    if err != nil {
        HandleError(err)
    }
}

```

Query the stream ingestion status of a LiveChannel

The following code provides an example on how to query the stream ingestion status of the specified LiveChannel:


```
package main
import (
    "fmt"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
    "os"
)
func HandleError(err error) {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com. Specify the endpoint based on your business requirements.
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        HandleError(err)
    }
    // Specify the name of the bucket.
    bucketName := "yourBucketName"
    bucket, err := client.Bucket(bucketName)
    // Specify the name of the LiveChannel.
    channelName := "mychannel"
    if err != nil {
        HandleError(err)
    }
    // Query the stream ingestion status of the LiveChannel.
    result, err := bucket.GetLiveChannelStat(channelName)
    if err != nil {
        HandleError(err)
    }
    fmt.Println(result)
}
```

Query the configurations of a LiveChannel

The following code provides an example on how to query the configurations of the specified LiveChannel:

```
package main
import (
    "fmt"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
    "os"
)
func HandleError(err error) {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com. Specify the endpoint based on your business requirements.
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        HandleError(err)
    }
    // Specify the name of the bucket.
    bucketName := "yourBucketName"
    bucket, err := client.Bucket(bucketName)
    // Specify the name of the LiveChannel.
    channelName := "mychannel"
    if err != nil {
        HandleError(err)
    }
    // Query the configurations of the LiveChannel.
    result, err := bucket.GetLiveChannelInfo(channelName)
    if err != nil {
        HandleError(err)
    }
    fmt.Println(result)
}
```

Generate a playlist for a LiveChannel

You can call the `PostVodPlaylist` operation to generate a VOD playlist for the specified LiveChannel. OSS queries the TS files that are generated by the streams ingested to the specified LiveChannel within the specified time period and converges the files into an m3u8 playlist.

The following code provides an example on how to generate a playlist for the specified LiveChannel:

```

package main
import (
    "fmt"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
    "os"
)
func HandleError(err error) {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com. Specify the endpoint based on your business requirements.
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        HandleError(err)
    }
    // Specify the name of the bucket.
    bucketName := "yourBucketName"
    bucket, err := client.Bucket(bucketName)
    // Specify the name of the LiveChannel.
    channelName := "mychannel"
    playlistName := "playlist.m3u8"
    // Specify the end time of the time range during which the TS files that you want to query are generated. By default, the end time is the same as the current time.
    endTime := time.Now().Add(time.Minute)
    // Specify the start time of the time range during which the TS files that you want to query are generated. By default, the start time is 60 minutes earlier than the current time.
    .
    startTime := endTime.Add(-60 * time.Minute)
    if err != nil {
        HandleError(err)
    }
    // Generate a playlist for the LiveChannel.
    err = bucket.PostVodPlaylist(channelName, playlistName, startTime, endTime)
    if err != nil {
        HandleError(err)
    }
}

```

Query the playlist of a LiveChannel

The following code provides an example on how to query the playlist that is generated by the streams ingested to the specified LiveChannel within the specified time range:

```

package main
import (
    "fmt"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
    "os"
)
func HandleError(err error) {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com. Specify the endpoint based on your business requirements.
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        HandleError(err)
    }
    // Specify the name of the bucket.
    bucketName := "yourBucketName"
    bucket, err := client.Bucket(bucketName)
    // Specify the name of the LiveChannel.
    channelName := "mychannel"
    // Specify the end time of the time range during which the TS files that you want to query are generated. By default, the end time is the same as the current time.
    endTime := time.Now().Add(time.Minute)
    // Specify the start time of the time range during which the TS files that you want to query are generated. By default, the start time is 60 minutes earlier than the current time
    .
    startTime := endTime.Add(-60 * time.Minute)
    if err != nil {
        HandleError(err)
    }
    // Query the playlist of the LiveChannel.
    result, err := bucket.GetVodPlaylist(channelName, startTime, endTime)
    if err != nil {
        HandleError(err)
    }
    fmt.Println(result)
}

```

Query the stream ingestion records of a LiveChannel

You can call the `GetLiveChannelHistory` operation to query up to the 10 most recent stream ingestion records of the specified LiveChannel.

The following code provides an example on how to query the stream ingestion records of the specified LiveChannel:

```
package main
import (
    "fmt"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
    "os"
)
func HandleError(err error) {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com. Specify the endpoint based on your business requirements.
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        HandleError(err)
    }
    // Specify the name of the bucket.
    bucketName := "yourBucketName"
    bucket, err := client.Bucket(bucketName)
    // Specify the name of the LiveChannel.
    channelName := "mychannel"
    if err != nil {
        HandleError(err)
    }
    // Query the stream ingestion records of the LiveChannel.
    result, err := bucket.GetLiveChannelHistory(channelName)
    if err != nil {
        HandleError(err)
    }
    fmt.Println(result)
}
```

Delete a LiveChannel

Notice

- If you send a DeleteLiveChannel request to delete a LiveChannel to which a client is ingesting streams, the request fails.
- The DeleteLiveChannel operation deletes only the LiveChannel and does not delete the files that are generated by the streams ingested to the LiveChannel.

The following code provides an example on how to delete the specified LiveChannel:

```
package main
import (
    "fmt"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
    "os"
)
func HandleError(err error) {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com. Specify the endpoint based on your business requirements.
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        HandleError(err)
    }
    // Specify the name of the bucket.
    bucketName := "yourBucketName"
    bucket, err := client.Bucket(bucketName)
    // Specify the name of the LiveChannel.
    channelName := "mychannel"
    if err != nil {
        HandleError(err)
    }
    // Delete the LiveChannel.
    err := bucket.DeleteLiveChannel(channelName)
    if err != nil {
        HandleError(err)
    }
}
```

References

- For more information about the complete sample code of LiveChannels, visit [GitHub](#).
- For more information about the API operation that you can call to create a LiveChannel, see [PutLiveChannel](#).
- For more information about the API operation that you can call to list LiveChannels, see [ListLiveChannel](#).
- For more information about the API operation that you can call to configure the LiveChannel status, see [PutLiveChannelStatus](#).
- For more information about the API operation that you can call to query the stream ingestion status of a LiveChannel, see [GetLiveChannelStatus](#).
- For more information about the API operation that you can call to query the configurations of a LiveChannel, see [GetLiveChannelInfo](#).

- For more information about the API operation that you can call to generate a playlist for a LiveChannel, see [PostVodPlaylist](#).
- For more information about the API operation that you can call to query the playlist of a LiveChannel, see [GetVodPlaylist](#).
- For more information about the API operation that you can call to query the stream ingestion records of a LiveChannel, see [GetLiveChannelHistory](#).
- For more information about the API operation that you can call to delete a LiveChannel, see [DeleteLiveChannel](#).

9.12. Data encryption

9.12.1. Server-side encryption

Configure server-side encryption for a bucket

```
package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Create an OSSClient instance.
    client, err := oss.New("<yourEndpoint>", "<yourAccessKeyId>", "<yourAccessKeySecret>")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Initialize an encryption rule. In this example, AES-256 encryption is used.
    config := oss.ServerEncryptionRule{SSEDefault: oss.SSEDefaultRule{SSEAlgorithm: "AES256"}}
    err = client.SetBucketEncryption("yourBucketName", config)
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
}
```

Obtain the server-side encryption configurations of a bucket

```
package main
import (
    "fmt"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Create an OSSClient instance.
    client, err := oss.New("<yourEndpoint>", "<yourAccessKeyId>", "<yourAccessKeySecret>")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Obtain the encryption rule configured for the bucket.
    ret, err := client.GetBucketEncryption("yourBucketName")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Display the encryption rule.
    fmt.Println("Encrypt rule", ret.SSEDefault.SSEAlgorithm)
}
```

Delete the server-side encryption configurations of a bucket

```
package main
import (
    "fmt"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Create an OSSClient instance.
    client, err := oss.New("<yourEndpoint>", "<yourAccessKeyId>", "<yourAccessKeySecret>")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Delete the encryption rule configured for the bucket.
    err = client.DeleteBucketEncryption("yourBucketName")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
}
```

9.12.2. Client-side encryption

When you perform client-side encryption, objects are encrypted on the local client before they are uploaded to OSS.

Background information

 Notice

- Client-side encryption supports multipart upload for objects larger than 5 GB in size. When you use multipart upload to upload an object, you must specify the total size of the object and the size of each part. The size of each part except for the last part must be the same and be a multiple of 16 bytes.
- After you upload objects encrypted on the client, object metadata related to client-side encryption is protected and cannot be modified by calling [CopyObject](#).

Encryption methods

Object metadata related to client-side encryption

Parameter	Description	Required
x-oss-meta-client-side-encryption-key	The encrypted data key, which is a string encrypted by using the CMK and encoded in Base64.	Yes
x-oss-meta-client-side-encryption-start	The initial value which is generated randomly for data encryption. The initial value is a string encrypted by using the CMK and encoded in Base64.	Yes
x-oss-meta-client-side-encryption-cek-alg	The encryption algorithm used to encrypt data.	Yes
x-oss-meta-client-side-encryption-wrap-alg	The encryption algorithm used to encrypt data keys.	Yes
x-oss-meta-client-side-encryption-matdesc	The description of the CMK in JSON format.  Warning We recommend that you configure a description for each CMK and store the mapping relationship between the CMK and its description. CMKs of which the description is not specified cannot be replaced.	No
x-oss-meta-client-side-encryption-unencrypted-content-length	The length of data before encryption. This parameter is not generated if Content-Length is not specified.	No
x-oss-meta-client-side-encryption-unencrypted-content-md5	The MD5 hash of the data before encryption. This parameter is not generated if Content-MD5 is not specified.	No
x-oss-meta-client-side-encryption-data-size	The total size of the data that needs to be encrypted for multipart upload when <code>init_multipart</code> is called.	Yes (for multipart upload)

Parameter	Description	Required
x-oss-meta-client-side-encryption-part-size	The size of each part that needs to be encrypted for multipart upload when <code>init_multipart</code> is called.  Note The size of each part must be an integral multiple of 16 bytes.	Yes (for multipart upload)

Use RSA-based CMKs managed by yourself to encrypt objects to upload by using simple upload or decrypt objects to download by using simple download

The following code provides an example on how to use an RSA-based CMK managed by yourself to encrypt an object to upload by using simple upload or decrypt an object to download by using simple download:

```
package main
import (
    "bytes"
    "fmt"
    "io/ioutil"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
    "github.com/aliyun/aliyun-oss-go-sdk/oss/crypto"
)
func main() {
    // Create an OSSClient instance.
    client, err := oss.New("<yourEndpoint>", "<yourAccessKeyId>", "<yourAccessKeySecret>")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Specify a description for the CMK. The description cannot be modified after being specified. You can specify only one description for a CMK.
    // If all objects share a CMK, you can leave the description of the CMK empty. In this case, the CMK cannot be replaced.
    // If the description of the CMK is empty, the client cannot determine which CMK to use for encryption or decryption.
    // We recommend that you configure a description (a JSON string) for each CMK and store the mapping relationship between the CMK and description on the client. The server does not store the relationship.
    // Convert the CMK description (a JSON string) into a mapping relationship.
    materialDesc := make(map[string]string)
    materialDesc["desc"] = "<your master encrypt key material describe information>"
    // Create a CMK object based on the CMK description.
    masterRsaCipher, err := osscrypto.CreateMasterRsa(materialDesc, "<your rsa public key>", "<your rsa private key>")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Create an encryption operation based on the CMK object and perform encryption in AES C
```

```

// Create an encryption operation based on the CMK object and perform encryption in AES CTR mode.
contentProvider := osscrypto.CreateAesCtrCipher(masterRsaCipher)
// Obtain an existing bucket to be used in client-side encryption.
// The usage of a bucket used in client-side encryption is similar to that of a common bucket.
cryptoBucket, err := osscrypto.GetCryptoBucket(client, "<yourBucketName>", contentProvider)
if err != nil {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
// The system performs encryption when PutObject is called.
err = cryptoBucket.PutObject("<yourObjectName>", bytes.NewReader([]byte("yourObjectValueByteArray")))
if err != nil {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
// The system performs decryption when GetObject is called.
body, err := cryptoBucket.GetObject("<yourObjectName>")
if err != nil {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
defer body.Close()
data, err := ioutil.ReadAll(body)
if err != nil {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
fmt.Println("data:", string(data))
}

```

Use RSA-based CMKs managed by yourself to encrypt objects to upload by using multipart upload

The following code provides an example on how to use an RSA-based CMK managed by yourself to encrypt an object to upload by using multipart upload:

```

package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
    "github.com/aliyun/aliyun-oss-go-sdk/oss/crypto"
)
func main() {
    // Create an OssClient instance.
    client, err := oss.New("<yourEndpoint>", "<yourAccessKeyId>", "<yourAccessKeySecret>")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
}

```

```

    // Specify a description for the CMK. The description cannot be modified after being specified. You can specify only one description for a CMK.
    // If all objects share a CMK, you can leave the description of the CMK empty. In this case, the CMK cannot be replaced.
    // If the description of the CMK is empty, the client cannot determine which CMK to use for encryption or decryption.
    // We recommend that you configure a description (a JSON string) for each CMK and store the mapping relationship between the CMK and description on the client. The server does not store the relationship.
    // Convert the CMK description (a JSON string) into a mapping relationship.
    materialDesc := make(map[string]string)
    materialDesc["desc"] = "<your master encrypt key material describe information>"
    // Create a CMK object based on the CMK description.
    masterRsaCipher, err := osscrypto.CreateMasterRsa(materialDesc, "<your rsa public key>", "<your rsa private key>")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Create an encryption operation based on the CMK object and perform encryption in AES CTR mode.
    contentProvider := osscrypto.CreateAesCtrCipher(masterRsaCipher)
    // Obtain an existing bucket to be used in client-side encryption.
    // The usage of a bucket used in client-side encryption is similar to that of a common bucket.
    cryptoBucket, err := osscrypto.GetCryptoBucket(client, "<yourBucketName>", contentProvider)
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    fileName := "<yourLocalFilePath>"
    fileInfo, err := os.Stat(fileName)
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    fileSize := fileInfo.Size()
    // Encrypt the context.
    var cryptoContext osscrypto.PartCryptoContext
    cryptoContext.DataSize = fileSize
    // Specify the expected number of parts. The actual number is based on the calculation result.
    expectPartCount := int64(10)
    // Specify the size of a part encrypted in AES CTR mode, which is an integral multiple of 16 bytes.
    cryptoContext.PartSize = (fileSize / expectPartCount / 16) * 16
    imur, err := cryptoBucket.InitiateMultipartUpload("<yourObjectName>", &cryptoContext)
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    chunks, err := oss.SplitFileByPartSize(fileName, cryptoContext.PartSize)
    if err != nil {

```

```

    fmt.Println("Error:", err)
    os.Exit(-1)
}
var partsUpload []oss.UploadPart
for _, chunk := range chunks {
    part, err := cryptoBucket.UploadPartFromFile(imur, fileName, chunk.Offset, chunk.Size,
(int)(chunk.Number), cryptoContext)
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    partsUpload = append(partsUpload, part)
}
// Complete multipart upload.
_, err = cryptoBucket.CompleteMultipartUpload(imur, partsUpload)
if err != nil {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
}
}

```

Decrypt objects that are encrypted by using different RSA-based CMKs managed by yourself

The following code provides an example on how to decrypt an object that is encrypted by using a different RSA-based CMK managed by yourself:

```

package main
import (
    "bytes"
    "fmt"
    "io/ioutil"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
    "github.com/aliyun/aliyun-oss-go-sdk/oss/crypto"
)
// Query the CMK based on the CMK description. This API is required if you need to decrypt
an object that is encrypted by using a different master key.
type MockRsaManager struct {
}
func (mg *MockRsaManager) GetMasterKey(matDesc map[string]string) ([]string, error) {
    keyList := []string{"<yourRsaPublicKey", "yourRsaPrivatKey"}
    return keyList, nil
}
// Decrypt the object that is encrypted by using a different CMK.
func main() {
    // Create an OSSClient instance.
    client, err := oss.New("<yourEndpoint>", "<yourAccessKeyId>", "<yourAccessKeySecret>")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Specify a description for the CMK. The description cannot be modified after being spec
ified. You can specify only one description for a CMK

```

vided. You can specify only one description for a CMK.

// If all objects share a CMK, you can leave the description of the CMK empty. In this case, the CMK cannot be replaced.

// If the description of the CMK is empty, the client cannot determine which CMK to use for encryption or decryption.

// We recommend that you configure a description (a JSON string) for each CMK and store the mapping relationship between the CMK and description on the client. The server does not store the relationship.

// Convert the CMK description (a JSON string) into a mapping relationship.

```
materialDesc := make(map[string]string)
```

```
materialDesc["desc"] = "<your master encrypt key material describe information>"
```

// Create a CMK object based on the CMK description.

```
masterRsaCipher, err := osscrypto.CreateMasterRsa(materialDesc, "<your rsa public key>", "<your rsa private key>")
```

```
if err != nil {
```

```
    fmt.Println("Error:", err)
```

```
    os.Exit(-1)
```

```
}
```

// Create an encryption operation based on the CMK object and perform encryption in AES CTR mode.

```
contentProvider := osscrypto.CreateAesCtrCipher(masterRsaCipher)
```

// To decrypt objects encrypted by using different CMKs, you must provide this operation.

```
var mockRsaManager MockRsaManager
```

```
var options []osscrypto.CryptoBucketOption
```

```
options = append(options, osscrypto.SetMasterCipherManager(&mockRsaManager))
```

// Obtain an existing bucket to be used in client-side encryption.

// The usage of a bucket used in client-side encryption is similar to that of a common bucket.

```
cryptoBucket, err := osscrypto.GetCryptoBucket(client, "<yourBucketName>", contentProvider, options...)
```

```
if err != nil {
```

```
    fmt.Println("Error:", err)
```

```
    os.Exit(-1)
```

```
}
```

// The system performs encryption when PutObject is called.

```
err = cryptoBucket.PutObject("<yourObjectName>", bytes.NewReader([]byte("yourObjectValueByteArray")))
```

```
if err != nil {
```

```
    fmt.Println("Error:", err)
```

```
    os.Exit(-1)
```

```
}
```

// The system performs decryption when GetObject is called.

```
body, err := cryptoBucket.GetObject("<otherObjectNameEncryptedWithOtherRsa>")
```

```
if err != nil {
```

```
    fmt.Println("Error:", err)
```

```
    os.Exit(-1)
```

```
}
```

```
defer body.Close()
```

```
data, err := ioutil.ReadAll(body)
```

```
if err != nil {
```

```
    fmt.Println("Error:", err)
```

```
    os.Exit(-1)
```

```
}
```

```
fmt.Println("data:", string(data))
```

```
}
```

Use CMKs stored in KMS to encrypt objects to be uploaded by using simple upload or decrypt objects to be downloaded by using simple download

The following code provides an example on how to use a CMK stored in KMS to encrypt an object to be uploaded by using simple upload or decrypt an object to be downloaded by using simple download:

```
package main
import (
    "bytes"
    "fmt"
    "io/ioutil"
    "os"

    kms "github.com/aliyun/alibaba-cloud-sdk-go/services/kms"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
    "github.com/aliyun/aliyun-oss-go-sdk/oss/crypto"
)
func main() {
    // Create an OSSClient instance.
    client, err := oss.New("<yourEndpoint>", "<yourAccessKeyId>", "<yourAccessKeySecret>")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Create a KMSClient instance.
    kmsClient, err := kms.NewClientWithAccessKey("<yourKmsRegion>", "<yourKmsAccessKeyId>", "<yourKmsAccessKeySecret>")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Specify a description for the CMK. The description cannot be modified after being specified. You can specify only one description for a CMK.
    // If all objects share a CMK, you can leave the description of the CMK empty. In this case, the CMK cannot be replaced.
    // If the description of the CMK is empty, the client cannot determine which CMK to use for encryption.
    // We recommend that you configure a description (a JSON string) for each CMK and store the mapping relationship between the CMK and description on the client. The server does not store the relationship.
    // Convert the CMK description (a JSON string) into a mapping relationship.
    materialDesc := make(map[string]string)
    materialDesc["desc"] = "<your kms encrypt key material describe information>"
    // Create a CMK object based on the CMK description.
    masterkmsCipher, err := osscrypto.CreateMasterAliKms(materialDesc, "<YourKmsId>", kmsClient)
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Create an encryption operation based on the CMK object and perform encryption in AES CTR mode.
```

```

    contentProvider := osscrypto.CreateAesCtrCipher(masterkmsCipher)
    // Obtain an existing bucket to be used in client-side encryption.
    // The usage of a bucket used in client-side encryption is similar to that of a common bucket.
    cryptoBucket, err := osscrypto.GetCryptoBucket(client, "<yourBucketName>", contentProvider)
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // The system performs encryption when PutObject is called.
    err = cryptoBucket.PutObject("<yourObjectName>", bytes.NewReader([]byte("yourObjectValueByteArray")))
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // The system performs decryption when GetObject is called.
    body, err := cryptoBucket.GetObject("<yourObjectName>")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    defer body.Close()
    data, err := ioutil.ReadAll(body)
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    fmt.Println("data:", string(data))
}

```

9.13. Single-connection bandwidth throttling

This topic describes how to add parameters in an object upload and download request to set the limit of upload or download bandwidth. This way, you can ensure other applications have sufficient bandwidth.

Configure single-connection bandwidth throttling for simple upload and download

The following code provides an example on how to configure single-connection bandwidth throttling for simple upload and download:

```

package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)

```



```

func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com. Specify your actual endpoint.
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Specify the name of the bucket. Example: examplebucket.
    bucket, err := client.Bucket("examplebucket")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Specify the full path of the local file to upload. Example: D:\\localpath\\examplefile.txt.
    // By default, if the path of the local file is not specified, the local file is uploaded from the path of the project to which the sample program belongs.
    fd, err := os.Open("D:\\localpath\\examplefile.txt")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    defer fd.Close()
    // Set the upload bandwidth limit. The value of the parameter must be a number. The value must be expressed in bit/s. In this example, the upload bandwidth limit is set to 5 MB/s.
    var traffic int64 = 41943040
    // Configure single-connection bandwidth throttling for the object upload.
    // Specify the full path of the object. The full path of the object cannot contain the bucket name.
    err = bucket.PutObject("exampledir/exampleobject.txt", fd, oss.TrafficLimitHeader(traffic))
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Configure single-connection bandwidth throttling for the object download.
    // Specify the full path of the object. Example: exampledir/exampleobject.txt. Then, specify the full path of the local file. Example: D:\\localpath\\exampleobject.txt.
    err = bucket.GetObjectToFile("exampledir/exampleobject.txt", "D:\\localpath\\exampleobject.txt", oss.TrafficLimitHeader(traffic))
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Upload objects in append upload.
    // localFileOne and localFileTwo indicates the full paths of the two local files to upload. After localFileOne is uploaded, localFileTwo is uploaded in append upload.

```

```

localFileOne := "localFileOne"
localFileTwo := "localFileTwo"
fd1, err := os.Open(localFileOne)
if err != nil {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
defer fd.Close()
fd2, err := os.Open(localFileTwo)
if err != nil {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
defer fd.Close()
var nextPos int64
nextPos, err = bucket.AppendObject("<yourObjectName>", fd1, nextPos, oss.TrafficLimitHeader(traffic))
if err != nil {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
nextPos, err = bucket.AppendObject("exampledir/exampleobject.txt", fd2, nextPos, oss.TrafficLimitHeader(traffic))
if err != nil {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
// Upload a large local file in multipart upload.
// You can split the local file into multiple parts based on the object size.
chunks, err := oss.SplitFileByPartNum("localFile", 3)
if err != nil {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
// Open the object.
fd, err = os.Open("fileName")
if err != nil {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
defer fd.Close()
// Initialize the multipart upload and configure single-connection bandwidth throttling.
imur, err := bucket.InitiateMultipartUpload("exampledir/exampleobject.txt")
if err != nil {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
// Upload parts and set the bandwidth limit.
var parts []oss.UploadPart
for _, chunk := range chunks {
    fd.Seek(chunk.Offset, os.SEEK_SET)
    part, err := bucket.UploadPart(imur, fd, chunk.Size, chunk.Number, oss.TrafficLimitHeader(traffic))
    if err != nil {
        fmt.Println("Error:", err)
    }
}

```

```

fmt.Println("Error:", err)
os.Exit(-1)
}
parts = append(parts, part)
}
// Complete the multipart upload.
_, err = bucket.CompleteMultipartUpload(imur, parts)
if err != nil {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
}Part

```

Configure single-connection bandwidth throttling for object uploads and downloads that use signed URLs

The following code provides an example on how to configure single-connection bandwidth throttling when you use signed URLs to upload or download objects:

```

package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com. Specify your actual endpoint.
    // Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Specify the name of the bucket. Example: examplebucket.
    bucket, err := client.Bucket("examplebucket")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Use the signed URL to upload an object.
    // Specify the full path of the local file to upload. Example: D:\\localpath\\exampleobject.txt.
    fd, err := os.Open("D:\\localpath\\exampleobject.txt")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    defer fd.Close()

```

```

// Set the upload bandwidth limit. The value of the parameter must be a number. The value
must be expressed in bit/s. In this example, the upload bandwidth limit is set to 5 MB/s.

var traffic int64 = 41943040
// Obtain the signed URL used to upload the object.
// Specify the full path of the object. The full path of the object cannot contain the bu
cket name.
strURL, err := bucket.SignURL("exemplerdir/exampleobject.txt", oss.HTTPPut, 60, oss.Traffi
cLimitParam(traffic))
if err != nil {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
// Upload the local file.
err = bucket.PutObjectWithURL(strURL, fd)
if err != nil {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
// Generate a signed URL used to download an object.
// Obtain the signed URL used to download the object.
strURL, err = bucket.SignURL("exemplerdir/exampleobject.txt", oss.HTTPGet, 60, oss.Traffic
LimitParam(traffic))
if err != nil {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
// Specify the full path of the local file to which you want the object to be downloaded.

err = bucket.GetObjectToFileWithURL(strURL, "D:\\localpath\\exampleobject.txt")
if err != nil {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
}

```

9.14. Data security

Object Storage Service (OSS) SDK for Java uses MD5 verification and CRC-64 to ensure data integrity when you upload, download, and copy objects.

MD5 verification

If you configure Content-MD5 in an object upload request, OSS calculates the MD5 hash of the uploaded object. If the calculated MD5 hash is different from the MD5 hash configured in the upload request, `InvalidDigest` is returned. This allows OSS to ensure data integrity for object upload. If `InvalidDigest` is returned, you must upload the object again.

The following code provides an example on how to perform MD5 verification for object upload operations:

```
package main
import (
    "crypto/md5"
    "encoding/base64"
    "fmt"
    "os"
    "strings"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Create an OSSClient instance.
    client, err := oss.New("<yourEndpoint>", "<yourAccessKeyId>", "<yourAccessKeySecret>")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Specify the name of the bucket to which to upload the object.
    bucket, err := client.Bucket("<yourBucketName>")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Obtain the content of the object to upload.
    content := "<yourObjectValue>"
    // Calculate the MD5 hash of the object content.
    h := md5.New()
    h.Write([]byte(content))
    strMd5 := base64.StdEncoding.EncodeToString(h.Sum(nil))
    // Upload the object by using simple upload.
    err = bucket.PutObject("<yourObjectName>", strings.NewReader(content), oss.ContentMD5(strMd5))
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
}
```

 **Note** MD5 verification can be used for putObject, getObject, appendObject, postObject, and uploadPart.

CRC-64

By default, CRC-64 is enabled to ensure data integrity when you upload, download, and copy an object.

- CRC-64 can be used for putObject, getObject, appendObject, and uploadPart. By default, CRC-64 is enabled when you upload an object. If the CRC-64 value calculated on the client is different from the CRC-64 value returned by the OSS server, an error is returned.
- CRC-64 is not supported in range download.
- CRC-64 consumes CPU resources and slows down uploads and downloads.

The following code provides an example on how to perform CRC-64 when you upload an object by using append upload:

```

package main
import (
    "fmt"
    "os"
    "strings"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func main() {
    // Create an OSSClient instance.
    client, err := oss.New("<yourEndpoint>", "<yourAccessKeyId>", "<yourAccessKeySecret>")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Specify the name of the bucket to which to upload the object.
    bucket, err := client.Bucket("<yourBucketName>")
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // If the object is appended for the first time, the append position is 0. The returned
    value is the position for the next append. The position to start the next append is the tot
    al length of appended bytes and the append object.
    request := &oss.AppendObjectRequest{
        ObjectKey: "<yourObjectName>",
        Reader:     strings.NewReader("<YourObjectAppendValue1>"),
        Position:   0,
    }
    // If you append the object for the first time, set the CRC-64 value to 0.
    options := []oss.Option{oss.InitCRC(0)}
    result, err := bucket.DoAppendObject(request, options)
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // The position to start the next append is the total length included in the response f
    or the previous append request.
    request = &oss.AppendObjectRequest{
        ObjectKey: "<yourObjectName>",
        Reader:     strings.NewReader("<YourObjectAppendValue2>"),
        Position:   result.NextPosition,
    }
    // The CRC-64 value in the next append request is the value included in the response fo
    r the previous append request.
    options = []oss.Option{oss.InitCRC(result.CRC)}
    result, err = bucket.DoAppendObject(request, options)
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // You can perform multiple AppendObject operations.
}

```

9.15. Authorized access

This topic describes how to authorize temporary access to Object Storage Service (OSS) by using Security Token Service (STS) or a signed URL.

Use STS to authorize temporary access

You can use Alibaba Cloud STS to authorize temporary access to OSS. STS is a web service that provides temporary access tokens for users. You can use STS to grant a set of temporary access credentials that have a custom validity period and custom permissions to a third-party application or a RAM user managed by you. For more information about STS, see [What is STS?](#)

STS provides the following benefits:

- You need only to generate an access token and send the access token to a third-party application, instead of exposing your AccessKey pair to the third-party application. You can define the access permissions and validity period of this token.
- The token automatically expires after the validity period. Therefore, you do not need to manually revoke the access permissions of a token.

 **Note** For more information about how to configure STS, see [Use a temporary credential provided by STS to access OSS](#). You can call the [AssumeRole](#) operation or use [STS SDKs for various programming languages](#) to obtain temporary access credentials. The temporary access credentials consist of an AccessKey pair and a security token. The AccessKey pair consists of an AccessKey ID and an AccessKey secret.

- Use STS to grant a third-party user permissions to upload files

The following code provides an example on how to use temporary access credentials obtained from STS to grant a third-party user permissions to upload files:


```
package main
import (
    "fmt"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
    "os"
)
func main() {
    // Specify the security token that is obtained from STS.
    securitytoken := "yourSecurityToken"
    // Specify the temporary AccessKey pair that is obtained from STS.
    // After you obtain temporary access credentials from STS, you can use the security token and temporary AccessKey pair that are contained in the credentials to create an OSSClient instance.
    // Create an OSSClient instance.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret", oss.SecurityToken("yourSecurityToken"))
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Specify the bucket name. Example: examplebucket.
    bucketName := "examplebucket"
    // Specify the full path of the object. The full path cannot contain the bucket name. Example: exampledir/exampleobject.txt.
    objectName := "exampledir/exampleobject.txt"
    // Specify the full path of the local file to upload. Example: D:\\localpath\\examplefile.txt.
    filepath := "D:\\localpath\\examplefile.txt"
    bucket, err := client.Bucket(bucketName)
    // Use the temporary access credentials obtained from STS to grant the third-party user permissions to upload files.
    err = bucket.PutObjectFromFile(objectName, filepath)
    fmt.Println(err)
}
```

- Use STS to grant a third-party user permissions to download objects

The following code provides an example on how to use temporary access credentials obtained from STS to grant a third-party user permissions to download objects:

```
package main
import (
    "fmt"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
    "os"
)
func main() {
    // Specify the security token that is obtained from STS.
    securityToken := "yourSecurityToken"
    // Specify the temporary AccessKey pair that is obtained from STS.
    // After you obtain temporary access credentials from STS, you can use the security token and temporary AccessKey pair that are contained in the credentials to create an OSSClient instance.
    // Create an OSSClient instance.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret", oss.SecurityToken("yourSecurityToken"))
    if err != nil {
        fmt.Println("Error:", err)
        os.Exit(-1)
    }
    // Specify the bucket name. Example: examplebucket.
    bucketName := "examplebucket"
    // Specify the full path of the object. The full path cannot contain the bucket name. Example: exampledir/exampleobject.txt.
    objectName := "exampledir/exampleobject.txt"
    // Download the object to the specified path on your local computer. If the specified local file exists, the downloaded object overwrites the file. Otherwise, a file is created.
    // If you do not specify the path for the downloaded object, the downloaded object is saved to the path of the project to which the sample program belongs.
    filepath := "D:\\localpath\\examplefile.txt"
    // Use the temporary access credentials obtained from STS to grant the third-party user permissions to download objects.
    err = bucket.GetObjectToFile(objectName, filepath)
    fmt.Println(err)
}
```

Use a signed URL to authorize temporary access

This section provides examples on how to use a signed URL to authorize temporary access. For the complete code that is used to authorize temporary access by using a signed URL, visit [GitHub](#).

 **Notice** A validity period must be set for an STS temporary access credential and a signed URL. When you use an STS temporary account credential to generate a signed URL that is used to perform operations such as object upload and download, the minimum validity period takes precedence. For example, you can set the validity period of your STS temporary access credential to 1,200 seconds and the validity period of your signed URL generated by using this credential to 3,600 seconds. In this case, the signed URL cannot be used to upload objects after the STS temporary access credential expires, even if the signed URL is within its validity period.

- Generate a signed URL

You can generate a signed URL and provide the URL to a visitor for temporary access. When you generate a signed URL, you can specify the validity period of the URL to limit the period of access from visitors. By default, the validity period of a signed URL is 3,600 seconds. The maximum validity period of a signed URL is 32,400 seconds.

You can add signature information to a URL and provide the URL to a third-party user for authorized access. For more information, see [Add signatures to URLs](#).

- Use a signed URL to upload a file

The following code provides an example on how to use a signed URL to upload a local file named `examplefile.txt` from the local path `D:\\localpath` to the `exampledir` directory of a bucket named `examplebucket` and store the file as an object named `exampleobject.txt`.

 **Notice** To use a signed URL that contains custom parameters to access an object in a browser, make sure that the value of the `ContentType` parameter contained in the URL is the same as the content type specified in the browser.

```
package main
import (
    "fmt"
    "os"
    "strings"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func HandleError(err error) {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
func main() {
    // After you obtain temporary access credentials from STS, you can use the security token and temporary AccessKey pair that are contained in the credentials to create an OSS client instance.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret", oss.SecurityToken("yourSecurityToken"))
    if err != nil {
        HandleError(err)
    }
    // Specify the bucket name. Example: examplebucket.
    bucketName := "examplebucket"
    // Specify the full path of the object. Example: exampledir/exampleobject.txt. The full path cannot contain the bucket name.
    objectName := "exampledir/exampleobject.txt"
    // Specify the full path of the local file to upload. Example: D:\\localpath\\examplefile.txt.
    localFilename := "D:\\localpath\\examplefile.txt"
    bucket, err := client.Bucket(bucketName)
    if err != nil {
        HandleError(err)
    }
    // Generate a signed URL to upload the file.
    signedURL, err := bucket.SignURL(objectName, oss.HTTPPut, 60)
    if err != nil {
        HandleError(err)
    }
}
```

```

    }
    var val = "Go with Alibaba Cloud"
    err = bucket.PutObjectWithURL(signedURL, strings.NewReader(val))
    if err != nil {
        HandleError(err)
    }
    // Use the signed URL that contains custom parameters to upload the file. Make sure t
    hat the value of the ContentType parameter contained in the URL is the same as the conten
    t type specified in the browser.
    options := []oss.Option{
        oss.Meta("myprop", "mypropval"),
        oss.ContentType("text/plain"),
    }
    signedURL, err = bucket.SignURL(objectName, oss.HTTPPut, 60, options...)
    if err != nil {
        HandleError(err)
    }
    err = bucket.PutObjectFromFileWithURL(signedURL, localFilename, options...)
    if err != nil {
        HandleError(err)
    }
}

```

 **Note** For more information about the custom parameters that you can configure in a signed URL, see [Manage object metadata](#).

- Use a signed URL to download an object

The following code provides an example on how to use a signed URL to download an object named `exampleobject.txt` from the `exampledir` directory of a bucket named `examplebucket` to the local path `D:\localpath` and store the object as a file named `examplefile.txt`.

```

package main
import (
    "fmt"
    "os"
    "io/ioutil"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func HandleError(err error) {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
func main() {
    // After you obtain temporary access credentials from STS, you can use the security t
    oken and temporary AccessKey pair that are contained in the credentials to create an OSSC
    lient instance.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret", oss.
    SecurityToken("yourSecurityToken"))
    if err != nil {
        HandleError(err)
    }
    // Specify the bucket name. Example: examplebucket.

```

```
bucketName := "examplebucket"
// Specify the full path of the object. Example: exampledir/exampleobject.txt. The full path cannot contain the bucket name.
objectName := "exampledir/exampleobject.txt"
// Download the object to the specified path on your local computer. If the specified local file exists, the downloaded object overwrites the file. Otherwise, a file is created.
// If you do not specify the path for the downloaded object, the downloaded object is saved to the path of the project to which the sample program belongs.
localDownloadedFilename := "D:\\localpath\\examplefile.txt"
bucket, err := client.Bucket(bucketName)
if err != nil {
    HandleError(err)
}
// Generate a signed URL to download the object to a stream.
signedURL, err := bucket.SignURL(objectName, oss.HTTPGet, 60)
if err != nil {
    HandleError(err)
}
body, err := bucket.GetObjectWithURL(signedURL)
if err != nil {
    HandleError(err)
}
// Read the content of the object.
data, err := ioutil.ReadAll(body)
body.Close()
data = data // Use the downloaded data.
// Use the signed URL to download the object and store the object as a local file.
err = bucket.GetObjectToFileWithURL(signedURL, localDownloadedFilename)
if err != nil {
    HandleError(err)
}
}
```

9.16. IMG

Image Processing (IMG) provided by Object Storage Service (OSS) is a secure, cost-effective, and highly reliable image processing service that can process large numbers of images. After you upload source images to OSS, you can call RESTful API operations to process the images on a device that is connected over the Internet anytime, anywhere.

For more information about IMG parameters, see [Parameters](#).

Use IMG parameters to process images

- Use a single IMG parameter to process an image and save the image to your local computer

```

package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func HandleError(err error) {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
func main() {
    // Create an OSSClient instance.
    client, err := oss.New("<yourEndpoint>", "<yourAccessKeyId>", "<yourAccessKeySecret>")
    if err != nil {
        HandleError(err)
    }
    // Specify the bucket in which the source image is stored.
    bucketName := "<yourBucketName>"
    bucket, err := client.Bucket(bucketName)
    if err != nil {
        HandleError(err)
    }
    // Specify the name of the source image. If the image is not stored in the root directory of the bucket, you must specify the path of the image. Example: example/example.jpg.
    sourceImageName := "<yourObjectName>"
    // Specify the name of the processed image.
    targetImageName := "<LocalFileName>"
    // Resize the image to a height and width of 100 pixels and save the image to your local computer.
    style := "image/resize,m_fixed,w_100,h_100"
    err = bucket.GetObjectToFile(sourceImageName, targetImageName, oss.Process(style))
    if err != nil {
        HandleError(err)
    }
}

```

- Use multiple IMG parameters to process an image at the same time and save the image to your local computer

The following code provides an example on how to use multiple IMG parameters to process an image at the same time. Separate the parameters with forward slashes (/).

```
package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func HandleError(err error) {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region in which the bucket is located. For
    // example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to htt
    // ps://oss-cn-hangzhou.aliyuncs.com. Specify the actual endpoint.
    // The AccessKey pair of an Alibaba Cloud account has permissions on all API operatio
    // ns. Using these credentials to access OSS is a high-risk operation. We recommend that you
    // use a RAM user to call API operations or perform routine O&M. To create a RAM user, log o
    // n to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        HandleError(err)
    }
    // Specify the name of the bucket in which the source image is stored. Example: exampl
    // ebucket.
    bucketName := "examplebucket"
    bucket, err := client.Bucket(bucketName)
    if err != nil {
        HandleError(err)
    }
    // Specify the name of the source image. If the image is not stored in the root direc
    // tory of the bucket, you must specify the full path of the image. Example: exampledir/exam
    // ple.jpg.
    sourceImageName := "exampledir/example.jpg"
    // Specify the name of the processed image.
    targetImageName := "exampledir/newexample.jpg"
    // Resize the image to a height and width of 100 pixels, rotate the image 90 degrees,
    // and save the image to your local computer.
    style := "image/resize,m_fixed,w_100,h_100/rotate,90"
    err = bucket.GetObjectToFile(sourceImageName, targetImageName, oss.Process(style))
    if err != nil {
        HandleError(err)
    }
}
```

Use an image style to process an image

You can encapsulate multiple IMG parameters within a style, and then use the style to process images. For more information, see [image styles](#). The following code provides an example on how to use an image style to process an image:

```

package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func HandleError(err error) {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com. Specify the actual endpoint.
    // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to access OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        HandleError(err)
    }
    // Specify the name of the bucket in which the source image is stored. Example: examplebucket.
    bucketName := "examplebucket"
    bucket, err := client.Bucket(bucketName)
    if err != nil {
        HandleError(err)
    }
    // Specify the name of the source image. If the image is not stored in the root directory of the bucket, you must specify the full path of the image. Example: example/example.jpg.
    sourceImageName := "example/example.jpg"
    // Name the processed image newexample.jpg and save the image to your local computer.
    targetImageName := "D:\\localpath\\newexample.jpg"
    // Use the image style to process the image. In this example, set yourCustomStyleName to the name of the image style that you create in the OSS console.
    style := "style/yourCustomStyleName"
    // Save the processed image to your local computer.
    err = bucket.GetObjectToFile(sourceImageName, targetImageName, oss.Process(style))
    if err != nil {
        HandleError(err)
    }
}

```

Save processed images

You can call the `ImgSaveAs` operation to save the processed images to the specified bucket. The following code provides an example on how to save a processed image:

- Save the processed image to the current bucket

```
package main
```



```

import (
    "fmt"
    "os"
    "encoding/base64"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)

func HandleError(err error) {
    fmt.Println("Error:", err)
    os.Exit(-1)
}

func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region in which the bucket is located. For
    // example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to htt
    // ps://oss-cn-hangzhou.aliyuncs.com. Specify the actual endpoint.
    // The AccessKey pair of an Alibaba Cloud account has permissions on all API operatio
    // ns. Using these credentials to access OSS is a high-risk operation. We recommend that you
    // use a RAM user to call API operations or perform routine O&M. To create a RAM user, log o
    // n to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        HandleError(err)
    }
    // Specify the name of the bucket in which the source image is stored. Example: srcbu
    // cket.
    bucketName := "srcbucket"
    bucket, err := client.Bucket(bucketName)
    if err != nil {
        HandleError(err)
    }
    // Specify the name of the source image. If the image is not stored in the root direc
    // tory of the bucket, you must specify the full path of the image. Example: example/example
    // .jpg.
    sourceImageName := "example/example.jpg"
    // Specify the bucket that stores the processed image. The bucket must be within the
    // same region as the bucket in which the source image is stored.
    targetBucketName := "destbucket"
    // Specify the name of the processed image. If the image is not stored in the root di
    // rectory of the bucket, you must specify the full path of the image. Example: exampledir/e
    // xample.jpg.
    targetImageName = "exampledir/example.jpg"
    // Resize the image to a height and width of 100 pixels and save the image to the spe
    // cified bucket.
    style = "image/resize,m_fixed,w_100,h_100"
    process = fmt.Sprintf("%s|sys/saveas,o_%v,b_%v", style, base64.URLEncoding.EncodeToStr
    // ing([]byte(targetImageName)), base64.URLEncoding.EncodeToString([]byte(targetBucketName)
    // ))
    result, err = bucket.ProcessObject(sourceImageName, process)
    if err != nil {
        HandleError(err)
    } else {
        fmt.Println(result)
    }
}

```

- Save the processed image to the specified bucket

```
package main
import (
    "fmt"
    "os"
    "encoding/base64"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func HandleError(err error) {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
func main() {
    // Create an OSSClient instance.
    client, err := oss.New("<yourEndpoint>", "<yourAccessKeyId>", "<yourAccessKeySecret>")
    if err != nil {
        HandleError(err)
    }
    // Specify the bucket in which the source image is stored.
    bucketName := "<yourSourceBucketName>"
    bucket, err := client.Bucket(bucketName)
    if err != nil {
        HandleError(err)
    }
    // Specify the name of the source image. If the image is not stored in the root directory of the bucket, you must specify the path of the image. Example: example/example.jpg.
    sourceImageName := "<yourObjectName>"
    // Specify the bucket that stores the processed image. The bucket must be within the same region as the bucket in which the source image is stored.
    targetBucketName := "<TargetBucketName>"
    // Specify the name of the processed image.
    targetImageName = "<TargetObjectName>"
    // Resize the image to a height and width of 100 pixels and save the image to the specified bucket.
    style = "image/resize,m_fixed,w_100,h_100"
    process = fmt.Sprintf("%s|sys/saveas,o_%v,b_%v", style, base64.URLEncoding.EncodeToString([]byte(targetImageName)), base64.URLEncoding.EncodeToString([]byte(targetBucketName)))
    result, err = bucket.ProcessObject(sourceImageName, process)
    if err != nil {
        HandleError(err)
    } else {
        fmt.Println(result)
    }
}
```

Generate a signed URL that includes IMG parameters for an object

URLs of private objects must be signed. IMG parameters cannot be added to a signed URL. If you want to process a private object, add IMG parameters to the signature. The following code provides an example on how to add IMG parameters to the signature:

```
package main
import (
    "fmt"
    "os"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
func HandleError(err error) {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
func main() {
    // Create an OSSClient instance.
    // Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com. Specify the actual endpoint.
    // The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to access OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
    client, err := oss.New("yourEndpoint", "yourAccessKeyId", "yourAccessKeySecret")
    if err != nil {
        HandleError(err)
    }
    // Specify the name of the bucket in which the image is stored. Example: examplebucket.

    bucketName := "examplebucket"
    bucket, err := client.Bucket(bucketName)
    if err != nil {
        HandleError(err)
    }
    // Specify the name of the image. If the image is not stored in the root directory of the bucket, you must specify the full path of the image. Example: exampledir/example.jpg.
    ossImageName := "exampledir/example.jpg"
    // Generate a signed URL that includes IMG parameters. Set the validity period of the signed URL to 600 seconds.
    signedURL, err := bucket.SignURL(ossImageName, oss.HTTPGet, 600, oss.Process("image/format,png"))
    if err != nil {
        HandleError(err)
    } else {
        fmt.Println(signedURL)
    }
}
```

9.17. Error handling

This topic lists the causes and solutions for some common errors you may experience when you use the OSS Go SDK.

Client errors

Call errors in Go SDK are returned through a unified error interface. The error interface is defined as follows:

```
type error interface {  
    Error() string  
}
```

Other errors are also returned through this error interface. For example, HTTP request errors are returned as follows:

```
//net.Error  
type Error interface {  
    error  
    Timeout() bool    // Is the error a timeout  
    Temporary() bool  // Is the error temporary  
}
```

Server errors

If the request errors occur when OSS Go SDK is used, corresponding errors are returned. In case of HTTP request errors and I/O errors, errors preset in Go are returned. In case of request processing errors of the OSS Server, the following error is returned through the error interface:

```
type ServiceError struct {  
    Code      string    // Error code returned by OSS to a user  
    Message   string    // Error details provided by OSS  
    RequestId string    // UUID that uniquely identifies the request  
    HostId    string    // OSS cluster that identifies the access  
    StatusCode int       // HTTP status code  
}
```

If the HTTP status code returned by OSS is not as expected, the following error is returned, also through the error interface:

```
type UnexpectedStatusCodeError struct {  
    allowed []int    // Expected HTTP status code returned by OSS  
    got     int       // Actual HTTP status code returned by OSS  
}
```

Example of error handling

The following code is used to demonstrate error handling:

```
package main
import (
    "fmt"
    "os"
    "strings"
    "github.com/aliyun/aliyun-oss-go-sdk/oss"
)
// Error handling function
func HandleError(err error) {
    fmt.Println("Error:", err)
    os.Exit(-1)
}
func main() {
    // Create an OSSClient instance.
    client, err := oss.New("<yourEndpoint>", "<yourAccessKeyId>", "<yourAccessKeySecret>")
    if err != nil {
        HandleError(err)
    }
    bucketName := "<yourBucketName>"
    objectName := "<yourObjectName>"
    // Obtain the bucket.
    bucket, err := client.Bucket(bucketName)
    if err != nil {
        HandleError(err)
    }
    // Simple upload: string upload
    err = bucket.PutObject(objectName, strings.NewReader("yourObjectValueContentString"))
    if err != nil {
        HandleError(err)
    }
}
```

OSS error codes

The following table lists the OSS error codes.

Error code	Description	HTTP status code
AccessDenied	The access is denied.	403
BucketAlreadyExists	The bucket already exists.	409
BucketNotEmpty	The bucket is not empty.	409
EntityTooLarge	The entity is too large.	400
EntityTooSmall	The entity is too small.	400
FileGroupTooLarge	The object group is too large.	400
InvalidLinkName	The name of the object link is the same as the linked object.	400

Error code	Description	HTTP status code
LinkPartNotExist	The object to which the object link is linked does not exist.	400
ObjectLinkTooLarge	The object link has too many objects.	400
FieldItemTooLong	The table field in the post request is too large.	400
FilePartIntegrity	The object part has been changed.	400
FilePartNotExist	The object part does not exist.	400
FilePartStale	The object part has expired.	400
IncorrectNumberOfFilesInPOSTRequest	The number of objects in the post request is incorrect.	400
InvalidArgument	The parameter format is invalid.	400
InvalidAccessKeyId	The AccessKeyId does not exist.	403
InvalidBucketName	The bucket name is invalid.	400
InvalidDigest	The digest is invalid.	400
InvalidEncryptionAlgorithmError	The specified entropy encryption algorithm is invalid.	400
InvalidObjectName	The object name is invalid.	400
InvalidPart	The part is invalid.	400
InvalidPartOrder	The part order is invalid.	400
InvalidPolicyDocument	The policy document is invalid.	400
InvalidTargetBucketForLogging	A target bucket for logging is invalid.	400
InternalServerError	An error occurs in OSS.	500
MalformedXML	The XML format is invalid.	400
MalformedPOSTRequest	The format of the body in the post request is invalid.	400
MaxPOSTPreDataLengthExceededError	The body, except the object content, uploaded in the post request is too large.	400

Error code	Description	HTTP status code
MethodNotAllowed	The method is not supported.	405
MissingArgument	A parameter is missing.	411
MissingContentLength	The content length is missing.	411
NoSuchBucket	The bucket does not exist.	404
NoSuchKey	The object does not exist.	404
NoSuchUpload	The multipart upload ID does not exist.	404
NotImplemented	The method cannot be implemented.	501
PreconditionFailed	The preconditioning fails.	412
RequestTimeTooSkewed	Client local time and OSS server time differ by more than 15 minutes.	403
RequestTimeout	The request times out.	400
RequestIsNotMultiPartContent	The content type in the post request is invalid.	400
SignatureDoesNotMatch	The signature is incorrect.	403
TooManyBuckets	The number of the user's buckets exceeds the limit.	400
InvalidEncryptionAlgorithmError	The specified entropy encryption algorithm is invalid.	400

 **Note** In the preceding table, the Error code is `OssServiceError.Code`, and the HTTP status code is `OssServiceError.StatusCode`.

10.iOS

10.1. Preface

This topic is based on OSS iOS SDK 8.0.

Overview

This topic assumes that you have activated OSS and created an AccessKey ID and AccessKey Secret.

If you have not activated or do not know about OSS, log on to the [OSS Product Homepage](#) for more help.

Environment requirements

- iOS V8.0 and later
- You must have registered an Alibaba Cloud account and activated OSS.

SDK source code and API documents

For the source code of the SDK, see [GitHub](#). For more information, see [OSS iOS SDK](#).

Download the SDK

- [Download from GitHub](#)
- Pod dependency: pod 'AliyunOSSiOS'

10.2. Installation

This topic describes how to install the OSS SDK for iOS.

Environment requirements

- iOS V8.0 and later
- macOS version: 10.10 and later

Directly import the framework

For more information about how to obtain the OSS iOS SDK framework, see [GitHub](#).

In Xcode, drag the framework and drop it to your required section. In the dialog box that appears, select **Copy items if needed**.

Pod dependency

If your project uses Cocoapods to manage dependencies, add the following dependency to the Podfile. In this case, you do not need to import the OSS iOS SDK framework.

```
pod 'AliyunOSSiOS'
```

 **Note** You can choose one of the preceding methods.

Import the header file to your project

```
#import <AliyunOSSiOS/OSSService.h>
```

 **Notice** After you import the framework, add `-ObjC` to `Other Linker Flags` of `Build Settings` in your project. If the `-force_load` option is configured for your project, add `-force_load <framework path>/AliyunOSSiOS`.

Compatible with IPv6-only networks

Domain name resolution in wireless networks is subject to hijackings. To address this issue, OSS SDKs for mobile devices imports HTTPDNS for domain name resolution and directly uses IP addresses to request the OSS server. In an IPv6-only network, compatibility issues may occur. Apple has officially issued the review requirements for applications, which requires applications to be IPv6-only network compatible. To meet this requirement, the SDK starts to be compatible from V2.5.0. In the later version, aside from -ObjC settings, two system libraries must be imported:

```
libresolv.tbd  
CoreTelephony.framework  
SystemConfiguration.framework
```

About the ATS policy of Apple

During WWDC 2016, Apple announced that all the applications in Apple App Store must implement App Transport Security (ATS) from January 1, 2017. In other words, all newly submitted applications are not allowed to use `NSAllowsArbitraryLoads` to bypass the ATS limitation. Additionally, the application submitters must make sure that all network requests from the application are encrypted by using HTTPS. Otherwise, the application may fail the review.

The OSS iOS SDK V 2.6.0 and later provide the support for this ATS policy. The SDK sends only HTTPS requests. The SDK supports the prefix of `https://` in the `Endpoint` field value. You need only to set the correct HTTPS `Endpoint` field. In this way, all network requests can meet the requirements.

 **Notice**

- When you set the `Endpoint` field, you must use the URL that is prefixed with `https://`.
- Make sure that the application initiates only HTTPS requests when you implement callbacks such as signing and obtaining STS tokens.

10.3. Quick start

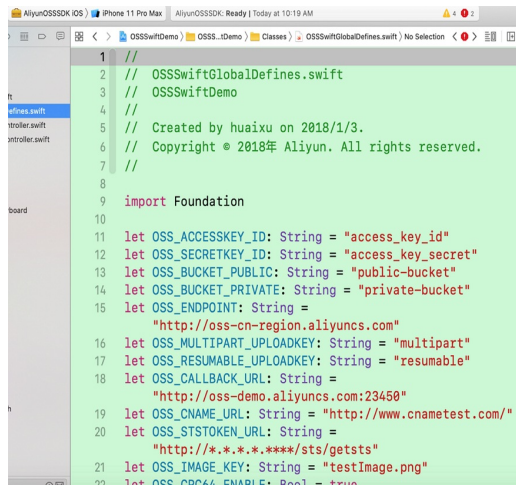
This topic describes how to use Object Storage Service (OSS) SDK for iOS to perform basic operations such as file upload and download.

Background information

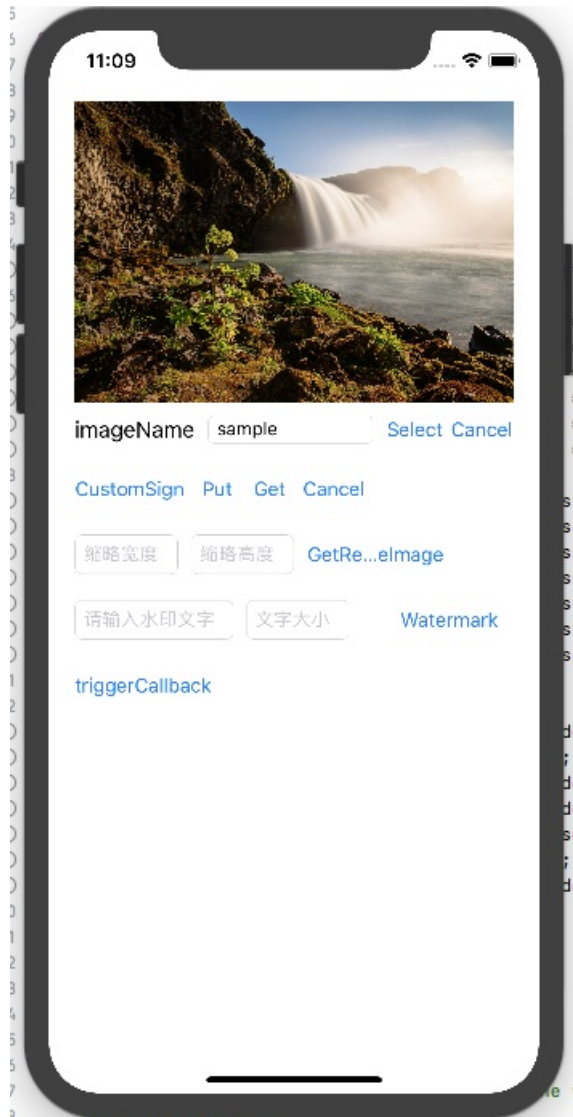
For more information about OSS SDK for iOS, see the following examples:

- [AliyunOSSSDK-iOS-Example](#)
- [AliyunOSSSDK-OSX-Example](#)
- [OSSSwiftDemo](#)
- [AliyunOSSiOSTests](#)

You can also clone the [project](#) by running the git clone command and configure necessary parameters such as OSS_ACCESSKEY_ID and OSS_SECRETKEY_ID.



Run the project demo as shown in the following figure.



Sample code

The following code shows how to upload and download objects on the iOS and macOS platforms:

1. Add the required reference

```
#import <AliyunOSSiOS/OSSService.h>
```

2. Initialize an OSSClient instance

To initialize an OSSClient instance, you must specify the endpoint, authentication mode, and client configurations. Three authentication modes are available: plaintext setting, self-signed, and security token service (STS) authentication. For more information about how to use the STS authentication mode, see [Authorized access](#).

Set the AccessKeyId, AccessKeySecret, and RoleArn parameters in the [sts.py](#) script file. Use OSS SDK for Python to start a local HTTP service. Use the client code to access the local service and obtain the StsToken.AccessKeyId, StsToken.AccessKeySecret, and StsToken.SecurityToken field values.

 Note

For more information about how to set up an STS, see [Use a temporary credential provided by STS to access OSS](#) in OSS Developer Guide. You can call the [AssumeRole](#) operation or use [STS SDKs for various programming languages](#) to obtain a temporary access credential. A temporary access credential contains a security token and a temporary AccessKey pair that consists of an AccessKey ID and an AccessKey secret.

```
NSString *endpoint = @"https://oss-cn-hangzhou.aliyuncs.com";
// We recommend that you use STS to initialize the OSSClient instance if you use a mobile device.
id<OSSCredentialProvider> credential = [[OSSFederationCredentialProvider alloc] initWithFederationTokenGetter:^OSSFederationToken * _Nullable{
    OSSFederationToken *token = [OSSFederationToken new];
    // Obtain a temporary AccessKey pair from STS.
    token.tAccessKey = @"AccessKeyId";
    token.tSecretKey = @"AccessKeySecret";
    // Obtain a security token from STS.
    token.tToken = @"SecurityToken";
    // Set the expiration time for the temporary access credential.
    token.expirationTimeInGMTFormat = @"Expiration";
    return token;
}];
client = [[OSSClient alloc] initWithEndpoint:endpoint credentialProvider:credential];
```

You can use the `OSSClient` instance to initiate multi-threaded upload and download requests to concurrently run multiple tasks.

3. Upload an object

If you have a bucket in the OSS console, an `OSSTask` is returned for each SDK-relevant operation. You can configure a continuation for `OSSTask` to be asynchronously completed or call `waitUntilFinished` to wait for the task to be completed.

```

OSSPutObjectRequest * put = [OSSPutObjectRequest new];
// Specify the name of the bucket. Example: examplebucket.
put.bucketName = @"examplebucket";
// Specify the full path of the object. The full path of the object cannot contain bucket names. Example: exampledir/testdir/exampleobject.txt.
put.objectKey = @"exampledir/testdir/exampleobject.txt";
// Directly upload NSData.
put.uploadingData = <NSData *>;
put.uploadProgress = ^(int64_t bytesSent, int64_t totalByteSent, int64_t totalBytesExpectedToSend) {
    NSLog(@"%lld, %lld, %lld", bytesSent, totalByteSent, totalBytesExpectedToSend);
};
OSSTask * putTask = [client putObject:put];
[putTask continueWithBlock:^id(OSSTask *task) {
    if (!task.error) {
        NSLog(@"upload object success!");
    } else {
        NSLog(@"upload object failed, error: %@", task.error);
    }
    return nil;
}];
// Wait until the task is completed.
// [putTask waitUntilFinished];

```

4. Download a specified object

The following code provides an example on how to download a specified object as `NSData` :

```

OSSGetObjectRequest * request = [OSSGetObjectRequest new];
// Specify the name of the bucket. Example: examplebucket.
request.bucketName = @"examplebucket";
// Specify the full path of the object. The full path of the object cannot contain bucket names. Example: exampledir/testdir/exampleobject.txt.
request.objectKey = @"exampledir/testdir/exampleobject.txt";
request.downloadProgress = ^(int64_t bytesWritten, int64_t totalBytesWritten, int64_t totalBytesExpectedToWrite) {
    NSLog(@"%lld, %lld, %lld", bytesWritten, totalBytesWritten, totalBytesExpectedToWrite);
};
OSSTask * getTask = [client getObject:request];
[getTask continueWithBlock:^id(OSSTask *task) {
    if (!task.error) {
        NSLog(@"download object success!");
        OSSGetObjectResult * getResult = task.result;
        NSLog(@"download result: %@", getResult.downloadedData);
    } else {
        NSLog(@"download object failed, error: %@", task.error);
    }
    return nil;
}];
// Wait until the task is completed.
// [task waitUntilFinished];

```

10.4. Initialization

The OSSClient is the iOS client of OSS. It provides the caller with a series of methods to operate and manage buckets and objects. Before you use OSS SDK for iOS to initiate a request to OSS, you need to first initialize an OSSClient instance and complete necessary settings.

Note

Ensure that the lifecycle of the OSSClient is consistent with that of the application. When an application is started, create an OSSClient instance. When the application is ended, release the instance.

Determine an endpoint

An endpoint is the URL of Alibaba Cloud OSS in a region. The following table describes the supported formats of OSS endpoints.

- OSS endpoints

When you use the endpoint of a bucket, you can query the endpoint in either of the following ways:

- For more information about how to query the relationship between the endpoint and region, see [Regions and endpoints](#).
- You can log on to the [OSS console](#). In the left-side navigation pane, locate the Buckets section, enter bucket-1 in the search box, and click the search icon. Click the bucket name to go to the Overview tab. In the Domain Names section, the suffix of `bucket-1.oss-cn-hangzhou.aliyuncs.com` is `oss-cn-hangzhou.aliyuncs.com`, which is the endpoint for Internet access.

- Custom domain name

Use CNAME to bind a custom domain name to a bucket. Use the custom domain name to access objects in the bucket.

For example, if you bind the `example.aliyundoc.com` domain name to the bucket named `examplebucket` in China (Hangzhou). Contact your DNS provider of `aliyundoc.com` to configure a new CNAME record. This record is used to resolve `https://example.aliyundoc.com` to `https://examplebucket.oss-cn-hangzhou.aliyuncs.com`. The record type is CNAME.

Configure an endpoint and credential

A mobile device is an untrusted environment. It is highly risky to store `AccessKey ID` and `AccessKey Secret` in a mobile device to sign requests. We recommend that you use STS Authentication Mode or Self-signed Mode. For more information, see [Authorized access](#) and [Set up direct data transfer for mobile apps](#).

 **Note** If the STS authentication mode is used, we recommend that you use `OSSAuthCredentialsProvider` to directly access STS. The token is automatically updated after it expires.

The following code provides an example on how to configure the Endpoint and CredentialProvider fields:

```
NSString *endpoint = @"https://oss-cn-hangzhou.aliyuncs.com";
// Use the AccessKey ID and AccessKey secret that are issued by Alibaba Cloud to create a credential provider.
// We recommend that you use OSSAuthCredentialProvider. The token is automatically updated after it expires.
id<OSSCredentialProvider> credential = [[OSSAuthCredentialProvider alloc] initWithAuthServerUrl:@"Application server address such as http://example.com:8080"];
client = [[OSSClient alloc] initWithEndpoint:endPoint credentialProvider:credential];
```

Set the Endpoint field to CNAME

If you bind CNAME to a bucket, you can set the Endpoint field to CNAME.

```
NSString *endpoint = @"https://example.aliyundoc.com";
// We recommend that you use OSSAuthCredentialProvider. The token is automatically updated after it expires.
id<OSSCredentialProvider> credential = [[OSSAuthCredentialProvider alloc] initWithAuthServerUrl:@"Application server address such as https://example.com:8080"];
client = [[OSSClient alloc] initWithEndpoint:endPoint credentialProvider:credential];
```

 **Notice** After Apple implemented the ATS standard, all `Endpoint` URLs must be HTTPS URLs. However, domain names set by `CNAME` do not support certificate settings. Therefore, you cannot use `CNAME` to set the `Endpoint` field.

Enable logging

Mobile devices are used in various environments. OSS SDKs cannot provide normal services in some regions or in a period of time. To further locate problems developers encounter, OSS SDKs record the log information locally after logging is enabled. Before you use the OSSClient, initialize the OSSClient instance and call the following method to enable logging:

```
// The log entry style.
//2017/10/25 11:05:43:863 [Debug]: the 17th: <NSThread: 0x7f8099108580>{number = 3, name = (null)}
//2017/10/25 11:05:43:863 [Debug]: the 15th:<NSThread: 0x7f80976052c0>
//2017/10/25 11:05:43:863 [Debug]: -----TestDebug-----
[OSSLog enableLog]; // Perform this method to enable logging.
```

Note

- Logs are stored in the Caches/OSSLogs folder of the sandbox.
- You can upload logs to the server to further track problems.

You can also access Alibaba Cloud [Log Service](#) to upload logs.

Configure ClientConfiguration

The following code provides an example on how to configure Client Configuration in detail during the initialization process:

```
NSString *endpoint = @"https://oss-cn-hangzhou.aliyuncs.com";
// We recommend that you use OSSAuthCredentialProvider. The token is automatically updated
after it expires.
id<OSSCredentialProvider> credential = [[OSSAuthCredentialProvider alloc] initWithAuthServe
rUrl:@"Application server address such as https://example.com:8080"];
OSSClientConfiguration * conf = [OSSClientConfiguration new];
conf.maxRetryCount = 3; // The number of retry attempts after the network request fails due
to exceptions.
conf.timeoutIntervalForRequest = 30; // The timeout period of a network request.
conf.timeoutIntervalForResource = 24 * 60 * 60; // The maximum timeout period for resource
transmission.
client = [[OSSClient alloc] initWithEndpoint:endPoint credentialProvider:credential clientC
onfiguration:conf];
```

OSSTask

An OSSTask is immediately returned for all OSS operations that are called. Example:

```
OSSTask * task = [client getObject:get];
```

To implement an asynchronous callback, you can configure a continuation for the task. Example:

```
[task continueWithBlock: ^(OSSTask *task) {
    // do something
    ...
    return nil;
}];
```

You can also wait until the task is complete (synchronous wait). Example:

```
[task waitUntilFinished];
```

10.5. Buckets

10.5.1. Create buckets

A bucket is a container used to store objects in Object Storage Service (OSS). Each object is contained in a bucket. This topic describes how to create a bucket.

Sample code

The following code provides an example on how to create a bucket named examplebucket:


```
id<OSSCredentialProvider> credentialProvider = [[OSSAuthCredentialProvider alloc] initWithAuthServerUrl:@"<StsServer>"];
OSSClientConfiguration *cfg = [[OSSClientConfiguration alloc] init];
cfg.maxRetryCount = 3;
cfg.timeoutIntervalForRequest = 15;
// If an OSSClient is released, all tasks in the session are canceled and the session becomes invalid. Make sure that the OSSClient is not released before the request is complete.
// Set Endpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set Endpoint to https://oss-cn-hangzhou.aliyuncs.com.
OSSClient *client = [[OSSClient alloc] initWithEndpoint:@"Endpoint" credentialProvider:credentialProvider clientConfiguration:cfg];
OSSCreateBucketRequest *create = [OSSCreateBucketRequest new];
// Set the bucket name to examplebucket.
create.bucketName = @"examplebucket";
// Set the access control list (ACL) of the bucket to private.
create.xOssACL = @"private";
// Set the storage class of the bucket to Infrequent Access (IA).
create.storageClass = OSSBucketStorageClassIA;
OSSTask *createTask = [client createBucket:create];
[createTask continueWithBlock:^id(OSSTask *task) {
    if (!task.error) {
        NSLog(@"create bucket success!");
    } else {
        NSLog(@"create bucket failed, error: %@", task.error);
    }
    return nil;
}];
```

References

- For more information about the complete sample code that is used to create a bucket, visit [GitHub](#).
- For more information about the API operation that you can call to create a bucket, see [PutBucket](#).

10.5.2. List buckets

A bucket is a container for objects stored in Object Storage Service (OSS). Every object is contained in a bucket. This topic describes how to list buckets that belong to an Alibaba Cloud account, including all buckets, buckets whose names contain a specific prefix, and a specific number of buckets.

List all buckets that belong to an Alibaba Cloud account

The following code provides an example on how to list all buckets that belong to an Alibaba Cloud account:

```

OSSGetServiceRequest * getService = [OSSGetServiceRequest new];
OSSTask * getServiceTask = [client getService:getService];
[getServiceTask continueWithBlock:^(id(OSSTask *task) {
    if (!task.error) {
        OSSGetServiceResult * result = task.result;
        NSLog(@"buckets: %@", result.buckets);
        NSLog(@"owner: %@", result.ownerId, result.ownerDispName);
        [result.buckets enumerateObjectsUsingBlock:^(id _Nonnull obj, NSUInteger idx, BOOL
* _Nonnull stop) {
            NSDictionary * bucketInfo = obj;
            NSLog(@"BucketName: %@", [bucketInfo objectForKey:@"Name"]);
            NSLog(@"CreationDate: %@", [bucketInfo objectForKey:@"CreationDate"]);
            NSLog(@"Location: %@", [bucketInfo objectForKey:@"Location"]);
        }];
    } else {
        NSLog(@"get service failed, error: %@", task.error);
    }
    return nil;
}]);

```

List buckets whose names contain a specific prefix

The following code provides an example on how to list buckets whose names contain the "bucket" prefix:

```

OSSGetServiceRequest * getService = [OSSGetServiceRequest new];
getService.prefix = @"bucket";
OSSTask * getServiceTask = [client getService:getService];
[getServiceTask continueWithBlock:^(id(OSSTask *task) {
    if (!task.error) {
        OSSGetServiceResult * result = task.result;
        NSLog(@"buckets: %@", result.buckets);
        NSLog(@"owner: %@", result.ownerId, result.ownerDispName);
        [result.buckets enumerateObjectsUsingBlock:^(id _Nonnull obj, NSUInteger idx, BOOL
* _Nonnull stop) {
            NSDictionary * bucketInfo = obj;
            NSLog(@"BucketName: %@", [bucketInfo objectForKey:@"Name"]);
            NSLog(@"CreationDate: %@", [bucketInfo objectForKey:@"CreationDate"]);
            NSLog(@"Location: %@", [bucketInfo objectForKey:@"Location"]);
        }];
    } else {
        NSLog(@"get service failed, error: %@", task.error);
    }
    return nil;
}]);

```

List buckets whose names are alphabetically after the bucket specified by marker

The marker parameter specifies a bucket name. Buckets whose names are alphabetically after the marker value are listed. The following code provides an example on how to list buckets whose names are alphabetically after examplebucket:

```

OSSGetServiceRequest * getService = [OSSGetServiceRequest new];
getService.marker = @"examplebucket";
OSSTask * getServiceTask = [client getService:getService];
[getServiceTask continueWithBlock:^(id(OSSTask *task) {
    if (!task.error) {
        OSSGetServiceResult * result = task.result;
        NSLog(@"buckets: %@", result.buckets);
        NSLog(@"owner: %@", result.ownerId, result.ownerDispName);
        [result.buckets enumerateObjectsUsingBlock:^(id _Nonnull obj, NSUInteger idx, BOOL
* _Nonnull stop) {
            NSDictionary * bucketInfo = obj;
            NSLog(@"BucketName: %@", [bucketInfo objectForKey:@"Name"]);
            NSLog(@"CreationDate: %@", [bucketInfo objectForKey:@"CreationDate"]);
            NSLog(@"Location: %@", [bucketInfo objectForKey:@"Location"]);
        }];
    } else {
        NSLog(@"get service failed, error: %@", task.error);
    }
    return nil;
}]);

```

List a specific number of buckets

The following code provides an example on how to list up to 20 buckets:

```

OSSGetServiceRequest * getService = [OSSGetServiceRequest new];
getService.maxKeys = 20;
OSSTask * getServiceTask = [client getService:getService];
[getServiceTask continueWithBlock:^(id(OSSTask *task) {
    if (!task.error) {
        OSSGetServiceResult * result = task.result;
        NSLog(@"buckets: %@", result.buckets);
        NSLog(@"owner: %@", result.ownerId, result.ownerDispName);
        [result.buckets enumerateObjectsUsingBlock:^(id _Nonnull obj, NSUInteger idx, BOOL
* _Nonnull stop) {
            NSDictionary * bucketInfo = obj;
            NSLog(@"BucketName: %@", [bucketInfo objectForKey:@"Name"]);
            NSLog(@"CreationDate: %@", [bucketInfo objectForKey:@"CreationDate"]);
            NSLog(@"Location: %@", [bucketInfo objectForKey:@"Location"]);
        }];
    } else {
        NSLog(@"get service failed, error: %@", task.error);
    }
    return nil;
}]);

```

10.5.3. Query bucket ACLs

A bucket is a container for objects stored in OSS. Every object is contained in a bucket. This topic describes how to query the ACL of a bucket.

Bucket ACL

The following table describes the ACLs of buckets.

ACL	Description	Value
Private	Only the owner or authorized users of this bucket can read and write objects in the bucket. Other users, including anonymous users cannot access the objects in the bucket without authorization.	private
Public read	Only the owner or authorized users of this bucket can write objects in the bucket. Other users, including anonymous users can only read objects in the bucket. We recommend that you exercise with caution when you set the bucket ACL to public read.	public-read
Public read/write	Any users, including anonymous users can read and write objects in the bucket. Exercise caution when set the bucket ACL to public read/write.	public-read-write

Examples

The following code provides an example on how to query the ACL of a bucket named `examplebucket`:

```
OSSGetBucketACLRequest *request = [OSSGetBucketACLRequest new];
request.bucketName = @"examplebucket";
OSSTask * getBucketACLTask = [client getBucketACL:request];
[getBucketACLTask continueWithBlock:^(id(OSSTask *task) {
    if (!task.error) {
        OSSGetBucketACLResult *result = task.result;
        NSLog(@"ACL%@", result.aclGranted);
    } else {
        NSLog(@"get bucket ACL failed, error: %@", task.error);
    }
    return nil;
}]);
```

10.5.4. Query bucket information

A bucket is a container for objects stored in OSS. Every object is contained in a bucket. This topic describes how to query the information about a bucket.

The following code provides an example on how to query the information about a bucket named `examplebucket`, including the region, creation date, and ACL of the bucket:

```
OSSGetBucketInfoRequest *request = [OSSGetBucketInfoRequest new];
request.bucketName = @"examplebucket";
OSSTask * getBucketInfoTask = [client getBucketInfo:request];
[getBucketInfoTask continueWithBlock:^id(OSSTask *task) {
    if (!task.error) {
        OSSGetBucketInfoResult *result = task.result;
        NSLog(@"Creation date:%@", result.creationDate);
        NSLog(@"Region:%@", result.location);
        NSLog(@"Storage class:%@", result.storageClass);
        NSLog(@"Owner:%@", result.owner.userName);
        NSLog(@"ACL:%@", result.acl.grant);
    } else {
        NSLog(@"get bucket info failed, error: %@", task.error);
    }
    return nil;
}];
```

10.5.5. Delete buckets

A bucket is a container for objects stored in OSS. Every object is contained in a bucket. This topic describes how to delete a bucket.

Note

The following code provides an example on how to delete a bucket named examplebucket:

```
OSSDeleteBucketRequest * delete = [OSSDeleteBucketRequest new];
delete.bucketName = @"examplebucket";
OSSTask * deleteTask = [client deleteBucket:delete];
[deleteTask continueWithBlock:^id(OSSTask *task) {
    if (!task.error) {
        NSLog(@"delete bucket success!");
    } else {
        NSLog(@"delete bucket failed, error: %@", task.error);
    }
    return nil;
}];
```

For more information about how to delete buckets, see [DeleteBucket](#).

10.6. Upload objects

10.6.1. Overview

This document describes the file upload methods provided by the OSS iOS SDK.

Objects are the basic data units for user operations in OSS. The OSS iOS SDK provides the following object upload methods:

- **Simple upload:** supports the upload of a file up to 5 GB in size. It includes local file upload and upload from the memory.

- **Multipart upload**: supports the upload of a file up to 48.8 TB in size. It applies to the upload of large files.
- **Append upload**: supports the upload of a file up to 5 GB in size.
- **Resumable upload**: supports the upload of a file up to 48.8 TB in size. It applies to the upload of large files. It supports concurrent upload and you can define the size of each part.

 **Note** For more information about the scenarios of each upload method, see the "Upload files" section in OSS Developer Guide.

After you complete the file upload, you can perform upload callback, as shown in [Upload callback](#).

10.6.2. Simple upload

This topic describes how to upload files from the memory or local disks. You can also perform MD5 verification to ensure data integrity in simple upload.

Upload a file from the memory or local disks

The following code provides an example on how to directly upload NSData or use an NSURL to upload a file:

```
OSSPutObjectRequest * put = [OSSPutObjectRequest new];
// Specify the bucket name. Example: examplebucket.
put.bucketName = @"examplebucket";
// Specify the full path of the object. Example: exampledir/exampleobject.txt. The full path of the object cannot contain the bucket name.
put.objectKey = @"exampledir/exampleobject.txt";
put.uploadingFileURL = [NSURL fileURLWithPath:@"<filePath>"];
// put.uploadingData = <NSData *>; // Directly upload NSData.
// (Optional) Specify the upload progress.
put.uploadProgress = ^(int64_t bytesSent, int64_t totalByteSent, int64_t totalBytesExpectedToSend) {
    // Specify the number of bytes that are being uploaded, the total number of bytes that are uploaded, and the total number of bytes to be uploaded.
    NSLog(@"%lld, %lld, %lld", bytesSent, totalByteSent, totalBytesExpectedToSend);
};
// Configure the optional fields.
// put.contentType = @"application/octet-stream";
// Specify Content-MD5.
// put.contentMd5 = @"eB5eJFlptWaXm4bijSPyxw==";
// Specify the method that is used to encode the object content.
// put.contentEncoding = @"identity";
// Specify the method that is used to display the object content.
// put.contentDisposition = @"attachment";
// Configure the object metadata or HTTP headers for the upload.
// NSMutableDictionary *meta = [NSMutableDictionary dictionary];
// Specify the object metadata.
// [meta setObject:@"value" forKey:@"x-oss-meta-name1"];
// Set the access control list (ACL) of the object to private.
// [meta setObject:@"private" forKey:@"x-oss-object-acl"];
// Set the storage class of the object to Standard.
// [meta setObject:@"Standard" forKey:@"x-oss-storage-class"];
// Specify to overwrite the existing object that has the same name as the file to upload.
```

```
// [meta setObject:@"true" forKey:@"x-oss-forbid-overwrite"];
// Specify tags for the destination object. You can specify multiple tags for the destination object at the same time.
// [meta setObject:@"a:1" forKey:@"x-oss-tagging"];
// Specify the server-side encryption algorithm that is used to encrypt the destination object when Object Storage Service (OSS) creates the object.
// [meta setObject:@"AES256" forKey:@"x-oss-server-side-encryption"];
// Specify the customer master key (CMK) that is managed by Key Management Service (KMS). This parameter takes effect only when x-oss-server-side-encryption is set to KMS.
// [meta setObject:@"9468da86-3509-4f8d-a61e-6eableac*****" forKey:@"x-oss-server-side-encryption-key-id"];
// put.objectMeta = meta;
OSSTask * putTask = [client putObject:put];
[putTask continueWithBlock:^id(OSSTask *task) {
    if (!task.error) {
        NSLog(@"upload object success!");
    } else {
        NSLog(@"upload object failed, error: %@", task.error);
    }
    return nil;
}];
// [putTask waitUntilFinished];
// [put cancel];
```

Upload a file to a directory

OSS uses a flat structure instead of a hierarchical structure to store objects. However, OSS allows you to emulate a folder hierarchy. If you want to create a directory in the OSS console, you can create an object whose name ends with a forward slash (/). Then, you can upload a file to the directory. The OSS console displays an object whose name ends with a forward slash (/) as a directory.

If you set objectKey to `folder/subfolder/file` when you upload a file, the file is uploaded to the `folder/subfolder/` directory.

 **Note** The default path is the root directory whose name does not start with a forward slash (/).

Configure contentType and enable MD5 verification during an upload

To ensure that the data received by the OSS server is the same as the data sent by a client, you can add the contentMd5 header to the request. The OSS server uses the value of this header to perform MD5 verification. You can explicitly specify the contentType field when you upload a file. If you do not explicitly specify contentType, the SDK automatically determines the content type based on the file name or the uploaded object name.

 **Notice** Take note that MD5 verification has a negative impact on OSS performance.

The SDK provides the method to obtain the Base64-encoded MD5 hash of the object content. The following code provides an example on how to obtain the Base64-encoded MD5 hash of the object content:

```

OSSPutObjectRequest * put = [OSSPutObjectRequest new];
// Specify the bucket name. Example: examplebucket.
put.bucketName = @"examplebucket";
// Specify the full path of the object. Example: exampledir/exampleobject.txt. The full path of the object cannot contain the bucket name.
put.objectKey = @"exampledir/exampleobject.txt";
put.uploadingFileURL = [NSURL fileURLWithPath:@"<filePath>"];
// Directly upload NSData.
// put.uploadingData = <NSData *>;
// (Optional) Configure the contentType field.
put.contentType = @"application/octet-stream";
// (Optional) Configure MD5 verification.
// Specify the MD5 hash of the file path.
put.contentMd5 = [OSSUtil base64Md5ForFilePath:@"<filePath>"];
// Specify the MD5 hash of the binary data.
// put.contentMd5 = [OSSUtil base64Md5ForData:<NSData *>];
// (Optional) Specify the upload progress.
put.uploadProgress = ^(int64_t bytesSent, int64_t totalByteSent, int64_t totalBytesExpectedToSend) {
    // Specify the number of bytes that are being uploaded, the total number of bytes that are uploaded, and the total number of bytes to be uploaded.
    NSLog(@"%lld, %lld, %lld", bytesSent, totalByteSent, totalBytesExpectedToSend);
};
OSSTask * putTask = [client putObject:put];
[putTask continueWithBlock:^id(OSSTask *task) {
    if (!task.error) {
        NSLog(@"upload object success!");
    } else {
        NSLog(@"upload object failed, error: %@", task.error);
    }
    return nil;
}];
// [putTask waitUntilFinished];
// [put cancel];

```

References

- For the complete sample code that is used to perform simple upload, visit [GitHub](#).
- For more information about the API operation that you can call to perform simple upload, see [Put Object](#).

10.6.3. Multipart upload

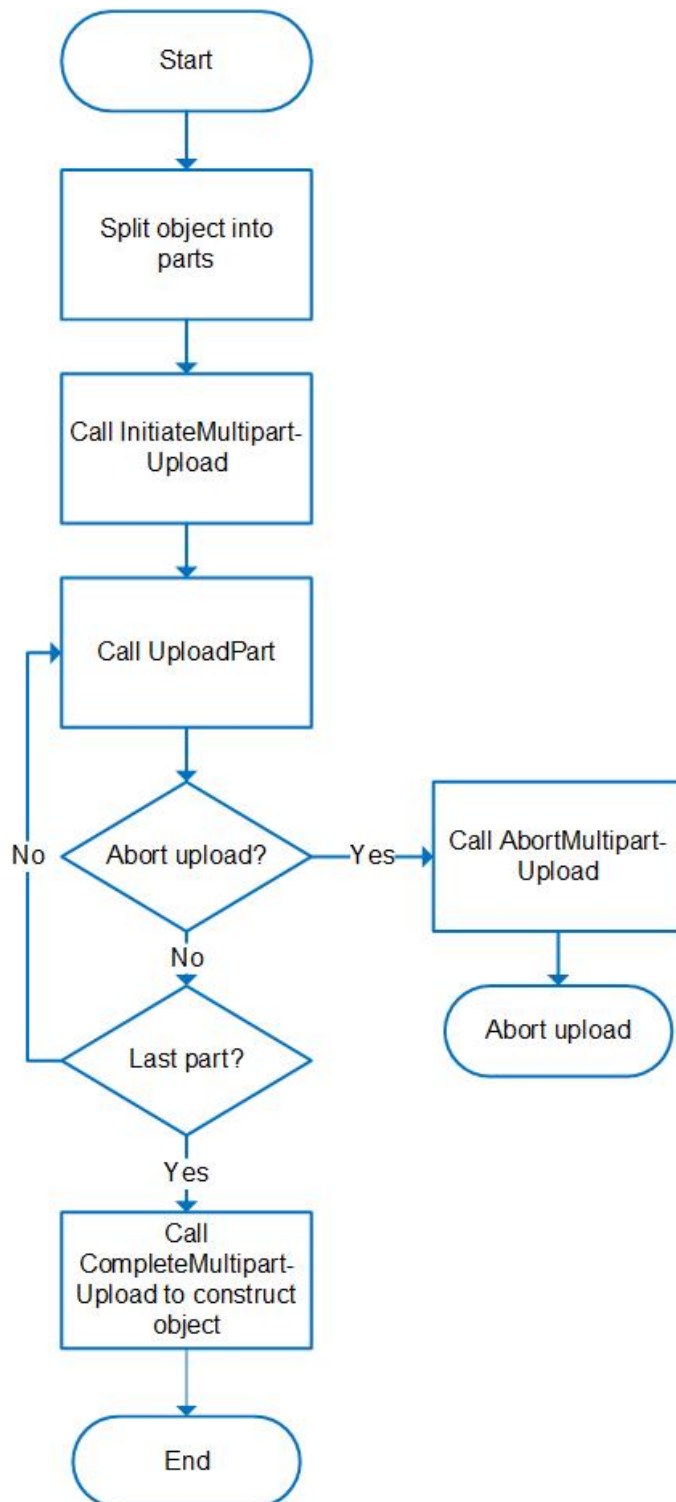
Object Storage Service (OSS) provides the multipart upload feature. Multipart upload allows you to split a large object into multiple parts that are simultaneously uploaded. After the upload is complete, you can call the CompleteMultipartUpload operation to combine these parts into a complete object. Multipart upload is similar to resumable upload.

Multipart upload process

To implement multipart upload, perform the following operations:

1. Split the object to upload into multiple parts.

2. Call the `InitiateMultipartUpload` operation to initiate a multipart upload task.
3. Call the `UploadPart` operation to upload the parts one by one or concurrently.
4. Call the `CompleteMultipartUpload` operation to complete the upload.



Before you start a multipart upload task, take note of the following items:

- Each part except the last part must be at least 100 KB in size. Otherwise, you may fail to call the `CompleteMultipartUpload` operation.

- The parts of the object that you want to upload are sorted based on the part numbers that you specify during the upload. However, parts are not uploaded in order, but are concurrently uploaded.

If you upload a large number of parts at the same time, the upload speed may vary based on network conditions and device loads. We recommend that you increase the size of each part when network conditions are good, and decrease the size of each part when network conditions are poor.

- By default, if you upload the parts of an object but you do not call the `CompleteMultipartUpload` operation to combine these parts, the uploaded parts are not automatically deleted. You can call the [AbortMultipartUpload](#) operation to terminate the upload task and delete the parts. For more information about how to automatically delete the uploaded parts, see [Manage lifecycle rules](#).

Initialize a multipart upload task

The following code provides an example on how to initialize a multipart upload task:

```
__block NSString * uploadId = nil;
__block NSMutableArray * partInfos = [NSMutableArray new];
NSString * uploadToBucket = @"<bucketName>";
// objectKey is equivalent to objectName and specifies the full path of the object that you
// want to upload to OSS. The path must include the file extension of the object. For example,
// you can set objectKey to abc/efg/123.jpg.
NSString * uploadObjectkey = @"<objectKey>";
// You can use OSSInitMultipartUploadRequest to specify the name of the object you want to
// upload and specify the name of the bucket to which the object belongs.
OSSInitMultipartUploadRequest * init = [OSSInitMultipartUploadRequest new];
init.bucketName = uploadToBucket;
init.objectKey = uploadObjectkey;
// init.contentType = @"application/octet-stream";
// The response to multipartUploadInit contains the upload ID. The upload ID is the unique
// ID of the multipart upload task.
OSSTask * initTask = [client multipartUploadInit:init];
[initTask waitUntilFinished];
if (!initTask.error) {
    OSSInitMultipartUploadResult * result = initTask.result;
    uploadId = result.uploadId;
} else {
    NSLog(@"multipart upload failed, error: %@", initTask.error);
    return;
}
```

Upload parts

The following code provides an example on how to upload parts:

```

// Specify the object you want to upload.
NSString * filePath = [docDir stringByAppendingPathComponent:@"***"];
// Query the access control list (ACL) of the object.
uint64_t fileSize = [[NSFileManager defaultManager] attributesOfItemAtPath:filePath error:
nil] fileSize];
// Specify the number of parts.
int chuckCount = *;
// Specify the size of each part.
uint64_t offset = fileSize/chuckCount;
for (int i = 1; i <= chuckCount; i++) {
    OSSUploadPartRequest * uploadPart = [OSSUploadPartRequest new];
    uploadPart.bucketName = uploadToBucket;
    uploadPart.objectkey = uploadObjectkey;
    uploadPart.uploadId = uploadId;
    uploadPart.partNumber = i; // part number start from 1
    NSFileHandle* readHandle = [NSFileHandle fileHandleForReadingAtPath:filePath];
    [readHandle seekToFileOffset:offset * (i -1)];
    NSData* data = [readHandle readDataOfLength:offset];
    uploadPart.uploadPartData = data;
    OSTask * uploadPartTask = [client uploadPart:uploadPart];
    [uploadPartTask waitUntilFinished];
    if (!uploadPartTask.error) {
        OSSUploadPartResult * result = uploadPartTask.result;
        uint64_t fileSize = [[NSFileManager defaultManager] attributesOfItemAtPath:uploadP
art.uploadPartFileURL.absoluteString error:nil] fileSize];
        [partInfos addObject:[OSSPartInfo partInfoWithPartNum:i eTag:result.eTag size:fileS
ize]];
    } else {
        NSLog(@"upload part error: %@", uploadPartTask.error);
        return;
    }
}

```

The preceding code uses `uploadPart` to upload each part.

- An upload ID and a part number must be specified for each multipart upload request.
- After a part is uploaded, the `uploadPart` operation does not immediately check the size of a part. This API operation checks the part size only if the multipart upload task is completed.
- The part number ranges from 1 to 10000. If a part number is not within this range, the `InvalidArgument` error code is returned.
- Each time you upload a part, make sure that the stream is directed to the start position of the part to upload.
- Each time a part is uploaded, OSS returns a response that contains the ETag value of the part. If the ETag value is the same as the MD5 hash, combine the ETag value with the part number into a PartETag. Then, save the PartETag for subsequent uploads of parts.

Complete the multipart upload task

In the following code, the `partInfos` field value specifies the list of PartETags saved during the multipart upload. After OSS receives the `partInfos` field value, OSS verifies the validity of each part. After OSS validates all parts of the object, OSS combines the parts into a complete object.

The following code provides an example on how to upload an object by using multipart upload:

```
OSSCompleteMultipartUploadRequest * complete = [OSSCompleteMultipartUploadRequest new];
complete.bucketName = uploadToBucket;
complete.objectKey = uploadObjectKey;
complete.uploadId = uploadId;
complete.partInfos = partInfos;
OSSTask * completeTask = [client completeMultipartUpload:complete];
[[completeTask continueWithBlock:^id(OSSTask *task) {
    if (!task.error) {
        OSSCompleteMultipartUploadResult * result = task.result;
        // ...
    } else {
        // ...
    }
    return nil;
}] waitUntilFinished];
```

You can set the `servercallback` parameter to complete the multipart upload request. After the multipart upload task is completed, a callback request is sent to the server address that you specify. You can view the `servercallback` result in `result.serverReturnJsonString` of the response.

```
OSSCompleteMultipartUploadRequest * complete = [OSSCompleteMultipartUploadRequest new];
complete.bucketName = @"<bucketName>";
complete.objectKey = @"<objectKey>";
complete.uploadId = uploadId;
complete.partInfos = partInfos;
complete.callbackParam = @{
    @"callbackUrl": @"<server address>",
    @"callbackBody": @"<test>"
};
complete.callbackVar = @{
    @"var1": @"value1",
    @"var2": @"value2"
};
OSSTask * completeTask = [client completeMultipartUpload:complete];
[[completeTask continueWithBlock:^id(OSSTask *task) {
    if (!task.error) {
        OSSCompleteMultipartUploadResult * result = task.result;
        NSLog(@"server call back return : %@", result.serverReturnJsonString);
    } else {
        // ...
    }
    return nil;
}] waitUntilFinished];
```

You can include the Content-MD5 field value in the upload request to check whether the object uploaded to OSS is the same as the local file. The OSS server helps you with the MD5 verification. The upload is successful only if the MD5 hash of the object received by the OSS server is the same as the Content-MD5 field value. This method can ensure the consistency of the uploaded object.

The following code provides an example on how to configure MD5 verification:

```
OSSUploadPartRequest * uploadPart = [OSSUploadPartRequest new];
uploadPart.bucketName = TEST_BUCKET;
uploadPart.uploadId = uploadId;
....
uploadPart.contentMd5 = [OSSUtil fileMD5String:filepath];
```

List parts

The following code provides an example on how to call the `listParts` method to obtain all uploaded parts of a multipart upload task:

```
OSSListPartsRequest * listParts = [OSSListPartsRequest new];
listParts.bucketName = @"<bucketName>";
listParts.objectKey = @"<objectkey>";
listParts.uploadId = @"<uploadid>";
OSSTask * listPartTask = [client listParts:listParts];
[listPartTask continueWithBlock:^id(OSSTask *task) {
    if (!task.error) {
        NSLog(@"list part result success!");
        OSSListPartsResult * listPartResult = task.result;
        for (NSDictionary * partInfo in listPartResult.parts) {
            NSLog(@"each part: %@", partInfo);
        }
    } else {
        NSLog(@"list part result error: %@", task.error);
    }
    return nil;
}];
```

 **Notice** By default, if a bucket contains more than 1,000 parts that are uploaded by using multipart upload, OSS returns the first 1,000 parts. In the response, the `IsTruncated` field is set to `false` and `NextPartNumberMarker` is returned to indicate the next starting position to continue reading data.

Cancel a multipart upload task

The following code provides an example on how to cancel a multipart upload request that is mapped to the specified `upload ID` :

```
OSSAbortMultipartUploadRequest * abort = [OSSAbortMultipartUploadRequest new];
abort.bucketName = @"<bucketName>";
abort.objectKey = @"<objectKey>";
abort.uploadId = uploadId;
OSSTask * abortTask = [client abortMultipartUpload:abort];
[abortTask waitUntilFinished];
if (!abortTask.error) {
    OSSAbortMultipartUploadResult * result = abortTask.result;
    uploadId = result.uploadId;
} else {
    NSLog(@"multipart upload failed, error: %@", abortTask.error);
    return;
}
```

10.6.4. Append upload

You can call the `AppendObject` operation of Object Storage Service (OSS) to append content to an appendable object. This is called append upload.

Usage notes

- If the object to which you want to append content does not exist, an appendable object is created when you call the `AppendObject` operation.
- When you call the `AppendObject` operation to append content to an object, you may encounter a few situations: If the object is an appendable object, and the specified position from which the append operation starts is equal to the current object length, the content is appended to the end of the object. If the specified position from which the append operation starts is not equal to the current object length, a `PositionNotEqualToLength` exception is thrown. If the object is not an appendable object, an `ObjectNotAppendable` exception is thrown.
- The `CopyObject` operation cannot be performed on appendable objects.

Sample code

The following code provides an example on how to perform append upload:

```

OSSAppendObjectRequest * append = [OSSAppendObjectRequest new];
// Configure the required fields. bucketName indicates the name of the bucket. objectKey is
// equivalent to objectName that indicates the full path of the object you want to upload to O
// SS by using append upload. The path must include the extension of the object. For example,
// you can set objectKey to abc/efg/123.jpg.
append.bucketName = @"<bucketName>";
append.objectKey = @"<objectKey>";
// Specify the position from which the first append operation starts.
append.appendPosition = 0;
NSString * docDir = [self getDocumentDirectory];
append.uploadingFileURL = [NSURL fileURLWithPath:@"<filepath>"];
// Configure the optional fields.
append.uploadProgress = ^(int64_t bytesSent, int64_t totalByteSent, int64_t totalBytesExpec
tedToSend) {
    NSLog(@"%lld, %lld, %lld", bytesSent, totalByteSent, totalBytesExpectedToSend);
};
// append.contentType = @"";
// append.contentMd5 = @"";
// append.contentEncoding = @"";
// append.contentDisposition = @"";
OSSTask * appendTask = [client appendObject:append];
[appendTask continueWithBlock:^id(OSSTask *task) {
    NSLog(@"objectKey: %@", append.objectKey);
    if (!task.error) {
        NSLog(@"append object success!");
        OSSAppendObjectResult * result = task.result;
        NSString * etag = result.eTag;
        long nextPosition = result.xOssNextAppendPosition;
    } else {
        NSLog(@"append object failed, error: %@", task.error);
    }
    return nil;
}];
}];

```

References

For more information about the API operation that you can call to perform append upload, see [AppendObject](#).

10.6.5. Resumable upload

In a wireless network environment, it takes a long time to upload a large object and the upload may fail due to poor network connectivity or network switches. If the upload fails, the entire object must be uploaded again. To resolve this problem, OSS SDK for iOS provides the resumable upload feature.

Background information

We recommend that you do not use resumable upload when you upload objects smaller than 5 GB in size from a mobile device. Resumable upload is implemented by using the multipart upload method. Resumable upload of a single object requires multiple network requests, which is inefficient. The following section provides the precautions you need to understand before and during resumable upload.

- Before resumable upload

Before you upload an object to OSS by using the resumable upload method, you can specify a folder for the checkpoint file that stores resumable upload records. The checkpoint file applies only to the current resumable upload task.

- If you do not specify the folder for the checkpoint file and part of a large object fails to be uploaded due to network issues, it takes a large amount of time and consumes a large amount of traffic to upload the entire large object again.
- If you specify the folder for the checkpoint file, a failed resumable upload task can be resumed from the position recorded in the checkpoint file.

- During resumable upload

When you use resumable upload, you must understand the following information:

- Resumable upload allows you to upload only local files. Resumable upload supports upload callback. You can use resumable upload in a similar way you use upload callback. For more information, see [Callback](#).
- Resumable upload is implemented by calling the following API operations: `InitMultipartUpload`, `UploadPart`, `ListParts`, `CompleteMultipartUpload`, and `AbortMultipartUpload`. If you want to use the STS authentication mode to perform resumable upload tasks, make sure that you are authorized to call the preceding API operations.
- By default, the MD5 hash of each part is verified in a resumable upload task. Therefore, you do not need to specify the `Content-Md5` header in the request.
- When a resumable upload task fails, the uploaded parts become useless in OSS. You can configure lifecycle rules for a bucket to delete expired parts. For more information, see [Manage lifecycle rules](#).

 **Notice** By default, if the current upload task is canceled, the parts uploaded to the server are deleted synchronously. If you want to retain resumable upload records, you must specify a folder to store the checkpoint file that contains the resumable upload records and modify the setting of the `deleteUploadIdOnCancelling` parameter. If resumable upload records are retained on the mobile device for an extended period of time but the parts uploaded to the server have been deleted by the configured lifecycle rules of the bucket, the resumable upload records on the server may be different from those on the mobile device.

Sample code

- The following code provides an example on how to call the `resumableUpload` method to perform a resumable upload task when the checkpoint file is permanently stored in the local device:

```
OSSResumableUploadRequest * resumableUpload = [OSSResumableUploadRequest new];
resumableUpload.bucketName = OSS_BUCKET_PRIVATE;
//...
NSString *cacheDir = [NSSearchPathForDirectoriesInDomains(NSCachesDirectory, NSUserDomainMask, YES) firstObject];
resumableUpload.recordDirectoryPath = cacheDir;
```


 **Note** When a resumable upload task fails, a checkpoint file is generated to record the resumable upload progress. The next upload starts from the position recorded in the checkpoint file to upload the entire object. After the object is uploaded, the checkpoint file is automatically deleted. By default, the checkpoint file is not retained permanently on the local device.

- The following code provides a complete example on how to perform a resumable upload task:

```
// Obtain an upload ID to upload an object.
OSSResumableUploadRequest * resumableUpload = [OSSResumableUploadRequest new];
resumableUpload.bucketName = <bucketName>;
// objectKey is equivalent to objectName that indicates the complete path of the object y
ou want to upload to OSS by using resumable upload. The path must include the extension o
f the object. For example, you can set objectKey to abc/efg/123.jpg.
resumableUpload.objectKey = <objectKey>;
resumableUpload.partSize = 1024 * 1024;
resumableUpload.uploadProgress = ^(int64_t bytesSent, int64_t totalByteSent, int64_t tota
lBytesExpectedToSend) {
    NSLog(@"%lld, %lld, %lld", bytesSent, totalByteSent, totalBytesExpectedToSend);
};
NSString *cachesDir = [NSSearchPathForDirectoriesInDomains(NSCachesDirectory, NSUserDomai
nMask, YES) firstObject];
// Specify the path to store the checkpoint file.
resumableUpload.recordDirectoryPath = cachesDir;
// Set the deleteUploadOnCancelling parameter to NO. NO indicates that the checkpoint fil
e is not deleted when the upload task fails. The next upload starts from the position rec
orded in the checkpoint file to upload the entire object. If you do not specify this para
meter, default value YES is used. YES indicates that the checkpoint file is deleted when
the upload task fails. The entire object is uploaded again during the next upload.
resumableUpload.deleteUploadIdOnCancelling = NO;
resumableUpload.uploadingFileURL = [NSURL fileURLWithPath:<your file path>];
OSSTask * resumeTask = [client resumableUpload:resumableUpload];
[resumeTask continueWithBlock:^(OSSTask *task) {
    if (task.error) {
        NSLog(@"error: %@", task.error);
        if ([task.error.domain isEqualToString:OSSClientErrorDomain] && task.error.code =
= OSSClientErrorCodeCannotResumeUpload) {
            // The task cannot be resumed. You must obtain a new upload ID to upload the
object.
        }
    } else {
        NSLog(@"Upload file success");
    }
    return nil;
}];
// [resumeTask waitUntilFinished];
// [resumableUpload cancel];
```

10.6.6. Upload callback

This topic describes how to configure upload callback.

For more information, see [Callback](#).

The following code provides an example on how to configure upload callback:

```
OSSPutObjectRequest * request = [OSSPutObjectRequest new];
request.bucketName = @"<bucketName>";
request.objectKey = @"<objectKey>";
request.uploadingFileURL = [NSURL URLWithString:@"<filepath>"];
// Configure callback parameters.
request.callbackParam = @{
    @"callbackUrl": @"<your server callback address>",
    @"callbackBody": @"<your callback body>"
};
// Configure custom variables.
request.callbackVar = @{
    @"<var1>": @"<value1>",
    @"<var2>": @"<value2>"
};
request.uploadProgress = ^(int64_t bytesSent, int64_t totalByteSent, int64_t totalBytesExpectedToSend) {
    NSLog(@"%lld, %lld, %lld", bytesSent, totalByteSent, totalBytesExpectedToSend);
};
OSSTask * task = [client putObject:request];
[task continueWithBlock:^id(OSSTask *task) {
    if (task.error) {
        OSSLogError(@"%@", task.error);
    } else {
        OSSPutObjectResult * result = task.result;
        NSLog(@"Result - requestId: %@, headerFields: %@, servercallback: %@",
            result.requestId,
            result.httpResponseHeaderFields,
            result.serverReturnJsonString);
    }
    return nil;
}];
```

10.6.7. Upload progress bars

You can use a progress bar to indicate the progress of an object that is being uploaded or downloaded. The PutObject operation is used in the example to describe how to display the progress bar of an object that is being uploaded.

The following code provides an example on how to display the progress bar when you upload a local file as an object in a bucket:

```
OSSPutObjectRequest * put = [OSSPutObjectRequest new];
// Specify the name of the bucket. Example: examplebucket.
put.bucketName = @"examplebucket";
// Specify the full path of the object. The path cannot contain the bucket name. Example: exampledir/exampleobject.txt.
put.objectKey = @"exampledir/exampleobject.txt";
// Specify the full path of the local file to upload.
// By default, if the path of the local file is not specified, the local file is uploaded from the path of the project to which the sample program belongs.
put.uploadingFileURL = [NSURL fileURLWithPath:@"filePath"];
// Configure the callback function to display the progress bar.
put.uploadProgress = ^(int64_t bytesSent, int64_t totalByteSent, int64_t totalBytesExpectedToSend) {
    // Specify the length of the segments that indicate the sizes of data being uploaded, data that is uploaded, and the total data to upload.
    NSLog(@"%lld, %lld, %lld", bytesSent, totalByteSent, totalBytesExpectedToSend);
};
OSSTask * putTask = [client putObject:put];
[putTask continueWithBlock:^id(OSSTask *task) {
    if (!task.error) {
        NSLog(@"upload object success!");
    } else {
        NSLog(@"upload object failed, error: %@", task.error);
    }
    return nil;
}];
```

10.7. Download objects

10.7.1. Overview

This document describes how to use the OSS iOS SDK to download objects.

The OSS iOS SDK provides the following download methods:

- [Simple download](#)
- [Streaming download](#)
- [Range download](#)
- [Resumable download](#)

10.7.2. Simple download

This topic describes how to perform simple download.

You can download an object to a local file or as NSData:

```
OSSGetObjectRequest * request = [OSSGetObjectRequest new];
// Configure required fields.
request.bucketName = @"<bucketName>";
request.objectKey = @"<objectKey>";
// Configure optional fields.
request.downloadProgress = ^(int64_t bytesWritten, int64_t totalBytesWritten, int64_t totalBytesExpectedToWrite) {
    // Specify the number of bytes that are being downloaded, the number of bytes that are
    // downloaded, and the total number of bytes that you want to download.
    NSLog(@"%lld, %lld, %lld", bytesWritten, totalBytesWritten, totalBytesExpectedToWrite);
};
// request.range = [[OSSRange alloc] initWithStart:0 withEnd:99]; // bytes=0-99. Specify th
// at the download range is from byte 0 to byte 99.
// request.downloadToFileURL = [NSURL fileURLWithPath:@"<filepath>"]; // If you need to dir
// ectly download the object to a file, specify the path of the file.
OSSTask * getTask = [client getObject:request];
[getTask continueWithBlock:^id(OSSTask *task) {
    if (!task.error) {
        NSLog(@"download object success!");
        OSSGetObjectResult * getResult = task.result;
        NSLog(@"download result: %@", getResult.downloadedData);
    } else {
        NSLog(@"download object failed, error: %@", task.error);
    }
    return nil;
}];
// [getTask waitUntilFinished];
// [request cancel];
```

10.7.3. Streaming download

If you have a large object that takes a long time to download in its entirety, you can use streaming download to download the object incrementally until the object is downloaded in full.

OSS SDK for iOS does not provide any streaming download APIs. Instead, it provides the multipart callback feature that is similar to the `didReceiveData` function in the `NSURLSession` library. Note: The download result does not contain the actual data if multipart callback is configured.

The following code provides an example on how to perform streaming download:

```
OSSGetObjectRequest * request = [OSSGetObjectRequest new];
// Configure the required fields. objectKey is equivalent to objectName and indicates the c
omplete path of the object that you want to download from OSS. The path must include the fi
le extension of the object. For example, you can set objectKey to abc/efg/123.jpg.
request.bucketName = @"<bucketName>";
request.objectKey = @"<objectKey>";
// Configure the multipart callback function.
request.onRecieveData = ^(NSData * data) {
    NSLog(@"Recieve data, length: %ld", [data length]);
};
OSSTask * getTask = [client getObject:request];
[getTask continueWithBlock:^id(OSSTask *task) {
    if (! task.error) {
        NSLog(@"download object success!") ;
    } else {
        NSLog(@"download object failed, error: %@", task.error);
    }
    return nil;
}];
// [getTask waitUntilFinished];
// [request cancel];
```

10.7.4. Range download

If you only need part of the data in an object, you can use range download to download specified content.

Range download is suitable for downloading large objects. If the Range parameter is specified in the request header, the response contains the length of the entire object and the range of data.

Run the following code for range download:

```

OSSGetObjectRequest * request = [OSSGetObjectRequest new];
request.bucketName = @"<bucketName>";
request.objectKey = @"<objectKey>";
request.range = [[OSSRange alloc] initWithStart:1 withEnd:99]; // bytes=1-99
// request.range = [[OSSRange alloc] initWithStart:-1 withEnd:99]; // bytes=-99
// request.range = [[OSSRange alloc] initWithStart:10 withEnd:-1]; // bytes=10-
request.downloadProgress = ^(int64_t bytesWritten, int64_t totalBytesWritten, int64_t total
BytesExpectedToWrite) {
    NSLog(@"%lld, %lld, %lld", bytesWritten, totalBytesWritten, totalBytesExpectedToWrite);
};
OSSTask * getTask = [client getObject:request];
[getTask continueWithBlock:^id(OSSTask *task) {
    if (! task.error) {
        NSLog(@"download object success!") ;
        OSSGetObjectResult * getResult = task.result;
        NSLog(@"download result: %@", getResult.downloadedData);
    } else {
        NSLog(@"download object failed, error: %@", task.error);
    }
    return nil;
}];
// [getTask waitUntilFinished];
// [request cancel];

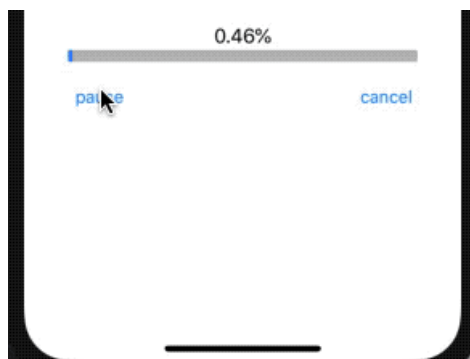
```

10.7.5. Resumable download

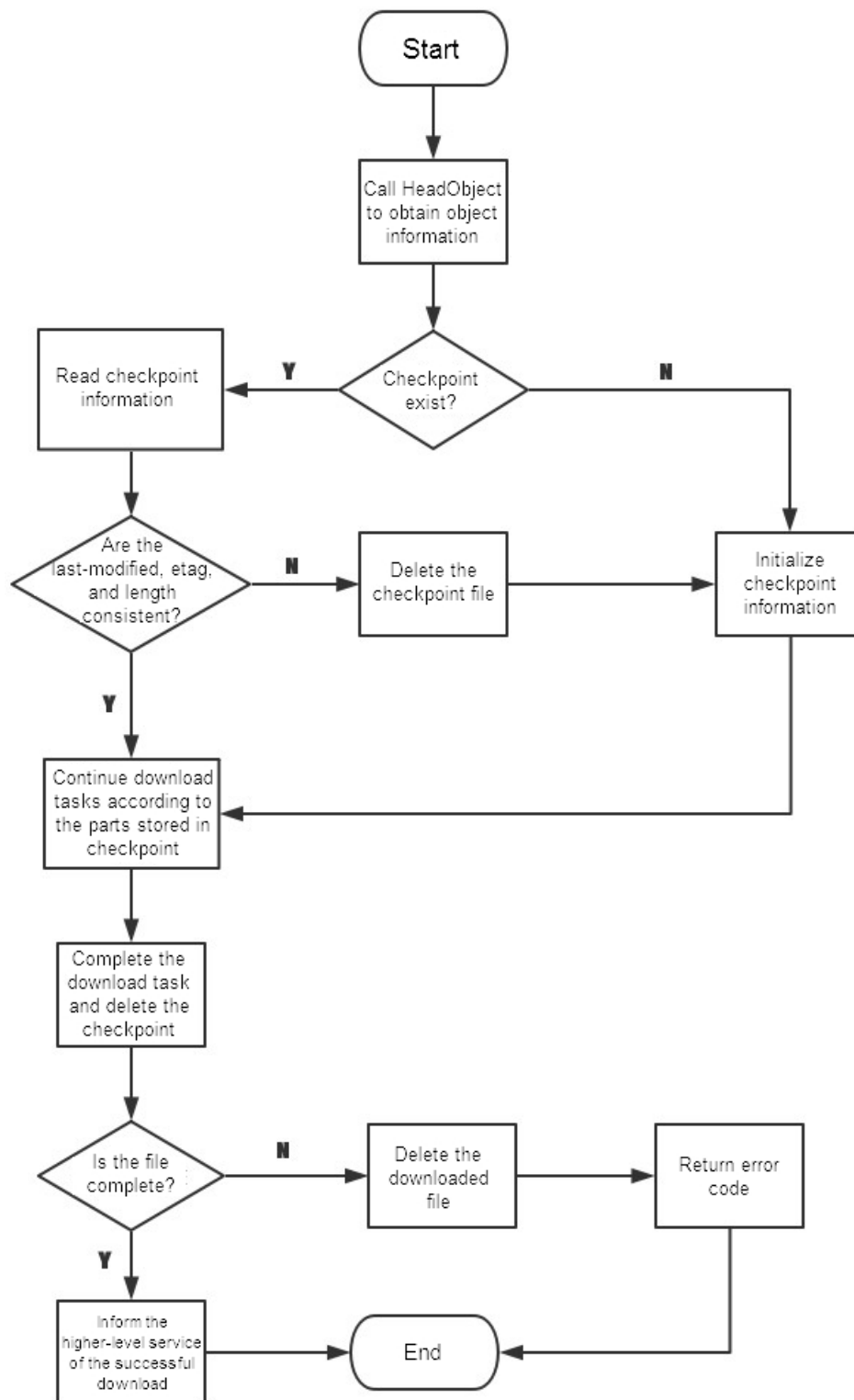
When a client downloads resources over a network, the download is interrupted for some reason. In this case, you can use resumable download to continue downloading the uncompleted parts. This method saves time and traffic.

Scenario: When you use a video application to download a video on a mobile phone, and the network switches from Wi-Fi to Cellular during the download, the application automatically interrupts the download. When you switch back to Wi-Fi, you can manually restart the download task to resume the download.

The following figure shows the effect of resumable download.



The following figure shows the process of resumable download.



Detail analysis

- The Range field is added to the HTTP 1.1 header to specify the scope within which to obtain data. The formats of the Range field value are as follows:
 - Range: bytes=100- : starts from the 101st Byte until the download is complete.

- Range: bytes=100-200 : starts the download from the 101st Byte and stops the download at the 201st Byte. The Range field value is counted from 0 by default. This method is typically used for multipart transmission of large objects, such as videos.
 - Range: bytes=-100 : The start position is not specified. The server downloads the last 100 Bytes, instead of the 100 Bytes starting from 0.
 - Range: bytes=0-100, 200-300 : downloads content within multiple ranges at the same time. This is untypical.
- During resumable download, the server requires to use the If-Match value in the header to verify whether the object on the server changes. The If-Match field value corresponds to the ETag value.
- When the client initiates a request, the Range and If-Match field values are included in the request. The OSS server verifies the ETag value that corresponds to the If-Match field value. If the received ETag value does not match the ETag value saved in OSS, OSS returns the 412 precondition status code.
- The OSS server currently supports the following fields for GetObject: Range, If-Match, If-None-Match, If-Modified-Since, If-Unmodified-Since. Therefore, you can use resumable download to download resources from OSS on your mobile device.

Specific implementation

Example:

 **Note** The following code is for reference only. We recommend that you do not use it in production projects. To download the code, you can download it by yourself or use third-party open-source download software.

```
#import "DownloadService.h"
#import "OSSTestMacros.h"
@implementation DownloadRequest
@end
@implementation Checkpoint
- (instancetype)copyWithZone:(NSZone *)zone {
    Checkpoint *other = [[[self class] allocWithZone:zone] init];
    other.etag = self.etag;
    other.totalExpectedLength = self.totalExpectedLength;
    return other;
}
@end
@interface DownloadService()<NSURLSessionTaskDelegate, NSURLSessionDataDelegate>
@property (nonatomic, strong) NSURLSession *session; //The network session.
@property (nonatomic, strong) NSURLSessionDataTask *dataTask; //The data request task.
@property (nonatomic, copy) DownloadFailureBlock failure; //The request error.
@property (nonatomic, copy) DownloadSuccessBlock success; //The request succeeds.
@property (nonatomic, copy) DownloadProgressBlock progress; //The download progress.
@property (nonatomic, copy) Checkpoint *checkpoint; //The checkpoint file.
@property (nonatomic, copy) NSString *requestURLString; //The object resource URL used in a download request.
@property (nonatomic, copy) NSString *headURLString; //The object resource URL used in a HEAD request.
@property (nonatomic, copy) NSString *targetPath; //The path to the object.
```



```

@property (nonatomic, assign) unsigned long long totalReceivedContentLength; //The size of
the downloaded content.
@property (nonatomic, strong) dispatch_semaphore_t semaphore;
@end

@implementation DownloadService
- (instancetype)init
{
    self = [super init];
    if (self) {
        NSURLSessionConfiguration *conf = [NSURLSessionConfiguration defaultSessionConfigur
ation];
        conf.timeoutIntervalForRequest = 15;
        NSOperationQueue *processQueue = [NSOperationQueue new];
        _session = [NSURLSession sessionWithConfiguration:conf delegate:self delegateQueue:
processQueue];
        _semaphore = dispatch_semaphore_create(0);
        _checkpoint = [[Checkpoint alloc] init];
    }
    return self;
}

+ (instancetype)downloadServiceWithRequest:(DownloadRequest *)request {
    DownloadService *service = [[DownloadService alloc] init];
    if (service) {
        service.failure = request.failure;
        service.success = request.success;
        service.requestURLString = request.sourceURLString;
        service.headURLString = request.headURLString;
        service.targetPath = request.downloadFilePath;
        service.progress = request.downloadProgress;
        if (request.checkpoint) {
            service.checkpoint = request.checkpoint;
        }
    }
    return service;
}

/**
 * Use HeadObject to obtain Object Meta. Compare the ETag value saved in OSS with that save
d in the local checkpoint file. The result is returned.
 */
- (BOOL)getFileInfo {
    __block BOOL resumable = NO;
    NSURL *url = [NSURL URLWithString:self.headURLString];
    NSMutableURLRequest *request = [[NSMutableURLRequest alloc] initWithURL:url];
    [request setHTTPMethod:@"HEAD"];
    NSURLSessionDataTask *task = [[NSURLSession sharedSession] dataTaskWithRequest:request
completionHandler:^(NSData * _Nullable data, NSURLResponse * _Nullable response, NSError *
_Nullable error) {
        if (error) {
            NSLog(@"Failed to obtain Object Meta. error: %@", error);
        } else {
            NSHTTPURLResponse *httpResponse = (NSHTTPURLResponse *)response;
            NSString *etag = [httpResponse.allHeaderFields objectForKey:@"ETag"];
            if ([self.checkpoint.etag isEqualToString:etag]) {
                resumable = YES;
            }
        }
    }];
}

```

```

        } else {
            resumable = NO;
        }
    }
    dispatch_semaphore_signal(self.semaphore);
}];
[task resume];
dispatch_semaphore_wait(self.semaphore, DISPATCH_TIME_FOREVER);
return resumable;
}
/**
 * Obtain the size of the local file.
 */
- (unsigned long long)fileSizeAtPath:(NSString *)filePath {
    unsigned long long fileSize = 0;
    NSFileManager *dfm = [NSFileManager defaultManager];
    if ([dfm fileExistsAtPath:filePath]) {
        NSError *error = nil;
        NSDictionary *attributes = [dfm attributesOfItemAtPath:filePath error:&error];
        if (! error && attributes) {
            fileSize = attributes.fileSize;
        } else if (error) {
            NSLog(@"error: %@", error);
        }
    }
    return fileSize;
}
- (void)resume {
    NSURL *url = [NSURL URLWithString:self.requestURLString];
    NSMutableURLRequest *request = [[NSMutableURLRequest alloc] initWithURL:url];
    [request setHTTPMethod:@"GET"];
    BOOL resumable = [self getFileInfo]; // If the resumable field value is NO, resumabl
e upload is unfeasible. Otherwise, continue the resumable upload logic.
    if (resumable) {
        self.totalReceivedContentLength = [self fileSizeAtPath:self.targetPath];
        NSString *requestRange = [NSString stringWithFormat:@"bytes=%llu-", self.totalRecei
vedContentLength];
        [request setValue:requestRange forHTTPHeaderField:@"Range"];
    } else {
        self.totalReceivedContentLength = 0;
    }
    if (self.totalReceivedContentLength == 0) {
        [[NSFileManager defaultManager] createFileAtPath:self.targetPath contents:nil attri
butes:nil];
    }
    self.dataTask = [self.session dataTaskWithRequest:request];
    [self.dataTask resume];
}
- (void)pause {
    [self.dataTask cancel];
    self.dataTask = nil;
}
- (void)cancel {
    [self.dataTask cancel];
    self.dataTask = nil;
}

```

```

        self.dataTask = nil;
        [self removeFileAtPath: self.targetPath];
    }
- (void)removeFileAtPath:(NSString *)filePath {
    NSError *error = nil;
    [[NSFileManager defaultManager] removeItemAtPath:self.targetPath error:&error];
    if (error) {
        NSLog(@"remove file with error : %@", error);
    }
}
#pragma mark - NSURLSessionDataDelegate
- (void)URLSession:(NSURLSession *)session task:(NSURLSessionTask *)task
didCompleteWithError:(nullable NSError *)error {
    NSHTTPURLResponse *httpResponse = (NSHTTPURLResponse *)task.response;
    if ([httpResponse isKindOfClass:[NSHTTPURLResponse class]]) {
        if (httpResponse.statusCode == 200) {
            self.checkpoint.etag = [[httpResponse allHeaderFields] objectForKey:@"Etag"];
            self.checkpoint.totalExpectedLength = httpResponse.expectedContentLength;
        } else if (httpResponse.statusCode == 206) {
            self.checkpoint.etag = [[httpResponse allHeaderFields] objectForKey:@"Etag"];
            self.checkpoint.totalExpectedLength = self.totalReceivedContentLength + httpRes
ponse.expectedContentLength;
        }
    }
    if (error) {
        if (self.failure) {
            NSMutableDictionary *userInfo = [NSMutableDictionary dictionaryWithDictionary:e
rror.userInfo];
            [userInfo oss_setObject:self.checkpoint forKey:@"checkpoint"];
            NSError *tError = [NSError errorWithDomain:error.domain code:error.code userInf
o:userInfo];
            self.failure(tError);
        }
    } else if (self.success) {
        self.success(@"status": @"success");
    }
}
- (void)URLSession:(NSURLSession *)session dataTask:(NSURLSessionDataTask *)dataTask didRec
eiveResponse:(NSURLResponse *)response completionHandler:(void (^)(NSURLSessionResponseDisp
osition))completionHandler
{
    NSHTTPURLResponse *httpResponse = (NSHTTPURLResponse *)dataTask.response;
    if ([httpResponse isKindOfClass:[NSHTTPURLResponse class]]) {
        if (httpResponse.statusCode == 200) {
            self.checkpoint.totalExpectedLength = httpResponse.expectedContentLength;
        } else if (httpResponse.statusCode == 206) {
            self.checkpoint.totalExpectedLength = self.totalReceivedContentLength + httpRe
sponse.expectedContentLength;
        }
    }
    completionHandler(NSURLSessionResponseAllow);
}
- (void)URLSession:(NSURLSession *)session dataTask:(NSURLSessionDataTask *)dataTask didRec
eiveData:(NSData *)data {
    NSFileHandle *fileHandle = [NSFileHandle fileHandleForWritingAtPath:self.targetPath];

```

```
[fileHandle seekToEndOfFile];
[fileHandle writeData:data];
[fileHandle closeFile];
self.totalReceivedContentLength += data.length;
if (self.progress) {
    self.progress(data.length, self.totalReceivedContentLength, self.checkpoint.totalExpectedLength);
}
}
@end
```

The preceding sample code shows the process logic when data is received over the network. The fields in the code are described as follows:

- `DownloadService`: the core of the download logic.
- `NSURLSession:dataTask:didReceiveData:`: The received network data is appended to the object. The download progress record is updated.
- `NSURLSession:task:didCompleteWithError:`: determines whether the download task is complete, and then returns the result to the upper layer business.
- `NSURLSession:dataTask:didReceiveResponse:completionHandler:`: processes object-relevant information, such as the ET ag value for the precheck of resumable download and the content-length field value to calculate the download progress.

`DownloadRequest` is defined as follows:

```

#import <Foundation/Foundation.h>
typedef void(^DownloadProgressBlock)(int64_t bytesReceived, int64_t totalBytesReceived, int
64_t totalBytesExpectToReceived);
typedef void(^DownloadFailureBlock)(NSError *error);
typedef void(^DownloadSuccessBlock)(NSDictionary *result);
@interface Checkpoint : NSObject<NSCopying>
@property (nonatomic, copy) NSString *etag; // The ETag value of the resource.
@property (nonatomic, assign) unsigned long long totalExpectedLength; //The total size o
f the object.
@end
@interface DownloadRequest : NSObject
@property (nonatomic, copy) NSString *sourceURLString; // The URL for the download.
@property (nonatomic, copy) NSString *headURLString; // The URL used to obtain the o
riginal information of the object.
@property (nonatomic, copy) NSString *downloadFilePath; // The local path to the object
.
@property (nonatomic, copy) DownloadProgressBlock downloadProgress; // The download progres
s.
@property (nonatomic, copy) DownloadFailureBlock failure; // The callback that is sent af
ter the download fails.
@property (nonatomic, copy) DownloadSuccessBlock success; // The callback that is sent af
ter the download succeeds.
@property (nonatomic, copy) Checkpoint *checkpoint; // The checkpoint file that sto
res the ETag value of the object.
@end
@interface DownloadService : NSObject
+ (instancetype)downloadServiceWithRequest:(DownloadRequest *)request;
/**
 * Start the download.
 */
- (void)resume;
/**
 * Pause the download.
 */
- (void)pause;
/**
 * Cancel the download.
 */
- (void)cancel;
@end

```

The invocation code for the upper-layer business is as follows:

```

- (void)initDownloadURLs {
    OSSPlainTextAKSKPairCredentialProvider *pCredential = [[OSSPlainTextAKSKPairCredentialP
rovider alloc] initWithPlainTextAccessKey:OSS_ACCESSKEY_ID secretKey:OSS_SECRETKEY_ID];
    _mClient = [[OSSClient alloc] initWithEndpoint:OSS_ENDPOINT credentialProvider:pCredent
ial];
    // Generate the signed URL for the GET request.
    OSTask *downloadURLTask = [_mClient presignConstrainURLWithBucketName:@"aliyun-dhc-sha
nghai" withObjectKey:OSS_DOWNLOAD_FILE_NAME withExpirationInterval:1800];
    [downloadURLTask waitUntilFinished];
    _downloadURLString = downloadURLTask.result;
    // Generate the signed URL for the GET request.
}

```

```

        // Generate the signed URL for the HEAD request.
        OSSTask *headURLTask = [_mClient presignConstrainURLWithBucketName:@"aliyun-dhc-shangha
i" withObjectKey:OSS_DOWNLOAD_FILE_NAME httpMethod:@"HEAD" withExpirationInterval:1800 with
Parameters:nil];
        [headURLTask waitUntilFinished];
        _headURLString = headURLTask.result;
    }
- (IBAction)resumeDownloadClicked:(id)sender {
    _downloadRequest = [DownloadRequest new];
    _downloadRequest.sourceURLString = _downloadURLString;          // Set the resource URL.
    _downloadRequest.headURLString = _headURLString;
    NSString *documentPath = NSSearchPathForDirectoriesInDomains(NSDocumentDirectory, NSUse
rDomainMask, YES).firstObject;
    _downloadRequest.downloadFilePath = [documentPath stringByAppendingPathComponent:OSS_DO
WNLOAD_FILE_NAME]; //Set the local path to the object you want to download.
    __weak typeof(self) weakSelf = self;
    _downloadRequest.downloadProgress = ^(int64_t bytesReceived, int64_t totalBytesReceived
, int64_t totalBytesExpectToReceived) {
        // totalBytesReceived is the number of Bytes already cached by the client. totalByt
esExpectToReceived is the total number of Bytes that need to be downloaded.
        dispatch_async(dispatch_get_main_queue(), ^{
            __strong typeof(self) sSelf = weakSelf;
            CGFloat fProgress = totalBytesReceived * 1.f / totalBytesExpectToReceived;
            sSelf.progressLab.text = [NSString stringWithFormat:@"%%. 2f%%", fProgress * 100
];

            sSelf.progressBar.progress = fProgress;
        });
    };
    _downloadRequest.failure = ^(NSError *error) {
        __strong typeof(self) sSelf = weakSelf;
        sSelf.checkpoint = error.userInfo[@"checkpoint"];
    };
    _downloadRequest.success = ^(NSDictionary *result) {
        NSLog(@"The download succeeds.");
    };
    _downloadRequest.checkpoint = self.checkpoint;
    NSString *titleText = [[_downloadButton titleLabel] text];
    if ([titleText isEqualToString:@"download"]) {
        [_downloadButton setTitle:@"pause" forState: UIControlStateNormal];
        _downloadService = [DownloadService downloadServiceWithRequest:_downloadRequest];
        [_downloadService resume];
    } else {
        [_downloadButton setTitle:@"download" forState: UIControlStateNormal];
        [_downloadService pause];
    }
}
- (IBAction)cancelDownloadClicked:(id)sender {
    [_downloadButton setTitle:@"download" forState: UIControlStateNormal];
    [_downloadService cancel];
}

```

 **Note** The checkpoint file can be obtained from the failure callback when the download is paused or cancelled. When you restart the download, you can upload the checkpoint file to `DownloadRequest`. Then, `DownloadService` uses the checkpoint file to verify the consistency.

10.7.6. Conditional download

You can specify one or more conditions to download objects. If the specified conditions are met, the object is downloaded. If the specified conditions are not met, an error is returned and the object is not downloaded.

Sample code

The following code provides an example on how to download the `exampleobject.txt` object from the `exampledir` directory of the `examplebucket` bucket. Multiple conditions are specified for the download:

```
OSSGetObjectRequest *get = [OSSGetObjectRequest new];
// Specify the bucket name. For more information about the naming conventions for buckets,
// see Bucket naming conventions.
get.bucketName = @"examplebucket";
// Specify the full path of the object. The path cannot contain the bucket name. For more i
nformation about the naming conventions for objects, see Object naming conventions.
get.objectKey = @"exampledir/exampleobject.txt";
NSMutableDictionary *headerFields = [NSMutableDictionary dictionary];
// Specify that objects that are modified after 14:47:53 Friday, October 13, 2021 in GMT ar
e downloaded.
[headerFields setValue:@"Fri, 13 Oct 2021 14:47:53 GMT" forKey:@"If-Modified-Since"];
// Specify that objects that are modified at the current time or earlier are downloaded.
[headerFields setValue:[NSDate new] oss_asStringValue forKey:@"If-Unmodified-Since"];
// Specify that objects whose ETag values do not match the ETag values that are specified i
n the requests are downloaded.
[headerFields setValue:@"5B3C1A2E0563E1B002CC607C*****" forKey:@"If-None-Match"];
// Specify that objects whose ETag values match the ETag values that are specified in the r
equests are downloaded.
[headerFields setValue:@"fba9dede5f27731c9771645a3986*****" forKey:@"If-Match"];
get.headerFields = headerFields;
[[[_client getObject:get] continueWithBlock:^(id _Nullable(OSSTask * _Nonnull task) {
    if (!task.error) {
        NSLog(@"get object success!");
    } else {
        NSLog(@"get object error: %@", task.error);
    }
    return nil;
}] waitUntilFinished];
```

References

For more information about the API operation that you can call to perform conditional download, see [GetObject](#).

10.7.7. Download progress bars

You can use a progress bar to indicate the progress of an object that is being uploaded or downloaded. The `GetObject` operation is used in the example to describe how to display the progress bar of an object that is being downloaded.

The following code provides an example on how to display the progress of an object named `exampleobject.txt` that is being downloaded from a bucket named `examplebucket`:

```
OSSGetObjectRequest * request = [OSSGetObjectRequest new];
// Specify the bucket name.
request.bucketName = @"examplebucket";
// Specify the full path of the object that you want to download. The full path of the object cannot contain bucket names.
request.objectKey = @"exampleobject.txt";
// Configure the callback function to display the progress bar.
request.downloadProgress = ^(int64_t bytesWritten, int64_t totalBytesWritten, int64_t totalBytesExpectedToWrite) {
    // Specify the length of the segments that indicate the sizes of data being downloaded, data that is downloaded, and the total data to download.
    NSLog(@"%lld, %lld, %lld", bytesWritten, totalBytesWritten, totalBytesExpectedToWrite);
};
OSSTask * getTask = [client getObject:request];
[getTask continueWithBlock:^id(OSSTask *task) {
    if (!task.error) {
        NSLog(@"download object success!");
    } else {
        NSLog(@"download object failed, error: %@", task.error);
    }
    return nil;
}];
```

10.8. Manage objects

10.8.1. Overview

You can use APIs to perform the following operations on objects in a bucket:

- Determine whether the specified object exists
- Obtain Object Meta
- List objects
- Copy an object
- Delete objects

10.8.2. Determine whether the specified object exists

The OSS iOS SDK provides a convenient synchronous API to check whether the specified object exists in the bucket.

Use the following code to determine whether an object exists:


```
NSError * error = nil;
BOOL isExist = [client doesObjectExistInBucket:TEST_BUCKET withObjectKey:@"file1m" withError:&error];
if (! error) {
    if(isExist) {
        NSLog(@"File exists.");
    } else {
        NSLog(@"File not exists.");
    }
} else {
    NSLog(@"Error!" );
}
```

10.8.3. Obtain object metadata

The `HeadObject` method can obtain only the object metadata but not the content of the object.

Note

- The object metadata includes HTTP headers and user metadata. For more information, see [Manage object metadata](#) in OSS Developer Guide.
- You cannot set or modify the object metadata by using the iOS SDK.

The following code provides an example on how to obtain the object metadata:

```
OSSHeadObjectRequest * request = [OSSHeadObjectRequest new];
request.bucketName = @"<bucketName>;
//objectKey indicates the name of the object, that is, the complete path of the object. objectKey must include the file extension of the object. For example, set objectKey to abc/efg/123.jpg.
request.objectKey = @"<objectKey>";
OSSTask * headTask = [client headObject:request];
[headTask continueWithBlock:^id(OSSTask *task) {
    if (! task.error) {
        NSLog(@"head object success!") ;
        OSSHeadObjectResult * result = task.result;
        NSLog(@"header fields: %@", result.httpResponseHeaderFields);
        for (NSString * key in result.objectMeta) {
            NSLog(@"ObjectMeta: %@ - %@", key, [result.objectMeta objectForKey:key]);
        }
    } else {
        NSLog(@"head object failed, error: %@", task.error);
    }
    return nil;
}];
```

10.8.4. List objects

This topic describes how to list objects in a bucket, including all objects, a specified number of objects, and objects whose names contain a specified prefix.

 **Note** For more information about how to list objects, see [GetBucket \(ListObjects\)](#).

List a specified number of objects

The following code provides an example on how to list 20 objects in a bucket named examplebucket:

```
OSSGetBucketRequest * getBucket = [OSSGetBucketRequest new];
// Specify the bucket name.
getBucket.bucketName = @"examplebucket";
// Specify the maximum number of returned objects. By default, the value of this parameter
is set to 100. The maximum value you can specify is 1000.
getBucket.maxKeys = 20;
OSSTask * getBucketTask = [client getBucket:getBucket];
[getBucketTask continueWithBlock:^id(OSSTask *task) {
    if (!task.error) {
        OSSGetBucketResult * result = task.result;
        NSLog(@"get bucket success!");
        for (NSDictionary * objectInfo in result.contents) {
            NSLog(@"list object: %@", objectInfo);
        }
    } else {
        NSLog(@"get bucket failed, error: %@", task.error);
    }
    return nil;
}];
```

List objects whose names contain a specified prefix

The following code provides an example on how to list objects whose names contain the "file" prefix in a bucket named examplebucket:

```
OSSGetBucketRequest * getBucket = [OSSGetBucketRequest new];
// Specify the bucket name.
getBucket.bucketName = @"examplebucket";
// Specify the prefix.
getBucket.prefix = @"file";
OSSTask * getBucketTask = [client getBucket:getBucket];
[getBucketTask continueWithBlock:^id(OSSTask *task) {
    if (!task.error) {
        OSSGetBucketResult * result = task.result;
        NSLog(@"get bucket success!");
        for (NSDictionary * objectInfo in result.contents) {
            NSLog(@"list object: %@", objectInfo);
        }
    } else {
        NSLog(@"get bucket failed, error: %@", task.error);
    }
    return nil;
}];
```

List objects whose names are alphabetically after the object specified by marker

The following code provides an example on how to list objects whose names are alphabetically after the object named `exampleobject.txt` in a bucket named `examplebucket`:

```
OSSGetBucketRequest * getBucket = [OSSGetBucketRequest new];
// Specify the bucket name.
getBucket.bucketName = @"examplebucket";
// Specify the value of marker. Objects whose names are alphabetically after the object specified by marker are returned.
// If the object specified by marker does not exist in the bucket, objects whose names are alphabetically after the value of marker are returned.
getBucket.marker = @"exampleobject.txt";
OSSTask * getBucketTask = [client getBucket:getBucket];
[getBucketTask continueWithBlock:^id(OSSTask *task) {
    if (!task.error) {
        OSSGetBucketResult * result = task.result;
        NSLog(@"get bucket success!");
        for (NSDictionary * objectInfo in result.contents) {
            NSLog(@"list object: %@", objectInfo);
        }
    } else {
        NSLog(@"get bucket failed, error: %@", task.error);
    }
    return nil;
}];
```

List all objects by page

The following code provides an example on how to list all objects in a bucket named `examplebucket` by page. Up to 20 objects can be listed on each page.

```
interface ... {
    NSString *_marker;
    BOOL _isCompleted;
}
@end
@implementation ...
// List all objects by page.
- (void)getAllObjects {
    do {
        OSSTask *task = [self getObjectList];
        // Wait until NextMarker is returned for the preceding request. Set the marker parameter in the current request to the value of NextMarker returned in the response to the preceding request. You do not need to set marker in the first request.
        // In this example, a loop function is used to list objects by page. Therefore, a request can be sent only after the NextMarker value for the preceding request is returned. You can determine whether to wait for the NextMarker value returned for the preceding request.
        [task waitUntilFinished];
    } while (!_isCompleted);
}
// List objects on a page.
```

```

// List objects on a page.
- (OSSTask *)getObjectList {
    OSSGetBucketRequest * getBucket = [OSSGetBucketRequest new];
    // Specify the bucket name.
    getBucket.bucketName = @"examplebucket";
    // Specify the maximum number of objects that can be listed on each page. By default, the
    // value of this parameter is set to 100. The maximum value you can specify is 1000.
    getBucket.maxKeys = 20;
    // marker is a global attribute.
    getBucket.marker = self.marker;
    OSSTask * getBucketTask = [client getBucket:getBucket];
    [getBucketTask continueWithBlock:^id(OSSTask *task) {
        if (!task.error) {
            OSSGetBucketResult * result = task.result;
            NSLog(@"get bucket success!");
            for (NSDictionary * objectInfo in result.contents) {
                NSLog(@"list object: %@", objectInfo);
            }
            if (result.isTruncated) {
                _marker = result.nextMarker;
            } else {
                _isCompleted = YES;
            }
        } else {
            _isCompleted = YES;
            NSLog(@"get bucket failed, error: %@", task.error);
        }
        return nil;
    }];
    return getBucketTask;
}

@end

@implementation ...
// List all objects by page.
- (void)getAllObjects {
    do {
        OSSTask *task = [self getObjectList];
        // Wait until NextMarker is returned for the preceding request. Set the marker parameter
        // in the current request to the value of NextMarker returned in the response to the preceding
        // request. You do not need to set marker in the first request.
        // In this example, a loop function is used to list objects by page. Therefore, a request
        // can be sent only after the NextMarker value for the preceding request is returned. You
        // can determine whether to wait for the NextMarker value returned for the preceding request.
        [task waitUntilFinished];
    } while (!_isCompleted);
}

// List objects on a page.
- (OSSTask *)getObjectList {
    OSSGetBucketRequest * getBucket = [OSSGetBucketRequest new];
    // Specify the bucket name.
    getBucket.bucketName = @"examplebucket";
    // Specify the maximum number of objects that can be listed on each page. By default, the
    // value of this parameter is set to 100. The maximum value you can specify is 1000.
    getBucket.maxKeys = 20;

```

```

// marker is a global attribute.
getBucket.marker = self.marker;
OSSTask * getBucketTask = [client getBucket:getBucket];
[getBucketTask continueWithBlock:^id(OSSTask *task) {
    if (!task.error) {
        OSSGetBucketResult * result = task.result;
        NSLog(@"get bucket success!");
        for (NSDictionary * objectInfo in result.contents) {
            NSLog(@"list object: %@", objectInfo);
        }
        if (result.isTruncated) {
            _marker = result.nextMarker;
        } else {
            _isCompleted = YES;
        }
    } else {
        _isCompleted = YES;
        NSLog(@"get bucket failed, error: %@", task.error);
    }
    return nil;
}];
return getBucketTask;
}
@end

```

List objects whose names contain a specified prefix by page

The following code provides an example on how to list objects whose names contain the "file" prefix in a bucket named examplebucket by page. Up to 20 objects can be listed on each page.

```

@interface ... {
    NSString *_marker;
    BOOL _isCompleted;
}
@end

@implementation ...
// List all objects by page.
- (void)getAllObjects {
    do {
        OSSTask *task = [self getObjectList];
        // Wait until NextMarker is returned for the preceding request. Set the marker parameter in the current request to the value of NextMarker returned in the response to the preceding request. You do not need to set marker in the first request.
        // In this example, a loop function is used to list objects by page. Therefore, a request can be sent only after the NextMarker value for the preceding request is returned. You can determine whether to wait for the NextMarker value returned for the preceding request.
        [task waitUntilFinished];
    } while (!_isCompleted);
}

// List objects on a page.
- (OSSTask *)getObjectList {
    OSSGetBucketRequest * getBucket = [OSSGetBucketRequest new];
    // Specify the bucket name.

```

```

// Specify the bucket name.
getBucket.bucketName = @"examplebucket";
// Specify the maximum number of objects that can be listed on each page. By default, the
// value of this parameter is set to 100. The maximum value you can specify is 1000.
getBucket.maxKeys = 20;
// Specify the prefix.
getBucket.prefix = @"file";
// marker is a global attribute.
getBucket.marker = self.marker;
OSSTask * getBucketTask = [client getBucket:getBucket];
[getBucketTask continueWithBlock:^(id(OSSTask *task) {
    if (!task.error) {
        OSSGetBucketResult * result = task.result;
        NSLog(@"get bucket success!");
        for (NSDictionary * objectInfo in result.contents) {
            NSLog(@"list object: %@", objectInfo);
        }
        if (result.isTruncated) {
            _marker = result.nextMarker;
        } else {
            _isCompleted = YES;
        }
    } else {
        _isCompleted = YES;
        NSLog(@"get bucket failed, error: %@", task.error);
    }
    return nil;
}]);
return getBucketTask;
}
@end

```

10.8.5. Copy objects

This topic describes how to copy an object within a bucket or across buckets in the same region.

Usage notes

- To copy an object, you must have read permissions on the source object and read and write permissions on the destination bucket.
- The source bucket and destination bucket must be in the same region. For example, objects in a bucket located in the China (Hangzhou) region cannot be copied to another bucket located in the China (Qingdao) region.
- The size of the object that you want to copy cannot exceed 1 GB.

Sample code

The following code provides an example on how to copy an object:

```

OSSCopyObjectRequest * copy = [OSSCopyObjectRequest new];
// Specify the name of the source bucket.
copy.sourceBucketName = @"sourcebucket";
// Specify the full path of the object in the source bucket.
copy.sourceObjectKey = @"dir1/xxxobject.txt";

```

```

copy.sourceObjectKey = @"dir1/srcobject.txt";
// Specify the name of the destination bucket.
copy.bucketName = @"destbucket";
// Specify the full path of the object in the destination bucket.
copy.objectKey = @"dir2/destobject.txt";
NSMutableDictionary *objectMeta = [NSMutableDictionary dictionary];
// Specify the access control list (ACL) of the destination object. In this example, the ACL of the destination object is set to private.
[objectMeta setValue:@"x-oss-object-acl" forKey:@"public-read"];
// Specify the storage class of the destination object. In this example, the storage class of the destination object is set to Standard.
[objectMeta setValue:@"x-oss-storage-class" forKey:@"Standard"];
// Specify whether to overwrite the object with the same name. By default, if x-oss-forbid-overwrite is not specified, existing objects are overwritten by the objects with the same names.
// If x-oss-forbid-overwrite is set to false, existing objects are overwritten by the objects with the same names. If x-oss-forbid-overwrite is set to true, existing objects are not overwritten by the objects with the same names. If an existing object has the same name as the copied object, an error is reported.
[objectMeta setValue:@"x-oss-forbid-overwrite" forKey:@"true"];
// If the ETag value of the source object is the same as the ETag value that you specified in the request, the object is copied.
[objectMeta setValue:@"x-oss-copy-source-if-match" forKey:@"5B3C1A2E053D763E1B002CC607C5***"];
// If the ETag value of the source object is different from the ETag value that you specified in the request, the object is copied.
[objectMeta setValue:@"x-oss-copy-source-if-none-match" forKey:@"5B3C1A2E053D763E1B002CC607C5*****"];
// If the object is modified at 07:01:56.000, December 09, 2021 or earlier in UTC, the object is copied.
[objectMeta setValue:@"x-oss-copy-source-if-unmodified-since" forKey:@"2021-12-09T07:01:56.000Z"];
// If the object is modified after 07:01:56.000, December 15, 2021 in UTC, the object is copied.
[objectMeta setValue:@"x-oss-copy-source-if-modified-since" forKey:@"2021-12-15T07:01:56.000Z"];
// Copy the metadata of the source object to the destination object.
[objectMeta setValue:@"x-oss-metadata-directive" forKey:@"COPY"];
// Copy the tags of the source object to the destination object.
[objectMeta setValue:@"x-oss-tagging-directive" forKey:@"Copy"];
// Specify the server-side encryption algorithm that is used to encrypt the destination object when OSS creates the destination object.
[objectMeta setValue:@"x-oss-server-side-encryption" forKey:@"KMS"];
// Specify the customer master key (CMK) that is managed by Key Management Service (KMS). This parameter takes effect only when you set x-oss-server-side-encryption to KMS.
[objectMeta setValue:@"x-oss-server-side-encryption-key-id" forKey:@"9468da86-3509-4f8d-a61e-6eableac****"];
copy.objectMeta = objectMeta;
OSSTask * task = [client copyObject:copy];
[task continueWithBlock:^(id(OSSTask *task) {
    if (!task.error) {
        NSLog(@"copy object success!");
    } else {
        NSLog(@"copy object failed, error: %@", task.error);
    }
}

```

```
,  
    return nil;  
}];
```

References

For more information about the API operation that you can call to copy an object, see [CopyObject](#).

10.8.6. Delete objects

This topic describes how to delete a single object or batch delete objects.



Warning Deleted objects cannot be recovered. Exercise caution when you delete objects.

Usage notes

To delete an object, you must have write permissions on the bucket in which the object is stored.

Delete a single object

The following code provides an example on how to delete an object named `exampleobject.txt` from a bucket named `examplebucket`:

```
OSSDeleteObjectRequest * delete = [OSSDeleteObjectRequest new];  
// Specify the bucket name.  
delete.bucketName = @"examplebucket";  
// Specify the full path of the object that you want to delete. The full path of the object  
cannot contain bucket names.  
delete.objectKey = @"exampleobject.txt";  
OSSTask * deleteTask = [client deleteObject:delete];  
[deleteTask continueWithBlock:^id(OSSTask *task) {  
    if (!task.error) {  
        // ...  
    }  
    return nil;  
}];  
// [deleteTask waitUntilFinished];
```

Delete multiple objects in a batch

You can batch delete up to 1,000 objects each time.

The result can be returned in the following two modes. Select a return mode based on your requirements.

- **verbose**: If `quiet` is set to `NO`, a list of all deleted objects is returned.
- **quiet**: If `quiet` is not specified or is set to `YES`, a list of objects that failed to be deleted is returned. This is the default return mode.

The following code provides an example on how to delete multiple specified objects from a bucket named `examplebucket` and return the result in the `quiet` mode:


```

OSSDeleteMultipleObjectsRequest *request = [OSSDeleteMultipleObjectsRequest new];
// Specify the bucket name.
request.bucketName = @"examplebucket";
// Specify the full paths of the objects that you want to delete. The full paths of the objects cannot contain bucket names.
request.keys = @[@"exampleobject.txt", @"testfolder/sampleobject.txt"];
// Set quiet to YES to return only a list of objects that failed to be deleted.
request.quiet = YES;
OSSTask * deleteMultipleObjectsTask = [client deleteMultipleObjects:request];
[deleteMultipleObjectsTask continueWithBlock:^(OSSTask *task) {
    if (!task.error) {
        OSSDeleteMultipleObjectsResult *result = task.result;
        NSLog(@"delete objects: %@", result.deletedObjects);
    } else {
        NSLog(@"delete objects failed, error: %@", task.error);
    }
    return nil;
}];

```

10.8.7. Convert the storage classes of objects

Object Storage Service (OSS) supports the Standard, Infrequent Access (IA), Archive, and Cold Archive storage classes. You can call the CopyObject operation to overwrite the source object. This way, the storage class of the destination object is specified as the storage class of the object.

Sample code

- Convert the storage class of an object from Standard or IA to Archive

The following code provides an example on how to convert the storage class of an object named `exampleobject.txt` from Standard or IA to Archive. In this example, the object is stored in the root directory of the bucket named `examplebucket`.

```

OSSCopyObjectRequest * copy = [OSSCopyObjectRequest new];
copy.sourceBucketName = @"examplebucket";
copy.sourceobjectKey = @"exampleobject.txt";
copy.bucketName = @"examplebucket";
copy.objectKey = @"exampleobject.txt";
// Set the storage class of the object named exampleobject.txt to Archive.
copy.objectMeta = @{@"x-oss-storage-class" : @"Archive"};
OSSTask * task = [client copyObject:copy];
[task continueWithBlock:^(OSSTask *task) {
    if (!task.error) {
        NSLog(@"copy object success!");
    } else {
        NSLog(@"copy object failed, error: %@", task.error);
    }
    return nil;
}];

```

- Convert the storage class of an object from Archive or Cold Archive to IA

The following code provides an example on how to convert the storage class of an object named `exampleobject.txt` from Archive or Cold Archive to IA. In this example, the object is stored in the root directory of the bucket named `examplebucket`.

```
NSString *bucketName = @"examplebucket";
NSString *objectKey = @"exampleobject.txt";
// Check whether the storage class of the object is Archive or Cold Archive. If the storage class of the object is Archive or Cold Archive, you must restore the object before you can change the storage class of the object.
__block bool isArchived = false;
OSSHeadObjectRequest *headRq = [OSSHeadObjectRequest new];
headRq.bucketName = bucketName;
headRq.objectKey = objectKey;
[[[client headObject:headRq] continueWithBlock:^id _Nullable(OSSTask * _Nonnull task) {
    if (!task.error) {
        OSSHeadObjectResult *headRs = task.result;
        isArchived = [headRs.httpResponseHeaderFields[@"x-oss-storage-class"] isEqualToString:@"Archive"];
    } else {
        NSLog(@"head object fail! error: %@", task.error);
    }
    return nil;
}] waitUntilFinished];
if (isArchived) {
    // Restore the object.
    OSSRestoreObjectRequest *restoreRequest = [OSSRestoreObjectRequest new];
    restoreRequest.bucketName = bucketName;
    restoreRequest.objectKey = objectKey;
    [[[client restoreObject:restoreRequest] continueWithBlock:^id _Nullable(OSSTask * _Nonnull task) {
        if (!task.error) {
            NSLog(@"restore object success!");
        } else {
            NSLog(@"restore object fail! error: %@", task.error);
        }
        return nil;
    }] waitUntilFinished];
    // Wait until the object is restored.
    __block bool isRestored = false;
    do {
        sleep(1);
        headRq = [OSSHeadObjectRequest new];
        headRq.bucketName = bucketName;
        headRq.objectKey = objectKey;
        [[[client headObject:headRq] continueWithBlock:^id _Nullable(OSSTask * _Nonnull task) {
            if (!task.error) {
                OSSHeadObjectResult *headRs = task.result;
                isRestored = (headRs.httpResponseHeaderFields[@"x-oss-restore"] != nil &&
                    ![headRs.httpResponseHeaderFields[@"x-oss-restore"] isEqualToString:@"ongoing-request=\\"true\\""]);
            } else {
                NSLog(@"head object fail! error: %@", task.error);
            }
        }
    }
}
```

```
        return nil;
    }
    } waitUntilFinished];
} while (!isRestored);
}

// Set the storage class of the object named exampleobject.txt to IA.
OSSCopyObjectRequest *copy = [OSSCopyObjectRequest new];
copy.bucketName = bucketName;
copy.objectKey = objectKey;
copy.sourceBucketName = bucketName;
copy.sourceObjectKey = objectKey;
copy.objectMeta = @{@"x-oss-storage-class": @"IA"};
[[[client copyObject:copy] continueWithBlock:^id(OSSTask *task) {
    if (!task.error) {
        NSLog(@"copy object success!");
    } else {
        NSLog(@"copy object failed, error: %@", task.error);
    }
    return nil;
}] waitUntilFinished];
```

References

For more information about the API operation that you can call to convert the storage class of an object, see [CopyObject](#).

10.8.8. Prevent objects from being overwritten by objects that have the same names

By default, if you upload an object that has the same name as an existing object on which you have access permissions, the existing object is overwritten by the uploaded object. This topic describes how to configure the `x-oss-forbid-overwrite` request header to prevent objects from being overwritten by objects with the same names when you copy objects or perform simple upload or multipart upload.

Simple upload

The following code provides an example on how to prevent objects from being overwritten by objects that have the same names in simple upload:

```

OSSPutObjectRequest * put = [OSSPutObjectRequest new];
// Specify the name of the bucket. Example: examplebucket. For more information about the naming conventions for buckets, see Bucket naming conventions.
put.bucketName = @"examplebucket";
// Specify the full path of the object. The full path of the object cannot contain the bucket name. Example: exampledir/exampleobject.txt. For more information about the naming conventions for objects, see Object naming conventions.
put.objectKey = @"exampledir/exampleobject.txt";
// Specify the full path of the local file to upload.
put.uploadingFileURL = [NSURL fileURLWithPath:@"/storage/emulated/0/oss/examplefile.txt"];
// Specify whether to overwrite the object with the same name.
// By default, if x-oss-forbid-overwrite is not specified, existing objects are overwritten by the objects with the same names.
// If x-oss-forbid-overwrite is set to false, existing objects are overwritten by the objects with the same names.
// If x-oss-forbid-overwrite is set to true, existing objects are not overwritten by the objects with the same names. If an existing object has the same name as the uploaded object, an error is reported.
put.objectMeta = @{@"x-oss-forbid-overwrite": @"true"};
// (Optional) Specify the upload progress.
put.uploadProgress = ^(int64_t bytesSent, int64_t totalByteSent, int64_t totalBytesExpectedToSend) {
    // Specify the number of bytes that are being uploaded, the total number of bytes that are uploaded, and the total number of bytes to be uploaded.
    NSLog(@"%lld, %lld, %lld", bytesSent, totalByteSent, totalBytesExpectedToSend);
};
OSSTask * putTask = [client putObject:put];
[putTask continueWithBlock:^id(OSSTask *task) {
    if (!task.error) {
        NSLog(@"upload object success!");
    } else {
        NSLog(@"upload object failed, error: %@", task.error);
    }
    return nil;
}];

```

Copy objects

The following code provides an example on how to prevent existing objects from being overwritten by newly copied objects that have the same names:

```

OSSCopyObjectRequest * copy = [OSSCopyObjectRequest new];
// Specify the name of the source bucket.
copy.sourceBucketName = @"srcbucket";
// Specify the full path of the object in the source bucket.
copy.sourceObjectKey = @"dir1/srcobject.txt";
// Specify the name of the destination bucket.
copy.bucketName = @"destbucket";
// Specify the full path of the object in the destination bucket.
copy.objectKey = @"dir2/destobject.txt";
// Specify whether to overwrite the object with the same name.
// By default, if x-oss-forbid-overwrite is not specified, existing objects are overwritten
by the objects with the same names.
// If x-oss-forbid-overwrite is set to false, existing objects are overwritten by the objects
with the same names.
// If x-oss-forbid-overwrite is set to true, existing objects are not overwritten by the objects
with the same names. If an existing object has the same name as the copied object, an
error is reported.
copy.objectMeta = @{@"x-oss-forbid-overwrite": @"true"};
OSSTask * task = [client copyObject:copy];
[task continueWithBlock:^id(OSSTask *task) {
    if (!task.error) {
        NSLog(@"copy object success!");
    } else {
        NSLog(@"copy object failed, error: %@", task.error);
    }
    return nil;
}];

```

Multipart upload

The following code provides an example on how to prevent objects from being overwritten by newly uploaded objects that have the same names in multipart upload:

```

__block NSString * uploadId = nil;
__block NSMutableArray * partInfos = [NSMutableArray new];
// Specify the name of the bucket. Example: examplebucket.
NSString * uploadToBucket = @"examplebucket";
// Specify the full path of the object. The full path of the object cannot contain the bucket
name. Example: exampledir/exampleobject.txt.
NSString * uploadObjectKey = @"exampledir/exampleobject.txt";
OSSInitMultipartUploadRequest * init = [OSSInitMultipartUploadRequest new];
init.bucketName = uploadToBucket;
init.objectKey = uploadObjectKey;
// Specify whether to overwrite the object with the same name.
// By default, if x-oss-forbid-overwrite is not specified, existing objects are overwritten
by the objects with the same names.
// If x-oss-forbid-overwrite is set to false, existing objects are overwritten by the objects
with the same names.
// If x-oss-forbid-overwrite is set to true, existing objects are not overwritten by the objects
with the same names. If an existing object has the same name as the uploaded object,
an error is reported.
init.objectMeta = @{@"x-oss-forbid-overwrite": @"true"};
// The response to multipartUploadInit contains the upload ID. The upload ID is the unique
ID of the multipart upload task. You can use the upload ID to cancel or query the multipart

```

```

ID of the multipart upload task. You can use the upload ID to cancel or query the multipart
upload task.
OSSTask * initTask = [client multipartUploadInit:init];
[initTask waitUntilFinished];
if (!initTask.error) {
    OSSInitMultipartUploadResult * result = initTask.result;
    uploadId = result.uploadId;
} else {
    NSLog(@"multipart upload failed, error: %@", initTask.error);
    return;
}
// Specify the full path of the local file to upload.
NSString * filePath = @"/storage/emulated/0/oss/examplefile.txt";
// Query the size of the file to upload.
uint64_t fileSize = [[NSFileManager defaultManager] attributesOfItemAtPath:filePath error:
nil] fileSize];
// Specify the part number for each part. The number starts from 1. Each part to upload has
a part number. Valid values: 1 to 10000.
int chunkCount = *;
// Specify the size of each part. Unit: bytes. Valid values: 100 KB to 5 GB.
uint64_t offset = fileSize/chunkCount;
for (int i = 1; i <= chunkCount; i++) {
    OSSUploadPartRequest * uploadPart = [OSSUploadPartRequest new];
    uploadPart.bucketName = uploadToBucket;
    uploadPart.objectKey = uploadObjectkey;
    uploadPart.uploadId = uploadId;
    uploadPart.partNumber = i; // part number start from 1
    NSFileHandle* readHandle = [NSFileHandle fileHandleForReadingAtPath:filePath];
    [readHandle seekToFileOffset:offset * (i -1)];
    NSData* data = [readHandle readDataOfLength:offset];
    uploadPart.uploadPartData = data;
    OSSTask * uploadPartTask = [client uploadPart:uploadPart];
    [uploadPartTask waitUntilFinished];
    if (!uploadPartTask.error) {
        OSSUploadPartResult * result = uploadPartTask.result;
        uint64_t fileSize = [[NSFileManager defaultManager] attributesOfItemAtPath:uploadP
art.uploadPartFileURL.absoluteString error:nil] fileSize];
        [partInfos addObject:[OSSPartInfo partInfoWithPartNum:i eTag:result.eTag size:fileS
ize]];
    } else {
        NSLog(@"upload part error: %@", uploadPartTask.error);
        return;
    }
}
OSSCompleteMultipartUploadRequest * complete = [OSSCompleteMultipartUploadRequest new];
complete.bucketName = uploadToBucket;
complete.objectKey = uploadObjectkey;
complete.uploadId = uploadId;
complete.partInfos = partInfos;
// Specify whether to overwrite the object with the same name when the multipart upload tas
k is complete.
// By default, if x-oss-forbid-overwrite is not specified, existing objects are overwritten
by the objects with the same names.
// If x-oss-forbid-overwrite is set to false, existing objects are overwritten by the objec
ts with the same names.

```

```

// If x-oss-forbid-overwrite is set to true, existing objects are not overwritten by the ob
jects with the same names. If an existing object has the same name as the uploaded object,
an error is reported.
complete.completeMetaHeader = @{@"x-oss-forbid-overwrite": @"true"};
OSSTask * completeTask = [client completeMultipartUpload:complete];
[[completeTask continueWithBlock:^id(OSSTask *task) {
    if (!task.error) {
        NSLog(@"multipart upload success!");
    } else {
        NSLog(@"multipart upload error: %@", task.error);
    }
    return nil;
}] waitUntilFinished];

```

References

- For more information about the API operation that you can call to perform simple upload, see [Put Object](#).
- For more information about the API operation that you can call to copy an object, see [Copy Object](#).
- The following API operations are required to perform multipart upload:
 - The API operation that you can call to initiate a multipart upload task. For more information, see [InitiateMultipart Upload](#).
 - The API operation that you can call to upload data by part. For more information, see [UploadPart](#).
 - The API operation that you can call to complete a multipart upload task. For more information, see [CompleteMultipart Upload](#).

10.8.9. Manage symbolic links

Create a symbolic link

The following code provides an example on how to create a symbolic link named `examplesymlink` that points to an object named `exampleobject.txt` in a bucket named `examplebucket`:

```

OSSPutSymlinkRequest *request = [OSSPutSymlinkRequest new];
// Specify the bucket name.
request.bucketName = @"examplebucket";
// Specify the name of the symbolic link that you want to create.
request.objectKey = @"examplesymlink";
// Specify the full path of the object to which the symbolic link points. The full path of
the object cannot contain bucket names.
request.targetObjectName = @"exampleobject.txt";
OSSTask *putSymlinkTask = [client putSymlink:request];
[putSymlinkTask continueWithBlock:^id _Nullable(OSSTask * _Nonnull task) {
    if (!task.error) {
        NSLog(@"put symlink success");
    } else {
        NSLog(@"put symlink failed, error: %@", task.error);
    }
    return nil;
}];

```

Query the name of the object to which a symbolic link points

To query the object to which a symbolic link points, you must have read permissions on the symbolic link.

The following code is used to obtain the name of the target file pointed to by the soft link `examplesymlink` in the `examplebucket` bucket.

```
OSSGetSymlinkRequest *request = [OSSGetSymlinkRequest new];
// Specify the bucket name.
request.bucketName = @"examplebucket";
// Specify the name of the symbolic link that you want to create.
request.objectKey = @"examplesymlink";
OSSTask *getSymlinkTask = [client getSymlink:request];
[getSymlinkTask continueWithBlock:^id _Nullable(OSSTask * _Nonnull task) {
    if (!task.error) {
        OSSGetSymlinkResult *result = task.result;
        NSLog(@"get symlink: %@", result.httpResponseHeaderFields[@"x-oss-symlink-target"]);
    } else {
        NSLog(@"get symlink failed, error: %@", task.error);
    }
    return nil;
}];
```

For more information about obtaining soft links, see [GetSymlink](#).

10.8.10. Restore Archive objects

You must restore an Archive object before you read the object. This topic describes how to restore an Archive object.

The following code provides an example on how to restore an Archive object named `exampleobject.txt`:

```
OSSRestoreObjectRequest *request = [OSSRestoreObjectRequest new];
// Specify the bucket name.
request.bucketName = @"examplebucket";
// Specify the full path of the Archive object. The full path of the object cannot contain
bucket names.
request.objectKey = @"exampleobject.txt";
OSSTask *restoreObjectTask = [client restoreObject:request];
[restoreObjectTask continueWithBlock:^id _Nullable(OSSTask * _Nonnull task) {
    if (!task.error) {
        NSLog(@"restore object success");
    } else {
        NSLog(@"restore object failed, error: %@", task.error);
    }
    return nil;
}];
```

For more information about the Archive storage class, see [Overview](#). For more information about the parameters that you can configure when you restore an Archive object, see [RestoreObject](#).

10.8.11. Manage object ACLs

You can set the ACL of an object to one of the following value: Inherited from bucket, Private, Public Read, and Public Read/Write. This topic describes how to manage the ACL of an object.

Object ACLs

Permission	Description	Value
Inherited from bucket	The ACL of an object complies with that of its bucket.	default
Private	Only the object owner or authorized users have read and write permissions on the object.	private
Public Read	Only the object owner or authorized users can write the object. Other users, including anonymous users can only read the object. Exercise caution when you use this operation.	public-read
Public Read/Write	Any users, including anonymous users can read and write the object. Exercise caution when you use this operation.	public-read-write

The ACL privileges of objects take precedence over those of buckets. For example, if the ACL of a bucket is private and the ACL of an object that is stored in this bucket is public, all users can read and write the object. If the ACL of an object is not configured, the ACL of the object is the same as that of the bucket in which the object is stored.

Configure the ACL of an object

The following code provides an example on how to set the ACL of an object named `exampleobject.txt` in a bucket named `examplebucket` to private:

```
OSSPutObjectACLRequest *request = [OSSPutObjectACLRequest new];
// Specify the bucket name.
request.bucketName = @"examplebucket";
// Specify the complete path of the object, which does not include the bucket name.
request.objectKey = @"exampleobject.txt";
/**
 * Configure the object ACL.
 * public-read
 * private
 * public-read-write
 * default: inherited from the bucket
 */
request.acl = @"private";
OSSTask * putObjectACLTask = [client putObjectACL:request];
[putObjectACLTask continueWithBlock:^id(OSSTask *task) {
    if (!task.error) {
        NSLog(@"put object ACL success!");
    } else {
        NSLog(@"put object ACL failed, error: %@", task.error);
    }
    return nil;
}];
```

For more information about object ACLs, see [PutObjectACL](#).

Query the ACL of an object

The following code provides an example on how to query the ACL of an object named `exampleobject.txt` in a bucket named `examplebucket`:

```
OSSGetObjectACLRequest *request = [OSSGetObjectACLRequest new];
// Specify the bucket name.
request.bucketName = @"examplebucket";
// Specify the complete path of the object, which does not include the bucket name.
request.objectName = @"exampleobject.txt";
OSSTask * getObjectACLTask = [client getObjectACL:request];
[getObjectACLTask continueWithBlock:^id(OSSTask *task) {
    if (!task.error) {
        OSSGetObjectACLResult *result = task.result;
        NSLog(@"objectACL: %@", result.grant);
    } else {
        NSLog(@"get object ACL failed, error: %@", task.error);
    }
    return nil;
}];
```

For more information about how to query object ACLs, see [GetObjectACL](#).

10.9. Data security

OSS SDK for iOS provides data integrity checks to ensure data security in uploads and downloads.

Data integrity checks in uploads

Due to the complex environment of mobile networks, an error may occur when data is transferred between the client and server. OSS provides end-to-end data integrity checks based on MD5 hashes and 64-bit CRC values.

- MD5 verification

If you specify the Content-MD5 header in the request when you upload an object, OSS performs MD5 verification on the object to ensure data integrity. The object is uploaded only when the MD5 hash of the received object is consistent with the Content-MD5 value specified in the request.

```
OSSPutObjectRequest * request = [OSSPutObjectRequest new];
request.bucketName = BUCKET_NAME;
...
request.contentMd5 = [base64Md5ForFilePath];
```

- CRC verification

The 64-bit CRC value of an object can be calculated when the object is being uploaded. The following code provides an example on how to perform CRC when you upload an object:

```
// Construct a request to upload the object.
OSSPutObjectRequest * request = [OSSPutObjectRequest new];
request.bucketName = BUCKET_NAME;
///....
request.crcFlag = OSSRequestCRCOpen;
// Enable CRC verification.
OSSTask * task = [_client putObject:request];
[[task continueWithBlock:^(id(OSSTask *task) {
    // If the data is inconsistent during upload, the CRC verification fails and the OSSC
    lientErrorCodeInvalidCRC error is returned.
    XCTAssertNil(task.error);
    return nil;
}] waitUntilFinished];
```

Data integrity checks in downloads

OSS SDK for iOS provides end-to-end data integrity checks based on 64-bit CRC values in downloads.

If CRC verification is enabled, OSS automatically checks data integrity after it reads data from a stream.

The following code provides an example on how to enable CRC verification in downloads:

```

OSSGetObjectRequest * request = [OSSGetObjectRequest new];
request.bucketName = BUCKET_NAME;
// Enable CRC verification.
request.crcFlag = OSSRequestCRCOpen;
OSSTask * task = [testProxyClient getObject:request];
[[task continueWithBlock:^(id(OSSTask *task) {
    // If an error occurs during transmission after CRC verification is enabled, OSSClientError
    rrorCodeInvalidCRC is returned.
    XCTAssertNil(task.error);
    return nil;
}]) waitUntilFinished];
// If the onDataReceived block is set, you must check whether the CRC value is the same.
OSSGetObjectRequest * request = [OSSGetObjectRequest new];
request.bucketName = BUCKET_NAME;
request.crcFlag = OSSRequestCRCOpen;
....
__block uint64_t localCrc64 = 0;
NSMutableData *receivedData = [NSMutableData data];
request.onReceiveData = ^(NSData *data) {
    if (data)
    {
        NSMutableData *mutableData = [data mutableCopy];
        void *bytes = mutableData.mutableBytes;
        localCrc64 = [OSSUtil crc64ecma:localCrc64 buffer:bytes length:data.length];
        [receivedData appendData:data];
    }
};
__block uint64_t remoteCrc64 = 0;
OSSTask * task = [_client getObject:request];
[[task continueWithBlock:^(id(OSSTask *task) {
    XCTAssertNil(task.error);
    OSSGetObjectResult *result = task.result;
    if (result.remoteCRC64ecma)
    {
        NSScanner *scanner = [NSScanner scannerWithString:result.remoteCRC64ecma];
        [scanner scanUnsignedLongLong:&remoteCrc64];
        if (remoteCrc64 == localCrc64)
        {
            NSLog(@"crc64 verification successful.") ;
        }
        else
        {
            NSLog(@"crc64 verification failed!") ;
        }
    }
}]) waitUntilFinished];

```

10.10. Signed URLs

OSS SDK for iOS supports the signed URLs that have a validity period or public URLs. This way, URLs can be forwarded to a third party for authorized access.

Generate a signed URL that has validity period for private resources

If the bucket or object ACL is Private, you must call the following API operations to obtain the signed URL:

```
NSString * constrainURL = nil;
// sign constrain url
OSSTask * task = [client presignConstrainURLWithBucketName:@"<bucket name>"
                                             withObjectKey:@"<object key>"
                                             withExpirationInterval: 30 * 60];

if (!task.error) {
    constrainURL = task.result;
} else {
    NSLog(@"error: %@", task.error);
}
```

Generate a signed public URL

If the bucket or object ACL is Public Read, you must call the following API operations to obtain the URL of the publicly accessible object:

```
NSString * publicURL = nil;
// sign public url
task = [client presignPublicURLWithBucketName:@"<bucket name>"
                                   withObjectKey:@"<object key>"];

if (!task.error) {
    publicURL = task.result;
} else {
    NSLog(@"sign url error: %@", task.error);
}
```

10.11. Authorized access

Object Storage Service (OSS) SDK for iOS provides two authentication modes to ensure the data security of mobile devices: Security Token Service (STS) authentication mode and self-signed mode.

Overview

When you use the STS authentication mode or self-signed mode, ensure that the callback function that you implement returns results for Token and Signature. If you must obtain the token and signature from the app server by sending a request in the callback function, we recommend that you call the synchronous API operations included in the network library. The callback function is run in the child thread of the request generated by the SDK and does not block the main thread.

STS authentication mode

 **Note** To use the STS authentication mode, you must first activate Alibaba Cloud Resource Access Management (RAM).

You can use Alibaba Cloud STS to authorize temporary access to OSS. STS is a web service that provides temporary access tokens for users. You can use STS to grant a set of temporary access credentials that have a custom validity period and custom permissions to a third-party application or a RAM user managed by you. For more information about STS, see [What is STS?](#)

STS provides the following benefits:

- You need only to generate an access token and send the access token to a third-party application, instead of exposing your AccessKey pair to the third-party application. You can define the access permissions and validity period of this token.
- The token automatically expires after the validity period. Therefore, you do not need to manually revoke the access permissions of a token.

 **Note** For more information about how to use STS, see [Use a temporary credential provided by STS to access OSS](#) in OSS User Guide. You can call the [AssumeRole](#) operation or use STS SDKs for various programming languages to obtain temporary access credentials. For more information, see [STS SDK overview](#). The temporary access credentials consist of an AccessKey pair and a security token. The AccessKey pair contains an AccessKey ID and an AccessKey secret. The minimum validity period of temporary access credentials is 900 seconds. The maximum validity period of temporary access credentials is the maximum session duration specified for the current role. For more information, see [Specify the maximum session duration for a RAM role](#).

- Configure an STS token

You can obtain an STS token in the app by a method, such as sending a request to the app server, and use the STS token to initialize the SDK. If you use this method, you must pay attention to the validity period of the STS token. When the STS token is about to expire, you must update the new STS token for the SDK.

The following code provides an example on how to use an STS token to initialize the SDK:

 **Note** To use `OSSAuthCredentialProvider` to initialize OSS SDK for iOS, see [Initialization](#).

```
id<OSSCredentialProvider> credential = [[OSSStsTokenCredentialProvider alloc] initWithAccessKeyId:@"<StsToken.AccessKeyId>" secretKeyId:@"<StsToken.SecretKeyId>" securityToken:@"<StsToken.SecurityToken>"];
client = [[OSSClient alloc] initWithEndpoint:endpoint credentialProvider:credential];
```

When the STS token is about to expire, you can create a new `OSSClient` or update `CredentialProvider` by using the following method:

```
id<OSSCredentialProvider> credential = [[OSSStsTokenCredentialProvider alloc] initWithAccessKeyId:@"<StsToken.AccessKeyId>" secretKeyId:@"<StsToken.SecretKeyId>" securityToken:@"<StsToken.SecurityToken>"];
client = [[OSSClient alloc] initWithEndpoint:endpoint credentialProvider:credential];
```

- Obtain an STS token by implementing callback

If you want the SDK to automatically update the STS token, you must implement callback in your app. The app uses the callback to obtain a federation token (STS token) and returns the token to the SDK. The SDK uses the STS token to generate signatures. When the STS token needs to be updated, the SDK calls the callback to obtain a new token.

```
id<OSSCredentialProvider> credential = [[OSSFederationCredentialProvider alloc] initWithFederationTokenGetter:^OSSFederationToken * {
    // Implement a method to obtain a FederationToken and return it as an OSSFederationToken object.
    // If the FederationToken is not obtained, nil is returned.
    OSSFederationToken * token;
    // Obtain a FederationToken from your server.
    ...
    return token;
}];
client = [[OSSClient alloc] initWithEndpoint:endpoint credentialProvider:credential];
```

 **Note** Additionally, if you obtain all fields required to generate a token in other ways, you can also return these fields by using the callback. In this case, you must manually update the token, and then reconfigure OSSCredentialProvider of the OSSClient instance.

Examples:

The URL of the server from which you obtain the token is `http://localhost:8080/distribute-token.json`. If you access this URL, the following similar data is returned:

```
{
    "StatusCode": 200,
    "AccessKeyId": "STS.iA645eTOXEqP3cg3****",
    "AccessKeySecret": "rV3VQrpFQ4BsyHSAvi5NVLpPIVffDJv4LojU****",
    "Expiration": "2015-11-03T09:52:59Z",
    "SecurityToken": "CAES7QIIARKAAZPlqaN9ILiQZPS+JDkS/GSZN45RLx4YS/p3OgaUC+oJl3XS1bJ7StKpQ****"
}
```

The following code provides an example on how to implement

```
OSSFederationCredentialProvider :
```

```

id<OSSCredentialProvider> credential2 = [[OSSFederationCredentialProvider alloc] initWith
FederationTokenGetter:^OSSFederationToken * {
    // Create a request to access your service server.
    NSURL * url = [NSURL URLWithString:@"http://localhost:8080/distribute-token.json"];
    // Use a request to set the parameters required by your server.
    NSURLRequest * request = [NSURLRequest requestWithURL:url];
    OSSTaskCompletionSource * tcs = [OSSTaskCompletionSource taskCompletionSource];
    NSURLSession * session = [NSURLSession sharedSession];
    // Send the request.
    NSURLSessionTask * sessionTask = [session dataTaskWithRequest:request
                                completionHandler:^(NSData *data, NSURLRe
sponse *response, NSError *error) {
        if (error) {
            [tcs setError:error];
            return;
        }
        [tcs setResult:data];
    }];

    [sessionTask resume];
    // Wait until the response to the request is returned.
    [tcs.task waitUntilFinished];
    // Parse the returned results.
    if (tcs.task.error) {
        NSLog(@"get token error: %@", tcs.task.error);
        return nil;
    } else {
        // The returned data is in the JSON format. Parse the data to obtain the values o
f the fields of the token.
        NSDictionary * object = [NSJSONSerialization JSONObjectWithData:tcs.task.result
                                options:kNilOptions
                                error:nil];

        OSSFederationToken * token = [OSSFederationToken new];
        token.tAccessKey = [object objectForKey:@"AccessKeyId"];
        token.tSecretKey = [object objectForKey:@"AccessKeySecret"];
        token.tToken = [object objectForKey:@"SecurityToken"];
        token.expirationTimeInGMTFormat = [object objectForKey:@"Expiration"];
        NSLog(@"get token: %@", token);
        return token;
    }
}];

```

Self-signed mode

You can perform the following operations to save the AccessKey ID and AccessKey secret on your own server:

1. Obtain the string-to-sign from the client and send the string to your own server.
 - i. Use the `signContent` method of `OSSCustomSignerCredentialProvider` provided by OSS SDK for iOS to obtain the string-to-sign when you create the request.
 - ii. Send the string-to-sign to your own server.
2. Sign the string on your own server and return the signed string to the client.

- i. Use the specified signature algorithm to sign the string. For more information about the signature algorithm, see [Include signatures in the Authorization header](#).

The signature is in the following format: `signature = "OSS " + AccessKeyId + ":" + base64(hmac-sha1(AccessKeySecret, content))`, in which content is the string that is concatenated based on the request parameters. Sample code:

```
id<OSSCredentialProvider> credential = [[OSSCustomSignerCredentialProvider alloc] initWithImplementedSigner:^(NSString *(NSString *contentToSign, NSError *__autoreleasing NSError *) {
    // Use the specified signature algorithm to sign a string, concatenate your AccessKey ID to the signed string, and then return the final string.
    // Send the signed string to your own server and return the signature.
    // If the string fails to be signed, the server returns nil with an error message.
    NSString *signature = [OSSUtil calculateBase64Sha1WithData:contentToSign withSecret:@"<your accessKeySecret>"]; // In this example, the string is signed on the client by using the tool provided by OSS SDK for iOS. We recommend that you sign the string on your business server.
    if (signature != nil) {
        *error = nil;
    } else {
        *error = [NSError errorWithDomain:@"<your domain>" code:-1001 userInfo:@{@"<your error info>"}];
        return nil;
    }
    return [NSString stringWithFormat:@"OSS %@:%@", @"<your accessKeyId>", signature];
}];
client = [[OSSClient alloc] initWithEndpoint:endpoint credentialProvider:credential];
```

- ii. Return the signed string to the client.
3. Send the signed string on the client to the OSS server for authentication.

10.12. IMG

OSS Image Processing (IMG) is a secure, cost-effective, and highly reliable image processing service that can process large amounts of data. After you upload source images to OSS, you can call RESTful API operations to process the images anytime, anywhere, and on any Internet device.

For more information about the IMG parameters, see [Parameters](#).

Usage

You can use standard HTTP GET requests to access IMG. All processing parameters are encoded in the query string of a URL.

If the ACL of an image object is private, only authorized users can access the image object.

- Anonymous access

```

OSSTask * task = [_client presignPublicURLWithBucketName:_publicBucketName
                    withObjectKey:@"hasky.jpeg"
                    withParameters:@{@"x-oss-process": @"image/resize,w_50"}];

```

- Authorized access

When you process images by using OSS SDKs, you can call the `request.setxOssProcess()` operation to configure IMG parameters in the following code. Examples:

```

OSSGetObjectRequest * request = [OSSGetObjectRequest new];
request.bucketName = @"<bucketName>";
request.objectKey = @"example.jpg";
// Process the image.
request.xOssProcess = @"image/resize,m_lfit,w_100,h_100";
OSSTask * getTask = [client getObject:request];
[getTask continueWithBlock:^(OSSTask *task) {
    if (! task.error) {
        NSLog(@"download image success!") ;
        OSSGetObjectResult * getResult = task.result;
        NSLog(@"download image data: %@", getResult.downloadedData);
    } else {
        NSLog(@"download object failed, error: %@", task.error);
    }
    return nil;
}];
// [getTask waitUntilFinished];
// [request cancel];

```

- Access by using SDKs

```

OSSGetObjectRequest * request = [OSSGetObjectRequest new];
request.bucketName = @"bucket-name";    // Specify the bucket name.
request.objectKey = @"image-name";      // Specify the name of the image object.
request.xOssProcess = @"image/resize,m_lfit,w_100,h_100"; // Scale the image to a fixed
width and height of 100.
OSSTask * task = [ossClient getObject:request];

```

Persistent storage of processed images

The following code provides an example on how to save processed images:

```

OSSImagePersistRequest *request = [OSSImagePersistRequest new];
request.fromBucket = _privateBucketName;
request.fromObject = OSS_IMAGE_KEY;
request.toBucket = _privateBucketName;
request.toObject = @"image_persist_key";
request.action = @"image/resize,w_100";
//request.action = @"resize,w_100";
[[[ossClient imageActionPersist:request] continueWithBlock:^(id _Nullable (OSSTask * _Nonnull
task) {
    return nil;
}]] waitUntilFinished];

```

10.13. Handle exceptions

There are two types of exceptions in the iOS SDK: `ClientError` and `ServerError`.

`ClientError` indicates parameter and network errors. `ServerError` indicates abnormal responses from the OSS server.

Error type	Error domain	Code	User Info	Description	Solution
ClientError	com.aliyun.oss.clientError	0	<code>OSSClientErrorCodeNetworkingFailWithResponseCode0</code>	The network connection is abnormal.	Check the network connection and try again.
		1	<code>OSSClientErrorCodeSignatureFailed</code>	A signature failure occurs.	For more information about troubleshooting, see FAQ .
		2	<code>OSSClientErrorCodeFileCantWrite</code>	The object fails to be written to the server.	The path of the checkpoint file for upload or download or the path of the object to be download may be invalid. Modify the corresponding path and try again.
		3	<code>OSSClientErrorCodeInvalidArgument</code>	The parameter is invalid.	The parameter format is incorrect. For more information about the parameter format, see List of operations by function .
		4	<code>OSSClientErrorCodeNilUploadid</code>	The upload ID for the resumable upload task fails to be obtained.	Check the parameters. For example, check whether the object metadata is correct. Try again.
		5	<code>OSSClientErrorCodeTaskCancelled</code>	The task is canceled.	Check whether the logic used to cancel the task is correct or whether the network connection is normal.

Error type	Error domain	Code	User Info	Description	Solution
		6	OSSClientErrorCodeNetworkError	The network is abnormal.	Check the network connection and try again.
		7	OSSClientErrorCodeInvalidCRC	The CRC check fails.	Data is inconsistent during transfer. Check whether the object is modified.
		8	OSSClientErrorCodeCannotResumeUpload	The resumable upload task fails.	The object is modified during upload, causing inconsistency in the size of the object. Do not modify the object during upload.
		9	OSSClientErrorCodeExceptionCaught	An exception is captured.	Troubleshoot the error based on the specific error information.
ServerError	com.aliyun.oss.serverError	(-1 * httpResponse.statusCode)	dict	The directory obtained by parsing the corresponding XML file.	An error may occur on the server, causing the server unable to complete the request. For more information about troubleshooting, see Error responses .

11.C++

11.1. Preface

This topic is written for OSS SDK for C++ V1.8.2.

SDK source code

For more information about SDK source code, visit [GitHub](#).

Sample code

OSS SDK for C++ provides a variety of sample code for your reference or use. The following table lists the sample code provided.

Sample file	Content
PutObjectFromFile	Upload objects
GetObjectToFile	Download objects
AppendObject	Append upload
MultipartUploadObject	Multipart upload
HeadObject	Manage object metadata
ListObjects	List objects
CopyObject	Copy objects
DeleteObject	Delete objects
PutObjectProgress	Upload progress bars
DownloadObjectProcess	Download progress bars
RestoreArchiveObject	Restore archive objects
BucketSample.cc	Configure Static website hosting, CORS, Lifecycle rules, Access logging, and Authorized access for a bucket
ObjectTrafficLimitTest.cc	Configure Single-connection bandwidth throttling
ObjectRequestPaymentTest.cc	Pay-by-requester
ObjectTaggingTest.cc	Add, query, and delete object tagging configurations and Object tagging and lifecycle rules

11.2. Installation

This topic describes how to install Object Storage Service (OSS) SDK for C++ and configure compilation options in different operating systems.

Prerequisites

- A compiler that supports C++ 11 and later is installed.
- Visual Studio 2013 or later is installed.
- GCC 4.8 or later is installed.
- Clang 3.3 or later is installed.

Download OSS SDK for C++

- [Download directly](#)
- [Download from GitHub](#)

Install OSS SDK for C++

You can install OSS SDK for C++ in Linux, Windows, Android, and Mac.

- In Linux:
 - i. Install CMake and generate the build script for Linux.

After you install CMake 3.1 or later, download the source code of SDK for C++ and generate the required file by using CMake.

Run the following command to compile the source code:

```
cd <path/to/alibabacloud-oss-cpp-sdk>
mkdir build
cd build
cmake ..
```

- ii. Install libcurl and openssl

Run the following command to install RedHat or CentOS:

```
yum -y install libcurl-devel openssl-devel
```

Run the following command to install Fedora:

```
sudo dnf install libcurl-devel openssl-devel
```

- iii. Install the SDK.

```
make && make install
```



Notice By default, RTTI is disabled in OSS SDK for C++. Therefore, add the following options when you use g++ to install the SDK: -std=c++11, -fno-rtti, -lalibabacloud-oss-cpp-sdk, -lcurl, -lcrypto, and -lpthread.

Examples:

```
g++ test.cpp -std=c++11 -fno-rtti -lalibabacloud-oss-cpp-sdk -lcurl -lcrypto -lpthread -o test.bin
```

- In Windows:

- i. Install CMake and generate the build script for Windows.

After you install CMake 3.1 or later, open cmd and go to the directory of the SDK. Create a directory named build and then run the `cmake ..` command to generate required files. The following figure provides you an example on the process.

```
D:\Item\C++\aliyun-oss-cpp-sdk\build>cmake ..
-- Selecting Windows SDK version 10.0.17134.0 to target Windows 6.1.7601.
-- Project version: 1.1.0
-- TARGET_OS: WINDOWS
-- Configuring done
-- Generating done
-- Build files have been written to: D:/Item/C++/aliyun-oss-cpp-sdk/build
```

 Note

- The SDK package does not include the project file `alibabacloud-oss-cpp-sdk.sln`. You must run the `cmake` command to generate required project files.
- To build a x64 structure, run the `cmake -A x64 ..` command.

- ii. Run VS developer command prompt as an administrator. Go to the build directory and run the following command to compile and install the SDK:

```
msbuild ALL_BUILD.vcxproj
msbuild INSTALL.vcxproj
```

You can also use Visual Studio to open the project file `alibabacloud-oss-cpp-sdk.sln` to generate a solution.

- In Android

Build a project based on the `android-ndk-r16` tool chain in Linux. Install third-party libraries `libcurl` and `libopenssl` in `$ANDROID_NDK/sysroot` and run the following command to compile and install the SDK:

```
cmake -DCMAKE_TOOLCHAIN_FILE=$ANDROID_NDK/build/cmake/android.toolchain.cmake \
-DANDROID_NDK=$ANDROID_NDK \
-DANDROID_ABI=armeabi-v7a \
-DANDROID_TOOLCHAIN=clang \
-DANDROID_PLATFORM=android-21 \
-DANDROID_STL=c++_shared ..
make
```

- In macOS:

In macOS, you can install dependency library by using the brew method.

Specify the installation path of OpenSSL in Mac. For example, if OpenSSL is installed in `/usr/local/Cellar/openssl/1.0.2p`, run the following command to compile and install the SDK:

```
cmake -DOPENSSL_ROOT_DIR=/usr/local/Cellar/openssl/1.0.2p \
-DOPENSSL_LIBRARIES=/usr/local/Cellar/openssl/1.0.2p/lib \
-DOPENSSL_INCLUDE_DIRS=/usr/local/Cellar/openssl/1.0.2p/include/ ..
make
```

Compilation options

You can configure different compilation options in different scenarios in the following format: `cmake -D${OptionName}=ON|OFF ...`

The following table describes the compilation options that you can configure.

Option	Description
BUILD_SHARED_LIBS	Build dynamic libraries. Default value: OFF. If you set this option to ON, dynamic libraries and static libraries are built at the same time, and the -static suffix is added to the name of static libraries. To build dynamic libraries in the compilation, run the <code>cmake -DBUILD_SHARED_LIBS=ON ..</code> command.
ENABLE_RTTI	Supports Run-Time Type Information. Default value: ON.
OSS_DISABLE_BUCKET	Does not build interfaces related to buckets. Default value: OFF. You can set this option to ON to reduce code size.
OSS_DISABLE_LIVECHANNEL	Does not build interfaces related to LiveChannels. Default value: OFF. You can set this option to ON to reduce code size.
OSS_DISABLE_RESUMABLE	Does not build interfaces related to resumable upload and download. Default value: OFF. You can set this option to ON to reduce code size.
OSS_DISABLE_ENCRYPTION	Does not build interfaces related to client-side encryption. Default value: OFF. You can set this option to ON to reduce code size.

11.3. Initialization

OSSClient is used to manage Object Storage Service (OSS) resources such as buckets and objects. To use OSS SDK for C++ to initiate a request, you must initialize an OSSClient instance and modify the default configuration items of ClientConfiguration.

Configure the OSSClient instance

ClientConfiguration is a configuration class of OSSClient. You can use ClientConfiguration to configure parameters such as userAgent, connectTimeoutMs, and maxConnections. The following table describes parameters that you can configure by using ClientConfiguration.

Parameter	Description
userAgent	The user agent (the User-Agent field in the HTTP header). Default value: aliyun-sdk-cpp/1.X.X.
maxConnections	The maximum number of connections. Default value: 16.
requestTimeoutMs	The request timeout period. Unit: milliseconds. The connection is closed if no data is received during the timeout period. Default value: 10000. Unit: milliseconds.

Parameter	Description
connectTimeoutMs	The timeout period to establish a connection. Default value: 5000. Unit: milliseconds.
retryStrategy	The maximum number of allowed retry attempts in the case of a request error.
proxyScheme	The proxy protocol. Default value: HTTP.
proxyPort	The port that is used to connect to the proxy server.
proxyPassword	The password that is used to log on to the proxy server.
proxyUserName	The username that is used to log on to the proxy server.
verifySSL	Specifies whether to enable SSL-based authentication. By default, SSL-based authentication is disabled.  Note By default, SSL-based authentication is enabled in OSS SDK for C++ V1.8.2 or later.
caPath	The root path of the CA certificate. This parameter is valid when verifySSL is set to true. This parameter is empty by default.
caFile	The path of the CA certificate. This parameter is valid when verifySSL is set to true. This parameter is empty by default.
enableCrc64	Specifies whether to enable CRC-64. By default, CRC-64 is enabled.
enableDateSkewAdjustment	Specifies whether to enable the automatic correction of HTTP request time. By default, the automatic correction of HTTP request time is enabled.
sendRateLimiter	The limit of the upload speed. Unit: KB/s.
recvRateLimiter	The limit of the download speed. Unit: KB/s.

Configure a timeout period

The following code provides an example on how to configure a timeout period:

```
#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize resources such as networks. */
    InitializeSdk();
    ClientConfiguration conf;
    /* Specify the maximum number of connections. The default value is 16. */
    conf.maxConnections = 20;
    /* Specify the request timeout period in milliseconds. The connection is closed if no data is received during the timeout period. The default value is 10000. */
    conf.requestTimeoutMs = 8000;
    /* Specify the timeout period in milliseconds to establish a connection. The default value is 5000. */
    conf.connectTimeoutMs = 8000;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    /* Release resources such as networks. */
    ShutdownSdk();
    return 0;
}
```

Configure SSL-based authentication

By default, SSL-based authentication is enabled in OSS SDK for C++ V1.8.2 or later. If SSL-based authentication fails, you must set the correct path of the SSL certificate, or disable SSL-based authentication.

The following code provides an example on how to configure SSL-based authentication:

```
#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize resources such as networks. */
    InitializeSdk();
    ClientConfiguration conf;
    /* Configure SSL-based authentication. The default value is true, which indicates that SSL-based authentication is enabled. */
    conf.verifySSL = true;
    /* Specify the root path of the CA certificate. This parameter is valid when verifySSL is set to true. This parameter is empty by default. */
    conf.caPath = "/etc/ssl/certs/";
    /* Specify the path of the CA certificate. This parameter is valid when verifySSL is set to true. This parameter is empty by default. */
    conf.caFile = "/etc/ssl/certs/ca-certificates.crt";
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    /* Release resources such as networks. */
    ShutdownSdk();
    return 0;
}
```

Configure bandwidth throttling

The following code provides an example on how to configure bandwidth throttling for uploads and downloads:

```
#include <alibabacloud/oss/OssClient.h>
#include <alibabacloud/oss/client/RateLimiter.h>
using namespace AlibabaCloud::OSS;
class UserRateLimiter : public RateLimiter
{
public:
    UserRateLimiter() :rate_(0) {};
    ~UserRateLimiter() {};
    virtual void setRate(int rate) { rate_ = rate; };
    virtual int Rate() const { return rate_; };
private:
    int rate_;
};
int main(void)
{
    /* Initialize the information about the account that is used to access OSS. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    std::string Endpoint = "yourEndpoint";
    std::string BucketName = "yourBucketName";
    std::string ObjectName = "yourObjectName";
    /* Initialize resources such as networks. */
    InitializeSdk();
    ClientConfiguration conf;
    auto sendrateLimiter = std::make_shared<UserRateLimiter>();
    auto recvrateLimiter = std::make_shared<UserRateLimiter>();
    conf.sendRateLimiter = sendrateLimiter;
    conf.recvRateLimiter = recvrateLimiter;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    /* Configure bandwidth throttling for the download. Unit: KB/s. */
    recvrateLimiter->setRate(256);
    /* Configure bandwidth throttling for the upload. Unit: KB/s. */
    sendrateLimiter->setRate(256);
    /* Upload the object. */
    auto outcome = client.PutObject(BucketName, ObjectName,"yourLocalFilename");
    /* Update the speed limit during the upload. Unit: KB/s. */
    sendrateLimiter->setRate(300);
    /* Release resources such as networks. */
    ShutdownSdk();
    return 0;
}
```

Configure a retry policy

The following code provides an example on how to configure a retry policy:

```
#include <alibabacloud/oss/OssClient.h>
#include <alibabacloud/oss/client/RetryStrategy.h>
using namespace AlibabaCloud::OSS;
class UserRetryStrategy : public RetryStrategy
{
    {
```

```

public:
    /* maxRetries indicates the maximum number of allowed retry attempts. scaleFactor indic
ates the factor that is used to calculate the time to wait before a retry is attempted. */
    UserRetryStrategy(long maxRetries = 3, long scaleFactor = 300) :
        m_scaleFactor(scaleFactor), m_maxRetries(maxRetries)
    {}
    /* You can define the shouldRetry function to determine whether to retry a request. */
    bool shouldRetry(const Error & error, long attemptedRetries) const;
    /* You can define the calcDelayTimeMs function to calculate the time to wait before a r
etry is attempted. */
    long calcDelayTimeMs(const Error & error, long attemptedRetries) const;
private:
    long m_scaleFactor;
    long m_maxRetries;
};

bool UserRetryStrategy::shouldRetry(const Error & error, long attemptedRetries) const
{
    if (attemptedRetries >= m_maxRetries)
        return false;
    long responseCode = error.Status();
    //http code
    if ((responseCode == 403 && error.Message().find("RequestTimeTooSkewed") != std::string
::npos) ||
        (responseCode > 499 && responseCode < 599)) {
        return true;
    }
    else {
        switch (responseCode)
        {
            //curl error code
            case (ERROR_CURL_BASE + 7): //CURLE_COULDNT_CONNECT
            case (ERROR_CURL_BASE + 18): //CURLE_PARTIAL_FILE
            case (ERROR_CURL_BASE + 23): //CURLE_WRITE_ERROR
            case (ERROR_CURL_BASE + 28): //CURLE_OPERATION_TIMEDOUT
            case (ERROR_CURL_BASE + 52): //CURLE_GOT_NOTHING
            case (ERROR_CURL_BASE + 55): //CURLE_SEND_ERROR
            case (ERROR_CURL_BASE + 56): //CURLE_RECV_ERROR
                return true;
            default:
                break;
        };
    }
    return false;
}

long UserRetryStrategy::calcDelayTimeMs(const Error & error, long attemptedRetries) const
{
    UNUSED_PARAM(error);
    return (1 << attemptedRetries) * m_scaleFactor;
}

int main(void)
{
    /* Initialize resources such as networks. */
    InitializeSdk();
    ClientConfiguration conf;

```

```
/* Configure the maximum number of allowed retry attempts in the case of a request error. The default value is 3. */
auto defaultRetryStrategy = std::make_shared<UserRetryStrategy>(5);
conf.retryStrategy = defaultRetryStrategy;
OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
/* Release resources such as networks. */
ShutdownSdk();
return 0;
}
```

11.4. Quick start

This topic describes how to use Object Storage Service (OSS) SDK for C++ to perform routine operations, such as creating buckets, uploading objects, and downloading objects.

 **Note** In the following sample code, `conf` is an instance of `ClientConfiguration`, and `ClientConfiguration` is a configuration class of an `OssClient`. You can use this configuration class to configure parameters such as proxies, connection timeout, and the maximum number of connections. For more information, see [Initialization](#).

Create buckets

A bucket is a global namespace in OSS. A bucket is a container for objects stored in OSS.

 **Note** For more information about the naming conventions for buckets, see the "bucket" section in [Terms](#). For more information about endpoints, see [Regions and endpoints](#).

The following code provides an example on how to create a bucket named `examplebucket`:

```
#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize the OSS account information. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    std::string Endpoint = "yourEndpoint";
    /* Specify the name of the bucket. Example: examplebucket. */
    std::string BucketName = "examplebucket";
    /* Initialize network resources. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    /* Specify the name, storage class, and the access control list (ACL) of the bucket. */
    CreateBucketRequest request(BucketName, StorageClass::IA, CannedAccessControlList::PublicReadWrite);
    /* Create the bucket. */
    auto outcome = client.CreateBucket(request);
    if (!outcome.IsSuccess()) {
        /* Handle exceptions. */
        std::cout << "CreateBucket fail" <<
            ",code:" << outcome.error().Code() <<
            ",message:" << outcome.error().Message() <<
            ",requestId:" << outcome.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }
    /* Release network resources. */
    ShutdownSdk();
    return 0;
}
```

For more information about how to create a bucket, see [Create buckets](#) in OSS Console User Guide.

Upload objects

The following code provides an example on how to upload a local file to OSS:

```
#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize the OSS account information. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    std::string Endpoint = "yourEndpoint";
    /* Specify the name of the bucket to which you want to upload an object. Example: examplebucket. */
    std::string BucketName = "examplebucket";
    /* Specify the full path of the object. Example: exampledir/exampleobject.txt. */ The path cannot contain bucket names. */
    std::string ObjectName = "exampledir/exampleobject.txt";
    /* Initialize network resources. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    /* Upload the local file. */
    /* Specify the full path of the local file. Example: D:\\localpath\\examplefile.txt, in which localpath indicates the local path of the local file named examplefile.txt. */
    auto outcome = client.PutObject(BucketName, ObjectName, "D:\\localpath\\examplefile.txt");
    if (!outcome.IsSuccess()) {
        /* Handle exceptions. */
        std::cout << "PutObject fail" <<
            ",code:" << outcome.error().Code() <<
            ",message:" << outcome.error().Message() <<
            ",requestId:" << outcome.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }
    /* Release network resources. */
    ShutdownSdk();
    return 0;
}
```

Download objects

The following code provides an example on how to download a specified object to a local file:

```

#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize the OSS account information. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    std::string Endpoint = "yourEndpoint";
    /* Specify the name of the bucket in which the object you want to download is stored. E
xample: examplebucket. */
    std::string BucketName = "examplebucket";
    /* Specify the full path of the object. Example: exampledir/exampleobject.txt. */ The p
ath cannot contain bucket names. */
    std::string ObjectName = "exampledir/exampleobject.txt";
    /* Specify the full path of the local file. Example: D:\\localpath\\examplefile.txt, in
which localpath indicates the local path of the local file named examplefile.txt. */
    std::string FileNameToSave = "D:\\localpath\\examplefile.txt";
    /* Initialize network resources. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    /* Download the object to a local file. */
    GetObjectRequest request(BucketName, ObjectName);
    request.setResponseStreamFactory([=]() {return std::make_shared<std::fstream>(FileNamet
oSave, std::ios_base::out | std::ios_base::in | std::ios_base::trunc | std::ios_base::binary
); });
    auto outcome = client.GetObject(request);
    if (outcome.isSuccess()) {
        std::cout << "GetObjectToFile success, ContentLength is" << outcome.result().Metada
ta().ContentLength() << std::endl;
    }
    else {
        /* Handle exceptions. */
        std::cout << "GetObjectToFile fail" <<
        ",code:" << outcome.error().Code() <<
        ",message:" << outcome.error().Message() <<
        ",requestId:" << outcome.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }
    /* Release network resources. */
    ShutdownSdk();
    return 0;
}

```

List objects

The following code provides an example on how to list objects stored in a bucket named examplebucket. By default, a maximum of 100 objects are listed.


```
#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize the OSS account information. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    std::string Endpoint = "yourEndpoint";
    /* Specify the name of the bucket in which the objects you want to list are stored. Example: examplebucket. */
    std::string BucketName = "examplebucket";
    /* Initialize network resources. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    /* List the objects. */
    ListObjectsRequest request(BucketName);
    auto outcome = client.ListObjects(request);
    if (!outcome.isSuccess()) {
        /* Handle exceptions. */
        std::cout << "ListObjects fail" <<
            ",code:" << outcome.error().Code() <<
            ",message:" << outcome.error().Message() <<
            ",requestId:" << outcome.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }
    else {
        for (const auto& object : outcome.result().ObjectSummaries()) {
            std::cout << "object"<<
                ",name:" << object.Key() <<
                ",size:" << object.Size() <<
                ",last modified time:" << object.LastModified() << std::endl;
        }
    }
    /* Release network resources. */
    ShutdownSdk();
    return 0;
}
```

Delete objects

The following code provides an example on how to delete an object:

```
#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize the OSS account information. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    std::string Endpoint = "yourEndpoint";
    /* Specify the name of the bucket in which the object you want to delete is stored. Example: examplebucket. */
    std::string BucketName = "examplebucket";
    /* Specify the full path of the object. Example: exampledir/exampleobject.txt. */ The path cannot contain bucket names. */
    std::string ObjectName = "exampledir/exampleobject.txt";
    /* Initialize network resources. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    DeleteObjectRequest request(BucketName, ObjectName);
    /* Delete the objects. */
    auto outcome = client.DeleteObject(request);
    if (!outcome.IsSuccess()) {
        /* Handle exceptions. */
        std::cout << "DeleteObject fail" <<
            ",code:" << outcome.error().Code() <<
            ",message:" << outcome.error().Message() <<
            ",requestId:" << outcome.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }
    /* Release network resources. */
    ShutdownSdk();
    return 0;
}
```

For more information about how to delete objects, see [Delete objects](#).

11.5. Buckets

11.5.1. Create buckets

A bucket is a container used to store objects in Object Storage Service (OSS). Each object is contained in a bucket. This topic describes how to create a bucket.

Sample code

The following code provides an example on how to create a bucket named examplebucket:

```

#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize the account information. */
    /* Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to
    o access OSS because the account has permissions on all API operations. We recommend that y
    ou use a RAM user to call API operations or perform routine operations and maintenance. To
    create a RAM user, log on to the RAM console. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    /* Set yourEndpoint to the endpoint of the region in which the bucket is located. For e
    xample, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https:
    //oss-cn-hangzhou.aliyuncs.com. */
    std::string Endpoint = "yourEndpoint";
    /* Specify the name of the bucket. Example: examplebucket. */
    std::string BucketName = "examplebucket";
    /* Initialize resources such as network resources. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    /*Specify the name, storage class, and access control list (ACL) of the bucket. */
    CreateBucketRequest request(BucketName, StorageClass::IA, CannedAccessControlList::Publ
icReadWrite);
    /* Set the redundancy type of the bucket to zone-redundant storage (ZRS). */
    //request.setDataRedundancyType(DataRedundancyType::ZRS);
    /* Create the bucket. */
    auto outcome = client.CreateBucket(request);
    if (!outcome.IsSuccess()) {
        /* Handle exceptions. */
        std::cout << "CreateBucket fail" <<
        ",code:" << outcome.error().Code() <<
        ",message:" << outcome.error().Message() <<
        ",requestId:" << outcome.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }
    /* Release resources such as network resources. */
    ShutdownSdk();
    return 0;
}

```

References

- For more information about the complete sample code that is used to create a bucket, visit [GitHub](#).
- For more information about the API operation that you can call to create a bucket, see [PutBucket](#).

11.5.2. List buckets

Buckets are listed in alphabetical order in Object Storage Service (OSS). This topic describes how to list all buckets that belong to the current Alibaba Cloud account.

Sample code

The following code provides an example on how to list all buckets that belong to the current Alibaba Cloud account:

```
#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize the account information. */
    /* Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to
    o access OSS because the account has permissions on all API operations. We recommend that y
    ou use a RAM user to call API operations or perform routine O&M. To create a RAM user, log
    on to the RAM console. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    /* Set yourEndpoint to the endpoint of the region in which the bucket is located. For e
    xample, if the bucket is located in the China (Hangzhou) region, set the endpoint to https:
    //oss-cn-hangzhou.aliyuncs.com. */
    std::string Endpoint = "yourEndpoint";
    /* Initialize resources such as network resources. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    /* List all buckets that belong to the current Alibaba Cloud account. */
    ListBucketsRequest request;
    auto outcome = client.ListBuckets(request);
    if (outcome.isSuccess()) {
        /* Display the information about the listed buckets. */
        std::cout << " success, and bucket count is" << outcome.result().Buckets().size() <<
std::endl;
        std::cout << "Bucket name is" << std::endl;
        for (auto result : outcome.result().Buckets())
        {
            std::cout << result.Name() << std::endl;
        }
    }
    else {
        /* Handle exceptions. */
        std::cout << "ListBuckets fail" <<
        ",code:" << outcome.error().Code() <<
        ",message:" << outcome.error().Message() <<
        ",requestId:" << outcome.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }
    /* Release resources such as network resources. */
    ShutdownSdk();
    return 0;
}
```

References

- For more information about the complete sample code that is used to list buckets, visit [GitHub](#).
-

11.5.3. Determine whether buckets exist

A bucket is a container used to store objects in Object Storage Service (OSS). Every object is contained in a bucket. This topic describes how to determine whether a bucket exists.

```
#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /*Initialize the OSS account information.*/
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    std::string Endpoint = "yourEndpoint";
    std::string BucketName = "yourBucketName";
    /*Initialize network resources.*/
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    /*Determine whether the bucket exists.*/
    auto outcome = client.DoesBucketExist(BucketName);
    if (outcome) {
        std::cout << " The Bucket exists" << std::endl;
    }
    else {
        std::cout << "The Bucket does not exist" << std::endl;
    }
    /*Release network resources.*/
    ShutdownSdk();
    return 0;
}
```

11.5.4. Query the regions of buckets

A bucket is a container for objects stored in Object Storage Service (OSS). Every object is contained in a bucket. This topic describes how to query the region of a bucket.

Sample code

The following sample code shows how to query the region or location of a bucket:

```

#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize OSS account information. */
    /* Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account t
o access OSS, because the account has permissions on all API operations. We recommend that
you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log
on to the RAM console. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    /* Set yourEndpoint to the endpoint of the region in which the bucket is located. For e
xample, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https:
//oss-cn-hangzhou.aliyuncs.com. */
    std::string Endpoint = "yourEndpoint";
    /* Set yourBucketName to the name of the bucket. */
    std::string BucketName = "yourBucketName";
    /* Initialize resources such as network resources. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    /* Query the region of the bucket. */
    GetBucketLocationRequest request(BucketName);
    auto outcome = client.GetBucketLocation(request);
    if (outcome.isSuccess()) {
        std::cout << "getBucketLocation success, location: " << outcome.result().Location()
<< std::endl;
    }
    else {
        /* Handle exceptions. */
        std::cout << "getBucketLocation fail" <<
            ",code:" << outcome.error().Code() <<
            ",message:" << outcome.error().Message() <<
            ",requestId:" << outcome.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }
    /* Release resources such as network resources. */
    ShutdownSdk();
    return 0;
}

```

References

- For more information about the complete sample code that is used to query the region of a bucket, visit [GitHub](#).
- For more information about the API operation that you can call to query the region of a bucket, see [GetBucketLocation](#).

11.5.5. Query bucket information

A bucket is a container for objects stored in Object Storage Service (OSS). Every object is contained in a bucket. This topic describes how to query the information about a bucket.

The following code provides an example on how to query the information about a bucket, including the region and creation date of the bucket:

```
#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /*Initialize the OSS account information.*/
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    std::string Endpoint = "yourEndpoint";
    std::string BucketName = "yourBucketName";
    /*Initialize network resources.*/
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    /*Query the bucket information.*/
    GetBucketInfoRequest request(BucketName);
    auto outcome = client.GetBucketInfo(request);
    /*View the type of redundant storage of the bucket.*/
    if (outcome.isSuccess()) {
        std::cout << "GetBucketInfo success, DataRedundancyType:" << outcome.result().DataRedundancyType() << std::endl;
    }
    else {
        /*Handle exceptions.*/
        std::cout << "GetBucketInfo fail" <<
            ",code:" << outcome.error().Code() <<
            ",message:" << outcome.error().Message() <<
            ",requestId:" << outcome.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }
    /*Release network resources.*/
    ShutdownSdk();
    return 0;
}
```

For more information about how to query bucket information, see [GetBucketInfo](#).

11.5.6. Manage bucket ACLs

A bucket is a container for objects stored in Object Storage Service (OSS). Every object is contained in a bucket. This topic describes how to configure and query the access control list (ACL) of a bucket.

Configure the ACL of a bucket

The following table describes the bucket ACLs.

ACL	Description	Method
private	Only the bucket owner and authorized users can perform read and write operations on objects in the bucket. Other users cannot access the objects in the bucket.	OSS_ACL_PRIVATE
public read	Only the bucket owner and authorized users of this bucket can perform read and write operations on objects in the bucket. Other users, including anonymous users, can only read objects in the bucket. Exercise caution when you set the bucket ACL to this value.	OSS_ACL_PUBLIC_READ
public read/write	All users, including anonymous users, can perform read and write operations on objects in the bucket. Exercise caution when you set the bucket ACL to this value.	OSS_ACL_PUBLIC_READ_WRITE

The following code provides an example on how to configure the ACL of a bucket:


```

#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize the account information. */
    /* Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to
    o access OSS because the account has permissions on all API operations. We recommend that y
    ou use a RAM user to call API operations or perform routine O&M. To create a RAM user, log
    on to the RAM console. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    /* Set yourEndpoint to the endpoint of the region in which the bucket is located. For e
    xample, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https:
    //oss-cn-hangzhou.aliyuncs.com. */
    std::string Endpoint = "yourEndpoint";
    /* Specify the name of the bucket. Example: examplebucket. */
    std::string BucketName = "examplebucket";
    /* Initialize resources such as network resources. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    /* Set the ACL of the bucket to private. */
    SetBucketAclRequest request(BucketName, CannedAccessControlList::Private);
    auto outcome = client.SetBucketAcl(request);
    if (outcome.IsSuccess()) {
        std::cout << " setBucketAcl to private success " << std::endl;
    }
    else {
        /* Handle exceptions. */
        std::cout << "SetBucketAcl fail" <<
            ",code:" << outcome.error().Code() <<
            ",message:" << outcome.error().Message() <<
            ",requestId:" << outcome.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }
    /* Release resources such as network resources. */
    ShutdownSdk();
    return 0;
}

```

For more information about how to configure the ACL of a bucket, see [PutBucketAcl](#).

Query the ACL of a bucket

The following code provides an example on how to query the ACL of a bucket:

```
#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize the account information. */
    /* Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to
    o access OSS because the account has permissions on all API operations. We recommend that y
    ou use a RAM user to call API operations or perform routine O&M. To create a RAM user, log
    on to the RAM console. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    /* Set yourEndpoint to the endpoint of the region in which the bucket is located. For e
    xample, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https:
    //oss-cn-hangzhou.aliyuncs.com. */
    std::string Endpoint = "yourEndpoint";
    /* Specify the name of the bucket. Example: examplebucket. */
    std::string BucketName = "examplebucket";
    /* Initialize resources such as network resources. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    /* Query the ACL of the bucket. */
    GetBucketAclRequest request(BucketName);
    auto outcome = client.GetBucketAcl(request);
    if (outcome.isSuccess()) {
        std::cout << "getBucketAcl success, acl: " << outcome.result().Acl() << std::endl;
    }
    else {
        /* Handle exceptions. */
        std::cout << "getBucketAcl fail" <<
            ",code:" << outcome.error().Code() <<
            ",message:" << outcome.error().Message() <<
            ",requestId:" << outcome.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }
    /* Release resources such as network resources. */
    ShutdownSdk();
    return 0;
}
```

For more information about how to query the ACL of a bucket, see [GetBucketAcl](#).

11.5.7. Delete a bucket

If you no longer use a bucket, delete the bucket to stop unexpected charges.



Warning Deleted buckets cannot be recovered. Exercise caution when you perform this operation.

Prerequisites

- All objects in the bucket are deleted.
 - For more information about how to manually delete a small number of objects, see [删除文件](#).
 - To delete a large number of objects, we recommend that you configure lifecycle rules to batch delete the objects. For more information about how to delete a large number of objects, see [Configure lifecycle rules](#).

 Notice

If versioning is enabled for the bucket that you want to delete, make sure that all versions of objects in the bucket are deleted. For more information about how to delete all versions of objects in a bucket, see [Configure versioning](#).

- Parts that are uploaded by multipart upload or resumable upload tasks in the bucket are deleted. For more information about how to delete parts in a bucket, see [Manage parts](#).
- All LiveChannels in the bucket are deleted. For more information about how to delete LiveChannels, see [DeleteLiveChannel](#).

Sample code

The following sample code shows how to delete a bucket named examplebucket:

```
#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize OSS account information. */
    /* The AccessKey pair of an Alibaba Cloud account has permissions on all API operations
     . Using these credentials to access OSS is a high-risk operation. We recommend that you use
     a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to t
     he RAM console. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    /* Set yourEndpoint to the endpoint of the region in which the bucket is located. For e
     xample, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https:
     //oss-cn-hangzhou.aliyuncs.com. */
    std::string Endpoint = "yourEndpoint";
    /* Specify the name of the bucket. Example: examplebucket. */
    std::string BucketName = "examplebucket";
    /* Initialize resources such as network resources. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    /* Delete the bucket. */
    DeleteBucketRequest request(BucketName);
    auto outcome = client.DeleteBucket(request);
    if (!outcome.isSuccess()) {
        /* Handle exceptions. */
        std::cout << "DeleteBucket fail" <<
            ",code:" << outcome.error().Code() <<
            ",message:" << outcome.error().Message() <<
            ",requestId:" << outcome.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }
    /* Release resources such as network resources. */
    ShutdownSdk();
    return 0;
}
```

References

- For more information about the complete sample code that is used to delete a bucket, visit [GitHub](#).
- For more information about the API operation that you can call to delete a bucket, see [DeleteBucket](#).

11.5.8. Configure bucket tagging

You can use tags to identify buckets used for different purposes and manage these buckets.

Usage notes

The bucket tagging feature uses a key-value pair to identify a bucket. You can add tags to buckets that are used for different purposes and manage the buckets by tags.

- Only the bucket owner and users who are granted the `oss:PutBucketTagging` permission can add

tags to the bucket or modify the tags of the bucket. If other users try to add tags to the bucket or modify the tags of the bucket, 403 Forbidden is returned with the `AccessDenied` error code.

- You can configure up to 20 tags for a bucket.
- The key and value of a tag must be UTF-8-encoded.
- The key of a tag can be up to 64 characters in length and cannot start with `http://`, `https://`, or `Aliyun`. In addition, the key of a tag cannot be empty.
- The value of a tag can be up to 128 characters in length and can be empty.

Configure tags for a bucket

The following code provides an example on how to configure tags for a bucket:

```

#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize information about the account used to access Object Storage Service (OSS)
    . */
    /* Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account t
    o access OSS because the account has permissions on all API operations. We recommend that y
    ou use a RAM user to call API operations or perform routine Q&M. To create a RAM user, log
    on to the RAM console. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    /* Set yourEndpoint to the endpoint of the region in which the bucket is located. For e
    xample, if the bucket is located in the China (Hangzhou) region, set the endpoint to https:
    //oss-cn-hangzhou.aliyuncs.com. */
    std::string Endpoint = "yourEndpoint";
    /* Specify the bucket name. Example: examplebucket. */
    std::string BucketName = "examplebucket";
    /* Initialize resources such as networks. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    /* Configure tags for the bucket. */
    SetBucketTaggingRequest request(BucketName);
    Tag tag1("yourTagkey1", "yourTagValue1");
    Tag tag2("yourTagkey2", "yourTagValue2");
    TagSet tagset;
    tagset.push_back(tag1);
    tagset.push_back(tag2);
    Tagging tagging;
    tagging.setTags(tagset);
    request.setTagging(tagging);
    auto outcome = client.SetBucketTagging(request);
    if (outcome.isSuccess()) {
        std::cout << " SetBucketTagging success " << std::endl;
    }
    else {
        /* Handle exceptions. */
        std::cout << "SetBucketTagging fail" <<
        ",code:" << outcome.error().Code() <<
        ",message:" << outcome.error().Message() <<
        ",requestId:" << outcome.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }
    /* Release resources such as networks. */
    ShutdownSdk();
    return 0;
}

```

Query the tags of a bucket

The following code provides an example on how to query tags of a bucket:

```

#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize information about the account used to access OSS. */
    /* Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account t
    o access OSS because the account has permissions on all API operations. We recommend that y
    ou use a RAM user to call API operations or perform routine Q&M. To create a RAM user, log
    on to the RAM console. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    /* Set yourEndpoint to the endpoint of the region in which the bucket is located. For e
    xample, if the bucket is located in the China (Hangzhou) region, set the endpoint to https:
    //oss-cn-hangzhou.aliyuncs.com. */
    std::string Endpoint = "yourEndpoint";
    /* Specify the bucket name. Example: examplebucket. */
    std::string BucketName = "examplebucket";
    /* Initialize resources such as networks. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    /* Query the tags of the bucket. */
    GetBucketTaggingRequest request(BucketName);
    auto outcome = client.GetBucketTagging(request);
    if (outcome.IsSuccess()) {
        std::cout << " GetBucketTagging success " << std::endl;
        for (const auto& tag : outcome.result().Tagging().Tags()) {
            std::cout << "tag key:" << tag.Key() <<
                "tag value:" << tag.Value() << std::endl;
        }
    }
    else {
        /* Handle exceptions. */
        std::cout << "GetBucketTagging fail" <<
            ",code:" << outcome.error().Code() <<
            ",message:" << outcome.error().Message() <<
            ",requestId:" << outcome.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }
    /* Release resources such as networks. */
    ShutdownSdk();
    return 0;
}

```

List buckets that have specified tags

The following code provides an example on how to list buckets that have the specified tag:

```
#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize information about the account used to access OSS. */
    /* Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account t
o access OSS because the account has permissions on all API operations. We recommend that y
ou use a RAM user to call API operations or perform routine Q&M. To create a RAM user, log
on to the RAM console. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    /* Set yourEndpoint to the endpoint of the region in which the bucket is located. For e
xample, if the bucket is located in the China (Hangzhou) region, set the endpoint to https:
//oss-cn-hangzhou.aliyuncs.com. */
    std::string Endpoint = "yourEndpoint";
    /* Initialize resources such as networks. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    /* List buckets that have specified tags. */
    ListBucketsRequest request;
    Tag tag1("yourTagkey1", "yourTagValue1");
    request.setTag(tag1);
    auto outcome = client.ListBuckets(request);
    if (outcome.isSuccess()) {
        std::cout << "ListBuckets success" << std::endl;
        for (const auto& bucket : outcome.result().Buckets()) {
            std::cout << "bucket name:" << bucket.Name() << std::endl;
        }
    }
    else {
        /* Handle exceptions. */
        std::cout << "ListBuckets fail" <<
            ",code:" << outcome.error().Code() <<
            ",message:" << outcome.error().Message() <<
            ",requestId:" << outcome.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }
    /* Release resources such as networks. */
    ShutdownSdk();
    return 0;
}
```

Delete the tags of a bucket

The following code provides an example on how to delete the tags of a bucket.


```

#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize information about the account used to access OSS. */
    /* Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account t
o access OSS because the account has permissions on all API operations. We recommend that y
ou use a RAM user to call API operations or perform routine Q&M. To create a RAM user, log
on to the RAM console. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    /* Set yourEndpoint to the endpoint of the region in which the bucket is located. For e
xample, if the bucket is located in the China (Hangzhou) region, set the endpoint to https:
//oss-cn-hangzhou.aliyuncs.com. */
    std::string Endpoint = "yourEndpoint";
    /* Specify the bucket name. Example: examplebucket. */
    std::string BucketName = "examplebucket";
    /* Initialize resources such as networks. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    /* Delete the tags of the bucket. */
    DeleteBucketTaggingRequest request(BucketName);
    auto outcome = client.DeleteBucketTagging(request);
    if (outcome.isSuccess()) {
        std::cout << " DeleteBucketTagging success " << std::endl;
    }
    else {
        /* Handle exceptions. */
        std::cout << "DeleteBucketTagging fail" <<
            ",code:" << outcome.error().Code() <<
            ",message:" << outcome.error().Message() <<
            ",requestId:" << outcome.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }
    /* Release resources such as networks. */
    ShutdownSdk();
    return 0;
}

```

Delete specified tags of a bucket

The following code provides an example on how to delete specified tags of a bucket:

```

#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize information about the account used to access OSS. */
    /* Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account
    to access OSS because the account has permissions on all API operations. We recommend that
    you use a RAM user to call API operations or perform routine Q&M. To create a RAM user, log
    on to the RAM console. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    /* Set yourEndpoint to the endpoint of the region in which the bucket is located. For
    example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https
    ://oss-cn-hangzhou.aliyuncs.com. */
    std::string Endpoint = "yourEndpoint";
    /* Specify the bucket name. Example: examplebucket. */
    std::string BucketName = "examplebucket";
    /* Initialize resources such as networks. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    /* Delete specified tags of the bucket by specifying the key. */
    DeleteBucketTaggingRequest request(BucketName);
    Tagging tagging;
    TagSet tagset;
    Tag tag("project","projectone");
    tagset.push_back(tag);
    tagging.setTags(tagset);
    request.setTagging(tagging);
    auto outcome = client.DeleteBucketTagging(request);
    if (outcome.IsSuccess()) {
        std::cout << " DeleteBucketTagging success " << std::endl;
    }
    else {
        /* Handle exceptions. */
        std::cout << "DeleteBucketTagging fail" <<
        ",code:" << outcome.error().Code() <<
        ",message:" << outcome.error().Message() <<
        ",requestId:" << outcome.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }
    /* Release resources such as networks. */
    ShutdownSdk();
    return 0;
}

```

References

- For more information about the API operation that you can call to configure tags for a bucket, see [PutBucketTags](#).
- For more information about the API operation that you can call to query tags of a bucket, see

[GetBucketTags](#).

- For more information about the API operation that you can call to delete tags of a bucket, see [DeleteBucketTags](#).

11.5.9. Bucket inventory

This topic describes how to add an inventory for a bucket and how to query, list, and delete inventories configured for a bucket.

Create an inventory for a bucket

The following code provides an example on how to create an inventory for a bucket:

```
#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize the information about the account that is used to access Object Storage Service (OSS). */
    /* Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    /* Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com. */
    std::string Endpoint = "yourEndpoint";
    /* Specify the name of the bucket. Example: examplebucket. */
    std::string BucketName = "examplebucket";
    /* Initialize resources such as networks. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    InventoryConfiguration inventoryConf;
    /* Specify the inventory name, which must be globally unique in the current bucket. */
    inventoryConf.setId("inventoryId");
    /* Specify whether inventory is enabled for the bucket. Valid values: true and false. */
    inventoryConf.setIsEnabled(true);
    /* Optional. Specify the prefix of object names to filter objects to include in the inventory lists. After you specify the prefix, the information about objects whose names contain the prefix is included in the inventory list. */
    inventoryConf.setFilter(InventoryFilter("objectPrefix"));
    InventoryOSSBucketDestination dest;
    /* Specify the format of the exported inventory lists. */
    dest.setFormat(InventoryFormat::CSV);
    /* Specify the account ID that is granted by the bucket owner. */
    dest.setAccountId("10988548*****");
    /* Specify the role ARN to which the bucket owner grants the operation permissions. */
    dest.setRoleArn("acs:ram::10988548*****:role/inventory-test");
    /* Specify the bucket in which you want to store the generated inventory lists. */
    dest.setBucket("yourDstBucketName");
```

```

    /* Specify the prefix of the path in which you want to store the generated inventory lists. */
    dest.setPrefix("yourPrefix");
    /* Optional. Specify the method for encrypting the inventory list object. You can use SSEOSS or SSEKMS. */
    //dest.setEncryption(InventoryEncryption(InventorySSEOSS()));
    //dest.setEncryption(InventoryEncryption(InventorySSEKMS("yourKmskeyId")));
    inventoryConf.setDestination(dest);
    /* Specify the frequency at which the inventory lists are exported. You can set Daily or Weekly. */
    inventoryConf.setSchedule(InventoryFrequency::Daily);
    /* Set the object versions to include in the inventory list. Valid values: All and Current. */
    inventoryConf.setIncludedObjectVersions(InventoryIncludedObjectVersions::All);
    /* Optional. Specify the fields to include in the inventory lists based on your requirements. */
    InventoryOptionalFields field {
        InventoryOptionalField::Size, InventoryOptionalField::LastModifiedDate,
        InventoryOptionalField::ETag, InventoryOptionalField::StorageClass,
        InventoryOptionalField::IsMultipartUploaded, InventoryOptionalField::EncryptionStatus
    };
    inventoryConf.setOptionalFields(field);
    /* Configure the inventory. */
    auto outcome = client.SetBucketInventoryConfiguration(
        SetBucketInventoryConfigurationRequest(BucketName, inventoryConf));
    if (!outcome.isSuccess()) {
        /* Handle exceptions. */
        std::cout << "Set Bucket Inventory fail" <<
            ",code:" << outcome.error().Code() <<
            ",message:" << outcome.error().Message() <<
            ",requestId:" << outcome.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }
    /* Release resources such as networks. */
    ShutdownSdk();
    return 0;
}

```

Query an inventory configured for a bucket

The following code provides an example on how to query inventories configured for a bucket:

```

#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize the information about the account that is used to access OSS. */
    /* Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console. */
    std::string AccessKeyId = "yourAccessKeyId";

```

```

    std::string AccessKeySecret = "yourAccessKeySecret";
    /* Set yourEndpoint to the endpoint of the region in which the bucket is located. For e
    xample, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https:
    //oss-cn-hangzhou.aliyuncs.com. */
    std::string Endpoint = "yourEndpoint";
    /* Specify the name of the bucket. Example: examplebucket. */
    std::string BucketName = "examplebucket";
    /* Specify the inventory name. */
    std::string InventoryId = "yourInventoryId";
    /* Initialize resources such as networks. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    /* Obtain the inventory configuration. */
    auto outcome = client.GetBucketInventoryConfiguration(
        GetBucketInventoryConfigurationRequest(BucketName, InventoryId));
    if (!outcome.IsSuccess()) {
        /* Handle exceptions. */
        std::cout << "Get Bucket Inventory fail" <<
            ",code:" << outcome.error().Code() <<
            ",message:" << outcome.error().Message() <<
            ",requestId:" << outcome.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }
    /* Display the information about the inventory configuration. */
    const auto& inventoryConf = outcome.result().InventoryConfiguration();
    std::cout << inventoryConf.Id() << std::endl;
    std::cout << inventoryConf.IsEnabled() << std::endl;
    std::cout << inventoryConf.Filter().Prefix() << std::endl;
    std::cout << inventoryConf.Destination().OSSBucketDestination().AccountId() << std::end
1;

    std::cout << inventoryConf.Destination().OSSBucketDestination().RoleArn() << std::endl;
    std::cout << inventoryConf.Destination().OSSBucketDestination().Bucket() << std::endl;
    std::cout << inventoryConf.Destination().OSSBucketDestination().Prefix() << std::endl;
    if (inventoryConf.Destination().OSSBucketDestination().Encryption().hasSSEKMS()) {
        std::cout << inventoryConf.Destination().OSSBucketDestination().Encryption().SSEKMS
().KeyId() << std::endl;
    }
    else if (inventoryConf.Destination().OSSBucketDestination().Encryption().hasSSEOSS()) {
        std::cout << "has sse-oss" << std::endl;
    }
    std::cout << static_cast<int>(inventoryConf.Schedule()) << std::endl;
    std::cout << static_cast<int>(inventoryConf.IncludedObjectVersions()) << std::endl;
    for (const auto& fiedl: inventoryConf.OptionalFields()) {
        std::cout << static_cast<int>(fiedl) << std::endl;
    }
    /* Release resources such as networks. */
    ShutdownSdk();
    return 0;
}

```

List multiple inventories configured for a bucket

 **Note** You can query up to 100 inventories by sending a request. To query more than 100 inventory rules, you must send multiple requests and use the token returned for each request as the parameter for the next request.

The following code provides an example on how to list multiple inventories configured for a bucket at a time:

```
#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize the information about the account that is used to access OSS. */
    /* Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to
    o access OSS because the account has permissions on all API operations. We recommend that y
    ou use a RAM user to call API operations or perform routine O&M. To create a RAM user, log
    on to the RAM console. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    /* Set yourEndpoint to the endpoint of the region in which the bucket is located. For e
    xample, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https:
    //oss-cn-hangzhou.aliyuncs.com. */
    std::string Endpoint = "yourEndpoint";
    /* Specify the name of the bucket. Example: examplebucket. */
    std::string BucketName = "examplebucket";
    /* Initialize resources such as networks. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    /* Set the request parameters. */
    std::string nextToken = "";
    bool isTruncated = false;
    do {
        /* List the inventories. Up to 100 inventories can be listed per request. */
        auto request = ListBucketInventoryConfigurationsRequest(BucketName);
        request.setContinuationToken(nextToken);
        auto outcome = client.ListBucketInventoryConfigurations(request);
        if (!outcome.isSuccess()) {
            /* Handle exceptions. */
            std::cout << "ListObjects fail" <<
                ",code:" << outcome.error().Code() <<
                ",message:" << outcome.error().Message() <<
                ",requestId:" << outcome.error().RequestId() << std::endl;
            break;
        }
        for (const auto& inventoryConf : outcome.result().InventoryConfigurationList()) {
            std::cout << inventoryConf.Id() << std::endl;
            std::cout << inventoryConf.IsEnabled() << std::endl;
            std::cout << inventoryConf.Filter().Prefix() << std::endl;
            std::cout << inventoryConf.Destination().OSSBucketDestination().AccountId() <<
            std::endl;
            std::cout << inventoryConf.Destination().OSSBucketDestination().RoleArn() << st
```

```

d::endl;
        std::cout << inventoryConf.Destination().OSSBucketDestination().Bucket() << std
::endl;
        std::cout << inventoryConf.Destination().OSSBucketDestination().Prefix() << std
::endl;
        if (inventoryConf.Destination().OSSBucketDestination().Encryption().hasSSEKMS()
) {
            std::cout << inventoryConf.Destination().OSSBucketDestination().Encryption(
).SSEKMS().KeyId() << std::endl;
        }
        else if (inventoryConf.Destination().OSSBucketDestination().Encryption().hasSSE
OSS()) {
            std::cout << "has sse-oss" << std::endl;
        }
        std::cout << static_cast<int>(inventoryConf.Schedule()) << std::endl;
        std::cout << static_cast<int>(inventoryConf.IncludedObjectVersions()) << std::e
endl;

        for (const auto& fiedl: inventoryConf.OptionalFields()) {
            std::cout << static_cast<int>(fiedl) << std::endl;
        }
    }
    nextToken = outcome.result().NextContinuationToken();
    isTruncated = outcome.result().IsTruncated();
} while (isTruncated);
/* Release resources such as networks. */
ShutdownSdk();
return 0;
}

```

Delete an inventory configured for a bucket

The following code provides an example on how to delete an inventory configured for a bucket:

```
#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize the information about the account that is used to access OSS. */
    /* Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to
    o access OSS because the account has permissions on all API operations. We recommend that y
    ou use a RAM user to call API operations or perform routine O&M. To create a RAM user, log
    on to the RAM console. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    /* Set yourEndpoint to the endpoint of the region in which the bucket is located. For e
    xample, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https:
    //oss-cn-hangzhou.aliyuncs.com. */
    std::string Endpoint = "yourEndpoint";
    /* Specify the name of the bucket. Example: examplebucket. */
    std::string BucketName = "examplebucket";
    /* Specify the inventory name. */
    std::string InventoryId = "yourInventoryId";
    /* Initialize resources such as networks. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    /* Delete the inventory. */
    auto outcome = client.DeleteBucketInventoryConfiguration(
        DeleteBucketInventoryConfigurationRequest(BucketName, InventoryId));
    if (!outcome.IsSuccess()) {
        /* Handle exceptions. */
        std::cout << "Delete Bucket Inventory fail" <<
            ",code:" << outcome.error().Code() <<
            ",message:" << outcome.error().Message() <<
            ",requestId:" << outcome.error().RequestId() << std::endl;
    }
    /* Release resources such as networks. */
    ShutdownSdk();
    return 0;
}
```

References

- For more information about the API operation that is called to create an inventory, see [Put Bucket Inventory](#).
- For more information about the API operation that is called to query inventories configured for a bucket, see [Get Bucket Inventory](#).
- For more information about the API operation that is called to list multiple inventories configured for a bucket, see [List Bucket Inventory](#).
- For more information about the API operation that is called to delete inventories configured for a bucket, see [Delete Bucket Inventory](#).

11.5.10. Bucket policy

This topic describes how to configure, query, and delete policies for a bucket.

Background information

Bucket policies provide resource-based authorization for users. Bucket policies apply to the following scenarios:

For more information about bucket policy configurations and use cases, see [Configure bucket policies to authorize other users to access OSS resources](#). For more information about bucket policy syntax, see [Policy structure and syntax](#).

Configure bucket policies

The following sample code shows how to configure a policy for a bucket:

```

#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize the OSS account information. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    std::string Endpoint = "yourEndpoint";
    std::string BucketName = "yourBucketName";
    /* Initialize network resources. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    /* Configure a bucket policy. For example, allow users to list or download objects prefixed with user1/ in the bucket.    std::string policy =
        R"(
        {
            "Statement": [
                {
                    "Action": [
                        "oss:GetObject",
                        "oss:ListObjects"
                    ],
                    "Effect" : "Allow",
                    "Resource" : ["acs:oss:*:*:*/*user1/*"]
                }
            ],
            "Version": "1"
        }
        )";
    SetBucketPolicyRequest request(BucketName);
    request.setPolicy(policy);
    auto outcome = client.SetBucketPolicy(request);
    if (!outcome.isSuccess()) {
        /* Handle exceptions. */
        std::cout << "Set Bucket Policy fail" <<
            ",code:" << outcome.error().Code() <<
            ",message:" << outcome.error().Message() <<
            ",requestId:" << outcome.error().RequestId() << std::endl;
    }
    /* Release network resources. */
    ShutdownSdk();
    return 0;
}

```

Query bucket policies

The following sample code shows how to query the policies configured for a bucket:

```
#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize the OSS account information. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    std::string Endpoint = "yourEndpoint";
    std::string BucketName = "yourBucketName";
    /* Initialize network resources. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    /* Query the policies configured for the bucket. */
    GetBucketPolicyRequest request(BucketName);
    auto outcome = client.GetBucketPolicy(request);
    if (!outcome.isSuccess()) {
        /* Handle exceptions. */
        std::cout << "Get Bucket Policy fail" <<
            ", code:" << outcome.error().Code() <<
            ", message:" << outcome.error().Message() <<
            ", requestId:" << outcome.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }
    /* Display the policies configured for the bucket. */
    std::cout << outcome.result().Policy() << std::endl;
    /* Release network resources. */
    ShutdownSdk();
    return 0;
}
```

Delete bucket policies

The following sample code shows how to delete the policies configured for a bucket:

```
#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize the OSS account information. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    std::string Endpoint = "yourEndpoint";
    std::string BucketName = "yourBucketName";
    /* Initialize network resources. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    /* Delete the policies configured for the bucket. */
    DeleteBucketPolicyRequest request(BucketName);
    auto outcome = client.DeleteBucketPolicy(request);
    if (!outcome.IsSuccess()) {
        /* Handle exceptions. */
        std::cout << "Delete Bucket Policy fail" <<
            ",code:" << outcome.error().Code() <<
            ",message:" << outcome.error().Message() <<
            ",requestId:" << outcome.error().RequestId() << std::endl;
    }
    /* Release network resources. */
    ShutdownSdk();
    return 0;
}
```

11.5.11. Retention policy

You can configure a time-based retention policy for a bucket. A retention policy has a protection period ranging from one day to 70 years. This topic describes how to create, query, and lock a retention policy.

Background information

Object Storage Service (OSS) supports the Write Once Read Many (WORM) feature, which prevents an object from being deleted or overwritten for a specified period of time.

If a retention policy is not locked within 24 hours after it is created, the retention policy becomes invalid. If the retention policy configured for a bucket is locked, you can upload objects to or read objects from the bucket. However, you cannot delete objects in the bucket or the retention policy within the retention period of the policy. The retention period of the policy can be extended but cannot be shortened. For more information about retention policies, see [Retention policies](#).

Create a retention policy

The following code provides an example on how to create a retention policy:

```
#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize the Alibaba Cloud account information.*/
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    std::string Endpoint = "yourEndpoint";
    std::string BucketName = "yourBucketName";
    /* Initialize network resources.*/
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    /* Create the retention policy and set the protection period to 10 days.*/
    auto outcome = client.InitiateBucketWorm(InitiateBucketWormRequest(BucketName, 10));
    if (outcome.isSuccess()) {
        std::cout << " InitiateBucketWorm success " << std::endl;
    }
    else {
        /* Handle exceptions.*/
        std::cout << "InitiateBucketWorm fail" <<
            ",code:" << outcome.error().Code() <<
            ",message:" << outcome.error().Message() <<
            ",requestId:" << outcome.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }
    /* Release network resources.*/
    ShutdownSdk();
    return 0;
}
```

For more information about how to create a retention policy, see [InitiateBucketWorm](#).

Cancel an unlocked retention policy

The following code provides an example on how to cancel an unlocked retention policy:

```
#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize the Alibaba Cloud account information.*/
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    std::string Endpoint = "yourEndpoint";
    std::string BucketName = "yourBucketName";
    /* Initialize network resources.*/
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    /* Cancel the unlocked retention policy.*/
    auto outcome = client.AbortBucketWorm(AbortBucketWormRequest(BucketName));
    if (outcome.IsSuccess()) {
        std::cout << " AbortBucketWorm success " << std::endl;
    }
    else {
        /* Handle exceptions.*/
        std::cout << "AbortBucketWorm fail" <<
            ",code:" << outcome.error().Code() <<
            ",message:" << outcome.error().Message() <<
            ",requestId:" << outcome.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }
    /* Release network resources.*/
    ShutdownSdk();
    return 0;
}
```

For more information about how to cancel an unlocked retention policy, see [AbortBucketWorm](#).

Lock a retention policy

The following code provides an example on how to lock a retention policy:

```
#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize the Alibaba Cloud account information.*/
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    std::string Endpoint = "yourEndpoint";
    std::string BucketName = "yourBucketName";
    std::string WormId = "yourWormId";
    /* Initialize network resources.*/
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    /* Lock the retention policy.*/
    auto outcome = client.CompleteBucketWorm(CompleteBucketWormRequest(BucketName, WormId));
    if (outcome.isSuccess()) {
        std::cout << " CompleteBucketWorm success " << std::endl;
    }
    else {
        /* Handle exceptions.*/
        std::cout << "CompleteBucketWorm fail" <<
            ",code:" << outcome.error().Code() <<
            ",message:" << outcome.error().Message() <<
            ",requestId:" << outcome.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }
    /* Release network resources.*/
    ShutdownSdk();
    return 0;
}
```

For more information about how to lock a retention policy, see [CompleteBucketWorm](#).

Query a retention policy

The following code provides an example on how to query a retention policy:

```
#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize the Alibaba Cloud account information.*/
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    std::string Endpoint = "yourEndpoint";
    std::string BucketName = "yourBucketName";
    /* Initialize network resources.*/
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    /* Query the retention policy.*/
    auto outcome = client.GetBucketWorm(GetBucketWormRequest(BucketName));
    if (outcome.isSuccess()) {
        std::cout << " GetBucketWorm success " << std::endl;
        std::cout << " CreationDate:" << outcome.result().CreationDate() <<
            ",State:" << outcome.result().State() <<
            ",WormId:" << outcome.result().WormId() << std::endl;
    }
    else {
        /* Handle exceptions.*/
        std::cout << "GetBucketWorm fail" <<
            ",code:" << outcome.error().Code() <<
            ",message:" << outcome.error().Message() <<
            ",requestId:" << outcome.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }
    /* Release network resources.*/
    ShutdownSdk();
    return 0;
}
```

For more information about how to query a retention policy, see [GetBucketWorm](#).

Extend the retention period of a retention policy

The following code provides an example on how to extend the retention period of a locked retention policy:


```
#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize the Alibaba Cloud account information.*/
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    std::string Endpoint = "yourEndpoint";
    std::string BucketName = "yourBucketName";
    std::string WormId = "yourWormId";
    /* Initialize network resources.*/
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    /* Extend the retention period of the locked retention policy.*/
    auto outcome = client.ExtendBucketWormWorm(ExtendBucketWormRequest(BucketName, WormId
, 20));
    if (outcome.IsSuccess()) {
        std::cout << " ExtendBucketWormWorm success " << std::endl;
    }
    else {
        /* Handle exceptions.*/
        std::cout << "ExtendBucketWormWorm fail" <<
            ",code:" << outcome.error().Code() <<
            ",message:" << outcome.error().Message() <<
            ",requestId:" << outcome.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }
    /* Release network resources.*/
    ShutdownSdk();
    return 0;
}
```

For more information about how to extend the retention period of a locked retention policy, see [ExtendBucketWorm](#).

11.5.12. Lifecycle

Object Storage Service (OSS) allows you to configure lifecycle rules to delete expired objects and parts or convert the storage class of expired objects to Infrequent Access (IA) or Archive to reduce your storage costs. This topic describes how to manage lifecycle rules for buckets.

Configure lifecycle rules

The following sample code provides an example on how to configure a lifecycle rule that matches objects by prefix and tag for the bucket named examplebucket:

```
#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize the account information. */
```

```

/* Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account t
o access OSS because the account has permissions on all API operations. We recommend that y
ou use a RAM user to call API operations or perform routine O&M. To create a RAM user, log
on to the RAM console. */
std::string AccessKeyId = "yourAccessKeyId";
std::string AccessKeySecret = "yourAccessKeySecret";
/* Set yourEndpoint to the endpoint of the region in which the bucket is located. For e
xample, if the bucket is located in the China (Hangzhou) region, set Endpoint to https://os
s-cn-hangzhou.aliyuncs.com. */
std::string Endpoint = "yourEndpoint";
/* Specify the name of the bucket. Example: examplebucket. */
std::string BucketName = "examplebucket";
/* Initialize resources such as network resources. */
InitializeSdk();
ClientConfiguration conf;
OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
SetBucketLifecycleRequest request(BucketName);
std::string date("2022-10-12T00:00:00.000Z");
/* Configure tagging. */
Tagging tagging;
tagging.addTag(Tag("key1", "value1"));
tagging.addTag(Tag("key2", "value2"));
/* Specify a lifecycle rule. */
auto rule1 = LifecycleRule();
rule1.setID("rule1");
rule1.setPrefix("test1/");
rule1.setStatus(RuleStatus::Enabled);
rule1.setExpiration(3);
rule1.setTags(tagging.Tags());
/* Specify the expiration date. */
auto rule2 = LifecycleRule();
rule2.setID("rule2");
rule2.setPrefix("test2/");
rule2.setStatus(RuleStatus::Disabled);
rule2.setExpiration(date);
/* rule3 is a lifecycle rule for a bucket that has versioning enabled. */
auto rule3 = LifecycleRule();
rule3.setID("rule3");
rule3.setPrefix("test3/");
rule3.setStatus(RuleStatus::Disabled);
/* Specify that the storage class of objects is converted to Archive 365 days after the
y are last modified. */
auto transition = LifecycleTransition();
transition.Expiration().setDays(365);
transition.setStorageClass(StorageClass::Archive);
rule3.addTransition(transition);
/* Specify that expired delete markers are automatically deleted. */
rule3.setExpiredObjectDeleteMarker(true);
/* Specify that the storage class of objects is converted to Infrequent Access (IA) 10
days after they become previous versions. */
auto transition1 = LifecycleTransition();
transition1.Expiration().setDays(10);
transition1.setStorageClass(StorageClass::IA);
/* Specify that the storage class of objects is converted to Archive 20 days after they

```

```

become previous versions. */
    auto transition2 = LifecycleTransition();
    transition2.Expiration().setDays(20);
    transition2.setStorageClass(StorageClass::Archive);
    /* Specify that objects are deleted 30 days after they become previous versions. */
    auto expiration = LifecycleExpiration(30);
    rule3.setNoncurrentVersionExpiration(expiration);
    LifecycleTransitionList noncurrentVersionStorageTransitions{transition1, transition2};
    rule3.setNoncurrentVersionTransitionList(noncurrentVersionStorageTransitions);
    /* Configure the lifecycle rules. */
    LifecycleRuleList list{rule1, rule2, rule3};
    request.setLifecycleRules(list);
    auto outcome = client.SetBucketLifecycle(request);
    if (!outcome.isSuccess()) {
        /* Handle exceptions. */
        std::cout << "SetBucketLifecycle fail" <<
            ",code:" << outcome.error().Code() <<
            ",message:" << outcome.error().Message() <<
            ",requestId:" << outcome.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }
    /* Release resources such as network resources. */
    ShutdownSdk();
    return 0;
}

```

Query lifecycle rules

The following code provides an example on how to query the lifecycle rules configured for the bucket named `examplebucket`:

```

#include <alibabacloud/oss/OssClient.h>
#include "../src/Utils/Utils.h"
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize the account information. */
    /* Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to
    o access OSS because the account has permissions on all API operations. We recommend that y
    ou use a RAM user to call API operations or perform routine O&M. To create a RAM user, log
    on to the RAM console. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    /* Set yourEndpoint to the endpoint of the region in which the bucket is located. For e
    xample, if the bucket is located in the China (Hangzhou) region, set Endpoint to https://os
    s-cn-hangzhou.aliyuncs.com. */
    std::string Endpoint = "yourEndpoint";
    /* Specify the name of the bucket. Example: examplebucket. */
    std::string BucketName = "examplebucket";
    /* Initialize resources such as network resources. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
}

```

```

/* View the lifecycle rules. */
auto outcome = client.GetBucketLifecycle(BucketName);
if (outcome.IsSuccess()) {
    std::cout << "GetBucketLifecycle success," << std::endl;
    for (auto const rule : outcome.result().LifecycleRules()) {
        std::cout << "rule:" << rule.ID() << "," << rule.Prefix() << "," << rule.Status
() << ","
        "hasExpiration:" << rule.hasExpiration() << "," <<
        "hasTransitionList:" << rule.hasTransitionList() << "," << std::endl;
        auto taglist = rule.Tags();
        for (const auto& tag : taglist)
        {
            std::cout <<"GetBucketLifecycle tag success, Key:"
            << tag.Key() << "; Value:" << tag.Value() << std::endl;
        }
        /* View the lifecycle rules to check whether expired delete markers are automat
ically deleted. */
        if (rule.ExpiredObjectDeleteMarker()) {
            std::cout << "rule expired delete marker: " << rule.ExpiredObjectDeleteMark
er() << std::endl;
        }
        /* View the configurations used to convert the storage class of previous versio
ns of the objects. */
        if (rule.hasNoncurrentVersionTransitionList()) {
            for (auto const lifeCycleTransition : rule.NoncurrentVersionTransitionList(
)) {
                std::cout << "rule noncurrent versions trans days:" << lifeCycleTransit
ion.Expiration() <<
                " trans storage class: " << ToStorageClassName(lifeCycleTransition.Stor
ageClass()) << std::endl;
            }
        }
        /* View the expiration configurations for the previous versions of the objects.
*/
        if (rule.hasNoncurrentVersionExpiration()) {
            std::cout << "rule noncurrent versions expiration days:" << rule.Noncurrent
VersionExpiration().Days() << std::endl;
        }
    }
}
else {
    /* Handle exceptions. */
    std::cout << "GetBucketLifecycle fail" <<
    ",code:" << outcome.error().Code() <<
    ",message:" << outcome.error().Message() <<
    ",requestId:" << outcome.error().RequestId() << std::endl;
    ShutdownSdk();
    return -1;
}
/* Release resources such as networks. */
ShutdownSdk();
return 0;
}

```

Delete lifecycle rules

The following code provides an example on how to delete lifecycle rules configured for the bucket named `examplebucket`:

```
#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize the account information. */
    /* Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to
    o access OSS because the account has permissions on all API operations. We recommend that y
    ou use a RAM user to call API operations or perform routine O&M. To create a RAM user, log
    on to the RAM console. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    /* Set yourEndpoint to the endpoint of the region in which the bucket is located. For e
    xample, if the bucket is located in the China (Hangzhou) region, set Endpoint to https://os
    s-cn-hangzhou.aliyuncs.com. */
    std::string Endpoint = "yourEndpoint";
    /* Specify the name of the bucket. Example: examplebucket. */
    std::string BucketName = "examplebucket";
    /* Initialize resources such as network resources. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    /* Delete the lifecycle rules. */
    DeleteBucketLifecycleRequest request(BucketName);
    auto outcome = client.DeleteBucketLifecycle(request);
    if (!outcome.IsSuccess()) {
        /* Handle exceptions. */
        std::cout << "DeleteBucketLifecycle fail" <<
            ",code:" << outcome.error().Code() <<
            ",message:" << outcome.error().Message() <<
            ",requestId:" << outcome.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }
    /* Release resources such as network resources. */
    ShutdownSdk();
    return 0;
}
```

References

- For more information about the complete sample code for lifecycle rules, visit [GitHub](#).
- For more information about the API operation that you can call to configure lifecycle rules, see [Put Bucket Lifecycle](#).
- For more information about the API operation that you can call to query lifecycle rules, see [Get Bucket Lifecycle](#).
- For more information about the API operation that you can call to delete lifecycle rules, see [Delete Bucket Lifecycle](#).

11.5.13. Pay-by-requester

When pay-by-requester is enabled for a bucket in Object Storage Service (OSS), the requester is charged the requests and traffic fees instead of the bucket owner. The bucket owner is charged only the storage fees. You can enable pay-by-requester for a bucket to share data in the bucket without paying for the request and traffic fees incurred by the access to your bucket.

Enable pay-by-requester

The following code provides an example on how to enable pay-by-requester for a bucket:

```
#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize OSS account information. */
    /* Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to
    o access OSS, because the account has permissions on all API operations. We recommend that
    you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log
    on to the RAM console. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    /* Set yourEndpoint to the endpoint of the region in which the bucket is located. For e
    xample, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https:
    //oss-cn-hangzhou.aliyuncs.com. */
    std::string Endpoint = "yourEndpoint";
    /* Specify the name of the bucket. Example: examplebucket. */
    std::string BucketName = "examplebucket";
    /* Initialize resources such as network resources. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    /* Enable pay-by-requester for the bucket. */
    SetBucketRequestPaymentRequest request(BucketName);
    request.setRequestPayer(RequestPayer::Requester);
    auto outcome = client.SetBucketRequestPayment(request);
    if (!outcome.isSuccess()) {
        /* Handle exceptions. */
        std::cout << "SetBucketRequestPayment fail" <<
        ",code:" << outcome.error().Code() <<
        ",message:" << outcome.error().Message() <<
        ",requestId:" << outcome.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }
    /* Release resources such as network resources. */
    ShutdownSdk();
    return 0;
}
```

Query pay-by-requester configurations

The following code provides an example on how to query the pay-by-requester configurations of a bucket:

```
#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize OSS account information. */
    /* Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to
    o access OSS, because the account has permissions on all API operations. We recommend that
    you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log
    on to the RAM console. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    /* Set yourEndpoint to the endpoint of the region in which the bucket is located. For e
    xample, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https:
    //oss-cn-hangzhou.aliyuncs.com. */
    std::string Endpoint = "yourEndpoint";
    /* Specify the name of the bucket. Example: examplebucket. */
    std::string BucketName = "examplebucket";
    /* Initialize resources such as network resources. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    /* Query the pay-by-requester configurations of the bucket. */
    GetBucketRequestPaymentRequest request(BucketName);
    auto outcome = client.GetBucketRequestPayment(request);
    if (outcome.isSuccess())
    {
        std::cout << "GetBucketRequestPayment success Payer:" << outcome.result().Payer() <
    < std::endl;
    }
    else {
        /* Handle exceptions. */
        std::cout << "GetBucketPayment fail" <<
        ",code:" << outcome.error().Code() <<
        ",message:" << outcome.error().Message() <<
        ",requestId:" << outcome.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }
    /* Release resources such as network resources. */
    ShutdownSdk();
    return 0;
}
```

Specify that third parties are charged for access to objects

If you specify that third parties are charged for access to objects in a bucket, requesters must include the `x-oss-request-payer:requester` header in the HTTP requests to perform operations on your objects. If this header is not included, an error is returned.

The following code provides an example on how to specify that third parties are charged when they access objects by calling the PutObject, GetObject, and DeleteObject operations. You can use the method to specify that third parties are charged when they perform read and write operations on objects by calling other API operations in the similar way.

```
#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize OSS account information. */
    /* Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to
    o access OSS, because the account has permissions on all API operations. We recommend that
    you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log
    on to the RAM console. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    /* Specify the endpoint of the region in which the bucket is located. For example, if the
    bucket is located in the China (Hangzhou) region, set Endpoint to https://oss-cn-hangzhou.
    aliyuncs.com. */
    std::string Endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
    /* Specify the AccessKey pair of the requester account. */
    std::string PayerAccessKeyId = "yourPayerAccessKeyId";
    std::string PayerAccessKeySecret = "yourPayerAccessKeySecret";
    /* Specify the endpoint of the region in which the bucket that the requester wants to access
    is located. */
    std::string PayerEndpoint = "yourPayerEndpoint";
    /* Specify the name of the bucket that the requester wants to access. Example: examplebucket. */
    std::string BucketName = "yourBucketName";
    /* Specify the UID of the requester. Example: 148562088256****. */
    std::string PayerUID = "148562088256****";
    /* Specify the full path of the object that the requester wants to access. The path can
    not contain the bucket name. Example: exampledir/exampleobject.txt. */
    std::string ObjectName = "exampledir/exampleobject.txt";
    /* Initialize resources such as network resources. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    OssClient payerClient(PayerEndpoint, PayerAccessKeyId, PayerAccessKeySecret, conf);
    /* Create the bucket. */
    CreateBucketOutcome outcome = client.CreateBucket(CreateBucketRequest(BucketName));
    /* Enable pay-by-requester for the bucket. */
    SetBucketRequestPaymentRequest request(BucketName);
    request.setRequestPayer(RequestPayer::Requester);
    auto payoutcome = client.SetBucketRequestPayment(request);
    /* Enable pay-by-requester for object uploads. */
    *content << "test cpp sdk";
    PutObjectRequest putrequest(BucketName, ObjectName, content);
    putrequest.setRequestPayer(RequestPayer::Requester);
    auto putoutcome = payerClient.PutObject(putrequest);
    /* Enable pay-by-requester for object downloads to the local memory. */
    GetObjectRequest getrequest(BucketName, ObjectName);
    getrequest.setRequestPayer(RequestPayer::Requester);
    auto getoutcome = payerClient.GetObject(getrequest);
}
```



```
/* Enable pay-by-requester for object deletion. */
DeleteObjectRequest delrequest(BucketName, ObjectName);
delrequest.setRequestPayer(RequestPayer::Requester);
auto deloutcome = payerClient.DeleteObject(delrequest);
/* Release resources such as network resources. */
ShutdownSdk();
return 0;
}
```

References

- For more information about the complete sample code that is used to enable pay-by-requester for a bucket, visit [GitHub](#).
- For more information about the API operation that you can call to enable pay-by-requester for a bucket, see [PutBucketRequestPayment](#).
- For more information about the API operation that you can call to query the pay-by-requester configurations of a bucket, see [GetBucketRequestPayment](#).

11.5.14. Hotlink protection

This topic describes how to use hotlink protection.

To prevent your data on OSS from being leached, OSS supports hotlink protection through the referer field settings in the HTTP header, including the following parameters:

- Referrer whitelist: Used to allow access only for specified domains to OSS data.
- Empty referer: Determines whether the referer can be empty. If it is not allowed, only requests with the referer filed in their HTTP or HTTPS headers can access OSS data.

For more information about hotlink protection, see [Hotlink protection](#) in OSS Developer Guide.

Configure hotlink protection

The following code provides an example on how to configure a Referrer whitelist for a bucket:

```
#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize OSS account information. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    std::string Endpoint = "yourEndpoint";
    std::string BucketName = "yourBucketName";
    /* Initialize resources such as network resources. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    /* Configure hotlink protection. */
    SetBucketRefererRequest request(BucketName);
    request.addReferer("http://example.com");
    request.addReferer("https://example.com");
    request.addReferer("https://www?.example.com");
    request.addReferer("https://www.*.cn");
    request.setAllowEmptyReferer(false);
    auto outcome = client.SetBucketReferer(request);
    if (!outcome.isSuccess()) {
        /* Handle exceptions. */
        std::cout << "SetBucketReferer fail" <<
            ",code:" << outcome.error().Code() <<
            ",message:" << outcome.error().Message() <<
            ",requestId:" << outcome.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }
    /* Release resources such as network resources. */
    ShutdownSdk();
    return 0;
}
```

Query hotlink protection configurations

The following code provides an example on how to query a Referer whitelist of a bucket:

```
#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize OSS account information. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    std::string Endpoint = "yourEndpoint";
    std::string BucketName = "yourBucketName";
    /* Initialize resources such as network resources. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    /* Query hotlink protection configurations. */
    GetBucketRefererRequest request(BucketName);
    auto outcome = client.GetBucketReferer(request);
    if (outcome.isSuccess()) {
        std::cout << " GetBucketReferer success, AllowEmptyReferer: " << outcome.result().AllowEmptyReferer() <<
        ",Referer size: " << outcome.result().RefererList().size() << std::endl;
    }
    else {
        /* Handle exceptions. */
        std::cout << "GetBucketReferer fail" <<
        ",code:" << outcome.error().Code() <<
        ",message:" << outcome.error().Message() <<
        ",requestId:" << outcome.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }
    /* Release resources such as network resources. */
    ShutdownSdk();
    return 0;
}
```

Delete hotlink protection configurations

The following code provides an example on how to clear a Referer whitelist of a bucket:

```
#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize OSS account information. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    std::string Endpoint = "yourEndpoint";
    std::string BucketName = "yourBucketName";
    /* Initialize resources such as network resources. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    /* You cannot directly delete hotlink protection configurations. To delete hotlink protection configurations, you must create a rule that allows empty Referer fields to replace the existing rule. */
    SetBucketRefererRequest request(BucketName);
    request.setAllowEmptyReferer(true);
    auto outcome = client.SetBucketReferer(request);
    if (!outcome.isSuccess()) {
        /* Handle exceptions. */
        std::cout << "CleanBucketReferer fail" <<
            ",code:" << outcome.error().Code() <<
            ",message:" << outcome.error().Message() <<
            ",requestId:" << outcome.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }
    /* Release resources such as network resources. */
    ShutdownSdk();
    return 0;
}
```

11.5.15. Logging

A large number of logs are generated when Object Storage Service (OSS) resources are accessed. After you enable and configure logging for a bucket, OSS generates logs every hour in accordance with predefined naming conventions and then stores the logs as objects in a specified bucket.

Enable logging for a bucket

The following code provides an example on how to enable logging for a bucket:

```

#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize the information about the account that is used to access Object Storage S
ervice (OSS). */
    /* The AccessKey pair of an Alibaba Cloud account has permissions on all API operations
. Using these credentials to perform operations in OSS is a high-risk operation. We recomme
nd that you use a RAM user to call API operations or perform routine O&M. To create a RAM u
ser, log on to the RAM console. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    /* Set yourEndpoint to the endpoint of the region in which the bucket is located. For e
xample, if the bucket is located in the China (Hangzhou) region, set the endpoint to https:
//oss-cn-hangzhou.aliyuncs.com. */
    std::string Endpoint = "yourEndpoint";
    /* Specify the name of the bucket for which you want to enable logging. Example: exampl
ebucket. */
    std::string BucketName = "examplebucket";
    /* Specify the name of the destination bucket used to store the log objects. The source
bucket and the destination bucket can be the same bucket or different buckets. */
    std::string TargetBucketName = "destbucket";
    /* Specify that the log objects are stored in the log/ directory. If you specify this p
arameter, log objects are stored in a specified directory of the destination bucket. If you
do not specify this parameter, log objects are stored in the root directory of the destinat
ion bucket. */
    std::string TargetPrefix = "log/";
    /* Initialize resources such as networks. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    /* Enable logging for the bucket. */
    SetBucketLoggingRequest request(BucketName, TargetBucketName, TargetPrefix);
    auto outcome = client.SetBucketLogging(request);
    if (!outcome.IsSuccess()) {
        /* Handle exceptions. */
        std::cout << "SetBucketLogging fail" <<
            ",code:" << outcome.error().Code() <<
            ",message:" << outcome.error().Message() <<
            ",requestId:" << outcome.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }
    /* Release resources such as networks. */
    ShutdownSdk();
    return 0;
}

```

View the logging configurations of a bucket

The following code provides an example on how to view the logging configurations of a bucket:

```

#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize the information about the account that is used to access OSS. */
    /* The AccessKey pair of an Alibaba Cloud account has permissions on all API operations
     . Using these credentials to perform operations in OSS is a high-risk operation. We recomme
     nd that you use a RAM user to call API operations or perform routine O&M. To create a RAM u
     ser, log on to the RAM console. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    /* Set yourEndpoint to the endpoint of the region in which the bucket is located. For e
     xample, if the bucket is located in the China (Hangzhou) region, set the endpoint to https:
     //oss-cn-hangzhou.aliyuncs.com. */
    std::string Endpoint = "yourEndpoint";
    /* Specify the bucket name. Example: examplebucket. */
    std::string BucketName = "examplebucket";
    /* Initialize resources such as networks. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    /* View the logging configurations of the bucket. */
    GetBucketLoggingRequest request(BucketName);
    auto outcome = client.GetBucketLogging(request);
    if (outcome.isSuccess()) {
        std::cout << " GetBucketLogging success, TargetBucket: " << outcome.result().TargetB
        ucket() <<
            ",TargetPrefix: " << outcome.result().TargetPrefix() << std::endl;
    }
    else {
        /* Handle exceptions. */
        std::cout << "GetBucketLogging fail" <<
            ",code:" << outcome.error().Code() <<
            ",message:" << outcome.error().Message() <<
            ",requestId:" << outcome.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }
    /* Release resources such as networks. */
    ShutdownSdk();
    return 0;
}

```

Disable logging for a bucket

The following code provides an example on how to disable logging for a bucket:

```
#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize the information about the account that is used to access OSS. */
    /* The AccessKey pair of an Alibaba Cloud account has permissions on all API operations
     . Using these credentials to perform operations in OSS is a high-risk operation. We recomme
     nd that you use a RAM user to call API operations or perform routine O&M. To create a RAM u
     ser, log on to the RAM console. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    /* Set yourEndpoint to the endpoint of the region in which the bucket is located. For e
     xample, if the bucket is located in the China (Hangzhou) region, set the endpoint to https:
     //oss-cn-hangzhou.aliyuncs.com. */
    std::string Endpoint = "yourEndpoint";
    /* Specify the bucket name. Example: examplebucket. */
    std::string BucketName = "examplebucket";
    /* Initialize resources such as networks. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    /* Disable logging for the bucket. */
    DeleteBucketLoggingRequest request(BucketName);
    auto outcome = client.DeleteBucketLogging(request);
    if (!outcome.IsSuccess()) {
        /* Handle exceptions. */
        std::cout << "DeleteBucketLogging fail" <<
            ",code:" << outcome.error().Code() <<
            ",message:" << outcome.error().Message() <<
            ",requestId:" << outcome.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }
    /* Release resources such as networks. */
    ShutdownSdk();
    return 0;
}
```

References

- For the complete sample code that is used to configure logging for a bucket, visit [GitHub](#).
- For more information about the API operation that you can call to enable logging for a bucket, see [PutBucketLogging](#).
- For more information about the API operation that you can call to view the logging configurations of a bucket, see [GetBucketLogging](#).
- For more information about the API operation that you can call to disable logging for a bucket, see [DeleteBucketLogging](#).

11.5.16. Static website hosting

You can enable static website hosting for buckets. After you host a static website on a bucket, you can access the bucket to visit the website. You are automatically redirected to the specified default homepage or default 404 page.

Configure static website hosting

The following code provides an example on how to configure static website hosting:

```
#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize the information about the account that is used to access Object Storage Service (OSS). */
    /* The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    /* Specify the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com. */
    std::string Endpoint = "yourEndpoint";
    /* Specify the bucket name. Example: examplebucket. */
    std::string BucketName = "examplebucket";
    /* Initialize resources such as networks. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    /* Configure static website hosting. */
    SetBucketWebsiteRequest request(BucketName);
    /* Set the default homepage to index.html for the static website hosted on the bucket. */
    request.setIndexDocument("index.html");
    /* Set the default 404 page to error.html for the static website hosted on the bucket. */
    request.setErrorDocument("error.html");
    auto outcome = client.SetBucketWebsite(request);
    if (!outcome.isSuccess()) {
        /* Handle exceptions. */
        std::cout << "SetBucketWebsite fail" <<
            ",code:" << outcome.error().Code() <<
            ",message:" << outcome.error().Message() <<
            ",requestId:" << outcome.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }
    /* Release resources such as networks. */
    ShutdownSdk();
    return 0;
}
```


Query static website hosting configurations

The following code provides an example on how to query static website hosting configurations:

```
#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize the information about the account that is used to access OSS. */
    /* The AccessKey pair of an Alibaba Cloud account has permissions on all API operations
     . Using these credentials to perform operations in OSS is a high-risk operation. We recomme
     nd that you use a RAM user to call API operations or perform routine O&M. To create a RAM u
     ser, log on to the RAM console. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    /* Specify the endpoint of the region in which the bucket is located. For example, if t
     he bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-han
     gzhou.aliyuncs.com. */
    std::string Endpoint = "yourEndpoint";
    /* Specify the bucket name. Example: examplebucket. */
    std::string BucketName = "examplebucket";
    /* Initialize resources such as networks. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    /* Query the static website hosting configurations. */
    GetBucketWebsiteRequest request(BucketName);
    auto outcome = client.GetBucketWebsite(request);
    if (outcome.isSuccess()) {
        std::cout << "GetBucketWebsite success, IndexDocument: " << outcome.result().IndexDo
        cument() <<
        " ,ErrorDocument: " << outcome.result().ErrorDocument() << std::endl;
    }
    else {
        /* Handle exceptions. */
        std::cout << "GetBucketWebsite fail" <<
        ",code:" << outcome.error().Code() <<
        ",message:" << outcome.error().Message() <<
        ",requestId:" << outcome.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }
    /* Release resources such as networks. */
    ShutdownSdk();
    return 0;
}
```

Delete static website hosting configurations

The following code provides an example on how to delete static website hosting configurations:

```
#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize the information about the account that is used to access OSS. */
    /* The AccessKey pair of an Alibaba Cloud account has permissions on all API operations
     . Using these credentials to perform operations in OSS is a high-risk operation. We recommen
     nd that you use a RAM user to call API operations or perform routine O&M. To create a RAM u
     ser, log on to the RAM console. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    /* Specify the endpoint of the region in which the bucket is located. For example, if t
     he bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-han
     gzhou.aliyuncs.com. */
    std::string Endpoint = "yourEndpoint";
    /* Specify the bucket name. Example: examplebucket. */
    std::string BucketName = "examplebucket";
    /* Initialize resources such as networks. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    /* Delete the static website hosting configurations. */
    DeleteBucketWebsiteRequest request(BucketName);
    auto outcome = client.DeleteBucketWebsite(request);
    if (!outcome.IsSuccess()) {
        /* Handle exceptions. */
        std::cout << "DeleteBucketWebsite fail" <<
            ",code:" << outcome.error().Code() <<
            ",message:" << outcome.error().Message() <<
            ",requestId:" << outcome.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }
    /* Release resources such as networks. */
    ShutdownSdk();
    return 0;
}
```

References

- For the complete sample code that is used to configure static website hosting, visit [GitHub](#).
- For more information about the API operation that you can call to configure static website hosting, see [PutBucketWebsite](#).
- For more information about the API operation that you can call to query static website hosting configurations, see [GetBucketWebsite](#).
- For more information about the API operation that you can call to delete static website hosting configurations, see [DeleteBucketWebsite](#).

11.5.17. CORS

Cross-origin resource sharing (CORS) is a standard cross-origin solution provided by HTML5 to allow web application servers to control cross-origin access and secure cross-origin data transmission.

Configure CORS rules

The following code provides an example on how to configure CORS rules for a specific bucket:

```
#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize OSS account information. */
    /* The AccessKey pair of an Alibaba Cloud account has permissions on all API operations
     . Using these credentials to access OSS is a high-risk operation. We recommend that you use
     a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to t
     he RAM console. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    /* Set yourEndpoint to the endpoint of the region in which the bucket is located. For e
     xample, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https:
     //oss-cn-hangzhou.aliyuncs.com. */
    std::string Endpoint = "yourEndpoint";
    /* Specify the name of the bucket. Example: examplebucket. */
    std::string BucketName = "examplebucket";
    /* Initialize resources such as network resources. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    SetBucketCorsRequest request(BucketName);
    /* Configure CORS rules. */
    auto rule1 = CORSRule();
    /* Specify the origins from which cross-origin requests are allowed. */
    rule1.addAllowedOrigin("http://example.com");
    /* Specify the HTTP methods that cross-origin requests are allowed to use, including GE
     T, PUT, POST, DELETE, and HEAD. */
    rule1.addAllowedMethod("POST");
    /* Specify whether the headers that are specified by Access-Control-Request-Headers in
     the OPTIONS preflight request are allowed. */
    rule1.addAllowedHeader("*");
    /* Specify the response headers for allowed access requests from applications. */
    rule1.addExposeHeader("x-oss-test");
    /* You can configure up to 10 CORS rules for the bucket. */
    request.addCORSRule(rule1);
    auto rule2 = CORSRule();
    rule2.addAllowedOrigin("http://example.net");
    rule2.addAllowedMethod("GET");
    rule2.addExposeHeader("x-oss-test2");
    rule2.setMaxAgeSeconds(100);
    request.addCORSRule(rule2);
    auto outcome = client.SetBucketCors(request);
    if (!outcome.isSuccess()) {
        /* Handle exceptions. */
        std::cout << "SetBucketCors fail" <<
            ",code:" << outcome.error().Code() <<
            ".message:" << outcome.error().Message() <<

```

```
        , message = "-----", message = "  
        ", requestId:" << outcome.error().RequestId() << std::endl;  
        ShutdownSdk();  
        return -1;  
    }  
    /* Release resources such as network resources. */  
    ShutdownSdk();  
    return 0;  
}
```

Query CORS rules

The following code provides an example on how to query CORS rules that are configured for a specific bucket:

```

#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize OSS account information. */
    /* The AccessKey pair of an Alibaba Cloud account has permissions on all API operations
     . Using these credentials to access OSS is a high-risk operation. We recommend that you use
     a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to t
     he RAM console. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    /* Set yourEndpoint to the endpoint of the region in which the bucket is located. For e
     xample, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https:
     //oss-cn-hangzhou.aliyuncs.com. */
    std::string Endpoint = "yourEndpoint";
    /* Specify the name of the bucket. Example: examplebucket. */
    std::string BucketName = "examplebucket";
    /* Initialize resources such as network resources. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, c onf);
    /* Query CORS rules. */
    auto outcome = client.GetBucketCors(BucketName);
    if (outcome.isSuccess()) {
        std::cout << "GetBucketCors success" << std::endl;
        for (auto const rule : outcome.result().CORSRules()) {
            std::cout << "Get Bucket Cors List:" << std::endl;
            for (auto const origin : rule.AllowedOrigins()) {
                std::cout << "Allowed origin:" << origin << std::endl;
            }
        }
    }
    else {
        /* Handle exceptions. */
        std::cout << "GetBucketCors fail" <<
            ",code:" << outcome.error().Code() <<
            ",message:" << outcome.error().Message() <<
            ",requestId:" << outcome.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }
    /* Release resources such as network resources. */
    ShutdownSdk();
    return 0;
}

```

Delete CORS rules

The following code provides an example on how to delete the CORS rules that are configured for a specific bucket:

```
#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize OSS account information. */
    /* The AccessKey pair of an Alibaba Cloud account has permissions on all API operations
     . Using these credentials to access OSS is a high-risk operation. We recommend that you use
     a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to t
     he RAM console. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    /* Set yourEndpoint to the endpoint of the region in which the bucket is located. For e
     xample, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https:
     //oss-cn-hangzhou.aliyuncs.com. */
    std::string Endpoint = "yourEndpoint";
    /* Specify the name of the bucket. Example: examplebucket. */
    std::string BucketName = "examplebucket";
    /* Initialize resources such as network resources. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    /* Delete CORS rules. */
    auto outcome = client.DeleteBucketCors(BucketName);
    if (!outcome.IsSuccess()) {
        /* Handle exceptions. */
        std::cout << "DeleteBucketCors fail" <<
            ",code:" << outcome.error().Code() <<
            ",message:" << outcome.error().Message() <<
            ",requestId:" << outcome.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }
    /* Release resources such as network resources. */
    ShutdownSdk();
    return 0;
}
```

References

- For more information about the complete sample code of CORS, visit [GitHub](#).
- For more information about the API operation that you can call to configure CORS rules, see [PutBucketCors](#).
- For more information about the API operation that you can call to query CORS rules, see [GetBucketCors](#).
- For more information about the API operation that you can call to delete CORS rules, see [DeleteBucketCors](#).

11.6. Upload objects

11.6.1. Overview

Objects are the basic unit for data operations in OSS. The OSS C++ SDK provides the following object upload methods:

- **Simple upload**: supports the upload of a file up to 5 GB in size. It includes upload from memory and local file upload.
- **Append upload**: supports the upload of a file up to 5 GB in size.
- **Resumable upload**: supports the upload of a file up to 48.8 TB in size. It applies to the upload of large files. It supports concurrent upload and you can define the size of each part.
- **Multipart upload**: supports the upload of files up to 48.8 TB in size. It applies to the upload of large files.

 **Note** For more information about the scenarios of each upload method, see the "Upload files" section in OSS Developer Guide.

During the file upload, you can set Object Meta and view upload progress, as shown in [Configure Object Meta](#) and [Use progress bars](#). After you upload the file, you can perform upload callback, as shown in [Upload callback](#).

11.6.2. Simple upload

This topic describes how to upload objects from memory or local disks.

Upload an object from memory

The following code provides an example on how to upload data from memory to an object named `exampleobject.txt` in the `exampledir` directory of a bucket named `examplebucket`:

```

#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize the information about the account that is used to access OSS. */
    /* Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account t
o access OSS, because the account has permissions on all API operations. We recommend that
you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log
on to the RAM console. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    /* Set yourEndpoint to the endpoint of the region in which the bucket is located. For e
xample, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https:
//oss-cn-hangzhou.aliyuncs.com. */
    std::string Endpoint = "yourEndpoint";
    /* Specify the name of the bucket. Example: examplebucket. */
    std::string BucketName = "examplebucket";
    /* Specify the full path of the object. The full path cannot contain the bucket name. E
xample: exampledir/exampleobject.txt. */
    std::string ObjectName = "exampledir/exampleobject.txt";
    /* Initialize resources such as network resources. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    std::shared_ptr<std::iostream> content = std::make_shared<std::stringstream>();
    *content << "Thank you for using Alibaba Cloud Object Storage Service!";
    PutObjectRequest request(BucketName, ObjectName, content);
    /* (Optional) Set the access control list (ACL) to private and the storage class to Sta
ndard for the object that you want to upload by referring to the following sample code. */
    //request.Metadata().addHeader("x-oss-object-acl", "private");
    //request.Metadata().addHeader("x-oss-storage-class", "Standard");
    /* Upload the object. */
    auto outcome = client.PutObject(request);
    if (!outcome.IsSuccess()) {
        /* Handle exceptions. */
        std::cout << "PutObject fail" <<
            ",code:" << outcome.error().Code() <<
            ",message:" << outcome.error().Message() <<
            ",requestId:" << outcome.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }
    /* Release resources such as network resources. */
    ShutdownSdk();
    return 0;
}

```

Upload a local file

The following code provides an example on how to upload a file named `examplefile.txt` from a directory named `D:\localpath` on the local disk to the `exampledir` directory of a bucket named `examplebucket`. The uploaded file is stored as an object named `exampleobject.txt` in OSS.


```

#include <alibabacloud/oss/OssClient.h>
#include <fstream>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize the information about the account that is used to access OSS. */
    /* Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to
    o access OSS, because the account has permissions on all API operations. We recommend that
    you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log
    on to the RAM console. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    /* Set yourEndpoint to the endpoint of the region in which the bucket is located. For e
    xample, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https:
    //oss-cn-hangzhou.aliyuncs.com. */
    std::string Endpoint = "yourEndpoint";
    /* Specify the name of the bucket. Example: examplebucket. */
    std::string BucketName = "examplebucket";
    /* Specify the full path of the object. The full path cannot contain the bucket name. E
    xample: exampledir/exampleobject.txt. */
    std::string ObjectName = "exampledir/exampleobject.txt";
    /* Initialize resources such as network resources. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    /* Specify the full path of the local file that you want to upload, for example, D:\\lo
    calpath\\examplefile.txt. In D:\\localpath\\examplefile.txt, localpath indicates the local
    path in which the file is stored. */
    std::shared_ptr<std::iostream> content = std::make_shared<std::fstream>("D:\\localpath\\
    \\examplefile.txt", std::ios::in | std::ios::binary);
    PutObjectRequest request(BucketName, ObjectName, content);
    /* (Optional) Set the ACL to private and the storage class to Standard for the object t
    hat you want to upload by referring to the following sample code. */
    //request.Metadata().addHeader("x-oss-object-acl", "private");
    //request.Metadata().addHeader("x-oss-storage-class", "Standard");
    auto outcome = client.PutObject(request);
    if (!outcome.IsSuccess()) {
        /* Handle exceptions. */
        std::cout << "PutObject fail" <<
        ",code:" << outcome.error().Code() <<
        ",message:" << outcome.error().Message() <<
        ",requestId:" << outcome.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }
    /* Release resources such as network resources. */
    ShutdownSdk();
    return 0;
}

```

References

- For more information about the complete sample code for simple upload, visit [GitHub](#).

- For more information about the API operation that you can call to perform simple upload, see [Put Object](#).

11.6.3. Append upload

You can call the `AppendObject` operation to append content to existing appendable objects.

Limits

- If the object to which you want to append content does not exist, an appendable object is created when you call the `AppendObject` operation.
- When you call the `AppendObject` operation to append content to an object, you may encounter a few situations: If the object is an appendable object, and the specified position from which the append operation starts is equal to the current object length, the content is appended to the end of the object. If the specified position from which the append operation starts is not equal to the current object length, a `PositionNotEqualToLength` exception is thrown. If the object is not an appendable object, an `ObjectNotAppendable` exception is thrown.
- The `CopyObject` operation cannot be performed on appendable objects.

Sample code

The following code provides an example on how to perform append upload operation:

```
#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize the information about the account that is used to access OSS. */
    /* Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account t
o access OSS because the account has permissions on all API operations. We recommend that y
ou use a RAM user to call API operations or perform routine O&M. To create a RAM user, log
on to the RAM console. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    /* Specify the endpoint of the region in which the bucket is located. For example, if t
he bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-han
gzhou.aliyuncs.com. */
    std::string Endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
    /* Specify the name of the bucket. Example: examplebucket. */
    std::string BucketName = "examplebucket";
    /* Specify the full path of the object. The full path cannot contain the bucket name. E
xample: exampledir/exampleobject.txt. */
    std::string ObjectName = "exampledir/exampleobject.txt";
    /* Initialize resources such as networks. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    auto meta = ObjectMetadata();
    meta.setContentType("text/plain");
    /* If the object is appended for the first time, set the position from which the append
operation starts to 0. The position for the next append operation is included in the respon
se. The position from which the next append operation starts is the current length of the o
bject. */
    std::shared_ptr<std::istream> content1 = std::make_shared<std::stringstream>();
```

```

std::shared_ptr<std::iostream> content1 = std::make_shared<std::stringstream>();
*content1 << "Thank you for using Aliyun Object Storage Service!";
AppendObjectRequest request(BucketName, ObjectName, content1, meta);
request.setPosition(0L);
/* Perform the first append operation. */
auto result = client.AppendObject(request);
if (!result.isSuccess()) {
    /* Handle exceptions. */
    std::cout << "AppendObject fail" <<
        ",code:" << result.error().Code() <<
        ",message:" << result.error().Message() <<
        ",requestId:" << result.error().RequestId() << std::endl;
    ShutdownSdk();
    return -1;
}

std::shared_ptr<std::iostream> content2 = std::make_shared<std::stringstream>();
*content2 << "Thank you for using Aliyun Object Storage Service!";
auto position = result.result().Length();
AppendObjectRequest appendObjectRequest(BucketName, ObjectName, content2);
appendObjectRequest.setPosition(position);
/* Perform the second append operation. */
auto outcome = client.AppendObject(appendObjectRequest);
if (!outcome.isSuccess()) {
    /* Handle exceptions. */
    std::cout << "AppendObject fail" <<
        ",code:" << outcome.error().Code() <<
        ",message:" << outcome.error().Message() <<
        ",requestId:" << outcome.error().RequestId() << std::endl;
    ShutdownSdk();
    return -1;
}

/* Release resources such as networks. */
ShutdownSdk();
return 0;
}

```

References

- For more information about the complete sample code that is used to perform append upload operation, visit [GitHub](#).
- For more information about the API operation that you can call to perform append upload, see [AppendObject](#).

11.6.4. Resumable upload

Before you upload an object to Object Storage Service (OSS) by using resumable upload, you can specify a directory for the checkpoint file that stores resumable upload records. If an object fails to upload due to a network exception or program error, the upload task is resumed from the position recorded in the checkpoint file to upload the remaining data.

Usage notes

- During resumable upload, the upload progress is recorded in a checkpoint file. If a part fails to be

uploaded, the next upload starts from the position recorded in the checkpoint file. After the object is uploaded, the checkpoint file is deleted.

- The upload progress is recorded in the checkpoint file. Make sure that you have write permissions on the checkpoint file.
- The checkpoint file contains a checksum. This checksum cannot be edited. If the checkpoint file is damaged, you must upload all parts of the object again.
- If the local file is modified during the upload, you must upload all parts of the object again.

Implementation method

You can call the `client.ResumableUploadObject` method to perform resumable upload. The following table describes the parameters you can configure for `UploadObjectRequest`.

Parameter	Description	Required	Default value	Configuration method
bucket	The name of the bucket.	Yes	None	Constructor
key	The name of the object uploaded to OSS.	Yes	None	
filePath	The name of the local file to upload.	No	None	
partSize	The size of each part to upload. Valid values: 100 KB to 5 GB.	No	8 MB	setPartSize
threadNum	The number of parts to upload simultaneously.	No	3	Constructor or setThreadNum
checkpointDir	The checkpoint file that records the results of the multipart upload task. This file stores information about upload progress. If a part cannot be uploaded, the task can be resumed based on the progress recorded in the checkpoint file. After the object is uploaded, the checkpoint file is deleted.	No	The directory in which the local file to upload is stored.	Constructor or setCheckpointDir

Sample code

The following code provides an example on how to perform resumable upload by using multipart upload:

```

#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize the information about the account used to access OSS. */
    /* Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to
    o access OSS because the account has permissions on all API operations. We recommend that y
    ou use a RAM user to call API operations or perform routine O&M. To create a RAM user, log
    on to the RAM console. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    /* Set yourEndpoint to the endpoint of the region in which the bucket is located. For e
    xample, if the bucket is located in the China (Hangzhou) region, set the endpoint to https:
    //oss-cn-hangzhou.aliyuncs.com. */
    std::string Endpoint = "yourEndpoint";
    /* Specify the name of the bucket. Example: examplebucket. */
    std::string BucketName = "examplebucket";
    /* Specify the full path of the object. The full path cannot contain the bucket name. E
    xample: exampledir/exampleobject.txt. */
    std::string ObjectName = "exampledir/exampleobject.txt";
    /* Specify the full path of the local file. Example: D:\\localpath\\examplefile.txt. By
    default, if you do not specify the full path of the local file, the local file is uploaded
    from the path of the project to which the sample program belongs. */
    std::string UploadFilePath = "D:\\localpath\\examplefile.txt";
    /* Specify the checkpoint file that records the results of the multipart upload task. T
    his file stores information about upload progress. If a part cannot be uploaded, the task c
    an be resumed based on the progress recorded in the checkpoint file. After the object is up
    loaded, the checkpoint file is deleted. */
    /* By default, if you do not specify the path of the checkpoint file, the checkpoint fi
    le and the local file to upload share the same path. */
    std::string CheckpointFilePath = "yourCheckpointFilepath"
    /* Initialize resources such as networks. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    /* Start resumable upload. */
    UploadObjectRequest request(BucketName, ObjectName, UploadFilePath, CheckpointFilePath)
;
    auto outcome = client.ResumableUploadObject(request);
    if (!outcome.IsSuccess()) {
        /* Handle exceptions. */
        std::cout << "ResumableUploadObject fail" <<
        ",code:" << outcome.error().Code() <<
        ",message:" << outcome.error().Message() <<
        ",requestId:" << outcome.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }
    /* Release resources such as networks. */
    ShutdownSdk();
    return 0;
}

```

References

For more information about the complete sample code for resumable upload, visit [GitHub](#).

11.6.5. Multipart upload

By using the multipart upload feature provided by OSS, you can split a large object into multiple parts and upload them separately. After all parts are uploaded, call the CompleteMultipartUpload operation to combine these parts into a single object to implement resumable upload.

Process

To implement multipart upload, perform the following steps:

1. Initiate a multipart upload task.

Call the InitiateMultipartUpload method to obtain a unique upload ID in Object Storage Service (OSS).

2. Upload parts.

Call the UploadPart method to upload the parts.

Note

- If parts are uploaded by a multipart upload task that has a specific upload ID, part numbers are used to identify the relative positions of the parts in an object. If you upload a part that has the same part number as an existing part, the existing part is overwritten by the uploaded part.
- OSS includes the MD5 hash of each uploaded part in the ETag header in the response.
- OSS calculates the MD5 hash of uploaded data and compares the MD5 hash with the MD5 hash that is calculated by the SDK. If the two hashes are different, OSS returns the InvalidDigest error code.

3. Complete the multipart upload task.

After all parts are uploaded, call the CompleteMultipartUpload method to combine the parts into a complete object.

Complete sample code

The following code provides a complete example that describes the process of multipart upload:

```
#include <alibabacloud/oss/OssClient.h>
int64_t getFileSize(const std::string& file)
{
    std::fstream f(file, std::ios::in | std::ios::binary);
    f.seekg(0, f.end);
    int64_t size = f.tellg();
    f.close();
    return size;
}
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize the information about the account used to access OSS. */
```

```

// Initialize the information about the account used to access OSS.
std::string AccessKeyId = "yourAccessKeyId";
std::string AccessKeySecret = "yourAccessKeySecret";
std::string Endpoint = "yourEndpoint";
/* Specify the name of the bucket. Example: examplebucket. */
std::string BucketName = "examplebucket";
/* Specify the full path of the object. The full path cannot contain the name of the bu
cket. Example: exampledir/exampleobject.txt. */
std::string ObjectName = "exampledir/exampleobject.txt";
/* Initialize resources such as networks. */
InitializeSdk();
ClientConfiguration conf;
OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
InitiateMultipartUploadRequest initUploadRequest(BucketName, ObjectName);
/* Optional. Specify the storage class. */
//initUploadRequest.Metadata().addHeader("x-oss-storage-class", "Standard");
/* Initiate the multipart upload task. */
auto uploadIdResult = client.InitiateMultipartUpload(initUploadRequest);
auto uploadId = uploadIdResult.result().UploadId();
std::string fileToUpload = "yourLocalFilename";
int64_t partSize = 100 * 1024;
PartList partETagList;
auto fileSize = getFileSize(fileToUpload);
int partCount = static_cast<int>(fileSize / partSize);
/* Calculate the number of parts. */
if (fileSize % partSize != 0) {
    partCount++;
}
/* Upload each part. */
for (int i = 1; i <= partCount; i++) {
    auto skipBytes = partSize * (i - 1);
    auto size = (partSize < fileSize - skipBytes) ? partSize : (fileSize - skipBytes);
    std::shared_ptr<std::iostream> content = std::make_shared<std::fstream>(fileToUploa
d, std::ios::in|std::ios::binary);
    content->seekg(skipBytes, std::ios::beg);
    UploadPartRequest uploadPartRequest(BucketName, ObjectName, content);
    uploadPartRequest.setContentLength(size);
    uploadPartRequest.setUploadId(uploadId);
    uploadPartRequest.setPartNumber(i);
    auto uploadPartOutcome = client.UploadPart(uploadPartRequest);
    if (uploadPartOutcome.isSuccess()) {
        Part part(i, uploadPartOutcome.result().ETag());
        partETagList.push_back(part);
    }
    else {
        std::cout << "uploadPart fail" <<
            "\n,code:" << uploadPartOutcome.error().Code() <<
            "\n,message:" << uploadPartOutcome.error().Message() <<
            "\n,requestId:" << uploadPartOutcome.error().RequestId() << std::endl;
    }
}
/* Complete the multipart upload task. */
CompleteMultipartUploadRequest request(BucketName, ObjectName);
request.setUploadId(uploadId);
request.setPartList(partETagList);

```

```
/* Optional. Specify the ACL of the object. */
//request.setAcl(CannedAccessControlList::Private);
auto outcome = client.CompleteMultipartUpload(request);
if (!outcome.IsSuccess()) {
    /* Handle exceptions. */
    std::cout << "CompleteMultipartUpload fail" <<
        ",code:" << outcome.error().Code() <<
        ",message:" << outcome.error().Message() <<
        ",requestId:" << outcome.error().RequestId() << std::endl;
    ShutdownSdk();
    return -1;
}
/* Release resources such as networks. */
ShutdownSdk();
return 0;
}
```

To call the `CompleteMultipartUpload` operation when you obtain uploaded parts, you must provide the ETag value of each part. You can obtain the ETag value of each part by using one of the following methods:

- When a part is uploaded, the ETag value of the part is included in the response. You can save and use this ETag value. In this example, the first method is used.
- Call the `ListParts` method to query the ETag values of the uploaded parts.

List uploaded parts

You can call the `ListParts` method to list all uploaded parts that use the specified upload ID.

- List all uploaded parts

 **Note** By default, `ListParts` can list up to 1,000 parts at a time. To list more than 1,000 parts, you must list the results by page in the multipart upload operation.

The following code provides an example on how to list all uploaded parts:


```

#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize the information about the account used to access OSS. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    std::string Endpoint = "yourEndpoint";
    /* Specify the name of the bucket. Example: examplebucket. */
    std::string BucketName = "examplebucket";
    /* Specify the full path of the object. The full path cannot contain the name of the
    bucket. Example: exampledir/exampleobject.txt. */
    std::string ObjectName = "exampledir/exampleobject.txt";
    /* Initialize resources such as networks. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    /* List the uploaded parts. By default, 1,000 parts are listed. */
    ListPartsRequest listuploadrequest(BucketName, ObjectName);
    listuploadrequest.setUploadId(uploadId);
    do {
        auto listuploadresult = client.ListParts(listuploadrequest);
        if (!listuploadresult.isSuccess()) {
            /* Handle exceptions. */
            std::cout << "ListParts fail" <<
                ",code:" << listuploadresult.error().Code() <<
                ",message:" << listuploadresult.error().Message() <<
                ",requestId:" << listuploadresult.error().RequestId() << std::endl;
            break;
        }
        else {
            for (const auto& part : listuploadresult.result().PartList()) {
                std::cout << "part"<<
                    ",name:" << part.PartNumber() <<
                    ",size:" << part.Size() <<
                    ",etag:" << part.ETag() <<
                    ",lastmodify time:" << part.LastModified() << std::endl;
            }
            listuploadrequest.setPartNumberMarker(listuploadresult.result().NextPartNumberMarker());
        } while (listuploadresult.result().IsTruncated());
        /* Release resources such as networks. */
        ShutdownSdk();
        return 0;
    }
}

```

- List all uploaded parts by page

The following code provides an example on how to specify the maximum number of parts to list on each page and list all parts by page:

```

#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize the information about the account used to access OSS. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    std::string Endpoint = "yourEndpoint";
    /* Specify the name of the bucket. Example: examplebucket. */
    std::string BucketName = "examplebucket";
    /* Specify the full path of the object. The full path cannot contain the name of the
    bucket. Example: exampledir/exampleobject.txt. */
    std::string ObjectName = "exampledir/exampleobject.txt";
    /* Initialize resources such as networks. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    /* List all uploaded parts by page. */
    /* Set the maximum number of parts to list on each page. */
    ListPartsRequest listuploadrequest(BucketName, ObjectName);
    listuploadrequest.setMaxParts(50);
    listuploadrequest.setUploadId(uploadId);
    do {
        listuploadresult = client.ListParts(listuploadrequest);
        if (!listuploadresult.isSuccess()) {
            /* Handle exceptions. */
            std::cout << "ListParts fail" <<
            ",code:" << listuploadresult.error().Code() <<
            ",message:" << listuploadresult.error().Message() <<
            ",requestId:" << listuploadresult.error().RequestId() << std::endl;
            break;
        }
        else {
            for (const auto& part : listuploadresult.result().PartList()) {
                std::cout << "part"<<
                ",name:" << part.PartNumber() <<
                ",size:" << part.Size() <<
                ",etag:" << part.ETag() <<
                ",lastmodify time:" << part.LastModified() << std::endl;
            }
        }
        listuploadrequest.setPartNumberMarker(listuploadresult.result().NextPartNumberMarker());
    } while (listuploadresult.result().IsTruncated());
    /* Release resources such as networks. */
    ShutdownSdk();
    return 0;
}

```

List multipart upload tasks

You can call the `ListMultipartUploads` method to list all ongoing multipart upload tasks. Ongoing multipart upload tasks are tasks that have been initiated but not completed or tasks that have been canceled.

- List all multipart upload tasks

 **Note** By default, the `ListMultipartUploads` method can list up to 1,000 multipart upload tasks at a time. To list more than 1,000 multipart upload tasks, you must list the results by page in the multipart upload operation..

The following code provides an example on how to list all multipart upload tasks:

```

#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize the information about the account used to access OSS. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    std::string Endpoint = "yourEndpoint";
    /* Specify the name of the bucket. Example: examplebucket. */
    std::string BucketName = "examplebucket";
    /* Initialize resources such as networks. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    /* List the completed multipart upload tasks. By default, 1,000 parts are listed. */
    ListMultipartUploadsRequest listmultiuploadrequest(BucketName);
    do {
        auto listresult = client.ListMultipartUploads(listmultiuploadrequest);
        if (!listresult.isSuccess()) {
            /* Handle exceptions. */
            std::cout << "ListMultipartUploads fail" <<
                ",code:" << listresult.error().Code() <<
                ",message:" << listresult.error().Message() <<
                ",requestId:" << listresult.error().RequestId() << std::endl;
            break;
        }
        else {
            for (const auto& part : listresult.result().MultipartUploadList()) {
                std::cout << "part"<<
                    ",name:" << part.Key <<
                    ",uploadid:" << part.UploadId <<
                    ",initiated time:" << part.Initiated << std::endl;
            }
            listmultiuploadrequest.setKeyMarker(listresult.result().NextKeyMarker());
            listmultiuploadrequest.setUploadIdMarker(listresult.result().NextUploadIdMarker());
        }
    } while (listresult.result().IsTruncated());
    /* Release resources such as networks. */
    ShutdownSdk();
    return 0;
}

```

- List all multipart upload tasks by page

The following code provides an example on how to list all multipart upload tasks by page:

```

#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize the information about the account used to access OSS. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    std::string Endpoint = "yourEndpoint";
    /* Specify the name of the bucket. Example: examplebucket. */
    std::string BucketName = "examplebucket";
    /* Initialize resources such as networks. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    /* List all multipart upload tasks by page. */
    /* Set the maximum number of multipart upload tasks to list on each page. */
    ListMultipartUploadsRequest listmultiuploadrequest(BucketName);
    listmultiuploadrequest.setMaxUploads(50);
    do {
        listresult = client.ListMultipartUploads(listmultiuploadrequest);
        if (!listresult.isSuccess()) {
            /* Handle exceptions. */
            std::cout << "ListMultipartUploads fail" <<
                ",code:" << listresult.error().Code() <<
                ",message:" << listresult.error().Message() <<
                ",requestId:" << listresult.error().RequestId() << std::endl;
            break;
        }
        else {
            for (const auto& part : listresult.result().MultipartUploadList()) {
                std::cout << "part"<<
                    ",name:" << part.Key <<
                    ",uploadid:" << part.UploadId <<
                    ",initiated time:" << part.Initiated << std::endl;
            }
        }
        listmultiuploadrequest.setKeyMarker(listresult.result().NextKeyMarker());
        listmultiuploadrequest.setUploadIdMarker(listresult.result().NextUploadIdMarker());
    };
    } while (listresult.result().IsTruncated());
    /* Release resources such as networks. */
    ShutdownSdk();
    return 0;
}

```

Cancel a multipart upload task

You can call the `client.AbortMultipartUpload` method to cancel a multipart upload task. If a multipart upload task is canceled, the upload ID cannot be used to upload parts. In addition, the uploaded parts are deleted.

The following code provides an example on how to cancel a multipart upload task:

```
#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize the information about the account used to access OSS. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    std::string Endpoint = "yourEndpoint";
    /* Specify the name of the bucket. Example: examplebucket. */
    std::string BucketName = "examplebucket";
    /* Specify the full path of the object. The full path cannot contain the name of the bucket. Example: exampledir/exampleobject.txt. */
    std::string ObjectName = "exampledir/exampleobject.txt";
    /* Initialize resources such as networks. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    InitiateMultipartUploadRequest initUploadRequest(BucketName, ObjectName);
    /* Initiate the multipart upload task. */
    auto uploadIdResult = client.InitiateMultipartUpload(initUploadRequest);
    auto uploadId = uploadIdResult.result().UploadId();
    /* Cancel the multipart upload task. */
    AbortMultipartUploadRequest abortUploadRequest(BucketName, ObjectName, uploadId);
    auto abortUploadIdResult = client.AbortMultipartUpload(abortUploadRequest);
    if (!abortUploadIdResult.isSuccess()) {
        /* Handle exceptions. */
        std::cout << "AbortMultipartUpload fail" <<
            ",code:" << outcome.error().Code() <<
            ",message:" << outcome.error().Message() <<
            ",requestId:" << outcome.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }
    /* Release resources such as networks. */
    ShutdownSdk();
    return 0;
}
```

References

- For more information about the complete sample code for multipart upload, visit [GitHub](#).
- The following API operations are required to perform multipart upload:
 - The API operation that you can call to initiate a multipart upload task. For more information, see [InitiateMultipartUpload](#).
 - The API operation that you can call to upload data by part. For more information, see [UploadPart](#).
 - The API operation that you can call to complete a multipart upload task. For more information, see [CompleteMultipartUpload](#).
- For more information about the API operation that you can call to cancel a multipart upload task, see [AbortMultipartUpload](#).
- For more information about the API operation that you can call to list the uploaded parts, see

List Parts.

- For more information about the API operation that you can call to list all ongoing multipart upload tasks, see [List Multipart Uploads](#). Ongoing multipart upload tasks include tasks that have been initiated but are not completed and tasks that are canceled.

11.6.6. Interrupt an upload task

You can interrupt an upload task at time when the object is being uploaded.

Use the following code to interrupt an upload task:

```
#include <alibabacloud/oss/OssClient.h>
#include <fstream>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize the OSS account information. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    std::string Endpoint = "yourEndpoint";
    std::string BucketName = "yourBucketName";
    std::string ObjectName = "yourObjectName ";
    /* Initialize network resources. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    std::vector<PutObjectOutcomeCallable> Callables;
    for (int i=0; i < 4; i++) {
        std::shared_ptr<std::iostream> content = std::make_shared<std::fstream>("yourLocalFilename", std::ios::in|std::ios::binary);
        PutObjectRequest request(BucketName, ObjectName, content);
        auto outcomeCallable = client.PutObjectCallable(request);
        Callables.emplace_back(std::move(outcomeCallable));
    }
    std::this_thread::sleep_for(std::chrono::milliseconds(1000));
    /* Interrupt the upload task. */
    client.DisableRequest();
    for (size_t i = 0; i < Callables.size(); i++) {
        auto outcome = Callables[i].get();
        if (outcome.error().Code() == "ClientError:100002" ||
            outcome.error().Code() == "ClientError:200042") {
            std::cout << "disable putobject success" << std::endl;
        }
    }
    /* Release network resources. */
    ShutdownSdk();
    return 0;
}
```

 **Note** An interrupted upload and a canceled multipart upload have the following differences:

- When you call the upload API to remotely upload an object to OSS, you can interrupt the upload task. In other words, you stopped the upload task. After you cancel the upload task, OSS no longer processes other multipart upload tasks that use the same upload ID as that of the canceled task.
- If you interrupt the upload task, you can still call the upload ID to continue uploading the content although the current upload task is interrupted. If you cancel a multipart upload task, the upload ID is unavailable for future uploads.
- Interrupted upload applies to all API operations. However, canceled multipart upload applies only to multipart upload.

11.6.7. Use a progress bar

Progress bars are used to indicate the progress of an upload or download.

The following code uses PutObject as an example to show you how to view progress information:


```

#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
void ProgressCallback(size_t increment, int64_t transfered, int64_t total, void* userData)
{
    std::cout << "ProgressCallback[" << userData << "] => " <<
        increment << " , " << transfered << " , " << total << std::endl;
}
int main(void)
{
    /* Initialize the OSS account information. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    std::string Endpoint = "yourEndpoint";
    std::string BucketName = "yourBucketName";
    std::string ObjectName = "yourObjectName";
    /* Initialize network resources. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    std::shared_ptr<std::iostream> content = std::make_shared<std::fstream>("yourLocalFilename", std::ios::in|std::ios::binary);
    PutObjectRequest request(BucketName, ObjectName, content);
    TransferProgress progressCallback = { ProgressCallback , nullptr };
    request.setTransferProgress(progressCallback);
    /* Upload the object. */
    auto outcome = client.PutObject(request);
    if (!outcome.isSuccess()) {
        /* Handle exceptions. */
        std::cout << "PutObject fail" <<
            ",code:" << outcome.error().Code() <<
            ",message:" << outcome.error().Message() <<
            ",requestId:" << outcome.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }
    /* Release network resources. */
    ShutdownSdk();
    return 0;
}

```

11.6.8. Upload callback

After an object is uploaded, Object Storage Service (OSS) can start a callback process for the application server. To implement callback, you need only to send a request that contains relevant callback parameters to OSS.

Sample code

The following code provides an example on how to configure upload callback:

```

#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize the information about the account that is used to access OSS. */
    /* Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account t
o access OSS because the account has permissions on all API operations. We recommend that y
ou use a RAM user to call API operations or perform routine O&M. To create a RAM user, log
on to the RAM console. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    /* Specify the endpoint of the region in which the bucket is located. For example, if t
he bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-han
gzhou.aliyuncs.com. */
    std::string Endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
    /* Specify the name of the bucket. Example: examplebucket. */
    std::string BucketName = "examplebucket";
    /* Specify the full path of the object. The full path of the object cannot contain the
bucket name. Example: exampledir/exampleobject.txt. */
    /* Specify the address of the application server that receives the callback request. Ex
ample: https://example.aliyundoc.com:23450. */
    std::string ServerName = "https://example.aliyundoc.com:23450";
    /* Initialize resources such as networks. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    std::shared_ptr<std::iostream> content = std::make_shared<std::stringstream>();
    *content << "Thank you for using Aliyun Object Storage Service!";
    /* Set the upload callback parameters. */
    std::string callbackBody = "bucket=${bucket}&object=${object}&etag=${etag}&size=${size}
&mimeType=${mimeType}&my_var1=${x:var1}";
    ObjectCallbackBuilder builder(ServerName, callbackBody, "", ObjectCallbackBuilder::Type
::URL);
    std::string value = builder.build();
    ObjectCallbackVariableBuilder varBuilder;
    varBuilder.addCallbackVariable("x:var1", "value1");
    std::string varValue = varBuilder.build();
    PutObjectRequest request(BucketName, ObjectName, content);
    request.Metadata().addHeader("x-oss-callback", value);
    request.Metadata().addHeader("x-oss-callback-var", varValue);
    auto outcome = client.PutObject(request);
    /* Release resources such as networks. */
    ShutdownSdk();
    return 0;
}

```

References

For more information about the API operations that you can call to configure upload callback, see [Callback](#).

11.7. Download objects

11.7.1. Overview

The OSS C++ SDK provides the following object download methods:

- [Download to a local file](#)
- [Download to local memory](#)
- [Range download](#)
- [Resumable download](#)

You can view the download progress, as show in [Use a progress bar](#).

11.7.2. Download objects to local files

This topic describes how to download an object from a bucket to your computer.

Sample code

The following code provides an example on how to download objects to local files:

```
#include <alibabacloud/oss/OssClient.h>
#include <memory>
#include <fstream>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize the information about the account that is used to access OSS. */
    /* Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account t
    o access OSS because the account has permissions on all API operations. We recommend that y
    ou use a RAM user to call API operations or perform routine O&M. To create a RAM user, log
    on to the RAM console. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    /* Specify the endpoint of the region in which the bucket is located. For example, if t
    he bucket is located in the China (Hangzhou) region, set Endpoint to https://oss-cn-hangzho
    u.aliyuncs.com. */
    std::string Endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
    /* Specify the name of the bucket. Example: examplebucket. */
    std::string BucketName = "examplebucket";
    /* Specify the full path of the object. The full path of the object cannot contain the
    bucket name. Example: exampledir/exampleobject.txt. */
    std::string ObjectName = "exampledir/exampleobject.txt";
    /* Download the object to D:\\localpath. After the object is downloaded, the local file
    is named examplefile.txt. If the specified local file exists, the object to download overwr
    ites the file. Otherwise, a file is created. */
    /* If the path for the object is not specified, the downloaded object is saved to the p
    ath of the project to which the sample program belongs. */
    std::string FileNameToSave = "D:\\localpath\\examplefile.txt";
    /* Initialize resources such as your network. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    /* Download the object to the local file. */
    GetObjectRequest request(BucketName, ObjectName);
    request.setResponseStreamFactory([=]() {return std::make_shared<std::fstream>(FileNameTo
```

```
oSave, std::ios_base::out | std::ios_base::in | std::ios_base::trunc | std::ios_base::binary
); });
    auto outcome = client.GetObject(request);
    if (outcome.IsSuccess()) {
        std::cout << "GetObjectToFile success" << outcome.result().Metadata().ContentLength
() << std::endl;
    }
    else {
        /* Handle exceptions. */
        std::cout << "GetObjectToFile fail" <<
        ",code:" << outcome.error().Code() <<
        ",message:" << outcome.error().Message() <<
        ",requestId:" << outcome.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }
    /* Release resources such as networks. */
    ShutdownSdk();
    return 0;
}
```

References

- For more information about the complete sample code that is used to download objects to local files, visit [GitHub](#).
- For more information about the API operation that you can call to download objects to local files, see [GetObject](#).

11.7.3. Download objects to the local memory

This topic describes how to download an object to the local memory.

The following code provides an example on how to download an object to the local memory:

```

#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize the account information. */
    /* Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to
    o log on to OSS because the account has permissions on all API operations. We recommend tha
    t you use your RAM user's credentials to call API operations or perform routine operations
    and maintenance. To create a RAM user, log on to the RAM console. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    /* Specify the endpoint of the region in which the bucket is located. For example, if y
    our bucket is in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.a
    liyuncs.com. */
    std::string Endpoint = "yourEndpoint";
    /* Specify the bucket name. */
    std::string BucketName = "yourBucketName";
    /* Specify the complete name of the object excluding the bucket name. Example: destfold
    er/exampleobject.txt. */
    std::string ObjectName = "yourObjectName";
    /* Initialize network resources. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    /* Download the object to local memory. */
    GetObjectRequest request(BucketName, ObjectName);
    auto outcome = client.GetObject(request);
    if (outcome.isSuccess()) {
        std::cout << "getObjectToBuffer" << " success, Content-Length:" << outcome.result().M
        etadata().ContentLength() << std::endl;
        /* Use the read operation to read data. */
        auto& stream = outcome.result().Content();
        char buffer[256];
        while (stream->good()) {
            stream->read(buffer, 256);
            auto count = stream->gcount();
            /* Process the data. */
        }
    }
    else {
        /* Handle exceptions. */
        std::cout << "getObjectToBuffer fail" <<
        ",code:" << outcome.error().Code() <<
        ",message:" << outcome.error().Message() <<
        ",requestId:" << outcome.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }
    /* Release network resources. */
    ShutdownSdk();
    return 0;
}

```

11.7.4. Range download

If you need only part of the data in an object, you can use range download to download data within a specified range.

Specify a valid range

The following code provides an example on how to specify a valid range to download data:

```
#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize the OSS account information. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    std::string Endpoint = "yourEndpoint";
    std::string BucketName = "yourBucketName";
    std::string ObjectName = "yourObjectName ";
    /* Initialize network resources. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf );
    /* Obtain the object. */
    GetObjectRequest request(BucketName, ObjectName);
    /* Set the download range. */
    request.setRange(0, 1);
    auto outcome = client.GetObject(request);
    if (!outcome.isSuccess ()) {
        /* Handle exceptions. */
        std::cout << "getObject fail" <<
            ",code:" << outcome.error().Code() <<
            ",message:" << outcome.error().Message() <<
            ",requestId:" << outcome.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }
    /* Release network resources. */
    ShutdownSdk();
    return 0;
}
```

An invalid range is specified

For an object whose size is 1,000 bytes, the valid range is from byte 0 to byte 999. If the specified range is not within the range from byte 0 to byte 999, the range does not take effect. Then, OSS returns HTTP status code 200 and the data of the entire object. The following list provides examples of invalid requests and the returned results:

- If you set Range: bytes to 500-2000, the value at the end of the range is invalid. Then, OSS returns HTTP status code 200 and the data of the entire object.
- If you set Range: bytes to 1000-2000, the value at the start of the range is invalid. Then, OSS returns HTTP status code 200 and the data of the entire object.

Compatible download

If you add `x-oss-range-behavior:standard` to the request header, the download behavior is modified when the specified range is not within the valid range. Example: Use range download to download an object whose size is 1,000 bytes.

- If you set `Range: bytes to 500-2000`, the value at the end of the range is invalid. Then, OSS returns HTTP status code 206 and the data that is within the range from byte 500 to byte 999.
- If you set `Range: bytes to 1000-2000`, the value at the start of the range is invalid. Then, OSS returns HTTP status code 416 and error code `InvalidRange`.

The following code provides an example on how to specify compatible behaviors to download data by range:

```
#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize the OSS account information. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    std::string Endpoint = "yourEndpoint";
    std::string BucketName = "yourBucketName";
    /* Specify an object whose size is 1,000 bytes. */
    std::string ObjectName = "yourObjectName ";
    /* Initialize network resources. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    /* Obtain the object. */
    GetObjectRequest request(BucketName, ObjectName);
    /* Set the download range. */
    /* Specify the compatible behavior. */
    /* When the value at the end of the range is invalid, OSS returns the data that is with
    in the range from byte 500 to byte 999. The HTTP status code 206 is also returned. */
    request.setRange(500, 2000, true);
    auto outcome = client.GetObject(request);
    /* When the value at the start of the range is invalid, exceptions are thrown. OSS returns
    the HTTP status code 416 and the InvalidRange error code. */
    request.setRange(1000, 2000, true);
    outcome = client.GetObject(request);
    if (!outcome.IsSuccess()) {
        /* Handle exceptions. */
        std::cout << "getObject fail" <<
            ",code:" << outcome.error().Code() <<
            ",message:" << outcome.error().Message() <<
            ",requestId:" << outcome.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }
    /* Release network resources. */
    ShutdownSdk();
    return 0;
}
```

11.7.5. Resumable download

You may fail to download a large object if the network is unstable or other exceptions occur. In some cases, you may still fail to download the object even after multiple attempts. To resolve this issue, Object Storage Service (OSS) SDK for C++ provides the resumable download feature. In resumable download, the object that you want to download is split into multiple parts and each part is separately downloaded. After all parts are downloaded, these parts are combined into a complete object.

Implementation method

You can use the `OssClient .ResumableDownloadObject` method to implement resumable download. The following table describes the parameters contained in `DownloadObjectRequest`.

Parameter	Description	Required	Default value	Configuration method
bucket	The name of the bucket.	Yes	None	Constructor
key	The full path of the OSS object to download.	Yes	None	Constructor
filePath	The full path of the local file to which the object is downloaded.	No	The name of the downloaded object	Constructor
partSize	The size of each part. Valid values: 100 KB to 5 GB.	No	8 MB	setPartSize
threadNum	The number of parts that can be concurrently downloaded.	No	3	Constructor or setThreadNum

Parameter	Description	Required	Default value	Configuration method
checkpointDir	The file that records the download results of parts. To enable resumable upload or download, you must specify this parameter. This file stores information about download progress. If a part fails to be downloaded, the next download operation continues from the recorded interruption position. After the object is downloaded, the checkpoint file is deleted.	No	The same directory as that of DownloadFile	Constructor or setCheckpointDir

Sample code

The following code provides an example on how to perform resumable download:

```

#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize the OSS account information. */
    /* Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to
    o access OSS because the account has permissions on all API operations. We recommend that y
    ou use a RAM user to call API operations or perform routine O&M. To create a RAM user, log
    on to the RAM console. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    /* Specify the endpoint of the region in which the bucket is located. For example, if t
    he bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-han
    gzhou.aliyuncs.com. */
    std::string Endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
    /* Specify the name of the bucket. Example: examplebucket. */
    std::string BucketName = "examplebucket";
    /* Specify the full path of the object. The full path of the object cannot contain the
    bucket name. Example: exampledir/exampleobject.txt. */
    std::string ObjectName = "exampledir/exampleobject.txt";
    /* Download the object to D:\\localpath. After the object is downloaded, the local file
    is named examplefile.txt. If the specified local file exists, the object to download overwr
    ites the file. Otherwise, a file is created. */
    /* If the path for the object is not specified, the downloaded object is saved to the p
    ath of the project to which the sample program belongs. */
    std::string DownloadFilePath = "D:\\localpath\\examplefile.txt";
    /* Specify the full path of the local file. Example: D:\\localpath\\examplefile.txt.dcp
    . */
    /* After the checkpoint file is generated because the object download is interrupted, y
    ou must set the full path of the checkpoint file only when you want to continue the downloa
    d of the object. After the object is downloaded, the checkpoint file is deleted. */
    std::string CheckpointFilePath = "D:\\localpath\\examplefile.txt.dcp";
    /* Initialize resources such as networks. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    /* Start the resumable download. */
    DownloadObjectRequest request(BucketName, ObjectName, DownloadFilePath, CheckpointFileP
    ath);
    auto outcome = client.ResumableDownloadObject(request);
    if (!outcome.IsSuccess()) {
        /* Handle exceptions. */
        std::cout << "ResumableDownloadObject fail" <<
        ",code:" << outcome.error().Code() <<
        ",message:" << outcome.error().Message() <<
        ",requestId:" << outcome.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }
    /* Release resources such as networks. */
    ShutdownSdk();
    return 0;
}

```

References

- For more information about the complete sample code that is used to perform resumable download, visit [Git Hub](#).
- For more information about the API operation that you can call to perform resumable download, see [Get Object](#).

11.7.6. Use a progress bar

Progress bars are used to indicate the progress of an upload or download.

For the complete code of download progress bars, see [Git Hub](#).

Use the following code to view the progress information:

```
#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
void ProgressCallback(size_t increment, int64_t transfered, int64_t total, void* userData)
{
    std::cout << "ProgressCallback[" << userData << "] => " <<
        increment << " , " << transfered << " , " << total << std::endl;
}
int main(void )
{
    /* Initialize the OSS account information. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    std::string Endpoint = "yourEndpoint";
    std::string BucketName = "yourBucketName";
    std::string ObjectName = "yourObjectName" ;
    /* Initialize network resources. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    /* Obtain the object and its download progress information. */
    GetObjectRequest request(BucketName, ObjectName);
    TransferProgress progressCallback = { ProgressCallback , nullptr };
    request.setTransferProgress(progressCallback);
    auto outcome = client.GetObject(request);
    if (!outcome.isSuccess()) {
        /* Handle exceptions. */
        std::cout << "getObject fail" <<
            ",code:" << outcome.error(). Code() <<
            ",message:" << outcome.error(). Message() <<
            ",requestId:" << outcome.error(). RequestId() << std::endl;
        ShutdownSdk ();
        return -1;
    }
    /* Release network resources. */
    ShutdownSdk();
    return 0;
}
```

11.8. Manage objects

11.8.1. Overview

You can use APIs to perform the following operations on objects in a bucket:

- [Manage the ACL of an object](#)
- [Manage Object Meta](#)
- [List objects](#)
- [Copy an object](#)
- [Delete objects](#)
- [Determine whether the specified object exists](#)
- [Restore an archive object](#)
- [Manage a symbolic link](#)

11.8.2. Determine whether the specified object exists

This topic describes how to determine whether an object exists.

Use the following code to determine whether an object exists:

```
#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize the OSS account information. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    std::string Endpoint = "yourEndpoint";
    std::string BucketName = "yourBucketName";
    std::string ObjectName = "yourObjectName";
    /* Initialize network resources. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    /* Determine whether the object exists. */
    auto outcome = client.DoesObjectExist(BucketName, ObjectName);
    if (outcome) {
        std::cout << "The Object exists!" << std::endl;
    }
    else {
        std::cout << "The Object does not exist!" << std::endl;
    }
    /* Release network resources. */
    ShutdownSdk();
    return 0;
}
```

11.8.3. Manage object ACLs

This topic describes how to manage the access control lists (ACLs) of objects.

The following table describes the ACLs that you can configure for an object.

Permission	Description	Value
Inherited from the bucket	The ACL of an object is the same as the ACL of the bucket in which the object is stored.	<code>CannedAccessControlList::Default</code>
Private	Only the object owner and authorized users have read and write permissions on the object. Other users cannot access the object.	<code>CannedAccessControlList::Private</code>
Public read	Only the object owner and authorized users have read and write permissions on the object. Other users have only read permissions on the object. Exercise caution when you use this permission.	<code>CannedAccessControlList::PublicRead</code>
Public read/write	All users have read and write permissions on the object. Exercise caution when you use this permission.	<code>CannedAccessControlList::PublicReadWrite</code>

The ACL of an object takes precedence over the ACL of the bucket in which the object is stored. For example, if the ACL of an object in a private bucket is public read/write, all users have read and write permissions on the object. If you do not configure the ACL of an object, the ACL of the object is the same as the ACL of the bucket in which the object is stored.

Configure the ACL of an object

The following code provides an example on how to configure the ACL of a specified object:

```
#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize the information about the account that is used to access Object Storage S
ervice (OSS). */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    std::string Endpoint = "yourEndpoint";
    std::string BucketName = "yourBucketName";
    std::string ObjectName = "yourObjectName";
    /* Initialize resources such as networks. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    /* Configure the ACL of the object. */
    SetObjectAclRequest request(BucketName, ObjectName);
    request.setAcl(CannedAccessControlList::Private);
    auto outcome = client.SetObjectAcl(request);
    if (!outcome.isSuccess()) {
        /* Handle exceptions. */
        std::cout << "SetObjectAcl fail" <<
            ",code:" << outcome.error().Code() <<
            ",message:" << outcome.error().Message() <<
            ",requestId:" << outcome.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }
    /* Release resources such as networks. */
    ShutdownSdk();
    return 0;
}
```

Obtain the ACL of an object.

The following code provides an example on how to query the ACL of a specified object:

```
#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize the information about the account that is used to access OSS. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    std::string Endpoint = "yourEndpoint";
    std::string BucketName = "yourBucketName";
    std::string ObjectName = "yourObjectName";
    /* Initialize resources such as networks. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    /*Obtain the ACL of the object. */
    GetObjectAclRequest request(BucketName, ObjectName);
    auto outcome = client.GetObjectAcl(request);
    if (!outcome.isSuccess()) {
        /* Handle exceptions. */
        std::cout << "GetObjectAcl fail" <<
            ",code:" << outcome.error().Code() <<
            ",message:" << outcome.error().Message() <<
            ",requestId:" << outcome.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }
    else {
        std::cout << " GetObjectAcl success, Acl:" << outcome.result().Acl() << std::endl;
    }
    /* Release resources such as networks. */
    ShutdownSdk();
    return 0;
}
```

11.8.4. Manage object metadata

Object metadata includes HTTP headers and user metadata. This topic describes how to configure, modify, and query object metadata.

Configure object metadata

The following code provides an example on how to configure object metadata:

```

#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize the information about the account that is used to access Object Storage Service (OSS). */
    /* Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    /* Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com. */
    std::string Endpoint = "yourEndpoint";
    /* Specify the bucket name. Example: examplebucket. */
    std::string BucketName = "examplebucket";
    /* Specify the full path of the object. The full path cannot contain the bucket name. Example: exampledir/exampleobject.txt. */
    std::string ObjectName = "exampledir/exampleobject.txt";
    /* Initialize resources such as networks. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    /* Specify the HTTP headers. */
    auto meta = ObjectMetadata();
    meta.setContentType("text/plain");
    meta.setCacheControl("max-age=3");
    /* Specify user metadata. */
    meta.UserMetadata()["meta"] = "meta-value";
    std::shared_ptr<std::iostream> content = std::make_shared<std::stringstream>();
    *content << "Thank you for using Aliyun Object Storage Service!";
    client.PutObject(BucketName, ObjectName, content, meta);
    if (!outcome.isSuccess()) {
        /* Handle exceptions. */
        std::cout << "PutObject fail" <<
            ",code:" << outcome.error().Code() <<
            ",message:" << outcome.error().Message() <<
            ",requestId:" << outcome.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }
    /* Release resources such as networks. */
    ShutdownSdk();
    return 0;
}

```

Query object metadata

The following table describes the methods that you can use to query object metadata.

Method	Description	Advantage
GetObjectMeta	Queries parts of object metadata such as the ETag, Size, and LastModified (the last modified time) values of the object.	More lightweight and faster
HeadObject	Queries all object metadata.	None

The following code provides an example on how to query object metadata:

```
#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize the information about the account that is used to access OSS. */
    /* Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to
    o access OSS because the account has permissions on all API operations. We recommend that y
    ou use a RAM user to call API operations or perform routine O&M. To create a RAM user, log
    on to the RAM console. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    /* Set Endpoint to the endpoint of the region in which the bucket is located. For exampl
    e, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://os
    s-cn-hangzhou.aliyuncs.com. */
    std::string Endpoint = "yourEndpoint";
    /* Specify the bucket name. Example: examplebucket. */
    std::string BucketName = "examplebucket";
    /* Specify the full path of the object. The full path cannot contain the bucket name. E
    xample: exampledir/exampleobject.txt. */
    std::string ObjectName = "exampledir/exampleobject.txt";
    /* Initialize resources such as networks. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    /* Query parts of the object metadata. */
    auto outcome = client.GetObjectMeta(BucketName, ObjectName);
    if (!outcome.IsSuccess()) {
        /* Handle exceptions. */
        std::cout << "GetObjectMeta fail" <<
        ",code:" << outcome.error().Code() <<
        ",message:" << outcome.error().Message() <<
        ",requestId:" << outcome.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }
    else {
        auto metadata = outcome.result();
        std::cout << " get metadata success, ETag:" << metadata.ETag() << "; LastModified:"

        << metadata.LastModified() << "; Size:" << metadata.ContentLength() << std::endl;
    }
    /* Query all object metadata. */
    outcome = client.HeadObject(BucketName, ObjectName);
```

```
if (!outcome.isSuccess()) {
    /* Handle exceptions. */
    std::cout << "HeadObject fail" <<
        ",code:" << outcome.error().Code() <<
        ",message:" << outcome.error().Message() <<
        ",requestId:" << outcome.error().RequestId() << std::endl;
    ShutdownSdk();
    return -1;
}
else {
    auto headMeta = outcome.result();
    std::cout <<"headMeta success, ContentType:"
        << headMeta.ContentType() << "; ContentLength:" << headMeta.ContentLength()
        << "; CacheControl:" << headMeta.CacheControl() << std::endl;
}
/* Release resources such as networks. */
ShutdownSdk();
return 0;
}
```

References

- For the complete sample code that is used to configure and query object metadata, visit [GitHub](#).
- For more information about the API operation that you can call to configure object metadata in simple upload, see [Put Object](#).
- For more information about the API operation that you can call to query object metadata, see [Get Object Meta](#).

11.8.5. Convert the storage class of an object

OSS provides the following storage classes to cover various data storage scenarios from hot data to cold data: Standard, Infrequent Access (IA), Archive and Cold Archive. This topic describes how to convert the storage class of an object.

For more information about these storage classes, see [Overview](#) and [Convert storage classes](#) in the OSS Developer Guide.

The following code provides an example of how to convert the storage class of an object.

- The following code provides an example of how to convert the storage class of an object from Standard or IA to Archive:

```
#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize the OSS account information. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    std::string Endpoint = "yourEndpoint";
    std::string SourceBucketName = "yourSourceBucketName";
    std::string SourceObjectName = "yourSourceObjectName";
    /* Initialize network resources. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    /* Set the storage class to Archive. */
    ObjectMetaData objectMeta;
    objectMeta.addHeader("x-oss-storage-class", "Archive");
    CopyObjectRequest request(SourceBucketName, SourceBucketName, objectMeta);
    request.setCopySource(SourceBucketName, SourceObjectName);
    /* Modify the storage class. */
    auto outcome = client.CopyObject(request);
    if (!outcome.isSuccess()) {
        /* Handle exceptions. */
        std::cout << "CopyObject fail" <<
            ",code:" << outcome.error().Code() <<
            ",message:" << outcome.error().Message() <<
            ",requestId:" << outcome.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }
    /* Release network resources. */
    ShutdownSdk();
    return 0;
}
```

- The following code provides an example of how to convert the storage class of an object from Archive to IA:

```
#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize the OSS account information. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    std::string Endpoint = "yourEndpoint";
    std::string SourceBucketName = "yourSourceBucketName";
    std::string SourceObjectName = "yourSourceObjectName";
    /* Initialize network resources. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    /* Check whether the storage class of the source object is Archive. If the storage class of the source object is Archive, you must restore the object before you can modify th
```

```

e storage class. */
auto restoreOutcome = client.RestoreObject(SourceBucketName, SourceObjectName);
if (! restoreOutcome.IsSuccess()) {
    /* Handle exceptions. */
    std::cout << "RestoreObject fail" <<
        ",code:" << restoreOutcome.error().Code() <<
        ",message:" << restoreOutcome.error().Message() <<
        ",requestId:" << restoreOutcome.error().RequestId() << std::endl;
    ShutdownSdk();
    return -1;
}
std::string onGoingRestore("ongoing-request=\"false\"");
int maxWaitTimeInSeconds = 600;
while (maxWaitTimeInSeconds > 0)
{
    auto meta = client.HeadObject(SourceBucketName, SourceObjectName);
    std::string restoreStatus = meta.result().HttpMetaData()["x-oss-restore"];
    std::transform(restoreStatus.begin(), restoreStatus.end(), restoreStatus.begin(),
::tolower);
    if (! restoreStatus.empty() &&
        restoreStatus.compare(0, onGoingRestore.size(), onGoingRestore)==0) {
        std::cout << " success, restore status:" << restoreStatus << std::endl;
        /* The Archive object is restored.*/
        break;
    }
    std::cout << " info, WaitTime:" << maxWaitTimeInSeconds
        << "; restore status:" << restoreStatus << std::endl;
    std::this_thread::sleep_for(std::chrono::seconds(10));
    maxWaitTimeInSeconds--;
}
/* Set the storage class to IA. */
ObjectMetaData objectMeta;
objectMeta.addHeader("x-oss-storage-class", "IA");
CopyObjectRequest request(SourceBucketName, SourceBucketName,objectMeta);
request.setCopySource(SourceBucketName, SourceObjectName);
/* Modify the storage class. */
auto outcome = client.CopyObject(request);
if (! outcome.IsSuccess()) {
    /* Handle exceptions. */
    std::cout << "CopyObject fail" <<
        ",code:" << outcome.error().Code() <<
        ",message:" << outcome.error().Message() <<
        ",requestId:" << outcome.error().RequestId() << std::endl;
    ShutdownSdk();
    return -1;
}
/* Release network resources. */
ShutdownSdk();
return 0;
}

```

For more information about the storage classes and storage fees, see the [Billing overview](#) section in [Billing items](#).

11.8.6. List objects

This topic describes how to list all objects in a bucket, a specified number of objects, and the sizes of objects in a specified directory.

Objects are listed alphabetically. You can use `OssClient.ListObjects` to list the objects in a bucket. `ListObjects` supports the following types of parameter formats:

- List Object Outcome `ListObjects(const std::string& bucket) const`: lists objects in a bucket. A maximum of 1,000 objects can be listed.
- List Object Outcome `ListObjects(const std::string& bucket, const std::string& prefix) const`: lists objects whose names contain the specified prefix in a bucket. A maximum of 1,000 objects can be listed.
- List Object Outcome `ListObjects(const ListObjectsRequest& request) const`: provides multiple filter conditions to query objects.

The following table describes the parameters for object listing.

Operation	Description
delimiter	The character that is used to group objects by name. Objects whose names contain the same string from the prefix and the next occurrence of the delimiter are grouped as a single result element in <code>commonPrefixes</code> .
prefix	The prefix that the names of returned objects must contain.
maxKeys	The maximum number of objects that can be listed at a time. The default value is 100, and the maximum value is 1000.
marker	The position from which the list operation starts. All objects are returned in lexicographic order starting from this position.

Simple list

The following code provides an example on how to list objects in a specified bucket. By default, a maximum of 100 objects can be listed.

```

#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize the information about the account that is used to access OSS. */
    /* Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to
    o access OSS because the account has permissions on all API operations. We recommend that y
    ou use a RAM user to call API operations or perform routine O&M. To create a RAM user, log
    on to the RAM console. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    /* Specify the endpoint of the region in which the bucket is located. For example, if t
    he bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-han
    gzhou.aliyuncs.com. */
    std::string Endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
    /* Specify the name of the bucket. Example: examplebucket. */
    std::string BucketName = "examplebucket";
    /* Initialize resources such as networks. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    /* List objects. */
    ListObjectsRequest request(BucketName);
    auto outcome = client.ListObjects(request);
    if (!outcome.isSuccess()) {
        /* Handle exceptions. */
        std::cout << "ListObjects fail" <<
        ",code:" << outcome.error().Code() <<
        ",message:" << outcome.error().Message() <<
        ",requestId:" << outcome.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }
    else {
        for (const auto& object : outcome.result().ObjectSummaries()) {
            std::cout << "object"<<
            ",name:" << object.Key() <<
            ",size:" << object.Size() <<
            ",lastmodify time:" << object.LastModified() << std::endl;
        }
    }
    /* Release resources such as networks. */
    ShutdownSdk();
    return 0;
}

```

List a specified number of objects

The following code provides an example on how to list a specified number of objects:

```

#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize the information about the account that is used to access OSS. */
    /* Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account t
o access OSS because the account has permissions on all API operations. We recommend that y
ou use a RAM user to call API operations or perform routine O&M. To create a RAM user, log
on to the RAM console. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    /* Specify the endpoint of the region in which the bucket is located. For example, if t
he bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-han
gzhou.aliyuncs.com. */
    std::string Endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
    /* Specify the name of the bucket. Example: examplebucket. */
    std::string BucketName = "examplebucket";
    /* Initialize resources such as networks. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf) ;
    /* List objects. */
    ListObjectsRequest request(BucketName);
    /* Set the maximum number of returned objects to 200. */
    request.setMaxKeys(200);
    auto outcome = client.ListObjects(request);
    if (!outcome.isSuccess() ) {
        /* Handle exceptions. */
        std::cout << "ListObjects fail" <<
            ",code:" << outcome.error().Code() <<
            ",message:" << outcome.error().Message() <<
            ",requestId:" << outcome.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }
    else {
        for (const auto& object : outcome.result().ObjectSummaries()) {
            std::cout << "object"<<
                ",name:" << object.Key() <<
                ",size:" << object.Size() <<
                ",lastmodify time:" << object.LastModified() << std::endl;
        }
    }
    /* Release resources such as networks. */
    ShutdownSdk();
    return 0;
}

```

List objects and subdirectories in a specified directory

The following code provides an example on how to list objects and subdirectories within a directory in a bucket:

```

#include <alibabacloud/oss/OssClient.h>

```

```

#include <alibabacloud/oss/ossclient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize the information about the account that is used to access OSS. */
    /* Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account t
    o access OSS because the account has permissions on all API operations. We recommend that y
    ou use a RAM user to call API operations or perform routine O&M. To create a RAM user, log
    on to the RAM console. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    /* Specify the endpoint of the region in which the bucket is located. For example, if t
    he bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-han
    gzhou.aliyuncs.com. */
    std::string Endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
    /* Specify the name of the bucket. Example: examplebucket. */
    std::string BucketName = "examplebucket";
    /* Specify the prefix that the objects to return must contain in their names.
    std::string keyPrefix = "yourkeyPrefix ";
    /* Initialize resources such as networks. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    std::string nextMarker = "";
    bool isTruncated = false;
    do {
        /* List objects. */
        ListObjectsRequest request(BucketName);
        /* Set the delimiter to a forward slash (/). */
        request.setDelimiter("/");
        request.setPrefix(keyPrefix);
        request.setMarker(nextMarker);
        auto outcome = client.ListObjects(request);
        if (!outcome.isSuccess ()) {
            /* Handle exceptions. */
            std::cout << "ListObjects fail" <<
            ",code:" << outcome.error().Code() <<
            ",message:" << outcome.error().Message() <<
            ",requestId:" << outcome.error().RequestId() << std::endl;
            break;
        }
        for (const auto& object : outcome.result().ObjectSummaries()) {
            std::cout << "object"<<
            ",name:" << object.Key() <<
            ",size:" << object.Size() <<
            ",lastmodify time:" << object.LastModified() << std::endl;
        }
        for (const auto& commonPrefix : outcome.result().CommonPrefixes()) {
            std::cout << "commonPrefix" << ",name:" << commonPrefix << std::endl;
        }
        nextMarker = outcome.result().NextMarker();
        isTruncated = outcome.result().IsTruncated();
    } while (isTruncated);
    /* Release resources such as networks. */
    ShutdownSdk();
}

```



```

        return 0;
    }
}

```

List sizes of objects in a specified directory

The following code provides an example on how to query the sizes of objects in a specified directory:

```

#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    static int64_t calculateFolderLength (const OssClient &client, const std::string &bucketName, const std::string &folder)
    {
        std::string nextMarker = "";
        bool isTruncated = false;
        int64_t size = 0;
        do {
            /* List objects. */
            ListObjectsRequest request(bucketName);
            request.setPrefix(folder);
            request.setMarker(nextMarker);
            auto outcome = client.ListObjects(request);
            if (!outcome.isSuccess()) {
                /* Handle exceptions. */
                std::cout << "ListObjects fail" <<
                    ",code:" << outcome.error().Code() <<
                    ",message:" << outcome.error().Message() <<
                    ",requestId:" << outcome.error().RequestId() << std::endl;
                break;
            }
            for (const auto& object : outcome.result().ObjectSummaries()) {
                size += object.Key()
            }
            nextMarker = outcome.result().NextMarker();
            isTruncated = outcome.result().IsTruncated();
        } while (isTruncated);
        return size;
    }

    /* Initialize the information about the account that is used to access OSS. */
    /* Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    /* Specify the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com. */
    std::string Endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
    /* Specify the name of the bucket. Example: examplebucket. */
    std::string BucketName = "examplebucket";
    /* Specify the prefix that the objects to return must contain in their names. */
    std::string KeyPrefix = "yourKeyPrefix";
}

```

```

std::string keyPrefix = "yourKeyPrefix" ,
/* Initialize resources such as networks. */
InitializeSdk();
ClientConfiguration conf;
OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
std::string nextMarker = "";
bool isTruncated = false;
do {
    /* List objects. */
    ListObjectsRequest request(BucketName);
    /* Set the delimiter to forward slashes (/). */
    request.setDelimiter("/");
    request.setPrefix(keyPrefix);
    request.setMarker(nextMarker);
    auto outcome = client.ListObjects(request);
    if (!outcome.IsSuccess()) {
        /* Handle exceptions. */
        std::cout << "ListObjects fail" <<
            ",code:" << outcome.error().Code() <<
            ",message:" << outcome.error().Message() <<
            ",requestId:" << outcome.error().RequestId() << std::endl;
        break;
    }
    for (const auto& object : outcome.result().ObjectSummaries()) {
        std::cout << "object" <<
            ",name:" << object.Key() <<
            ",size:" << object.Size() << std::endl;
    }
    for (const auto& commonPrefix : outcome.result().CommonPrefixes()) {
        int64_t foldersize = calculateFolderLength(client, BucketName, commonPrefix);
        std::cout << "folder" <<
            ",name:" << commonPrefix <<
            ",size:" << foldersize << std::endl;
    }
    nextMarker = outcome.result().NextMarker();
    isTruncated = outcome.result().IsTruncated();
} while (isTruncated);
/* Release resources such as networks. */
ShutdownSdk();
return 0;
}

```

List objects whose names contain a specified prefix

The following code provides an example on how to list objects whose names contain a specified prefix:

```

#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize the information about the account that is used to access OSS. */
    /* Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account
    to access OSS because the account has permissions on all API operations. We recommend that
    you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log
    on to the RAM console. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    /* Specify the endpoint of the region in which the bucket is located. For example, if
    the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-ha
    ngzhou.aliyuncs.com. */
    std::string Endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
    /* Specify the name of the bucket. Example: examplebucket. */
    std::string BucketName = "examplebucket";
    /* Specify the prefix that the objects to return must contain in their names.
    std::string keyPrefix = "yourkeyPrefix ";
    /* Initialize resources such as networks. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    std::string nextMarker = "";
    bool isTruncated = false;
    do {
        /* List objects. */
        ListObjectsRequest request(BucketName);
        request.setPrefix(keyPrefix);
        request.setMarker(nextMarker);
        auto outcome = client.ListObjects(request);
        if (!outcome.isSuccess()) {
            /* Handle exceptions. */
            std::cout << "ListObjects fail" <<
            ",code:" << outcome.error().Code() <<
            ",message:" << outcome.error().Message() <<
            ",requestId:" << outcome.error().RequestId() << std::endl;
            break;
        }
        for (const auto& object : outcome.result().ObjectSummaries()) {
            std::cout << "object"<<
            ",name:" << object.Key() <<
            ",size:" << object.Size() <<
            ",lastmodify time:" << object.LastModified() << std::endl;
        }
        nextMarker = outcome.result().NextMarker();
        isTruncated = outcome.result().IsTruncated();
    } while (isTruncated);
    /* Release resources such as networks. */
    ShutdownSdk();
    return 0;
}

```

List objects whose names are alphabetically after the object specified by marker

You can configure the marker parameter to specify the name of the object after which the list operation starts. The following code provides an example on how to list objects whose names are alphabetically after the object specified by marker:

```

#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize the information about the account that is used to access OSS. */
    /* Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account
    to access OSS because the account has permissions on all API operations. We recommend that
    you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log
    on to the RAM console. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    /* Specify the endpoint of the region in which the bucket is located. For example, if
    the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-ha
    ngzhou.aliyuncs.com. */
    std::string Endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
    /* Specify the name of the bucket. Example: examplebucket. */
    std::string BucketName = "examplebucket";
    /* Specify marker as the name of the object after which the list begins.
    std::string YourMarker = "yourMarker";
    /* Initialize resources such as networks. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    ListObjectOutcome outcome;
    do {
        /* List the objects. */
        ListObjectsRequest request(BucketName);
        /* Specify the name of the object after which the list begins. */
        request.setMarker(YourMarker);
        outcome = client.ListObjects(request);
        if (!outcome.IsSuccess()) {
            /* Handle exceptions. */
            std::cout << "ListObjects fail" <<
            ",code:" << outcome.error().Code() <<
            ",message:" << outcome.error().Message() <<
            ",requestId:" << outcome.error().RequestId() << std::endl;
            break;
        }
        YourMarker = outcome.result().NextMarker();
        for (const auto& object : outcome.result().ObjectSummaries()) {
            std::cout << "object"<<
            ",name:" << object.Key() <<
            ",size:" << object.Size() <<
            ",lastmodify time:" << object.LastModified() << std::endl;
        }
    } while (outcome.result().IsTruncated());
    /* Release network resources. */
    ShutdownSdk();
    return 0;
}

```

List objects and specify the encoding type of the object names

An object name that contains one of the following special characters needs to be encoded before the object is transmitted. Currently, OSS supports only URL encoding.

- Single quotation marks (')
- Double quotation marks (")
- Ampersands (&)
- Angle brackets (<>)
- Pause markers
- Chinese characters

The following code provides an example on how to encode a specified object name:

```

#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize the information about the account that is used to access OSS. */
    /* Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account t
o access OSS because the account has permissions on all API operations. We recommend that y
ou use a RAM user to call API operations or perform routine O&M. To create a RAM user, log
on to the RAM console. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    /* Specify the endpoint of the region in which the bucket is located. For example, if t
he bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-han
gzhou.aliyuncs.com. */
    std::string Endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
    /* Specify the name of the bucket. Example: examplebucket. */
    std::string BucketName = "examplebucket";
    /* Initialize network resources. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    std::string nextMarker = "";
    bool isTruncated = false;
    do {
        /* List objects. */
        ListObjectsRequest request(BucketName) ;
        /* Specify the encoding type of the object names. */
        request.setEncodingType("url");
        request.setMarker(nextMarker);
        auto outcome = client.ListObjects(request);
        if (!outcome.isSuccess() ) {
            /* Handle exceptions. */
            std::cout << "ListObjects fail" <<
                ",code:" << outcome.error().Code() <<
                ",message:" << outcome.error().Message() <<
                ",requestId:" << outcome.error().RequestId() << std::endl;
            break;
        }
        for (const auto& object : outcome.result().ObjectSummaries()) {
            std::cout << "object"<<
                ",name:" << object.Key() <<
                ",size:" << object.Size() <<
                ",lastmodify time:" << object.LastModified() << std::endl;
        }
        nextMarker = outcome.result().NextMarker();
        isTruncated = outcome.result().IsTruncated();
    } while (isTruncated);
    /* Release resources such as networks. */
    ShutdownSdk();
    return 0;
}

```

List objects in a directory

OSS uses a flat structure for objects instead of a hierarchical structure to store objects. A directory is an object whose size is 0 and whose name ends with a forward slash (/). You can upload and download this object. By default, objects whose names end with a forward slash (/) are displayed as directories in the OSS console. For the complete sample code that is used to create directories, visit [GitHub](#).

You can specify the delimiter and prefix parameters to list objects by directory.

- If you set prefix to a directory name in the request, objects and subdirectories whose names contain the prefix are listed.
- If you also set delimiter to a forward slash (/) in the request, the objects and subdirectories whose names start with the specified prefix in the directory are listed. Each subdirectory is listed as a single result element in commonPrefixes. The objects and directories in these subdirectories are not listed.

For more information about directories, see [List objects](#). For more information about the complete code used to create a directory, visit [GitHub](#).

Example: The following four objects are stored in a bucket: `oss.jpg`, `fun/test.jpg`, `fun/movie/001.avi`, and `fun/movie/007.avi`. The forward slash (/) is specified as the directory delimiter. The following code provides examples on how to list objects by directory.

- List all objects in a bucket

The following code provides an example on how to list all objects in a bucket:


```

#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize the information about the account that is used to access OSS. */
    /* Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account
    to access OSS because the account has permissions on all API operations. We recommend tha
    t you use a RAM user to call API operations or perform routine O&M. To create a RAM user,
    log on to the RAM console. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    /* Specify the endpoint of the region in which the bucket is located. For example, if
    the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-
    hangzhou.aliyuncs.com. */
    std::string Endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
    /* Specify the name of the bucket. Example: examplebucket. */
    std::string BucketName = "examplebucket";
    /* Initialize resources such as networks. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    std::string nextMarker = "";
    bool isTruncated = false;
    do {
        /* List objects. */
        ListObjectsRequest request(BucketName);
        request.setMarker(nextMarker);
        auto outcome = client.ListObjects(request);
        if (!outcome.isSuccess()) {
            /* Handle exceptions. */
            std::cout << "ListObjects fail" <<
            ",code:" << outcome.error().Code() <<
            ",message:" << outcome.error().Message() <<
            ",requestId:" << outcome.error().RequestId() << std::endl;
            ShutdownSdk();
            return -1;
        }
        else {
            for (const auto& object : outcome.result().ObjectSummaries()) {
                std::cout << "object"<<
                ",name:" << object.Key() <<
                ",size:" << object.Size() <<
                ",lastmodify time:" << object.LastModified() << std::endl;
            }
        }
        nextMarker = outcome.result().NextMarker();
        isTruncated = outcome.result().IsTruncated();
    } while (isTruncated);
    /* Release resources such as networks. */
    ShutdownSdk();
    return 0;
}

```

The following result is returned:

```
Objects:
fun/movie/001.avi
fun/movie/007.avi
fun/test.jpg
oss.jpg
CommonPrefixes:
```

- List all objects in the specified directory

The following code provides an example on how to list all objects in the specified directory:

```

#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize the information about the account that is used to access OSS. */
    /* Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account
    to access OSS because the account has permissions on all API operations. We recommend tha
    t you use a RAM user to call API operations or perform routine O&M. To create a RAM user,
    log on to the RAM console. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    /* Specify the endpoint of the region in which the bucket is located. For example, if
    the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-
    hangzhou.aliyuncs.com. */
    std::string Endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
    /* Specify the name of the bucket. Example: examplebucket. */
    std::string BucketName = "examplebucket";
    /* Initialize resources such as networks. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    std::string nextMarker = "";
    bool isTruncated = false;
    do {
        /* List objects. */
        ListObjectsRequest request(BucketName);
        request.setPrefix("fun/");
        request.setMarker(nextMarker);
        auto outcome = client.ListObjects(request);
        if (!outcome.isSuccess()) {
            /* Handle exceptions. */
            std::cout << "ListObjects fail" <<
            ",code:" << outcome.error().Code() <<
            ",message:" << outcome.error().Message() <<
            ",requestId:" << outcome.error().RequestId() << std::endl;
            break;
        }
        for (const auto& object : outcome.result().ObjectSummaries()) {
            std::cout << "object"<<
            ",name:" << object.Key() <<
            ",size:" << object.Size() <<
            ",lastmodify time:" << object.LastModified() << std::endl;
        }
        for (const auto& commonPrefix : outcome.result().CommonPrefixes()) {
            std::cout << "commonPrefix" << ",name:" << commonPrefix << std::endl;
        }
        nextMarker = outcome.result().NextMarker();
        isTruncated = outcome.result().IsTruncated();
    } while (isTruncated);
    /* Release resources such as networks. */
    ShutdownSdk();
    return 0;
}

```

The following result is returned:

```
Objects:
fun/movie/001.avi
fun/movie/007.avi
fun/test.jpg
CommonPrefixes:
```

- List objects and subdirectories in a specified directory

The following code provides an example on how to list objects and subdirectories in a specified directory:

```
#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize the information about the account that is used to access OSS. */
    /* Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account
    to access OSS because the account has permissions on all API operations. We recommend tha
    t you use a RAM user to call API operations or perform routine O&M. To create a RAM user,
    log on to the RAM console. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    /* Specify the endpoint of the region in which the bucket is located. For example, if
    the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-
    hangzhou.aliyuncs.com. */
    std::string Endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
    /* Specify the name of the bucket. Example: examplebucket. */
    std::string BucketName = "examplebucket";
    /* Initialize resources such as networks. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    std::string nextMarker = "";
    bool isTruncated = false;
    do {
        /* List objects. */
        ListObjectsRequest request(BucketName);
        /* Set the delimiter to a forward slash (/). */
        request.setDelimiter("/");
        request.setPrefix("fun/");
        request.setMarker(nextMarker);
        auto outcome = client.ListObjects(request);
        if (!outcome.isSuccess()) {
            /* Handle exceptions. */
            std::cout << "ListObjects fail" <<
                ",code:" << outcome.error().Code() <<
                ",message:" << outcome.error().Message() <<
                ",requestId:" << outcome.error().RequestId() << std::endl;
            break;
        }
        for (const auto& object : outcome.result().ObjectSummaries()) {
            std::cout << "object"<<
                ",name:" << object.Key() <<
```

```

        ",size:" << object.Size() <<
        ",lastmodify time:" << object.LastModified() << std::endl;
    }
    for (const auto& commonPrefix : outcome.result().CommonPrefixes()) {
        std::cout << "commonPrefix" << ",name:" << commonPrefix << std::endl;
    }
    nextMarker = outcome.result().NextMarker();
    isTruncated = outcome.result().IsTruncated();
} while (isTruncated);
/* Release resources such as networks. */
ShutdownSdk();
return 0;
}

```

The following result is returned:

```

Objects:
fun/test.jpg
CommonPrefixes:
fun/movie/

```

- List sizes of objects in a specified directory

The following code provides an example on how to query the sizes of objects in a specified directory:

```

#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
static int64_t calculateFolderLength (const OssClient &client, const std::string &bucketName, const std::string &folder)
{
    std::string nextMarker = "";
    bool isTruncated = false;
    int64_t size = 0;
    do {
        /* List objects. */
        ListObjectsRequest request(bucketName);
        request.setPrefix(folder);
        request.setMarker(nextMarker);
        auto outcome = client.ListObjects(request);
        if (!outcome.isSuccess()) {
            /* Handle exceptions. */
            std::cout << "ListObjects fail" <<
                ",code:" << outcome.error().Code() <<
                ",message:" << outcome.error().Message() <<
                ",requestId:" << outcome.error().RequestId() << std::endl;
            break;
        }
        for (const auto& object : outcome.result().ObjectSummaries()) {
            size += object.Size();
        }
        nextMarker = outcome.result().NextMarker();
        isTruncated = outcome.result().IsTruncated();
    } while (isTruncated);
    return size;
}

```

```

}
int main(void)
{
    /* Initialize the information about the account that is used to access OSS. */
    /* Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account
    to access OSS because the account has permissions on all API operations. We recommend tha
    t you use a RAM user to call API operations or perform routine O&M. To create a RAM user,
    log on to the RAM console. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    /* Specify the endpoint of the region in which the bucket is located. For example, if
    the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-
    hangzhou.aliyuncs.com. */
    std::string Endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
    /* Specify the name of the bucket. Example: examplebucket. */
    std::string BucketName = "examplebucket";
    /* Initialize network resources. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    std::string nextMarker = "";
    bool isTruncated = false;
    do {
        /* List objects. */
        ListObjectsRequest request(BucketName);
        /* Set the delimiter to forward slashes (/). */
        request.setDelimiter("/");
        request.setPrefix("fun/");
        request.setMarker(nextMarker);
        auto outcome = client.ListObjects(request);
        if (!outcome.isSuccess()) {
            /* Handle exceptions. */
            std::cout << "ListObjects fail" <<
            ",code:" << outcome.error().Code() <<
            ",message:" << outcome.error().Message() <<
            ",requestId:" << outcome.error().RequestId() << std::endl;
            break;
        }
        for (const auto& object : outcome.result().ObjectSummaries()) {
            std::cout << "object" <<
            ",name:" << object.Key() <<
            ",size:" << object.Size() << std::endl;
        }
        for (const auto& commonPrefix : outcome.result().CommonPrefixes()) {
            int64_t foldersize = calculateFolderLength(client, BucketName, commonPrefix);
            std::cout << "folder" <<
            ",name:" << commonPrefix <<
            ",size:" << foldersize << std::endl;
        }
        nextMarker = outcome.result().NextMarker();
        isTruncated = outcome.result().IsTruncated();
    } while (isTruncated);
    /* Release resources such as networks. */
    ShutdownSdk();
    return 0;
}

```

```
return 0;  
}
```

11.8.7. Delete objects

You can delete a single object, multiple specified objects, objects whose names contain a specified prefix, or a specified directory and all objects in the directory.



Warning Deleted objects cannot be recovered. Exercise caution when you delete objects.

Delete a single object

The following code provides an example on how to delete an object named *exampleobject.txt* from a bucket named *examplebucket*:

```

#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize the information about the account that is used to access Object Storage S
ervice (OSS). */
    /* The AccessKey pair of an Alibaba Cloud account has permissions on all API operations
. Using these credentials to perform operations in OSS is a high-risk operation. We recomme
nd that you use a RAM user to call API operations or perform routine O&M. To create a RAM u
ser, log on to the RAM console. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    /* Specify the endpoint of the region in which the bucket is located. For example, if t
he bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-han
gzhou.aliyuncs.com. */
    std::string Endpoint = "yourEndpoint";
    /* Specify the bucket name. Example: examplebucket. */
    std::string BucketName = "examplebucket";
    /* Specify the full path of the object. Example: exampleobject.txt. The full path of th
e object cannot contain the bucket name. */
    /* If you want to delete a directory, set ObjectName to the directory name. If the dire
ctory contains objects, you must delete all objects from the directory before you can delet
e the directory. */
    std::string ObjectName = "exampleobject.txt";
    /* Initialize resources such as networks. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    DeleteObjectRequest request(BucketName, ObjectName);
    /* Delete the object. */
    auto outcome = client.DeleteObject(request);
    if (!outcome.IsSuccess()) {
        /* Handle exceptions. */
        std::cout << "DeleteObject fail" <<
            ",code:" << outcome.error().Code() <<
            ",message:" << outcome.error().Message() <<
            ",requestId:" << outcome.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }
    /* Release resources such as networks. */
    ShutdownSdk();
    return 0;
}

```

Delete multiple objects

You can delete up to 1,000 objects at a time. You can delete multiple specified objects, objects whose names contain a specified prefix, or a specified directory and all objects in the directory.

When you delete multiple objects, the result can be returned in one of the following modes:

- **verbose:** A list of all deleted objects is returned. This is the default return mode.

- `quiet`: A list of objects that fail to be deleted is returned.

You can also configure lifecycle rules to automatically delete objects. For more information, see [Lifecycle rules based on the last modified time](#).

Delete multiple specified objects

The following code provides an example on how to delete multiple specified objects:

```
#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize the information about the account that is used to access OSS. */
    /* The AccessKey pair of an Alibaba Cloud account has permissions on all API operations
     . Using these credentials to perform operations in OSS is a high-risk operation. We recomme
     nd that you use a RAM user to call API operations or perform routine O&M. To create a RAM u
     ser, log on to the RAM console. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    /* Specify the endpoint of the region in which the bucket is located. For example, if t
     he bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-han
     gzhou.aliyuncs.com. */
    std::string Endpoint = "yourEndpoint";
    /* Specify the bucket name. Example: examplebucket. */
    std::string BucketName = "examplebucket";
    /* Specify the full path of the object. Example: exampleobject.txt. The full path of th
     e object cannot contain the bucket name. */
    std::string ObjectName = "exampleobject.txt";
    /* Initialize resources such as networks. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    DeleteObjectsRequest request(BucketName);
    /* Specify the full paths of the objects that you want to delete. */
    request.addKey(ObjectName);
    /* Delete the objects. */
    auto outcome = client.DeleteObjects(request);
    if (!outcome.isSuccess()) {
        /* Handle exceptions. */
        std::cout << "DeleteObjects fail" <<
            ",code:" << outcome.error().Code() <<
            ",message:" << outcome.error().Message() <<
            ",requestId:" << outcome.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }
    /* Release resources such as networks. */
    ShutdownSdk();
    return 0;
}
```

Delete multiple objects with a specified prefix or in a specified directory

The following code provides an example on how to delete multiple objects with a specified prefix and how to delete a specified directory and all objects in the directory.

 **Warning** If the value of `keyPrefix` is an empty string or is set to `NULL` in the following code, all objects in the bucket are deleted. Exercise caution when you configure this value.

```
#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize the information about the account that is used to access OSS. */
    /* The AccessKey pair of an Alibaba Cloud account has permissions on all API operations
     . Using these credentials to perform operations in OSS is a high-risk operation. We recommen
     nd that you use a RAM user to call API operations or perform routine O&M. To create a RAM u
     ser, log on to the RAM console. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    /* Specify the endpoint of the region in which the bucket is located. For example, if t
     he bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-han
     gzhou.aliyuncs.com. */
    std::string Endpoint = "yourEndpoint";
    /* Specify the bucket name. Example: examplebucket. */
    std::string BucketName = "examplebucket";
    /* Specify the prefix of the objects that you want to delete. If you want to delete all
     objects whose names contain the src prefix, set the prefix to src. After you set the prefix
     to src, all non-directory objects whose names contain the src prefix, the src directory, an
     d all objects in the directory are deleted. */
    std::string keyPrefix = "src";
    /* If you want to delete only the src directory and all objects in the directory, set t
     he prefix to src/. */
    /* std::string keyPrefix = "src/"; */
    /* Initialize resources such as networks. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    std::string nextMarker = "";
    bool isTruncated = false;
    do {
        /* List objects. */
        ListObjectsRequest request(BucketName);
        request.setPrefix(keyPrefix);
        request.setMarker(nextMarker);
        auto outcome = client.ListObjects(request);
        if (!outcome.isSuccess()) {
            /* Handle exceptions. */
            std::cout << "ListObjects fail" <<
                ",code:" << outcome.error().Code() <<
                ",message:" << outcome.error().Message() <<
                ",requestId:" << outcome.error().RequestId() << std::endl;
            break;
        }
        for (const auto& object : outcome.result().ObjectSummaries()) {
            DeleteObjectRequest request(BucketName, object.Key());
```

```
        /* Delete the objects. */
        auto delResult = client.DeleteObject(request);
    }
    nextMarker = outcome.result().NextMarker();
    isTruncated = outcome.result().IsTruncated();
} while (isTruncated);
/* Release resources such as networks. */
ShutdownSdk();
return 0;
}
```

References

- For the complete sample code that is used to delete a single object or multiple objects, visit [GitHub](#).
- For more information about the API operation that you can call to delete a single object, see [DeleteObject](#).
- For more information about the API operation that you can call to delete multiple objects, see [DeleteMultipleObjects](#).

11.8.8. Copy objects

This topic describes how to copy an object within a bucket or across buckets in the same region.

Precautions

- To copy an object, you must have read permissions on the source object and read and write permissions on the destination bucket.
- The source bucket and the destination bucket must have no retention policies configured. Otherwise, the error message `The object you specified is immutable.` is returned.
- The source bucket and the destination bucket must be in the same region. For example, objects cannot be copied from a bucket located in the China (Hangzhou) region to another bucket located in the China (Qingdao) region.

Copy a small object

You can call the `client.CopyObject` method to copy an object smaller than 1 GB from a source bucket to a destination bucket within the same region. The following table describes how to configure parameters when you call `client.CopyObject`.

Configuration method	Description
<code>CopyObjectOutcome OssClient::CopyObject(const CopyObjectRequest &request)</code>	Allows you to specify the metadata of the destination object and copy conditions. If the URLs of the source object and destination object are the same, the metadata that is specified in the request replaces the metadata of the destination object.

The following code provides an example on how to copy a small object:

```

#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize the information about the account that is used to access Object Storage S
ervice (OSS). */
    /* Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account t
o access OSS because the account has permissions on all API operations. We recommend that y
ou use a RAM user to call API operations or perform routine O&M. To create a RAM user, log
on to the RAM console. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    /* Specify the endpoint of the region in which the bucket is located. For example, if t
he bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-han
gzhou.aliyuncs.com. */
    std::string Endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
    /* Specify the name of the source bucket. Example: srcexamplebucket. */
    std::string SourceBucketName = "srcexamplebucket";
    /* Specify the name of the destination bucket. The destination bucket must be located i
n the same region as the source bucket. Example: destbucket. */
    std::string CopyBucketName = "destbucket";
    /* Specify the full path of the source object. The full path cannot contain the bucket
name. Example: srcdir/scrobject.txt. */
    std::string SourceObjectName = "srcdir/scrobject.txt";
    /* Specify the full path of the destination object. The full path cannot contain the bu
cket name. Example: destdir/destobject.txt. */
    std::string CopyObjectName = "destdir/destobject.txt";
    /* Initialize resources such as networks. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    CopyObjectRequest request(CopyBucketName, CopyObjectName);
    request.setCopySource(SourceBucketName, SourceObjectName);
    /* Copy the object. */
    auto outcome = client.CopyObject(request);
    if (!outcome.isSuccess()) {
        /* Handle exceptions. */
        std::cout << "CopyObject fail" <<
            ",code:" << outcome.error().Code() <<
            ",message:" << outcome.error().Message() <<
            ",requestId:" << outcome.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }
    /* Release resources such as networks. */
    ShutdownSdk();
    return 0;
}

```

Copy a large object

To copy an object that is larger than 1 GB in size, you must split the object into parts and sequentially copy the parts by using UploadPartCopy. To implement multipart copy, perform the following steps:

1. Use `client.InitiateMultipartUpload` to initiate a multipart copy task.
2. Use `client.UploadPartCopy` to copy the parts. All parts must be larger than 100 KB, except for the very last part.
3. Use `client.CompleteMultipartUpload` to complete the multipart copy task.

The following code provides an example on how to copy a large object:

```
#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize the information about the account that is used to access OSS. */
    /* Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    /* Specify the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com. */
    std::string Endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
    /* Specify the name of the source bucket. Example: srcexamplebucket. */
    std::string SourceBucketName = "srcexamplebucket";
    /* Specify the name of the destination bucket. The destination bucket must be located in the same region as the source bucket. Example: destbucket. */
    std::string CopyBucketName = "destbucket";
    /* Specify the full path of the source object. The full path cannot contain the bucket name. Example: srcdir/scrobjct.txt. */
    std::string SourceObjectName = "srcdir/scrobjct.txt";
    /* Specify the full path of the destination object. The full path cannot contain the bucket name. Example: destdir/destobjct.txt. */
    std::string CopyObjectName = "destdir/destobjct.txt";
    /* Initialize resources such as networks. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    auto getObjectMetaReq = GetObjectMetaRequest(SourceBucketName, SourceObjectName);
    auto getObjectMetaResult = GetObjectMeta(getObjectMetaReq);
    if (!getObjectMetaResult.isSuccess()) {
        std::cout << "GetObjectMeta fail" <<
            ",code:" << outcome.error().Code() <<
            ",message:" << outcome.error().Message() <<
            ",requestId:" << outcome.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }
    /* Obtain the size of the source object. */
    auto objectSize = getObjectMetaResult.result().ContentLength();
    /* Copy the large object. */
    InitiateMultipartUploadRequest initUploadRequest(CopyBucketName, CopyObjectName);
    /* Initiate a multipart copy task. */
    auto uploadIdResult = client.InitiateMultipartUpload(initUploadRequest);
    auto uploadId = uploadIdResult.result().UploadId();
```

```

int64_t partSize = 100 * 1024;
PartList partETagList;
int partCount = static_cast<int>(objectSize / partSize);
/* Calculate the number of parts. */
if (objectSize % partSize != 0) {
    partCount++;
}
/* Copy each part sequentially. */
for (int i = 1; i <= partCount; i++) {
    auto skipBytes = partSize * (i - 1);
    auto size = (partSize < objectSize - skipBytes) ? partSize : (objectSize - skipBytes);

    auto uploadPartCopyReq = UploadPartCopyRequest(CopyBucketName, CopyObjectName, SourceBucketName, SourceObjectName, uploadId, i + 1);
    uploadPartCopyReq.setCopySourceRange(skipBytes, skipBytes + size - 1);
    auto uploadPartOutcome = client.UploadPartCopy(uploadPartCopyReq);
    if (uploadPartOutcome.isSuccess()) {
        Part part(i, uploadPartOutcome.result().ETag());
        partETagList.push_back(part);
    }
    else {
        std::cout << "UploadPartCopy fail" <<
            ",code:" << outcome.error().Code() <<
            ",message:" << outcome.error().Message() <<
            ",requestId:" << outcome.error().RequestId() << std::endl;
    }
}

/* Complete the multipart copy task. */
CompleteMultipartUploadRequest request(CopyBucketName, CopyObjectName, partETagList, uploadId);
auto outcome = client.CompleteMultipartUpload(request);
if (!outcome.isSuccess()) {
    /* Handle exceptions. */
    std::cout << "CompleteMultipartUpload fail" <<
        ",code:" << outcome.error().Code() <<
        ",message:" << outcome.error().Message() <<
        ",requestId:" << outcome.error().RequestId() << std::endl;
    ShutdownSdk();
    return -1;
}

/* Release resources such as networks. */
ShutdownSdk();
return 0;
}

```

References

- For the complete sample code that is used to copy a large object and a small object, visit [GitHub](#).
- For more information about the API operation that you can call to copy a small object, see [CopyObject](#).
- For more information about the API operation that you can call to copy a large object, see [UploadPartCopy](#).

11.8.9. Disable overwrite for objects with the same name

By default, if you upload an object that has the same name as an existing object, the existing object is overwritten by the uploaded object. This topic describes how to configure the `x-oss-forbid-overwrite` request header to prevent objects from being overwritten by objects with the same names when you copy objects or perform simple upload or multipart upload.

Simple upload

The following code provides an example on how to prevent objects from being overwritten by objects that have the same names in simple upload:

```

#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /*Initialize the OSS account information.*/
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    std::string Endpoint = "yourEndpoint";
    std::string BucketName = "yourBucketName";
    std::string ObjectName = "yourObjectName";
    /*Initialize network resources.*/
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    std::shared_ptr<std::iostream> content = std::make_shared<std::stringstream>();
    *content << "test cpp sdk";
    /*Specify whether to overwrite the object with the same name.*/
    /*By default, if x-oss-forbid-overwrite is not specified, the object with the same name
is overwritten.*/
    /*If x-oss-forbid-overwrite is set to false, the object with the same name is overwritt
en.*/
    /*If x-oss-forbid-overwrite is set to true, the object with the same name is not overwr
itten. If an object with the same name already exists, an error is reported.*/
    ObjectMetaData meta;
    meta.addHeader("x-oss-forbid-overwrite", "true");
    PutObjectRequest request(BucketName, ObjectName, content, meta);
    /*Upload the object.*/
    auto outcome = client.PutObject(request);
    if (! outcome.IsSuccess()) {
        /*Handle exceptions.*/
        std::cout << "PutObject fail" <<
            ",code:" << outcome.error().Code() <<
            ",message:" << outcome.error().Message() <<
            ",requestId:" << outcome.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }
    /*Release network resources.*/
    ShutdownSdk();
    return 0;
}

```

Copy objects

- Copy a small object

The following code provides an example on how to copy a small object without overwriting the object with the same object name:


```

#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /*Initialize the OSS account information.*/
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    std::string Endpoint = "yourEndpoint";
    std::string SourceBucketName = "yourSourceBucketName";
    std::string CopyBucketName = "yourCopyBucketName";
    std::string SourceObjectName = "yourSourceObjectName";
    std::string CopyObjectName = "yourCopyObjectName";
    /*Initialize network resources.*/
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    /*Specify whether to overwrite the object with the same name.*/
    /*By default, if x-oss-forbid-overwrite is not specified, the object with the same na
me is overwritten.*/
    /*If x-oss-forbid-overwrite is set to false, the object with the same name is overwri
tten.*/
    /*If x-oss-forbid-overwrite is set to true, the object with the same name is not over
written. If an object with the same name already exists, an error is reported.*/
    ObjectMetaData meta;
    meta.addHeader("x-oss-forbid-overwrite", "true");
    CopyObjectRequest request(CopyBucketName, CopyObjectName, meta);
    request.setCopySource(SourceBucketName, SourceObjectName);
    /*Copy the object.*/
    auto outcome = client.CopyObject(request);
    if (! outcome.IsSuccess()) {
        /*Handle exceptions.*/
        std::cout << "CopyObject fail" <<
            ",code:" << outcome.error().Code() <<
            ",message:" << outcome.error().Message() <<
            ",requestId:" << outcome.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }
    /*Release network resources.*/
    ShutdownSdk();
    return 0;
}

```

- Copy a large object

The following code provides an example on how to prevent an existing object from being overwritten by a large object with the same name when you copy the large object by using multipart copy:

```

#include <alibabacloud/oss/OssClient.h>
#include <sstream>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /*Initialize the OSS account information.*/

```

```

/* Initialize the OSS account information. */
std::string AccessKeyId = "yourAccessKeyId";
std::string AccessKeySecret = "yourAccessKeySecret";
std::string Endpoint = "yourEndpoint";
std::string SourceBucketName = "yourSourceBucketName";
std::string CopyBucketName = "yourCopyBucketName";
std::string SourceObjectName = "yourSourceObjectName";
std::string CopyObjectName = "yourCopyObjectName";
/*Initialize network resources.*/
InitializeSdk();
ClientConfiguration conf;
OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
auto getObjectMetaReq = GetObjectMetaRequest(SourceBucketName, SourceObjectName);
auto getObjectMetaResult = GetObjectMeta(getObjectMetaReq);
if (! getObjectMetaResult.isSuccess()) {
    std::cout << "GetObjectMeta fail" <<
        ",code:" << outcome.error().Code() <<
        ",message:" << outcome.error().Message() <<
        ",requestId:" << outcome.error().RequestId() << std::endl;
    ShutdownSdk();
    return -1;
}

/*Query the size of the object to copy.*/
auto objectSize = getObjectMetaResult.result().ContentLength();
/*Copy the large object.*/
ObjectMetaData metaData;
std::stringstream ssDesc;
ssDesc << "/" << SourceBucketName << "/" << SourceObjectName;
std::string value = ssDesc.str();
metaData.addHeader("x-oss-copy-source", value);
/*Specify whether to overwrite the object with the same name.*/
/*By default, if x-oss-forbid-overwrite is not specified, the object with the same na
me is overwritten.*/
/*If x-oss-forbid-overwrite is set to false, the object with the same name is overwri
tten.*/
/*If x-oss-forbid-overwrite is set to true, the object with the same name is not over
written. If an object with the same name already exists, an error is reported.*/
metaData.addHeader("x-oss-forbid-overwrite", "true");
InitiateMultipartUploadRequest initUploadRequest(BucketName, ObjectName, metaData);
/*Initiate a multipart copy task.*/
auto uploadIdResult = client.InitiateMultipartUpload(initUploadRequest);
auto uploadId = uploadIdResult.result().UploadId();
int64_t partSize = 100 * 1024;
PartList partETagList;
int partCount = static_cast<int>(objectSize / partSize);
/*Calculate the number of parts.*/
if (objectSize % partSize != 0) {
    partCount++;
}
/*Copy each part sequentially.*/
for (int i = 1; i <= partCount; i++) {
    auto skipBytes = partSize * (i - 1);
    auto size = (partSize < objectSize - skipBytes) ? partSize : (objectSize - skipBy
tes);
    auto uploadPartCopyReq = UploadPartCopyRequest(CopyBucketName, CopyObjectName, So

```

```

    urceBucketName, SourceObjectName,uploadId, i + 1);
    uploadPartCopyReq.setCopySourceRange(skipBytes, skipBytes + size -1);
    auto uploadPartOutcome = client.UploadPartCopy(uploadPartCopyReq);
    if (uploadPartOutcome.isSuccess()) {
        Part part(i, uploadPartOutcome.result().ETag());
        partETagList.push_back(part);
    }
    else {
        std::cout << "UploadPartCopy fail" <<
            ",code:" << outcome.error().Code() <<
            ",message:" << outcome.error().Message() <<
            ",requestId:" << outcome.error().RequestId() << std::endl;
    }
}
/*Complete the multipart copy task.*/
CompleteMultipartUploadRequest request(CopyBucketName, CopyObjectName, partETagList,
uploadId);
/*Specify whether to overwrite the object with the same name.*/
/*By default, if x-oss-forbid-overwrite is not specified, the object with the same na
me is overwritten.*/
/*If x-oss-forbid-overwrite is set to false, the object with the same name is overwri
tten.*/
/*If x-oss-forbid-overwrite is set to true, the object with the same name is not over
written. If an object with the same name already exists, an error is reported.*/
request.Metadata().addHeader("x-oss-forbid-overwrite", "true");
auto outcome = client.CompleteMultipartUpload(request);
if (! outcome.isSuccess()) {
    /*Handle exceptions.*/
    std::cout << "CompleteMultipartUpload fail" <<
        ",code:" << outcome.error().Code() <<
        ",message:" << outcome.error().Message() <<
        ",requestId:" << outcome.error().RequestId() << std::endl;
    ShutdownSdk();
    return -1;
}
/*Release network resources.*/
ShutdownSdk();
return 0;
}

```

Multipart upload

The following code provides an example on how to prevent objects from being overwritten by newly uploaded objects that have the same names in multipart upload:

```

#include <alibabacloud/oss/OssClient.h>
int64_t getFileSize(const std::string& file)
{
    std::fstream f(file, std::ios::in | std::ios::binary);
    f.seekg(0, f.end);
    int64_t size = f.tellg();
    f.close();
    return size;
}

```

```

using namespace AlibabaCloud::OSS;
int main(void)
{
    /*Initialize the OSS account information.*/
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    std::string Endpoint = "yourEndpoint";
    std::string BucketName = "yourBucketName";
    std::string ObjectName = "yourObjectName";
    /*Initialize network resources.*/
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    /*Specify whether to overwrite the object with the same name.*/
    /*By default, if x-oss-forbid-overwrite is not specified, the object with the same name
is overwritten.*/
    /*If x-oss-forbid-overwrite is set to false, the object with the same name is overwritt
en.*/
    /*If x-oss-forbid-overwrite is set to true, the object with the same name is not overwr
itten. If an object with the same name already exists, an error is reported.*/
    ObjectMetaData meta;
    data.addHeader("x-oss-forbid-overwrite", "true");
    InitiateMultipartUploadRequest initUploadRequest(BucketName, ObjectName, data);
    /*Initiate the multipart upload task.*/
    auto uploadIdResult = client.InitiateMultipartUpload(initUploadRequest);
    auto uploadId = uploadIdResult.result().UploadId();
    std::string fileToUpload = "yourLocalFilename";
    int64_t partSize = 100 * 1024;
    PartList partETagList;
    auto fileSize = getFileSize(fileToUpload);
    int partCount = static_cast<int>(fileSize / partSize);
    /*Calculate the number of parts.*/
    if (fileSize % partSize != 0) {
        partCount++;
    }
    /*Upload each part sequentially.*/
    for (int i = 1; i <= partCount; i++) {
        auto skipBytes = partSize * (i - 1);
        auto size = (partSize < fileSize - skipBytes) ? partSize : (fileSize - skipBytes);
        std::shared_ptr<std::iostream> content = std::make_shared<std::fstream>(fileToUploa
d, std::ios::in|std::ios::binary);
        content->seekg(skipBytes, std::ios::beg);
        UploadPartRequest uploadPartRequest(BucketName, ObjectName, content);
        uploadPartRequest.setContentLength(size);
        uploadPartRequest.setUploadId(uploadId);
        uploadPartRequest.setPartNumber(i);
        auto uploadPartOutcome = client.UploadPart(uploadPartRequest);
        if (uploadPartOutcome.isSuccess()) {
            Part part(i, uploadPartOutcome.result().ETag());
            partETagList.push_back(part);
        }
        else {
            std::cout << "uploadPart fail" <<
                ",code:" << uploadPartOutcome.error().Code() <<

```

```

        ",message:" << uploadPartOutcome.error().Message() <<
        ",requestId:" << uploadPartOutcome.error().RequestId() << std::endl;
    }
}
/*Complete the multipart upload task.*/
CompleteMultipartUploadRequest request(BucketName, ObjectName);
request.setUploadId(uploadId);
request.setPartList(partETagList);
/*Specify whether to overwrite the object with the same name.*/
/*By default, if x-oss-forbid-overwrite is not specified, the object with the same name
is overwritten.*/
/*If x-oss-forbid-overwrite is set to false, the object with the same name is overwritt
en.*/
/*If x-oss-forbid-overwrite is set to true, the object with the same name is not overwr
itten. If an object with the same name already exists, an error is reported.*/
request.Metadata().addHeader("x-oss-forbid-overwrite", "true");
auto outcome = client.CompleteMultipartUpload(request);
if (! outcome.IsSuccess()) {
    /*Handle exceptions.*/
    std::cout << "CompleteMultipartUpload fail" <<
    ",code:" << outcome.error().Code() <<
    ",message:" << outcome.error().Message() <<
    ",requestId:" << outcome.error().RequestId() << std::endl;
    ShutdownSdk();
    return -1;
}
/*Release network resources.*/
ShutdownSdk();
return 0;
}

```

11.8.10. Restore objects

Notice

Restoration

If you want to read an Archive object or a Cold Archive object, you must restore the object in advance. It takes several minutes to restore an Archive object and takes several hours to restore a Cold Archive object.

The following section describes the status of an Archive object or a Cold Archive object throughout the restoration process:

1. By default, an Archive or a Cold Archive object is in the frozen state before restoration.
2. After you submit a restore request, the object is in the restoring state.
3. After the server completes the restore task, the object enters the restored state and you can read the object.
 - o Archive objects

For Archive objects, the restored state lasts 24 hours by default. During the 24 hours, if you call `RestoreObject` again, the restored state is extended by 24 hours. You can extend the restored state to seven days by calling `RestoreObject` six times during the first 24 hours of restored state. You can also configure the duration of the restored state in days by calling `RestoreObject` once. You can specify a duration of at most seven days.

- o Cold Archive objects

For Cold Archive objects, you can specify the duration of restored state and restoration priority. The duration of the restored state must be at least one day and at most 365 days. The time required to restore a Cold Archive object to the readable state is determined based on the restoration priority of the object:

- Expedited: The object is restored within one hour.
- Standard: The object is restored within two to five hours. If the `JobParameters` element is not passed in, the default restoration priority is Standard.
- Bulk: The object is restored within five to twelve hours.

4. After the restored state expires, the object returns to the frozen state.

Restore an Archive object

The following code provides an example on how to restore an Archive object:

```
#include <alibabacloud/oss/OssClient.h>
#include <thread>
#include <chrono>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize OSS account information. */
    /* Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to
    o access OSS, because the account has permissions on all API operations. We recommend that
    you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log
    on to the RAM console. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    /* Set yourEndpoint to the endpoint of the region in which the bucket is located. For e
    xample, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https:
    //oss-cn-hangzhou.aliyuncs.com. */
    std::string Endpoint = "yourEndpoint";
    /* Specify the name of the bucket. Example: examplebucket. */
    std::string BucketName = "examplebucket";
    /* Specify the full path of the object. The path cannot contain the bucket name. */
    std::string ObjectName = "yourObjectName";
    /* Initialize resources such as network resources. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    /* Restore the Archive object. */
    auto outcome = client.RestoreObject(BucketName, ObjectName);
    /* You cannot restore objects of the non-Archive storage class. */
    if (!outcome.IsSuccess()) {
        /* Handle exceptions. */
        std::cout << "RestoreObject fail" <<
```

```

        ",code:" << outcome.error().Code() <<
        ",message:" << outcome.error().Message() <<
        ",requestId:" << outcome.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }
    std::string onGoingRestore("ongoing-request=\"false\"");
    int maxWaitTimeInSeconds = 600;
    while (maxWaitTimeInSeconds > 0)
    {
        auto meta = client.HeadObject(BucketName, ObjectName);
        std::string restoreStatus = meta.result().HttpMetaData()["x-oss-restore"];
        std::transform(restoreStatus.begin(), restoreStatus.end(), restoreStatus.begin(), :
:tolower);
        if (!restoreStatus.empty() &&
            restoreStatus.compare(0, onGoingRestore.size(), onGoingRestore)==0) {
            std::cout << " success, restore status:" << restoreStatus << std::endl;
            /* The Archive object is restored. */
            break;
        }
        std::cout << " info, WaitTime:" << maxWaitTimeInSeconds
        << "; restore status:" << restoreStatus << std::endl;
        std::this_thread::sleep_for(std::chrono::seconds(10));
        maxWaitTimeInSeconds--;
    }
    if (maxWaitTimeInSeconds == 0)
    {
        std::cout << "RestoreObject fail, TimeoutException" << std::endl;
    }
    /* Release resources such as network resources. */
    ShutdownSdk();
    return 0;
}

```

Restore a Cold Archive object

The following code provides an example on how to restore a Cold Archive object:

```

#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize OSS account information. */
    /* Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to
    o access OSS, because the account has permissions on all API operations. We recommend that
    you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log
    on to the RAM console. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    /* Set yourEndpoint to the endpoint of the region in which the bucket is located. For e
    xample, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https:
    //oss-cn-hangzhou.aliyuncs.com. */
    std::string Endpoint = "yourEndpoint";
    /* Specify the name of the bucket. Example: examplebucket. */
    std::string BucketName = "examplebucket";
    /* Specify the full path of the object. The path cannot contain the bucket name. */
    std::string ObjectName = "yourObjectName";
    /* Initialize resources such as network resources. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    /* Refer to the following code if you set the storage class of the object that you want
    to upload to Cold Archive. */
    //auto content = std::make_shared<std::stringstream>("test");
    //PutObjectRequest putRequest(BucketName, ObjectName, content);
    //putRequest.Metadata().addHeader("x-oss-storage-class", "ColdArchive");
    //auto putOutcome = client.PutObject(putRequest);
    RestoreObjectRequest request(BucketName, ObjectName);
    /* Specify the duration for which the object can remain in the restored state. Unit: da
    ys. Default value: 1. */
    request.setDays(2);
    /* Set the restoration priority of the Cold Archive object. TierType::Expedited indicat
    es that the object is restored within 1 hour. */
    request.setTierType(TierType::Expedited);
    auto outcome = client.RestoreObject(request);
    if (!outcome.isSuccess()) {
        /* Handle exceptions. */
        std::cout << "Delete Bucket Inventory fail" <<
        ",code:" << outcome.error().Code() <<
        ",message:" << outcome.error().Message() <<
        ",requestId:" << outcome.error().RequestId() << std::endl;
    }
    /* Release resources such as network resources. */
    ShutdownSdk();
    return 0;
}

```

References

- For more information about the complete sample code that is used to restore Archive and Cold Archive objects, visit [GitHub](#).

-

11.8.11. Manage symbolic links

A symbolic link is a special object that points to an object. It is similar to a short cut used in Windows. This topic describes how to create a symbolic link and obtain the name of the object to which the symbolic link points.

Note

- To delete symbolic links, see [Delete objects](#).
- After you delete the object to which a symbolic link points, the symbolic link still exists, but the object cannot be accessed by using the symbolic link.

Create a symbolic link

The following code provides an example on how to create symbolic links. When you create a symbolic link, you can specify user metadata of the symbolic link.

```

#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to
    o access Object Storage Service (OSS) because the account has permissions on all API operations.
    We recommend that you use a Resource Access Management (RAM) user to call API operations or
    perform routine O&M. To create a RAM user, log on to the RAM console. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    /* Set yourEndpoint to the endpoint of the region in which the bucket is located. For example,
    if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https://oss-cn-hangzhou.aliyuncs.com. */
    std::string Endpoint = "yourEndpoint";
    /* Specify the name of the bucket. Example: examplebucket. */
    std::string BucketName = "examplebucket";
    /* Specify the full path of the object. Example: exampledir/exampleobject.txt. The full path
    of the object cannot contain the bucket name. */
    std::string ObjectName = "exampledir/exampleobject.txt";
    /* Specify the full path of the symbolic link. Example: shortcut/myobject.txt. */
    std::string LinkName = "shortcut/myobject.txt";
    /* Initialize resources such as networks. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    /* Set the HTTP headers. */
    auto meta = ObjectMetaData();
    meta.setContentType("text/plain");
    /* Set user metadata. */
    meta.UserMetaData()["meta"] = "meta-value";
    /* Create the symbolic link. */
    CreateSymlinkRequest request(BucketName, ObjectName, meta);
    request.SetSymlinkTarget(LinkObjectName);
    auto outcome = client.CreateSymlink(request);
    if (!outcome.isSuccess()) {
        /* Handle exceptions. */
        std::cout << "CreateSymlink fail" <<
        ",code:" << outcome.error().Code() <<
        ",message:" << outcome.error().Message() <<
        ",requestId:" << outcome.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }
    /* Release resources such as networks. */
    ShutdownSdk();
    return 0;
}

```

For more information about how to create symbolic links, see [PutSymlink](#).

Query the name of the object to which a symbolic link points

To query the name of the object to which a symbolic link points, you must have the permission to read the symbolic link.

The following code provides an example on how to obtain the name of the object to which the symbolic link points:

```
#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to
    o access OSS because the account has permissions on all API operations. We recommend that y
    ou use a RAM user to call API operations or perform routine operations and maintenance. To
    create a RAM user, log on to the RAM console. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    /* Set yourEndpoint to the endpoint of the region in which the bucket is located. For e
    xample, if the bucket is located in the China (Hangzhou) region, set yourEndpoint to https:
    //oss-cn-hangzhou.aliyuncs.com. */
    std::string Endpoint = "yourEndpoint";
    /* Specify the name of the bucket. Example: examplebucket. */
    std::string BucketName = "examplebucket";
    /* Specify the full path of the symbolic link. Example: shortcut/myobject.txt. */
    std::string LinkName = "shortcut/myobject.txt";
    /* Initialize resources such as networks. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    /* Obtain the name of the object to which the symbolic link points. */
    GetSymlinkRequest request(BucketName, LinkName);
    auto outcome = client.GetSymlink(request);
    if (!outcome.IsSuccess()) {
        /* Handle exceptions. */
        std::cout << "GetSymlink fail" <<
            ",code:" << outcome.error().Code() <<
            ",message:" << outcome.error().Message() <<
            ",requestId:" << outcome.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }
    else {
        std::cout << " GetSymlink success Symlink name:" << outcome.result().SymlinkTarget(
        ) << std::endl;
    }
    /* Release resources such as networks. */
    ShutdownSdk();
    return 0;
}
```

For more information about how to obtain symbolic links, see [GetSymlink](#).

11.9. Versioning

11.9.1. Versioning

The versioning state of a bucket applies to all of the objects in the bucket.

Versioning allows you to restore objects in a bucket to any previous point in time, and protects your data from being accidentally overwritten or deleted. A bucket can be in any one of the following versioning states: unversioned (default), versioning-enabled, or versioning-suspended.

For more information about versioning, see [Overview](#) in OSS Developer Guide.

Set the versioning state of a bucket

The following code provides an example on how to set the versioning state of a bucket to Enabled or Suspended:

```
#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /*Initialize the OSS account information.*/
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    std::string Endpoint = "yourEndpoint";
    std::string BucketName = "yourBucketName";
    /*Initialize network resources.*/
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    /*Set the versioning state of the bucket to Enabled or Suspended.*/
    SetBucketVersioningRequest setrequest(BucketName, VersioningStatus::Enabled);
    auto outcome = client.SetBucketVersioning(setrequest);
    if (!outcome.isSuccess()) {
        /*Handle exceptions.*/
        std::cout << "SetBucketVersioning fail" <<
            ",code:" << outcome.error().Code() <<
            ",message:" << outcome.error().Message() <<
            ",requestId:" << outcome.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }
    /*Release network resources.*/
    ShutdownSdk();
    return 0;
}
```

For more information about how to configure versioning for a bucket, see [PutBucketVersioning](#).

Query information about the versioning state of a bucket

The following code provides an example on how to query information about the versioning state of a bucket:

```
#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /*Initialize the OSS account information.*/
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    std::string Endpoint = "yourEndpoint";
    std::string BucketName = "yourBucketName";
    /*Initialize network resources.*/
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    /*Query information about the versioning state of the bucket.*/
    auto outcome = client.GetBucketVersioning(GetBucketVersioningRequest(BucketName));
    if (!outcome.isSuccess()) {
        /*Handle exceptions.*/
        std::cout << "GetBucketVersioning fail" <<
            ",code:" << outcome.error().Code() <<
            ",message:" << outcome.error().Message() <<
            ",requestId:" << outcome.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }
    /*Release network resources.*/
    ShutdownSdk();
    return 0;
}
```

For more information about how to obtain the versioning state of a bucket, see [GetBucketVersioning](#).

List the version information of all objects in a bucket

The following code provides an example on how to list the version information of all objects in a specified bucket, including delete markers:

```

#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /*Initialize the OSS account information.*/
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    std::string Endpoint = "yourEndpoint";
    std::string BucketName = "yourBucketName";
    std::string ObjectName = "yourObjectName";
    /*Initialize network resources.*/
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    ListObjectVersionsRequest request(BucketName);
    bool IsTruncated = false;
    do {
        auto outcome = client.ListObjectVersions(request);
        if (outcome.isSuccess()) {
            /*View the version information of listed objects whose version IDs are delete markers.*/
            for (auto const &marker : outcome.result().DeleteMarkerSummarys()) {
                std::cout << "marker key:" << marker.Key() << ",marker versionid:" << marker.VersionId() << std::endl;
            }
            /*View the version information of all listed objects.*/
            for (auto const &obj : outcome.result().ObjectVersionSummarys()) {
                std::cout << "object key:" << obj.Key() << ",object versionid:" << obj.VersionId() << std::endl;
            }
        }
        else {
            std::cout << "ListObjectVersions fail" <<
                ",code:" << outcome.error().Code() <<
                ",message:" << outcome.error().Message() <<
                ",requestId:" << outcome.error().RequestId() << std::endl;
            break;
        }
        request.setKeyMarker(outcome.result().NextKeyMarker());
        request.setVersionIdMarker(outcome.result().NextVersionIdMarker());
        IsTruncated = outcome.result().IsTruncated();
    } while (IsTruncated);
    /*Release network resources.*/
    ShutdownSdk();
    return 0;
}

```

For more information about how to list the versions of all objects in a bucket, including delete markers, see [GetBucketVersions\(ListObjectVersions\)](#).

11.9.2. Upload objects

This topic describes how to upload objects to a versioning-enabled bucket.

Simple upload

The following code provides an example for simple upload:

```
#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /*Initialize the OSS account information.*/
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    std::string Endpoint = "yourEndpoint";
    std::string BucketName = "yourBucketName";
    std::string ObjectName = "yourObjectName";
    /*Initialize network resources.*/
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    std::shared_ptr<std::iostream> content = std::make_shared<std::stringstream>();
    *content << "test cpp sdk";
    PutObjectRequest request(BucketName, ObjectName, content);
    /*Upload the object.*/
    auto outcome = client.PutObject(request);
    if (outcome.isSuccess()) {
        std::cout << "versionid:" << outcome.result().VersionId() << std::endl;
    }
    else {
        /*Handle exceptions.*/
        std::cout << "PutObject fail" <<
            ",code:" << outcome.error().Code() <<
            ",message:" << outcome.error().Message() <<
            ",requestId:" << outcome.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }
    /*Release network resources.*/
    ShutdownSdk();
    return 0;
}
```

For more information about simple upload, see [PutObject](#).

Append upload

In a bucket with versioning enabled or suspended, the AppendObject operation can be performed only on objects of which the current version is an appendable object.

 Note

- When you perform the AppendObject operation on an object of which the current version is an appendable object, OSS does not generate a historical version for the appendable object.
- When you perform the PutObject or DeleteObject operation on an object of which the current version is an appendable object, OSS stores the appendable object as a historical version and prevents the object from being further appended.
- You cannot perform the AppendObject operation on objects of which the current version is not an appendable object (such as a normal object or a delete marker).

The following code provides an example for append upload:

```
#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /*Initialize the OSS account information.*/
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    std::string Endpoint = "yourEndpoint";
    std::string BucketName = "yourBucketName";
    std::string ObjectName = "yourObjectName";
    /*Initialize network resources.*/
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    auto meta = ObjectMetaData();
    meta.setContentType("text/plain");
    /*If the object is appended for the first time, the append position is 0, and the returned value is the position for the next append. The position to start the next append is the total length of appended bytes and the append object. */
    std::shared_ptr<std::iostream> content1 = std::make_shared<std::stringstream>();
    *content1 << "Thank you for using Aliyun Object Storage Service!" ;
    AppendObjectRequest request(BucketName, ObjectName, content1, meta);
    request.setPosition(0L);
    /*Start the first append.*/
    auto result = client.AppendObject(request);
    /*View the version ID of the current append object.*/
    listresult.isSuccess() {
        std::cout << "versionid:" << result.result().VersionId() << std::endl;
    }
    else {
        /*Handle exceptions.*/
        std::cout << "AppendObject fail" <<
            ",code:" << result.error().Code() <<
            ",message:" << result.error().Message() <<
            ",requestId:" << result.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }
    std::shared_ptr<std::iostream> content2 = std::make_shared<std::stringstream>();
    *content2 << "Thank you for using Aliyun Object Storage Service!" ;
```



```

// ...
auto position = result.result().Length();
AppendObjectRequest appendObjectRequest(BucketName, ObjectName, content2);
appendObjectRequest.setPosition(position);
/*Start the second append.*/
auto outcome = client.AppendObject(appendObjectRequest);
if (outcome.isSuccess()) {
    std::cout << "versionid:" << outcome.result().VersionId() << std::endl;
}
else {
    /*Handle exceptions.*/
    std::cout << "AppendObject fail" <<
        ",code:" << outcome.error().Code() <<
        ",message:" << outcome.error().Message() <<
        ",requestId:" << outcome.error().RequestId() << std::endl;
    ShutdownSdk();
    return -1;
}
/*Release network resources.*/
ShutdownSdk();
return 0;
}

```

For more information about append upload, see [AppendObject](#).

Multipart upload

If you call `CompleteMultipartUpload` to complete the MultipartUpload task for an object in a bucket with versioning enabled or suspended, OSS generates a unique version ID for the object and returns it as the `x-oss-version-id` field in the response header.

The following code provides an example for multipart upload:

```

#include <alibabacloud/oss/OssClient.h>
int64_t getFileSize(const std::string& file)
{
    std::fstream f(file, std::ios::in | std::ios::binary);
    f.seekg(0, f.end);
    int64_t size = f.tellg();
    f.close();
    return size;
}
using namespace AlibabaCloud::OSS;
int main(void)
{
    /*Initialize the OSS account information.*/
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    std::string Endpoint = "yourEndpoint";
    std::string BucketName = "yourBucketName";
    std::string ObjectName = "yourObjectName";
    /*Initialize network resources.*/
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
}

```

```

InitiateMultipartUploadRequest initUploadRequest(BucketName, ObjectName);
/*Initiate the multipart upload task.*/
auto uploadIdResult = client.InitiateMultipartUpload(initUploadRequest);
auto uploadId = uploadIdResult.result().UploadId();
std::string fileToUpload = "yourLocalFilename";
int64_t partSize = 100 * 1024;
PartList partETagList;
auto fileSize = getFileSize(fileToUpload);
int partCount = static_cast<int>(fileSize / partSize);
/*Calculate the number of upload parts.*/
if (fileSize % partSize != 0) {
    partCount++;
}
/*Upload each part sequentially.*/
for (int i = 1; i <= partCount; i++) {
    auto skipBytes = partSize * (i - 1);
    auto size = (partSize < fileSize - skipBytes) ? partSize : (fileSize - skipBytes);
    std::shared_ptr<std::iostream> content = std::make_shared<std::fstream>(fileToUpload, std::ios::in|std::ios::binary);
    content->seekg(skipBytes, std::ios::beg);
    UploadPartRequest uploadPartRequest(BucketName, ObjectName, content);
    uploadPartRequest.setContentLength(size);
    uploadPartRequest.setUploadId(uploadId);
    uploadPartRequest.setPartNumber(i);
    auto uploadPartOutcome = client.UploadPart(uploadPartRequest);
    if (uploadPartOutcome.isSuccess()) {
        Part part(i, uploadPartOutcome.result().ETag());
        partETagList.push_back(part);
    }
    else {
        std::cout << "uploadPart fail" <<
            ",code:" << uploadPartOutcome.error().Code() <<
            ",message:" << uploadPartOutcome.error().Message() <<
            ",requestId:" << uploadPartOutcome.error().RequestId() << std::endl;
    }
}
/*Complete multipart upload.*/
CompleteMultipartUploadRequest request(BucketName, ObjectName);
request.setUploadId(uploadId);
request.setPartList(partETagList);
auto outcome = client.CompleteMultipartUpload(request);
if (outcome.isSuccess()) {
    std::cout << "versionid:" << outcome.result().VersionId() << std::endl;
}
else {
    /*Handle exceptions.*/
    std::cout << "CompleteMultipartUpload fail" <<
        ",code:" << outcome.error().Code() <<
        ",message:" << outcome.error().Message() <<
        ",requestId:" << outcome.error().RequestId() << std::endl;
    ShutdownSdk();
    return -1;
}
/*Release network resources.*/

```

```
ShutdownSdk();  
return 0;  
}
```

For more information about multipart upload, see [CompleteMultipartUpload](#).

11.9.3. Download objects

By default, calling the `GetObject` operation on an object in a versioning-enabled bucket returns only the current version of the object.

When you perform the `GetObject` operation on an object in a bucket:

- If the current version of the object is a delete marker, the 4040 Not Found error is returned.
- If you specify the `versionId` of the target object in the request, the specified version of the object is returned. For example, if the `versionId` is specified as null in the request, the version of which the `versionId` is null is returned.
- If the version specified in the request is a delete marker, the 405 Method Not Allowed error is returned.

The following code provides an example on how to download an object:

```

#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /*Initialize the OSS account information.*/
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    std::string Endpoint = "yourEndpoint";
    std::string BucketName = "yourBucketName";
    std::string ObjectName = "yourObjectName";
    /*Initialize network resources.*/
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    /*Download an object to local memory.*/
    GetObjectRequest request(BucketName, ObjectName);
    request.setVersionId("yourObjectVersionId");
    auto outcome = client.GetObject(request);
    if (outcome.isSuccess()) {
        std::cout << "getObjectToBuffer" << " success, Content-Length:" << outcome.result()
        .Metadata().ContentLength() << std::endl;
        /*Read the content of the downloaded object.*/
        std::string content;
        *(outcome.result().Content()) >> content;
        std::cout << "getObjectToBuffer" << "content:" << content << std::endl;
        /*View the version ID of the downloaded object.*/
        std::cout << "versionid:" << outcome.result().VersionId() << std::endl;
    }
    else {
        /*Handle exceptions.*/
        std::cout << "getObjectToBuffer fail" <<
        ",code:" << outcome.error().Code() <<
        ",message:" << outcome.error().Message() <<
        ",requestId:" << outcome.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }
    /*Release network resources.*/
    ShutdownSdk();
    return 0;
}

```

For more information about downloading objects, see [GetObject](#).

11.9.4. Copy objects

This topic describes how to copy objects in a versioned bucket. You can use `CopyObject` to copy an object that is up to 1 GB in size and use `UploadPartCopy` to copy an object larger than 1 GB.

Copy a small object

You can use `CopyObject` to copy an object that is up to 1 GB in size from a source bucket to a destination bucket within the same region.

- By default, the x-oss-copy-source header in a CopyObject request specifies the current version of the object to copy. If the current version of the object is a delete marker, Object Storage Service (OSS) returns 404 Not Found, which indicates that the object does not exist. You can add a version ID to x-oss-copy-source to copy the specified object version. Delete markers cannot be copied.
- You can copy a previous version of an object to the same bucket. The copied previous version becomes the new current version of the object. This way, the previous version of the object is recovered.
- If versioning is enabled for the destination bucket, OSS generates a unique version ID for the destination object. The version ID is returned in the response as the x-oss-version-id header value. If versioning is disabled or suspended for the destination bucket, OSS generates a version whose version ID is null for the destination object and overwrites the original version whose version ID is null.
- Appendable objects in a bucket for which versioning is enabled or suspended cannot be copied.

The following code provides an example on how to copy a small object:

```
#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize the OSS account information. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    std::string Endpoint = "yourEndpoint";
    std::string SourceBucketName = "yourSourceBucketName";
    std::string CopyBucketName = "yourCopyBucketName";
    std::string SourceObjectName = "yourSourceObjectName";
    std::string CopyObjectName = "yourCopyObjectName";
    /* Initialize network resources. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    CopyObjectRequest request(CopyBucketName, CopyObjectName);
    request.setCopySource(SourceBucketName, SourceObjectName);
    request.setVersionId("yourSourceObjectVersionId");
    /* Copy the specified version of the source object.*/
    auto outcome = client.CopyObject(request);
    if (outcome.isSuccess()) {
        std::cout << "versionid:" << outcome.result().VersionId() << std::endl;
    }
    else {
        /* Handle exceptions. */
        std::cout << "CopyObject fail" <<
            ",code:" << outcome.error().Code() <<
            ",message:" << outcome.error().Message() <<
            ",requestId:" << outcome.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }
    /* Release network resources. */
    ShutdownSdk();
    return 0;
}
```

For more information about how to copy small objects, see [CopyObject](#).

Copy a large object

To copy an object that is larger than 1 GB in size, you must split the object into parts and copy the parts by using UploadPartCopy.

By default, the UploadPartCopy operation uploads a part by copying data from the current version of an existing object. You can add versionId to the x-oss-copy-source request header to copy the specified version of the object. Example: x-oss-copy-source : /SourceBucketName/SourceObjectName?versionId=CAEQMxiBgICAof2D0BYiIDJhMGE3N2M1YTl1NDQzOGY5NTkyNTl3MGYyMzJm****.

 **Note** The value of SourceObjectName must be URL-encoded. The version ID of the copied object is returned in the response as the value of the x-oss-copy-source-version-id header.

If versionId is not specified in the request and the current version of the source object is a delete marker, OSS returns 404 Not Found. If versionId is specified in the request and the specified version of the source object is a delete marker, OSS returns 400 Bad Request.

The following code provides an example on how to use multipart copy to copy large objects:

```
#include <alibabacloud/oss/OssClient.h>
#include <sstream>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize the OSS account information. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    std::string Endpoint = "yourEndpoint";
    std::string SourceBucketName = "yourSourceBucketName";
    std::string CopyBucketName = "yourCopyBucketName";
    std::string SourceObjectName = "yourSourceObjectName";
    std::string CopyObjectName = "yourCopyObjectName";
    /* Initialize network resources. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    auto getObjectMetaReq = GetObjectMetaRequest(SourceBucketName, SourceObjectName);
    getObjectMetaReq.setVersionId("yourSourceObjectVersionId");
    auto getObjectMetaResult = GetObjectMeta(getObjectMetaReq);
    if (!getObjectMetaResult.isSuccess()) {
        std::cout << "GetObjectMeta fail" <<
            ",code:" << outcome.error().Code() <<
            ",message:" << outcome.error().Message() <<
            ",requestId:" << outcome.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }
    /* Query the size of the source object.*/
    auto objectSize = getObjectMetaResult.result().ContentLength();
    /* Copy the large source object.*/
    InitiateMultipartUploadRequest initUploadRequest(BucketName, ObjectName, metaData);
    /* Initiate a multipart copy task.*/
    auto uploadIdResult = client.InitiateMultipartUpload(initUploadRequest);
```

```

auto uploadId = uploadIdResult.result().UploadId();
int64_t partSize = 100 * 1024;
PartList partETagList;
int partCount = static_cast<int>(objectSize / partSize);
/* Calculate the number of parts to upload. */
if (objectSize % partSize != 0) {
    partCount++;
}
/* Copy each part sequentially.*/
for (int i = 1; i <= partCount; i++) {
    auto skipBytes = partSize * (i - 1);
    auto size = (partSize < objectSize - skipBytes) ? partSize : (objectSize - skipBytes);

    auto uploadPartCopyReq = UploadPartCopyRequest(CopyBucketName, CopyObjectName, SourceBucketName, SourceObjectName, uploadId, i + 1);
    uploadPartCopyReq.setCopySourceRange(skipBytes, skipBytes + size - 1);
    uploadPartCopyReq.setVersionId("yourSourceObjectVersionId");
    auto uploadPartOutcome = client.UploadPartCopy(uploadPartCopyReq);
    if (uploadPartOutcome.isSuccess()) {
        Part part(i, uploadPartOutcome.result().ETag());
        partETagList.push_back(part);
    }
    else {
        std::cout << "UploadPartCopy fail" <<
            ",code:" << outcome.error().Code() <<
            ",message:" << outcome.error().Message() <<
            ",requestId:" << outcome.error().RequestId() << std::endl;
    }
}
/* Complete the multipart copy task. */
CompleteMultipartUploadRequest request(CopyBucketName, CopyObjectName, partETagList, uploadId);
auto outcome = client.CompleteMultipartUpload(request);
if (outcome.isSuccess()) {
    std::cout << "versionid:" << outcome.result().VersionId() << std::endl;
}
else {
    /* Handle exceptions. */
    std::cout << "CompleteMultipartUpload fail" <<
        ",code:" << outcome.error().Code() <<
        ",message:" << outcome.error().Message() <<
        ",requestId:" << outcome.error().RequestId() << std::endl;
    ShutdownSdk();
    return -1;
}
/* Release network resources. */
ShutdownSdk();
return 0;
}

```

For more information about multipart copy, see [UploadPartCopy](#).

11.9.5. Delete objects

This topic describes how to delete a single object, multiple objects, or objects whose names contain a specified prefix from a versioned bucket.

Delete operations in a versioned bucket

When you delete an object from a versioned bucket, you must determine whether to specify a version ID in the request.

- Delete an object without specifying a version ID (temporary deletion)

By default, if you do not specify the version ID of the object that you want to delete in the request, Object Storage Service (OSS) does not delete the current version of the object but adds a delete marker to the object as the new current version. If you perform the `GetObject` operation on the object, OSS identifies the current version of the object as a delete marker and returns `404 Not Found`. Additionally, the `x-oss-delete-marker` header and the `x-oss-version-id` header that indicates the version ID of the created delete marker are included in the response to the `DeleteObject` request.

The value of the `x-oss-delete-marker` header in the response is true, which indicates that the `x-oss-version-id` value is the version ID of a delete marker.

- Delete an object by specifying a version ID (permanent deletion)

If you specify the version ID of the object that you want to delete in the request, OSS permanently deletes the version specified by the `versionId` field in the `params` parameter. To delete a version whose ID is null, add `params['versionId'] = "null"` to `params` in the request. OSS identifies the string "null" as the ID of the version to delete and deletes the version whose ID is null.

Delete a single object

The following examples describe how to permanently or temporarily delete a single object from a versioned bucket.

- Permanently delete an object from a versioned bucket

The following code provides an example on how to permanently delete an object with a specified version ID from a versioned bucket:


```

#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize the OSS account information. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    std::string Endpoint = "yourEndpoint";
    std::string BucketName = "yourBucketName";
    std::string ObjectName = "yourObjectName";
    /* Initialize network resources. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    /* Delete the object with the specified version ID or the object whose current version is a delete marker with the specified version ID. */
    auto outcome = client.DeleteObject(DeleteObjectRequest(BucketName, ObjectName, "yourObjectVersionIdOrDeleteMarkerVersionId"));
    /* If the version ID of an object is specified in the request, the value of the delete_marker header in the response is None and the value of the versionId header in the response is the version ID specified in the request. */
    /* If the version ID of a delete marker is specified in the request, the value of the delete_marker header in the response is True and the value of the versionId header in the response is the version ID specified in the request. */
    if (!outcome.IsSuccess()) {
        /* Handle exceptions. */
        std::cout << "DeleteObject fail" <<
            ",code:" << outcome.error().Code() <<
            ",message:" << outcome.error().Message() <<
            ",requestId:" << outcome.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }
    /* Release network resources. */
    ShutdownSdk();
    return 0;
}

```

- Temporarily delete an object from a versioned bucket

The following code provides an example on how to temporarily delete an object from a versioned bucket:

```

#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize the OSS account information. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    std::string Endpoint = "yourEndpoint";
    std::string BucketName = "yourBucketName";
    std::string ObjectName = "yourObjectName";
    /* Initialize network resources. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    DeleteObjectRequest request(BucketName, ObjectName);
    /* Temporarily delete an object without specifying its version ID. A delete marker is
added to the object. */
    auto outcome = client.DeleteObject(request);
    /* Display the version ID of the added delete marker. */
    if (outcome.isSuccess()) {
        std::cout << "versionid:" << outcome.result().VersionId() << ",DeleteMarker:" <<
outcome.result().DeleteMarker() << std::endl;
    }
    else {
        /* Handle exceptions. */
        std::cout << "PutObject fail" <<
        ",code:" << outcome.error().Code() <<
        ",message:" << outcome.error().Message() <<
        ",requestId:" << outcome.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }
    /* Release network resources. */
    ShutdownSdk();
    return 0;
}

```

For more information about how to delete a single object, see [DeleteObject](#).

Delete multiple objects

The following examples describe how to permanently or temporarily delete multiple objects from a versioned bucket.

- Permanently delete multiple objects from a versioned bucket

The following code provides an example on how to delete the objects with specified version IDs or the objects whose current versions are delete markers with specified version IDs:

```

#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize the OSS account information. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    std::string Endpoint = "yourEndpoint";
    std::string BucketName = "yourBucketName";
    /* Initialize network resources. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    DeleteObjectVersionsRequest request(BucketName);
    /* Specify the version IDs of the objects or delete markers you want to delete. */
    ObjectIdentifier obj1("yourObject1Name");
    obj1.setVersionId("yourVersionId");
    ObjectIdentifier obj2("yourObject2Name");
    obj2.setVersionId("obj2_del_marker_versionid");
    request.addObject(obj1);
    request.addObject(obj2);
    /* Delete the objects with specified version IDs or the objects whose current version
s are delete markers with specified version IDs. */
    auto outcome = client.DeleteObjectVersions(request);
    if (!outcome.isSuccess()) {
        /* Handle exceptions. */
        std::cout << "DeleteObjectVersions fail" <<
            ",code:" << outcome.error().Code() <<
            ",message:" << outcome.error().Message() <<
            ",requestId:" << outcome.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }
    /* Release network resources. */
    ShutdownSdk();
    return 0;
}

```

For more information about how to delete multiple objects, see [DeleteMultipleObjects](#).

- Temporarily delete multiple objects from a versioned bucket

The following code provides an example on how to temporarily delete multiple objects from a versioned bucket:

```

#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize the OSS account information. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    std::string Endpoint = "yourEndpoint";
    std::string BucketName = "yourBucketName";
    /* Initialize network resources. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    DeleteObjectVersionsRequest request(BucketName);
    /* Specify the names of the objects that you want to delete. */
    ObjectIdentifier obj1(ObjectName1);
    ObjectIdentifier obj2(ObjectName2);
    ObjectIdentifier obj3(ObjectName3);
    request.addObject(obj1);
    request.addObject(obj2);
    request.addObject(obj3);
    /* Delete the objects. */
    auto outcome = client.DeleteObjectVersions(request);
    if (outcome.isSuccess()) {
        for (auto const &obj : outcome.result().DeletedObjects()) {
            std::cout << "versionid:" << obj.VersionId() << ",DeleteMarker:" << obj.DeleteMarker() << std::endl;
        }
    }
    else {
        /* Handle exceptions. */
        std::cout << "DeleteObjectVersions fail" <<
            ",code:" << outcome.error().Code() <<
            ",message:" << outcome.error().Message() <<
            ",requestId:" << outcome.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }
    /* Release network resources. */
    ShutdownSdk();
    return 0;
}

```

Delete objects whose names contain a specified prefix

The following code provides an example on how to delete objects whose names contain a specified prefix:

```

#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize the OSS account information. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    std::string Endpoint = "yourEndpoint";
    std::string BucketName = "yourBucketName";
    /* Initialize network resources. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    ListObjectVersionsRequest request(BucketName);
    bool IsTruncated = false;
    do {
        request.setPrefix("yourkeyPrefix");
        auto outcome = client.ListObjectVersions(request);
        if (outcome.isSuccess()) {
            /* Display the versions of the listed objects and delete markers. */
            for (auto const &marker : outcome.result().DeleteMarkerSummarys()) {
                client.DeleteObject(DeleteObjectRequest(bucket, marker.Key(), marker.VersionId()));
            }
            /* List all versions of the objects whose names contain the specified prefixes and delete the versions. */
            for (auto const &obj : outcome.result().ObjectVersionSummarys()) {
                client.DeleteObject(DeleteObjectRequest(bucket, obj.Key(), obj.VersionId()));
            }
        }
        else {
            std::cout << "ListObjectVersions fail" <<
                ",code:" << outcome.error().Code() <<
                ",message:" << outcome.error().Message() <<
                ",requestId:" << outcome.error().RequestId() << std::endl;
            break;
        }
        request.setKeyMarker(outcome.result().NextKeyMarker());
        request.setVersionIdMarker(outcome.result().NextVersionIdMarker());
        IsTruncated = outcome.result().IsTruncated();
    } while (IsTruncated);
    /* Release network resources. */
    ShutdownSdk();
    return 0;
}

```

11.9.6. Restore objects

In a versioning-enabled bucket, different versions of the objects may have different storage classes. By default, the `RestoreObject` operation restores the current version of an object. To restore a specific version of the object, specify the version ID.

The following code provides an example on how to restore an object :

```
#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /*Initialize the OSS account information.*/
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    std::string Endpoint = "yourEndpoint";
    std::string BucketName = "yourBucketName";
    std::string ObjectName = "yourObjectName";
    /*Initialize network resources.*/
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    /*Restore an object with the Archive storage class.*/
    RestoreObjectRequest request(BucketName, ObjectName);
    request.setVersionId("yourObjectVersionId");
    auto outcome = client.RestoreObject(request);
    /*View the version ID of the object to restore.*/
    if (outcome.isSuccess()) {
        std::cout << "versionid:" << outcome.result().VersionId() << std::endl;
    }
    else {
        /*Handle exceptions.*/
        std::cout << "RestoreObject fail" <<
            ",code:" << outcome.error().Code() <<
            ",message:" << outcome.error().Message() <<
            ",requestId:" << outcome.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }
    /*Release network resources.*/
    ShutdownSdk();
    return 0;
}
```

For more information about restoring an object, see [RestoreObject](#).

11.9.7. Obtain object metadata

By default, calling the HeadObject operation on an object in a versioning-enabled bucket returns only metadata of the current version of the object.

 **Note** If the current version of the object is a delete marker, OSS returns 404 Not Found. If you specify the version ID in your request, OSS returns metadata of the object with the specified version ID.

The following code provides an example on how to obtain object metadata:

```
#include <alibabacloud/oss/OssClient.h>
#include <alibabacloud/oss/model/ObjectMeta.h>
```

```
using namespace AlibabaCloud::OSS;
int main(void)
{
    /*Initialize the OSS account information.*/
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    std::string Endpoint = "yourEndpoint";
    std::string BucketName = "yourBucketName";
    std::string ObjectName = "yourObjectName";
    /*Initialize network resources.*/
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    /*Obtain some metadata of the object.*/
    GetObjectMetaRequest request(BucketName, ObjectName);
    request.setVersionId("yourObjectVersionId");
    auto outcome = client.GetObjectMeta(request);
    if (!outcome.isSuccess()) {
        /*Handle exceptions.*/
        std::cout << "GetObjectMeta fail" <<
            ",code:" << outcome.error().Code() <<
            ",message:" << outcome.error().Message() <<
            ",requestId:" << outcome.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }
    else {
        auto metadata = outcome.result();
        std::cout << " get metadata success, VersionId:" << metadata.VersionId() << std
            ::endl;
    }
    /*Obtain all metadata of the object.*/
    HeadObjectRequest request1(BucketName, ObjectName);
    request1.setVersionId("yourObjectVersionId");
    outcome = client.HeadObject(request1);
    if (!outcome.isSuccess()) {
        /*Handle exceptions.*/
        std::cout << "HeadObject fail" <<
            ",code:" << outcome.error().Code() <<
            ",message:" << outcome.error().Message() <<
            ",requestId:" << outcome.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }
    else {
        auto headMeta = outcome.result();
        std::cout << " get headMeta success, VersionId:" << headMeta.VersionId() << std
            ::endl;
    }
    /*Release network resources.*/
    ShutdownSdk();
    return 0;
}
```

For more information about obtaining object metadata, see [HeadObject](#) and [GetObjectMeta](#).

11.9.8. Manage object ACLs

This topic describes how to manage ACLs of objects in a versioning-enabled bucket.

Configure the ACL of an object

PutObjectACL sets the ACL for the current version of the target object by default. If the current version of the target object is a delete marker, the 404 Not Found error is returned. You can set the versionId in the request to set the ACL for a specified version of the target object.

The following code provides an example on how to configure the ACL of an object:

```
#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /*Initialize the OSS account information.*/
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    std::string Endpoint = "yourEndpoint";
    std::string BucketName = "yourBucketName";
    std::string ObjectName = "yourObjectName";
    /*Initialize network resources.*/
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    /*Configure the ACL of the object.*/
    SetObjectAclRequest request(BucketName, ObjectName);
    request.setAcl(CannedAccessControlList::Private);
    request.setVersionId("yourObjectVersionId");
    auto outcome = client.SetObjectAcl(request);
    /*View the version ID of the object to configure the ACL.*/
    if (outcome.isSuccess()) {
        std::cout << "versionid:" << outcome.result().VersionId() << std::endl;
    }
    else {
        /*Handle exceptions.*/
        std::cout << "SetObjectAcl fail" <<
            ",code:" << outcome.error().Code() <<
            ",message:" << outcome.error().Message() <<
            ",requestId:" << outcome.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }
    /*Release network resources.*/
    ShutdownSdk();
    return 0;
}
```

For more information about setting the ACL for an object, see [PutObjectACL](#).

Obtain the ACL of an object

GetObjectACL obtains the ACL for the current version of the target object by default. If the current version of the object is a delete marker, the 404 Not Found error is returned. You can specify the versionId in the request to obtain the ACL for a specified version of the target object.

The following code provides an example on how to obtain the ACL of an object:

```
#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /*Initialize the OSS account information.*/
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    std::string Endpoint = "yourEndpoint";
    std::string BucketName = "yourBucketName";
    std::string ObjectName = "yourObjectName";
    /*Initialize network resources.*/
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    /*Obtain the ACL of the object.*/
    GetObjectAclRequest request(BucketName, ObjectName);
    request.setVersionId("yourObjectVersionId");
    auto outcome = client.GetObjectAcl(request);
    /*View the version ID of the object to obtain the ACL.*/
    if (outcome.isSuccess()) {
        std::cout << "GetObjectAcl success versionid:"
        << outcome.result().VersionId()
        << "Acl:" << outcome.result().Acl() << std::endl;
    }
    else {
        /*Handle exceptions.*/
        std::cout << "GetObjectAcl fail" <<
        ",code:" << outcome.error().Code() <<
        ",message:" << outcome.error().Message() <<
        ",requestId:" << outcome.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }
    /*Release network resources.*/
    ShutdownSdk();
    return 0;
}
```

For more information about obtaining the ACL for an object, see [GetObjectACL](#).

11.9.9. Manage symbolic links

This topic describes how to manage symbolic links in a versioning-enabled bucket.

Create a symbolic link

You can create a symbol link that directs to the current version of the target object.

 **Note** In a bucket with versioning enabled or suspended, you cannot create a symbol link for a delete marker.

A symbol link can have multiple versions that direct to different target objects. The version ID of the symbol link is automatically generated and is returned as the x-oss-version-id field in the response header.

The following code provides an example on how to create a symbolic link:

```
#include <alibabacloud/oss/OssClient.h>
#include <alibabacloud/oss/model/ObjectMetaData.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /*Initialize the OSS account information.*/
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    std::string Endpoint = "yourEndpoint";
    std::string BucketName = "yourBucketName";
    std::string ObjectName = "yourObjectName";
    std::string LinkName = "yourLinkName";
    /*Initialize network resources.*/
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    /*Set HTTP headers.*/
    auto meta = ObjectMetaData();
    meta.setContentType("text/plain");
    /*Set user metadata.*/
    meta.UserMetaData()["meta"] = "meta-value";
    /*Create the symbolic link.*/
    CreateSymlinkRequest request(BucketName, ObjectName, meta);
    request.SetSymlinkTarget(LinkObjectName);
    auto outcome = client.CreateSymlink(request);
    /*View the version ID of the created symbolic link.*/
    if (outcome.isSuccess()) {
        std::cout << "versionid:" << outcome.result().VersionId() << std::endl;
    }
    else {
        /*Handle exceptions.*/
        std::cout << "CreateSymlink fail" <<
            ",code:" << outcome.error().Code() <<
            ",message:" << outcome.error().Message() <<
            ",requestId:" << outcome.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }
    /*Release network resources.*/
    ShutdownSdk();
    return 0;
}
```

For more information about creating a symbol link, see [PutSymlink](#).

Obtain symbolic links

GetSymlink obtains the current version of the target symbol link by default. You can specify the versionId in the request to obtain the specified version of a symbol link. If the current version of the target symbol link is a delete marker, OSS returns the 404 Not Found error and includes `x-oss-delete-marker = true` and `x-oss-version-id` in the response header.

 **Note** You must have the read permission on a symbol link to obtain it.

The following code provides an example on how to obtain a symbolic link:

```
#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize the OSS account information. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    std::string Endpoint = "yourEndpoint";
    std::string BucketName = "yourBucketName";
    std::string LinkName = "yourLinkName";
    /* Initialize network resources. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    /* Obtain the symbolic link with the specified version ID. */
    GetSymlinkRequest request(BucketName, LinkName);
    request.setVersionId("yourObjectVersionId");
    auto outcome = client.GetSymlink(request);
    /* View the returned version ID of the symbolic link. */
    if (outcome.isSuccess()) {
        std::cout << "versionid:" << outcome.result().VersionId()
        << "Symlink name:" << outcome.result().SymlinkTarget() << std::endl;
    }
    else {
        /* Handle exceptions. */
        std::cout << "GetSymlink fail" <<
        ",code:" << outcome.error().Code() <<
        ",message:" << outcome.error().Message() <<
        ",requestId:" << outcome.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }
    /* Release network resources. */
    ShutdownSdk();
    return 0;
}
```

For more information about obtaining a symbol link, see [GetSymlink](#).

11.10. Object tagging

11.10.1. Configure object tagging

Object Storage Service (OSS) allows you to configure object tags to classify objects. You can configure lifecycle rules and control access to objects based on tags.

 **Note** For more information about object tags, see [Object tagging](#).

Add tags to an object when you upload the object

- Add tags to an object when you upload the object by using simple upload

The following code provides an example on how to add tags to an object when you upload the object by using simple upload:

```

#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize the information about the account that is used to access Object Storage
    Service (OSS). */
    /* The AccessKey pair of an Alibaba Cloud account has permissions on all API operations.
    Using these credentials to access OSS is a high-risk operation. We recommend that you
    use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on
    to the RAM console. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    /* Set yourEndpoint to the endpoint of the region in which the bucket is located. For
    example, if the bucket is located in the China (Hangzhou) region, set the endpoint to
    https://oss-cn-hangzhou.aliyuncs.com. */
    std::string Endpoint = "yourEndpoint";
    /* Specify the bucket name. Example: examplebucket. */
    std::string BucketName = "examplebucket";
    /* Specify the full path of the object. The full path cannot contain the bucket name.
    Example: exampledir/exampleobject.txt. */
    std::string ObjectName = "exampledir/exampleobject.txt";
    /* Initialize resources such as networks. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    std::shared_ptr<std::iostream> content = std::make_shared<std::stringstream>();
    *content << "test cpp sdk";
    PutObjectRequest request(BucketName, ObjectName, content);
    /* Specify the tags that you want to add to the object. */
    Tagging tagging;
    tagging.addTag(Tag("key1", "value1"));
    tagging.addTag(Tag("key2", "value2"));
    request.setTagging(tagging.toQueryParameters());
    /* Upload the object. */
    auto outcome = client.PutObject(request);
    if (!outcome.isSuccess()) {
        /* Handle exceptions. */
        std::cout << "PutObject fail" <<
        ",code:" << outcome.error().Code() <<
        ",message:" << outcome.error().Message() <<
        ",requestId:" << outcome.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }
    /* Release resources such as networks. */
    ShutdownSdk();
    return 0;
}

```

- Add tags to an object when you upload the object by using multipart upload

The following code provides an example on how to add tags to an object when you upload the object by using multipart upload:

```

#include <alibabacloud/oss/OssClient.h>

```

```

#include <alibabacloud/oss/ossclient.h>
int64_t getFileSize(const std::string& file)
{
    std::fstream f(file, std::ios::in | std::ios::binary);
    f.seekg(0, f.end);
    int64_t size = f.tellg();
    f.close();
    return size;
}
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize the information about the account that is used to access OSS. */
    /* The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to access OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    /* Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com. */
    std::string Endpoint = "yourEndpoint";
    /* Specify the bucket name. Example: examplebucket. */
    std::string BucketName = "examplebucket";
    /* Specify the full path of the object. The full path cannot contain the bucket name. Example: exampledir/exampleobject.txt. */
    std::string ObjectName = "exampledir/exampleobject.txt";
    /* Initialize resources such as networks. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    InitiateMultipartUploadRequest initUploadRequest(BucketName, ObjectName);
    /* Specify the tags that you want to add to the object. */
    Tagging tagging;
    tagging.addTag(Tag("key1", "value1"));
    tagging.addTag(Tag("key2", "value2"));
    initUploadRequest.setTagging(tagging.toQueryParameters());
    /* Initiate a multipart upload task. */
    auto uploadIdResult = client.InitiateMultipartUpload(initUploadRequest);
    auto uploadId = uploadIdResult.result().UploadId();
    std::string fileToUpload = "yourLocalFilename";
    int64_t partSize = 100 * 1024;
    PartList partETagList;
    auto fileSize = getFileSize(fileToUpload);
    int partCount = static_cast<int>(fileSize / partSize);
    /* Calculate the number of parts. */
    if (fileSize % partSize != 0) {
        partCount++;
    }
    /* Upload each part sequentially. */
    for (int i = 1; i <= partCount; i++) {
        auto skipBytes = partSize * (i - 1);
        auto size = (partSize < fileSize - skipBytes) ? partSize : (fileSize - skipBytes)
;

```

```

        std::shared_ptr<std::iostream> content = std::make_shared<std::fstream>(fileToUpload, std::ios::in|std::ios::binary);
        content->seekg(skipBytes, std::ios::beg);
        UploadPartRequest uploadPartRequest(BucketName, ObjectName, content);
        uploadPartRequest.setContentLength(size);
        uploadPartRequest.setUploadId(uploadId);
        uploadPartRequest.setPartNumber(i);
        auto uploadPartOutcome = client.UploadPart(uploadPartRequest);
        if (uploadPartOutcome.isSuccess()) {
            Part part(i, uploadPartOutcome.result().ETag());
            partETagList.push_back(part);
        }
        else {
            std::cout << "uploadPart fail" <<
                ",code:" << uploadPartOutcome.error().Code() <<
                ",message:" << uploadPartOutcome.error().Message() <<
                ",requestId:" << uploadPartOutcome.error().RequestId() << std::endl;
        }
    }

    /* Complete the multipart upload task. */
    CompleteMultipartUploadRequest request(BucketName, ObjectName);
    request.setUploadId(uploadId);
    request.setPartList(partETagList);
    auto outcome = client.CompleteMultipartUpload(request);
    if (!outcome.isSuccess()) {
        /* Handle exceptions. */
        std::cout << "CompleteMultipartUpload fail" <<
            ",code:" << outcome.error().Code() <<
            ",message:" << outcome.error().Message() <<
            ",requestId:" << outcome.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }

    /* Release resources such as networks. */
    ShutdownSdk();
    return 0;
}

```

- Add tags to an object when you upload the object by using append upload

The following code provides an example on how to add tags to an object when you upload the object by using append upload:

```

#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize the information about the account that is used to access OSS. */
    /* The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to access OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    /* Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com. */
    std::string Endpoint = "yourEndpoint";
    /* Specify the bucket name. Example: examplebucket. */
    std::string BucketName = "examplebucket";
    /* Specify the full path of the object. The full path cannot contain the bucket name. Example: exampledir/exampleobject.txt. */
    std::string ObjectName = "exampledir/exampleobject.txt";
    /* Initialize resources such as networks. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    auto meta = ObjectMeta();
    meta.setContentType("text/plain");
    /* Append data to the object. The default value of position from which the append operation starts is 0. */
    std::shared_ptr<std::iostream> content1 = std::make_shared<std::stringstream>();
    *content1 << "Thank you for using Aliyun Object Storage Service!";
    AppendObjectRequest request(BucketName, ObjectName, content1, meta);
    /* Specify the tags that you want to add to the object. */
    Tagging tagging;
    tagging.addTag(Tag("key1", "value1"));
    tagging.addTag(Tag("key2", "value2"));
    request.setTagging(tagging.toQueryParameters());
    /* Perform the append operation. */
    auto result = client.AppendObject(request);
    if (!result.isSuccess()) {
        /* Handle exceptions. */
        std::cout << "AppendObject fail" <<
            ",code:" << result.error().Code() <<
            ",message:" << result.error().Message() <<
            ",requestId:" << result.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }
    /* Release resources such as networks. */
    ShutdownSdk();
    return 0;
}

```

- Add tags to an object when you upload the object by using resumable upload

The following code provides an example on how to add tags to an object when you upload the object by using resumable upload:

```
#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize the information about the account that is used to access OSS. */
    /* The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to access OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    /* Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com. */
    std::string Endpoint = "yourEndpoint";
    /* Specify the bucket name. Example: examplebucket. */
    std::string BucketName = "examplebucket";
    /* Specify the full path of the object. The full path cannot contain the bucket name. Example: exampledir/exampleobject.txt. */
    std::string ObjectName = "exampledir/exampleobject.txt";
    std::string UploadFilePath = "yourUploadfilePath";
    /* Initialize resources such as networks. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    /* Start resumable upload. */
    UploadObjectRequest request(BucketName, ObjectName, YourUploadFileName);
    /* Add tags to the object. */
    Tagging tagging;
    tagging.addTag(Tag("key1", "value1"));
    tagging.addTag(Tag("key2", "value2"));
    request.setTagging(tagging.toQueryParameters());
    auto outcome = client.ResumableUploadObject(request);
    if (!outcome.isSuccess()) {
        /* Handle exceptions. */
        std::cout << "ResumableUploadObject fail" <<
            ",code:" << outcome.error().Code() <<
            ",message:" << outcome.error().Message() <<
            ",requestId:" << outcome.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }
    /* Release resources such as networks. */
    ShutdownSdk();
    return 0;
}
```

Add tags to an uploaded object or modify the tags of the object

The following code provides an example on how to add tags to an uploaded object or modify the tags of the object:

```

#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize the information about the account that is used to access OSS. */
    /* The AccessKey pair of an Alibaba Cloud account has permissions on all API operations
     . Using these credentials to access OSS is a high-risk operation. We recommend that you use
     a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to t
     he RAM console. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    /* Set yourEndpoint to the endpoint of the region in which the bucket is located. For e
     xample, if the bucket is located in the China (Hangzhou) region, set the endpoint to https:
     //oss-cn-hangzhou.aliyuncs.com. */
    std::string Endpoint = "yourEndpoint";
    /* Specify the bucket name. Example: examplebucket. */
    std::string BucketName = "examplebucket";
    /* Specify the full path of the object. The full path cannot contain the bucket name. E
     xample: exampledir/exampleobject.txt. */
    std::string ObjectName = "exampledir/exampleobject.txt";
    /* Initialize resources such as networks. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    std::shared_ptr<std::iostream> content = std::make_shared<std::stringstream>();
    *content << "test cpp sdk";
    PutObjectRequest request(BucketName, ObjectName, content);
    /* Upload the object. */
    auto outcome = client.PutObject(request);
    /* Specify the tags that you want to add to the object. */
    Tagging tagging;
    tagging.addTag(Tag("key1", "value1"));
    tagging.addTag(Tag("key2", "value2"));
    auto putTaggingOutcome = client.SetObjectTagging(SetObjectTaggingRequest(BucketName, Ob
    jectName, tagging));
    if (!putTaggingOutcome.isSuccess()) {
        /* Handle exceptions. */
        std::cout << "SetObjectTagging fail" <<
        ",code:" << putTaggingOutcome.error().Code() <<
        ",message:" << putTaggingOutcome.error().Message() <<
        ",requestId:" << putTaggingOutcome.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }
    /* Release resources such as networks. */
    ShutdownSdk();
    return 0;
}

```

Add tags to an object when you copy the object

- The following code provides an example on how to add tags to an object smaller than 1 GB in size when you copy the object by using simple copy:

```

#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize the information about the account that is used to access OSS. */
    /* The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to access OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    /* Set yourEndpoint to the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com. */
    std::string Endpoint = "yourEndpoint";
    /* Specify the bucket name. Example: examplebucket. */
    std::string BucketName = "examplebucket";
    /* Specify the full path of the destination object. The full path cannot contain the bucket name. Example: exampledir/exampleobject.txt. */
    std::string ObjectName = "exampledir/exampleobject.txt";
    /* Specify the full path of the source object. */
    std::string CopyObjectName = "sourcedir/scrobject.txt";
    /* Initialize resources such as networks. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    std::shared_ptr<std::iostream> content = std::make_shared<std::stringstream>();
    *content << "test cpp sdk";
    PutObjectRequest request(BucketName, ObjectName, content);
    /* Specify the tags that you want to add to the object. */
    Tagging tagging;
    tagging.addTag(Tag("key1", "value1"));
    tagging.addTag(Tag("key2", "value2"));
    request.setTagging(tagging.toQueryParameters());
    /* Upload the object. */
    auto outcome = client.PutObject(request);
    /* Add tags to the object when you copy the object. */
    CopyObjectRequest cpRequest(yourBucketName, CopyObjectName);
    cpRequest.setCopySource(yourBucketName, yourObjectName);
    Tagging tagging2;
    tagging2.addTag(Tag("key1-2", "value1-2"));
    tagging2.addTag(Tag("key2-2", "value2-2"));
    tagging2.addTag(Tag("key3-2", "value3-2"));
    cpRequest.setTagging(tagging2.toQueryParameters());
    /* Replace the tags of the destination object with the specified tags in the request. */
    cpRequest.setTaggingDirective(CopyActionList::Replace);
    /* Copy the object. */
    auto copyoutcome = client.CopyObject(cpRequest);
    if (!copyoutcome.isSuccess()) {
        /* Handle exceptions. */
        std::cout << "CopyObject fail" <<
            ",code:" << copyoutcome.error().Code() <<
            ",message:" << copyoutcome.error().Message() <<

```

```

        ",requestId:" << copyoutcome.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }
    /* Release resources such as networks. */
    ShutdownSdk();
    return 0;
}

```

- The following code provides an example on how to add tags to an object larger than 1 GB in size when you copy the object by using multipart copy:

```

#include <alibabacloud/oss/OssClient.h>
#include <sstream>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize the information about the account that is used to access OSS. */
    /* The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to access OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    /* Specify the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com. */
    std::string Endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
    /* Specify the name of the source bucket. Example: srcexamplebucket. */
    std::string SourceBucketName = "srcexamplebucket";
    /* Specify the name of the destination bucket. The destination bucket must be located in the same region as the source bucket. Example: destbucket. */
    std::string CopyBucketName = "destbucket";
    /* Specify the full path of the source object. The full path cannot contain the bucket name. Example: srcdir/scrobject.txt. */
    std::string SourceObjectName = "srcdir/scrobject.txt";
    /* Specify the full path of the destination object. The full path cannot contain the bucket name. Example: destdir/destobject.txt. */
    std::string CopyObjectName = "destdir/destobject.txt";
    /* Initialize resources such as networks. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    auto getObjectMetaReq = GetObjectMetaRequest(SourceBucketName, SourceObjectName);
    auto getObjectMetaResult = GetObjectMeta(getObjectMetaReq);
    if (!getObjectMetaResult.isSuccess()) {
        std::cout << "GetObjectMeta fail" <<
            ",code:" << outcome.error().Code() <<
            ",message:" << outcome.error().Message() <<
            ",requestId:" << outcome.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }
    /* Query the size of the object that you want to copy. */
}

```

```

auto objectSize = getObjectMetaResult.result().ContentLength();
/* Copy the large object. */
ObjectMetaData metaData;
std::stringstream ssDesc;
ssDesc << "/" << SourceBucketName << "/" << SourceObjectName;
std::string value = ssDesc.str();
metaData.addHeader("x-oss-copy-source", value);
InitiateMultipartUploadRequest initUploadRequest(BucketName, ObjectName, metaData);
/* Specify the tags that you want to add to the object. */
Tagging tagging;
tagging.addTag(Tag("key1", "value1"));
tagging.addTag(Tag("key2", "value2"));
initUploadRequest.setTagging(tagging.toQueryParameters());
/* Initiate a multipart copy task. */
auto uploadIdResult = client.InitiateMultipartUpload(initUploadRequest);
auto uploadId = uploadIdResult.result().UploadId();
int64_t partSize = 100 * 1024;
PartList partETagList;
int partCount = static_cast<int>(objectSize / partSize);
/* Calculate the number of parts. */
if (objectSize % partSize != 0) {
    partCount++;
}
/* Copy each part sequentially. */
for (int i = 1; i <= partCount; i++) {
    auto skipBytes = partSize * (i - 1);
    auto size = (partSize < objectSize - skipBytes) ? partSize : (objectSize - skipBytes);
    auto uploadPartCopyReq = UploadPartCopyRequest(CopyBucketName, CopyObjectName, SourceBucketName, SourceObjectName, uploadId, i + 1);
    uploadPartCopyReq.setCopySourceRange(skipBytes, skipBytes + size - 1);
    auto uploadPartOutcome = client.UploadPartCopy(uploadPartCopyReq);
    if (uploadPartOutcome.isSuccess()) {
        Part part(i, uploadPartOutcome.result().ETag());
        partETagList.push_back(part);
    }
    else {
        std::cout << "UploadPartCopy fail" <<
            ",code:" << outcome.error().Code() <<
            ",message:" << outcome.error().Message() <<
            ",requestId:" << outcome.error().RequestId() << std::endl;
    }
}
/* Complete the multipart copy task. */
CompleteMultipartUploadRequest request(CopyBucketName, CopyObjectName, partETagList, uploadId);
auto outcome = client.CompleteMultipartUpload(request);
if (!outcome.isSuccess()) {
    /* Handle exceptions. */
    std::cout << "CompleteMultipartUpload fail" <<
        ",code:" << outcome.error().Code() <<
        ",message:" << outcome.error().Message() <<
        ",requestId:" << outcome.error().RequestId() << std::endl;
    ShutdownSdk();
    return -1;
}

```

```
        return -1;
    }
    /* Release resources such as networks. */
    ShutdownSdk();
    return 0;
}
```

Add tags to a symbolic link

The following code provides an example on how to add tags to a symbolic link:

```

#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize the information about the account that is used to access OSS. */
    /* The AccessKey pair of an Alibaba Cloud account has permissions on all API operations
     . Using these credentials to access OSS is a high-risk operation. We recommend that you use
     a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to t
     he RAM console. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    /* Set yourEndpoint to the endpoint of the region in which the bucket is located. For e
     xample, if the bucket is located in the China (Hangzhou) region, set the endpoint to https:
     //oss-cn-hangzhou.aliyuncs.com. */
    std::string Endpoint = "yourEndpoint";
    /* Specify the bucket name. Example: examplebucket. */
    std::string BucketName = "examplebucket";
    /* Specify the full path of the object. The full path cannot contain the bucket name. E
     xample: exampledir/exampleobject.txt. */
    std::string ObjectName = "exampledir/exampleobject.txt";
    /* Specify the name of the symbolic link. Example: symlink.txt.
    std::string LinkName = "symlink.txt";
    /* Initialize resources such as networks. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    std::shared_ptr<std::iostream> content = std::make_shared<std::stringstream>();
    *content << "test cpp sdk";
    PutObjectRequest request(BucketName, ObjectName, content);
    /* Upload the object. /
    auto outcome = client.PutObject(request);
    /* Create the symbolic link. */
    CreateSymlinkRequest csRequest(BucketName, ObjectName);
    csRequest.SetSymlinkTarget(LinkName);
    /* Specify the tags that you want to add to the symbolic link. */
    Tagging tagging;
    tagging.addTag(Tag("key1", "value1"));
    tagging.addTag(Tag("key2", "value2"));
    csRequest.setTagging(tagging.toQueryParameters());
    auto csoutcome = client.CreateSymlink(csRequest);
    if (!csoutcome.isSuccess()) {
        /* Handle exceptions. */
        std::cout << "CreateSymlink fail" <<
            ",code:" << csoutcome.error().Code() <<
            ",message:" << csoutcome.error().Message() <<
            ",requestId:" << csoutcome.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }
    /* Release resources such as networks. */
    ShutdownSdk();
    return 0;
}

```

References

- For the complete sample code that is used to configure object tagging, visit [GitHub](#).
- For more information about the API operation that you can call to configure object tagging, see [PutObjectTagging](#).

11.10.2. Obtain the tags added to an object

This topic describes how to obtain the tags added to an object.

The object tagging function uses a key-value pair to tag an object. For more information about the object tagging function, see [Object tagging](#).

Run the following code to obtain the tags added to an object:


```

#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initializes the Alibaba Cloud OSS account information. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    std::string Endpoint = "yourEndpoint";
    std::string BucketName = "yourBucketName";
    std::string ObjectName = "yourObjectName";
    /* Initializes resources such as networks. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    std::shared_ptr<std::iostream> content = std::make_shared<std::stringstream>();
    *content << "test cpp sdk";
    PutObjectRequest request(BucketName, ObjectName, content);
    /* Uploads the object. */
    auto outcome = client.PutObject(request);
    /* Configures the tags to be added. */
    Tagging tagging;
    tagging.addTag(Tag("key1", "value1"));
    tagging.addTag(Tag("key2", "value2"));
    auto putTaggingOutcome = client.SetObjectTagging(SetObjectTaggingRequest(BucketName, ObjectName, tagging));
    /* Obtains the tags added to the object. */
    getTaggingOutcome = client.GetObjectTagging(GetObjectTaggingRequest(BucketName, key));
    if (!getTaggingOutcome.isSuccess()) {
        /* Handles exceptions. */
        std::cout << "getTaggingOutcome fail" <<
            ",code:" << getTaggingOutcome.error().Code() <<
            ",message:" << getTaggingOutcome.error().Message() <<
            ",requestId:" << getTaggingOutcome.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }
    else {
        auto taglist = getTaggingOutcome.result().Tagging();
        for (const auto& tag : taglist)
        {
            std::cout <<"GetObjectTagging success, Key:"
                << tag.Key() << "; Value:" << tag.Value() << std::endl;
        }
    }
    /* Releases resources such as networks. */
    ShutdownSdk();
    return 0;
}

```

For more information about how to obtain the tags added to an object, see [GetObjectTagging](#).

11.10.3. Delete the tags added to an object

This topic describes how to delete the tags added to an object.

The object tagging function uses a key-value pair to tag an object. For more information about the object tagging function, see [Object tagging](#).

Run the following code to delete the tags added to an object:

```
#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initializes the Alibaba Cloud OSS account information. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    std::string Endpoint = "yourEndpoint";
    std::string BucketName = "yourBucketName";
    std::string ObjectName = "yourObjectName";
    /* Initializes resources such as networks. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    std::shared_ptr<std::iostream> content = std::make_shared<std::stringstream>();
    *content << "test cpp sdk";
    PutObjectRequest request(BucketName, ObjectName, content);
    /* Uploads the object */
    auto outcome = client.PutObject(request);
    /* Configures the tags to be added. */
    Tagging tagging;
    tagging.addTag(Tag("key1", "value1"));
    tagging.addTag(Tag("key2", "value2"));
    auto putTaggingOutcome = client.SetObjectTagging(SetObjectTaggingRequest(BucketName, ObjectName, tagging));
    /* Deletes the tags added to the object. */
    auto delTaggingOutcome = client.DeleteObjectTagging(DeleteObjectTaggingRequest(BucketName, ObjectName));
    if (!delTaggingOutcome.isSuccess()) {
        /* Handles exceptions. */
        std::cout << "delTaggingOutcome fail" <<
            ",code:" << delTaggingOutcome.error().Code() <<
            ",message:" << delTaggingOutcome.error().Message() <<
            ",requestId:" << delTaggingOutcome.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }
    /* Releases resources such as networks. */
    ShutdownSdk();
    return 0;
}
```

For more information about how to delete the tags added to an object, see [DeleteObjectTagging](#).

11.10.4. Object tagging and lifecycle rules

Lifecycle rules can take effect on objects with specified prefixes or tags. You can also specify prefixes and tags as the condition of a lifecycle rule at the same time.

 **Note** If you set a specified tag as the condition of a lifecycle rule, the rule applies to an object only when the key and value in the tag of the object both match the specified tag. If you specify a prefix and multiple tags as the condition of a lifecycle rule, the rule applies to an object only when the prefix and tags of the object match the prefix and all tags specified in the rule.

Set tags as the matching condition of a lifecycle rule

Run the following code to set tags as the matching condition of a lifecycle rule:

```
#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initializes the Alibaba Cloud OSS account information. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    std::string Endpoint = "yourEndpoint";
    std::string BucketName = "yourBucketName";
    /* Initializes resources such as networks. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    SetBucketLifecycleRequest request(BucketName);
    /* Configures the tags to be set as the matching condition. */
    Tagging tagging;
    tagging.addTag(Tag("key1", "value1"));
    tagging.addTag(Tag("key2", "value2"));
    /* Specifies the lifecycle rule for the object. */
    auto rule1 = LifecycleRule();
    rule1.setID("rule1");
    rule1.setPrefix("test1/");
    rule1.setStatus(RuleStatus::Enabled);
    rule1.setExpiration(3);
    rule1.setTags(tagging.Tags());
    /* Sets the tags as the matching condition of the lifecycle rule. */
    LifecycleRuleList list{rule1};
    request.setLifecycleRules(list);
    auto outcome = client.SetBucketLifecycle(request);
    if (!outcome.isSuccess()) {
        /* Handles exceptions. */
        std::cout << "SetBucketLifecycle fail" <<
            ",code:" << outcome.error().Code() <<
            ",message:" << outcome.error().Message() <<
            ",requestId:" << outcome.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }
    /* Releases resources such as networks. */
    ShutdownSdk();
    return 0;
}
```

View the tags set as the matching condition of a lifecycle rule

Run the following code to view the tags set as the matching condition of a lifecycle rule:

```

#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initializes the Alibaba Cloud OSS account information. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    std::string Endpoint = "yourEndpoint";
    std::string BucketName = "yourBucketName";
    /* Initializes resources such as networks. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    /* Views the lifecycle rule that includes the tags set as the matching condition. */
    auto outcome = client.GetBucketLifecycle(BucketName);
    if (outcome.isSuccess()) {
        std::cout << "GetBucketLifecycle success," << std::endl;
        for (auto const rule : outcome.result().LifecycleRules()) {
            std::cout << "rule:" << rule.ID() << "," << rule.Prefix() << "," << rule.Status
() << "," <<
                "hasExpiration:" << rule.hasExpiration() << "," <<
                "hasTransitionList:" << rule.hasTransitionList() << "," << std::endl;
            auto taglist = rule.Tags();
            for (const auto& tag : taglist)
            {
                std::cout << "GetBucketLifecycle tag success, Key:"
                << tag.Key() << "; Value:" << tag.Value() << std::endl;
            }
        }
    }
    else {
        /* Handles exceptions. */
        std::cout << "GetBucketLifecycle fail" <<
            ",code:" << outcome.error().Code() <<
            ",message:" << outcome.error().Message() <<
            ",requestId:" << outcome.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }
    /* Releases resources such as networks. */
    ShutdownSdk();
    return 0;
}

```

11.11. Data encryption

11.11.1. Server-side encryption

OSS supports server-side encryption for uploaded data. When you upload data, OSS encrypts the data and stores the encrypted data. When you download data, OSS decrypts the data and returns the original data. The returned HTTP request header declares that the data has been encrypted on the server.

OSS provides two methods of server-side encryption:

- Server-side encryption with CMKs stored in KMS (SSE-KMS)

When you upload objects to OSS, you can use a specified CMK ID or a CMK stored in KMS to encrypt data. This method is cost-effective because you do not need to send user data to the KMS server over networks for encryption and decryption.

 **Notice** You are charged for making API calls when you use CMKs to encrypt or decrypt data.

- Server-side encryption with OSS-managed keys (SSE-OSS)

When you upload objects to OSS, the OSS server uses the fully managed 256-bit AES keys to encrypt data. Each object is encrypted with a unique data key. The data key is further encrypted using a master key that is rotated on a regular basis to ensure the security of the data.

 **Notice**

- Only one server-side encryption method can be used for an object at a time.
- If you configure server-side encryption for a bucket, you can still configure an encryption method for individual objects in the bucket when uploading or copying objects. In this case, the encryption method configured for the object takes precedence. For more information, see [Put Object](#).
- For more information about server-side encryption, see [Server-side encryption](#).

Configure server-side encryption for a bucket

You can use the following code to configure the default encryption method for a bucket. The encryption method of the bucket applies to all uploaded objects that do not have the encryption method configured.

```
#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /*Initialize the OSS account information.*/
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    std::string Endpoint = "yourEndpoint";
    std::string BucketName = "yourBucketName";
    /*Initialize network resources.*/
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    SetBucketEncryptionRequest setrequest(BucketName);
    setrequest.setSSEAlgorithm(SSEAlgorithm::KMS);
    undefined /*Configure KMS-based server-side encryption for a bucket.*/
    auto outcome = client.SetBucketEncryption(setrequest);
    if (! outcome.isSuccess()) {
        /* Handle exceptions. */
        std::cout << "SetBucketEncryption fail" <<
            ",code:" << outcome.error().Code() <<
            ",message:" << outcome.error().Message() <<
            ",requestId:" << outcome.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }
    /*Release network resources.*/
    ShutdownSdk();
    return 0;
}
```

Obtain the encryption configurations of a bucket

Use the following code to obtain the encryption configurations of a bucket:

```
#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /*Initialize the OSS account information.*/
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    std::string Endpoint = "yourEndpoint";
    std::string BucketName = "yourBucketName";
    /*Initialize network resources.*/
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    /*Obtain the server-side encryption configurations of a bucket.*/
    GetBucketEncryptionRequest request(BucketName);
    auto outcome = client.GetBucketEncryption(request);
    if (!outcome.isSuccess()) {
        /* Handle exceptions. */
        std::cout << "GetBucketEncryption fail" <<
            ",code:" << outcome.error().Code() <<
            ",message:" << outcome.error().Message() <<
            ",requestId:" << outcome.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }
    /*Release network resources.*/
    ShutdownSdk();
    return 0;
}
```

Delete the encryption configurations for a bucket

Use the following code to delete the encryption configurations for a bucket:


```
#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /*Initialize the OSS account information.*/
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    std::string Endpoint = "yourEndpoint";
    std::string BucketName = "yourBucketName";
    /*Initialize network resources.*/
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    /*Delete the server-side encryption configurations for a bucket.*/
    DeleteBucketEncryptionRequest request(BucketName);
    auto outcome = client.DeleteBucketEncryption(request);
    if (!outcome.IsSuccess()) {
        /* Handle exceptions. */
        std::cout << "DeleteBucketEncryption fail" <<
            ",code:" << outcome.error().Code() <<
            ",message:" << outcome.error().Message() <<
            ",requestId:" << outcome.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }
    /*Release network resources.*/
    ShutdownSdk();
    return 0;
}
```

11.11.2. Client-side encryption

When you perform client-side encryption, objects are encrypted on the local client before they are uploaded to OSS.

Background information

Notice

- Client-side encryption supports multipart upload for objects larger than 5 GB in size. When you use multipart upload to upload an object, you must specify the total size of the object and the size of each part. The size of each part except for the last part must be the same and be a multiple of 16 bytes.
- After you upload objects encrypted on the client, object metadata related to client-side encryption is protected and cannot be modified by calling [CopyObject](#).

Encryption methods

Object metadata related to client-side encryption

Parameter	Description	Required
x-oss-meta-client-side-encryption-key	The encrypted data key, which is a string encrypted by using the CMK and encoded in Base64.	Yes
x-oss-meta-client-side-encryption-start	The initial value which is randomly generated for data encryption. The initial value is a string encrypted by using the CMK and encoded in Base64.	Yes
x-oss-meta-client-side-encryption-cek-alg	The encryption algorithm used to encrypt data.	Yes
x-oss-meta-client-side-encryption-wrap-alg	The encryption algorithm used to encrypt data keys.	Yes
x-oss-meta-client-side-encryption-matdesc	The description of the CMK in JSON format.  Warning We recommend that you configure a description for each CMK and store the mapping relationship between the CMK and its description. CMKs of which the description is not specified cannot be replaced.	No
x-oss-meta-client-side-encryption-unencrypted-content-length	The length of data before encryption. This parameter is not generated if Content-Length is not specified.	No
x-oss-meta-client-side-encryption-unencrypted-content-md5	The MD5 hash of the data before encryption. This parameter is not generated if Content-MD5 is not specified.	No
x-oss-meta-client-side-encryption-data-size	The total size of the data that needs to encrypt for multipart upload when init_multipart is called.	Yes (for multipart upload)

Parameter	Description	Required
x-oss-meta-client-side-encryption-part-size	The size of each part that needs to encrypt for multipart upload when <code>init_multipart</code> is called.  Note The size of each part must be an integral multiple of 16 bytes.	Yes (for multipart upload)

The following section provides complete sample code on how to use RSA-based CMKs managed by yourself to perform client-side encryption in the following scenarios: upload objects from memory or local files, resumable upload, multipart upload, and download objects to local files.

Encrypt objects to upload from memory

The following code provides an example on how to use an RSA-based CMK managed by yourself to encrypt an object to upload from memory:

```

#include <alibabacloud/oss/OssEncryptionClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize the OSS account information.*/
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    std::string Endpoint = "yourEndpoint";
    std::string BucketName = "yourBucketName";
    std::string ObjectName = "yourObjectName";
    /* Specify the CMK and the description. */
    std::string RSAPublicKey = "your rsa public key";
    std::string RSAPrivateKey = "your rsa private key";
    std::map<std::string, std::string> desc;
    desc["comment"] = "your comment";
    /* Initialize network resources.*/
    InitializeSdk();
    ClientConfiguration conf;
    CryptoConfiguration cryptoConf;
    auto materials = std::make_shared<SimpleRSAEncryptionMaterials>(RSAPublicKey, RSAPrivat
eKey, desc);
    OssEncryptionClient client(Endpoint, AccessKeyId, AccessKeySecret, conf, materials, cry
ptoConf);
    std::shared_ptr<std::iostream> content = std::make_shared<std::stringstream>();
    *content << "Thank you for using Aliyun Object Storage Service!" ;
    PutObjectRequest request(BucketName, ObjectName, content);
    /* Upload the object */
    auto outcome = client.PutObject(request);
    if (! outcome.IsSuccess()) {
        /* Handle exceptions.*/
        std::cout << "PutObject fail" <<
            ",code:" << outcome.error().Code() <<
            ",message:" << outcome.error().Message() <<
            ",requestId:" << outcome.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }
    /* Release network resources.*/
    ShutdownSdk();
    return 0;
}

```

Encrypt objects to upload from a local file

The following code provides an example on how to use an RSA-based CMK managed by yourself to encrypt an object to upload from a local file:

```

#include <alibabacloud/oss/OssEncryptionClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize the OSS account information.*/
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    std::string Endpoint = "yourEndpoint";
    std::string BucketName = "yourBucketName";
    std::string ObjectName = "yourObjectName";
    /* Specify the CMK and the description. */
    std::string RSAPublicKey = "your rsa public key";
    std::string RSAPrivateKey = "your rsa private key";
    std::map<std::string, std::string> desc;
    desc["comment"] = "your comment";
    /* Initialize network resources.*/
    InitializeSdk();
    ClientConfiguration conf;
    CryptoConfiguration cryptoConf;
    auto materials = std::make_shared<SimpleRSAEncryptionMaterials>(RSAPublicKey, RSAPrivateKey, desc);
    OssEncryptionClient client(Endpoint, AccessKeyId, AccessKeySecret, conf, materials, cryptoConf);
    /* Upload the object. */
    auto outcome = client.PutObject(BucketName, ObjectName, "yourLocalFilename");
    if (!outcome.isSuccess()) {
        /* Handle exceptions.*/
        std::cout << "PutObject fail" <<
            ",code:" << outcome.error().Code() <<
            ",message:" << outcome.error().Message() <<
            ",requestId:" << outcome.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }
    /* Release network resources.*/
    ShutdownSdk();
    return 0;
}

```

Encrypt objects to upload by using resumable upload

The following code provides an example on how to use an RSA-based CMK managed by yourself to encrypt an object to upload by using resumable upload:

```

#include <alibabacloud/oss/OssEncryptionClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize the OSS account information.*/
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    std::string Endpoint = "yourEndpoint";
    std::string BucketName = "yourBucketName";
    std::string ObjectName = "yourObjectName";
    std::string UploadFilePath = "yourUploadfilePath";
    std::string CheckpointFilePath = "yourCheckpointfilePath";
    /* Specify the CMK and the description. */
    std::string RSAPublicKey = "your rsa public key";
    std::string RSAPrivateKey = "your rsa private key";
    std::map<std::string, std::string> desc;
    desc["comment"] = "your comment";
    /* Initialize network resources.*/
    InitializeSdk();
    ClientConfiguration conf;
    CryptoConfiguration cryptoConf;
    auto materials = std::make_shared<SimpleRSAEncryptionMaterials>(RSAPublicKey, RSAPrivateKey, desc);
    OssEncryptionClient client(Endpoint, AccessKeyId, AccessKeySecret, conf, materials, cryptoConf);
    /* Start resumable upload. */
    UploadObjectRequest request(BucketName, ObjectName, UploadFilePath, CheckpointFilePath);
    ;
    auto outcome = client.ResumableUploadObject(request);
    if (!outcome.IsSuccess()) {
        /* Handle exceptions.*/
        std::cout << "ResumableUploadObject fail" <<
            ",code:" << outcome.error().Code() <<
            ",message:" << outcome.error().Message() <<
            ",requestId:" << outcome.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }
    /*Release network resources.*/
    ShutdownSdk();
    return 0;
}

```

Encrypt objects to upload by using multipart upload

The following code provides an example on how to use an RSA-based CMK managed by yourself to encrypt an object to upload by using multipart upload:

```

#include <alibabacloud/oss/OssEncryptionClient.h>
#include <fstream>
using namespace AlibabaCloud::OSS;
static int64_t getFileSize(const std::string& file)
{
    std::fstream f(file, std::ios::in | std::ios::binary);

```

```

        std::istream f(file, std::ios::in | std::ios::binary);
        f.seekg(0, f.end);
        int64_t size = f.tellg();
        f.close();
        return size;
    }
}

int main(void)
{
    /* Initialize the OSS account information. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    std::string Endpoint = "yourEndpoint";
    std::string BucketName = "yourBucketName";
    std::string ObjectName = "yourObjectName";
    std::string fileToUpload = "yourLocalFilename";
    /* Specify the CMK and the description. */
    std::string RSAPublicKey = "your rsa public key";
    std::string RSAPrivateKey = "your rsa private key";
    std::map<std::string, std::string> desc;
    desc["comment"] = "your comment";
    /*Initialize network resources.*/
    InitializeSdk();
    ClientConfiguration conf;
    CryptoConfiguration cryptoConf;
    auto materials = std::make_shared<SimpleRSAEncryptionMaterials>(RSAPublicKey, RSAPrivateKey, desc);
    OssEncryptionClient client(Endpoint, AccessKeyId, AccessKeySecret, conf, materials, cryptoConf);

    /* Initialize the context for encryption in multipart upload. */
    /* The part size must be an integral multiple of 16 bytes. */
    int64_t partSize = 100 * 1024;
    auto fileSize = getFileSize(fileToUpload);
    MultipartUploadCryptoContext cryptoCtx;
    cryptoCtx.setPartSize(partSize);
    cryptoCtx.setDataSize(fileSize);
    /* Initiate the multipart upload task. */
    InitiateMultipartUploadRequest initUploadRequest(BucketName, ObjectName);
    auto uploadIdResult = client.InitiateMultipartUpload(initUploadRequest, cryptoCtx);
    auto uploadId = uploadIdResult.result().UploadId();
    PartList partETagList;
    int partCount = static_cast<int>(fileSize / partSize);
    /* Calculate the number of parts to upload.*/
    if (fileSize % partSize != 0) {
        partCount++;
    }
    /* Upload each part sequentially. */
    for (int i = 1; i <= partCount; i++) {
        auto skipBytes = partSize * (i - 1);
        auto size = (partSize < fileSize - skipBytes) ? partSize : (fileSize - skipBytes);
        std::shared_ptr<std::istream> content = std::make_shared<std::fstream>(fileToUpload, std::ios::in|std::ios::binary);
        content->seekg(skipBytes, std::ios::beg);
        UploadPartRequest uploadPartRequest(BucketName, ObjectName, content);
        uploadPartRequest.setContentLength(size);
        uploadPartRequest.setUploadId(uploadId);
    }
}

```

```

        uploadPartRequest.setPartNumber(i);
        auto uploadPartOutcome = client.UploadPart(uploadPartRequest, cryptoCtx);
        if (uploadPartOutcome.isSuccess()) {
            Part part(i, uploadPartOutcome.result().ETag());
            partETagList.push_back(part);
        }
        else {
            std::cout << "uploadPart fail" <<
                ",code:" << uploadPartOutcome.error().Code() <<
                ",message:" << uploadPartOutcome.error().Message() <<
                ",requestId:" << uploadPartOutcome.error().RequestId() << std::endl;
        }
    }

    /* Complete the multipart upload. */
    CompleteMultipartUploadRequest request(BucketName, ObjectName);
    request.setUploadId(uploadId);
    request.setPartList(partETagList);
    auto outcome = client.CompleteMultipartUpload(request, cryptoCtx);
    if (!outcome.isSuccess()) {
        /*Handle exceptions.*/
        std::cout << "CompleteMultipartUpload fail" <<
            ",code:" << outcome.error().Code() <<
            ",message:" << outcome.error().Message() <<
            ",requestId:" << outcome.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }

    /* Release network resources.*/
    ShutdownSdk();
    return 0;
}

```

Decrypt objects to download to local files

The following code provides an example on how to use an RSA-based CMK managed by yourself to decrypt an object to download to a local file:

```

#include <alibabacloud/oss/OssEncryptionClient.h>
#include <fstream>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize the OSS account information. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    std::string Endpoint = "yourEndpoint";
    std::string BucketName = "yourBucketName";
    std::string ObjectName = "yourObjectName";
    std::string FileNameToSave = "yourFileName";
    /* Specify the CMK and the description. */
    std::string RSAPublicKey = "your rsa public key";
    std::string RSAPrivateKey = "your rsa private key";
    std::map<std::string, std::string> desc;
    desc["comment"] = "your comment";
}

```



```

desc["comment"] = "your comment";
/*Initialize network resources.*/
InitializeSdk();
ClientConfiguration conf;
CryptoConfiguration cryptoConf;
auto materials = std::make_shared<SimpleRSAEncryptionMaterials>(RSAPublicKey, RSAPrivateKey, desc);
/* To decrypt content that is encrypted by using other CMKs, you must provide the information about the CMKs. */
//std::string RSAPublicKey2 = "your rsa public key";
//std::string RSAPrivateKey2 = "your rsa private key";
//std::map<std::string, std::string> desc2;
//desc2["comment"] = "your comment";
//materials.addEncryptionMaterial(RSAPublicKey2, RSAPrivateKey2, desc2);
OssEncryptionClient client(Endpoint, AccessKeyId, AccessKeySecret, conf, materials, cryptoConf);
/* Download the object to a local file. */
GetObjectRequest request(BucketName, ObjectName);
request.setResponseStreamFactory([=]() {return std::make_shared<std::fstream>(FileNameToSave, std::ios_base::out | std::ios_base::in | std::ios_base::trunc | std::ios_base::binary); });
auto outcome = client.GetObject(request);
if (outcome.IsSuccess()) {
    std::cout << "GetObjectToFile success" << outcome.result().Metadata().ContentLength() << std::endl;
}
else {
    /* Handle exceptions.*/
    std::cout << "GetObjectToFile fail" <<
        ",code:" << outcome.error().Code() <<
        ",message:" << outcome.error().Message() <<
        ",requestId:" << outcome.error().RequestId() << std::endl;
    ShutdownSdk();
    return -1;
}
/* Release network resources.*/
ShutdownSdk();
return 0;
}

```

Decrypt objects to download to local memory

The following code provides an example on how to use an RSA-based CMK managed by yourself to decrypt an object to download to local memory:

```

#include <alibabacloud/oss/OssEncryptionClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize the OSS account information. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    std::string Endpoint = "yourEndpoint";
    std::string BucketName = "yourBucketName";

```

```

std::string ObjectName = "yourObjectName";
std::string RSAPublicKey = "your rsa public key";
std::string RSAPrivateKey = "your rsa private key";
/* Specify the CMK and the description. */
std::string RSAPublicKey = "your rsa public key";
std::string RSAPrivateKey = "your rsa private key";
std::map<std::string, std::string> desc;
desc["comment"] = "your comment";
/* Initialize network resources.*/
InitializeSdk();
ClientConfiguration conf;
CryptoConfiguration cryptoConf;
auto materials = std::make_shared<SimpleRSAEncryptionMaterials>(RSAPublicKey, RSAPrivateKey, desc);
/* To decrypt content that is encrypted by using different CMKs, you must provide the information about the CMKs. */
//std::string RSAPublicKey2 = "your rsa public key";
//std::string RSAPrivateKey2 = "your rsa private key";
//std::map<std::string, std::string> desc2;
//desc2["comment"] = "your comment";
//materials.addEncryptionMaterial(RSAPublicKey2, RSAPrivateKey2, desc2);
OssEncryptionClient client(Endpoint, AccessKeyId, AccessKeySecret, conf, materials, cryptoConf);
/* Download the object to local memory. */
GetObjectRequest request(BucketName, ObjectName);
auto outcome = client.GetObject(request);
if (outcome.IsSuccess()) {
    std::cout << "getObjectToBuffer" << " success, Content-Length:" << outcome.result().Metadata().ContentLength() << std::endl;
    /* Display the downloaded content. */
    std::string content;
    *(outcome.result().Content()) >> content;
    std::cout << "getObjectToBuffer" << "content:" << content << std::endl;
}
else {
    /*Handle exceptions.*/
    std::cout << "getObjectToBuffer fail" <<
    ",code:" << outcome.error().Code() <<
    ",message:" << outcome.error().Message() <<
    ",requestId:" << outcome.error().RequestId() << std::endl;
    ShutdownSdk();
    return -1;
}
/* Release network resources. */
ShutdownSdk();
return 0;
}

```

Decrypt objects to download by range

The following code provides an example on how to use an RSA-based CMK managed by yourself to decrypt an object to download by range:

```

#include <alibabacloud/oss/OssEncryptionClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /*Initialize the OSS account information.*/
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    std::string Endpoint = "yourEndpoint";
    std::string BucketName = "yourBucketName";
    std::string ObjectName = "yourObjectName ";
    /* Specify the CMK and the description. */
    std::string RSAPublicKey = "your rsa public key";
    std::string RSAPrivateKey = "your rsa private key";
    std::map<std::string, std::string> desc;
    desc["comment"] = "your comment";
    /*Initialize network resources.*/
    InitializeSdk();
    ClientConfiguration conf;
    CryptoConfiguration cryptoConf;
    auto materials = std::make_shared<SimpleRSAEncryptionMaterials>(RSAPublicKey, RSAPrivateKey, desc);
    /* To decrypt content that is encrypted by using different CMKs, you must provide the information about the CMKs. */
    //std::string RSAPublicKey2 = "your rsa public key";
    //std::string RSAPrivateKey2 = "your rsa private key";
    //std::map<std::string, std::string> desc2;
    //desc2["comment"] = "your comment";
    //materials.addEncryptionMaterial(RSAPublicKey2, RSAPrivateKey2, desc2);
    OssEncryptionClient client(Endpoint, AccessKeyId, AccessKeySecret, conf, materials, cryptoConf);
    /* Set the download range. */
    GetObjectRequest request(BucketName, ObjectName);
    request.setRange(0, 1);
    auto outcome = client.GetObject(request);
    if (!outcome.IsSuccess()) {
        /* Handle exceptions.*/
        std::cout << "getObject fail" <<
            ",code:" << outcome.error().Code() <<
            ",message:" << outcome.error().Message() <<
            ",requestId:" << outcome.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }
    /* Release network resources.*/
    ShutdownSdk();
    return 0;
}

```

Decrypt objects to download by using resumable download

The following code provides an example on how to use an RSA-based CMK managed by yourself to decrypt an object to download by using resumable download:

```

#include <alibabacloud/oss/OssEncryptionClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /*Initialize the OSS account information.*/
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    std::string Endpoint = "yourEndpoint";
    std::string BucketName = "yourBucketName";
    std::string ObjectName = "yourObjectName";
    std::string DownloadFilePath = "yourDownloadFilePath";
    std::string CheckpointFilePath = "yourCheckpointFilepath";
    /* Specify the CMK and the description. */
    std::string RSAPublicKey = "your rsa public key";
    std::string RSAPrivateKey = "your rsa private key";
    std::map<std::string, std::string> desc;
    desc["comment"] = "your comment";
    /*Initialize network resources.*/
    InitializeSdk();
    ClientConfiguration conf;
    CryptoConfiguration cryptoConf;
    auto materials = std::make_shared<SimpleRSAEncryptionMaterials>(RSAPublicKey, RSAPrivateKey, desc);
    /* To decrypt content that is encrypted by using different CMKs, you must provide the information about the CMKs. */
    //std::string RSAPublicKey2 = "your rsa public key";
    //std::string RSAPrivateKey2 = "your rsa private key";
    //std::map<std::string, std::string> desc2;
    //desc2["comment"] = "your comment";
    //materials.addEncryptionMaterial(RSAPublicKey2, RSAPrivateKey2, desc2);
    OssEncryptionClient client(Endpoint, AccessKeyId, AccessKeySecret, conf, materials, cryptoConf);
    /* Start resumable download. */
    DownloadObjectRequest request(BucketName, ObjectName, DownloadFilePath, CheckpointFilePath);
    auto outcome = client.ResumableDownloadObject(request);
    if (!outcome.IsSuccess()) {
        /* Handle exceptions.*/
        std::cout << "ResumableDownloadObject fail" <<
            ",code:" << outcome.error().Code() <<
            ",message:" << outcome.error().Message() <<
            ",requestId:" << outcome.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }
    /* Release network resources.*/
    ShutdownSdk();
    return 0;
}

```

11.12. Single-connection bandwidth throttling

This topic describes how to add parameters in an object upload or download request to set the limit of upload or download bandwidth. This ensures sufficient bandwidth for other applications.

Configure bandwidth throttling for simple upload and download

The following sample code provides an example on how to configure single-connection bandwidth throttling for simple upload and download:

```

#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize the information about the account that is used to access Object Storage S
ervice (OSS). */
    /* The AccessKey pair of an Alibaba Cloud account has permissions on all API operations
. Using these credentials to access OSS is a high-risk operation. We recommend that you use
a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to t
he RAM console. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    /* Specify the endpoint of the region in which the bucket is located. For example, if t
he bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-han
gzhou.aliyuncs.com. */
    std::string Endpoint = "yourEndpoint";
    /* Specify the bucket name. Example: examplebucket. */
    std::string BucketName = "examplebucket";
    /* Specify the full path of the object. The full path of the object cannot contain the
bucket name. Example: exampledir/exampleobject.txt. */
    std::string ObjectName = "exampledir/exampleobject.txt";
    /* Initialize resources such as networks. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    /* Upload the object. */
    std::shared_ptr<std::iostream> content = std::make_shared<std::fstream>("yourLocalFileN
ame", std::ios::in|std::ios::binary);
    PutObjectRequest putrequest(BucketName, ObjectName, content);
    /* Set the upload bandwidth limit to 100 KB/s. */
    putrequest.setTrafficLimit(819200);
    auto putoutcome = client.PutObject(putrequest);
    /* Download the object to local memory. */
    GetObjectRequest getrequest(BucketName, ObjectName);
    /* Set the download bandwidth limit to 100 KB/s. */
    getrequest.setTrafficLimit(819200);
    auto getoutcome = client.GetObject(getrequest);
    /* Release resources such as networks. */
    ShutdownSdk();
    return 0;
}

```

Configure bandwidth throttling for upload and download that use signed URLs

The following sample code provides an example on how to configure single-connection bandwidth throttling when you use signed URLs to upload or download objects:

```

#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize the information about the account that is used to access OSS. */
    /* The AccessKey pair of an Alibaba Cloud account has permissions on all API operations
     . Using these credentials to access OSS is a high-risk operation. We recommend that you use
     a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to t
     he RAM console. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    /* Specify the endpoint of the region in which the bucket is located. For example, if t
     he bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-han
     gzhou.aliyuncs.com. */
    std::string Endpoint = "yourEndpoint";
    /* Specify the bucket name. Example: examplebucket. */
    std::string BucketName = "examplebucket";
    /* Specify the full path of the object. The full path of the object cannot contain the
     bucket name. Example: exampledir/exampleobject.txt. */
    std::string ObjectName = "exampledir/exampleobject.txt";
    /* Initialize resources such as networks. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    /* Set the validity period of the signed URL to 1,200 seconds. */
    std::time_t t = std::time(nullptr) + 1200;
    /* Generate the signed URL that is used to upload the object. */
    GeneratePresignedUrlRequest putrequest(BucketName, PutobjectUrlName, Http::Put);
    putrequest.setExpires(expires);
    /* Set the upload bandwidth limit to 100 KB/s. */
    putrequest.setTrafficLimit(819200);
    auto genOutcome = client.GeneratePresignedUrl(putrequest);
    std::shared_ptr<std::iostream> content = std::make_shared<std::stringstream>();
    *content << "test cpp sdk";
    /* Use the signed URL to upload the object. */
    auto outcome = client.PutObjectByUrl(genOutcome.result(), content);
    /* Generate the signed URL that is used to download the object. */
    GeneratePresignedUrlRequest getrequest(BucketName, PutobjectUrlName, Http::Get);
    getrequest.setExpires(expires);
    /* Set the download bandwidth limit to 100 KB/s. */
    getrequest.setTrafficLimit(819200);
    genOutcome = client.GeneratePresignedUrl(getrequest);
    /* Use the signed URL to download the object. */
    outcome = client.GetObjectByUrl(genOutcome.result());
    /* Release resources such as networks. */
    ShutdownSdk();
    return 0;
}

```

11.13. Data security

OSS SDK for C++ uses MD5 verification and CRC-64 to ensure data integrity when you upload, download, and copy objects.

MD5 verification

If you configure Content-MD5 in an object upload request, OSS calculates the MD5 hash of the uploaded object. If the calculated MD5 hash is different from the MD5 hash configured in the upload request, `InvalidDigest` is returned. This allows OSS to ensure data integrity for object upload. If `InvalidDigest` is returned, you must upload the object again.

The following code provides an example on how to perform MD5 verification for object upload operations:

```
#include <alibabacloud/oss/OssClient.h>
#include "../src/Utils/Utils.h"
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize the OSS account information. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    std::string Endpoint = "yourEndpoint";
    std::string BucketName = "yourBucketName";
    std::string ObjectName = "yourObjectName";
    /* Initialize network resources. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    std::shared_ptr<std::iostream> content = std::make_shared<std::stringstream>();
    *content << "test cpp sdk";
    PutObjectRequest request(BucketName, ObjectName, content);
    /* Configure MD5 verification. */
    std::string contentMd5 = ComputeContentMD5(*content);
    request.setContentMd5(contentMd5);
    /* Upload the object. */
    auto outcome = client.PutObject(request);
    if (!outcome.isSuccess()) {
        /* Handle exceptions. */
        std::cout << "PutObject fail" <<
            ",code:" << outcome.error().Code() <<
            ",message:" << outcome.error().Message() <<
            ",requestId:" << outcome.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }
    /* Release network resources. */
    ShutdownSdk();
    return 0;
}
```

 **Note** MD5 verification can be used for `putObject`, `getObject`, `appendObject`, `postObject`, and `uploadPart`.

CRC-64

By default, CRC-64 is enabled to ensure data integrity when you upload, download, and copy an object.

Note

- CRC-64 can be used for putObject, getObject, appendObject, and uploadPart. By default, CRC-64 is enabled when you upload an object. If the CRC-64 value calculated on the client is different from the CRC-64 value returned by the OSS server, an error is returned.
- CRC-64 is not supported in range download.
- CRC-64 consumes CPU resources and slows down uploads and downloads.

The following code provides an example on how to perform CRC-64 in object download:

```
#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize the OSS account information. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    std::string Endpoint = "yourEndpoint";
    std::string BucketName = "yourBucketName";
    std::string ObjectName = "yourObjectName";
    /* Initialize network resources. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    /* Download the object to local memory. */
    GetObjectRequest request(BucketName, ObjectName);
    /* By default, CRC-64 is enabled when you use OSS SDK for C++. */
    /* If CRC-64 fails, a Crc64 validation failed error is returned. */
    auto outcome = client.GetObject(request);
    if (outcome.isSuccess()) {
        std::cout << "getObjectToBuffer" << "server crc:" << outcome.result().Metadata().HttpMetadata().at("x-oss-hash-crc64ecma-by-client") << std::endl;
    }
    else {
        /* Handle exceptions. */
        std::cout << "getObjectToBuffer fail" <<
            ",code:" << outcome.error().Code() <<
            ",message:" << outcome.error().Message() <<
            ",requestId:" << outcome.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }
    /* Release network resources. */
    ShutdownSdk();
    return 0;
}
```

11.14. Authorized access

This topic describes how to authorize temporary access to Object Storage Service (OSS) by using Security Token Service (STS) or a signed URL.

Use STS to authorize temporary access

You can use Alibaba Cloud STS to authorize temporary access to OSS. STS is a web service that provides temporary access tokens for users. You can use STS to grant a set of temporary access credentials that have a custom validity period and custom permissions to a third-party application or a RAM user managed by you. For more information about STS, see [What is STS?](#)

STS provides the following benefits:

- You need only to generate an access token and send the access token to a third-party application, instead of exposing your AccessKey pair to the third-party application. You can define the access permissions and validity period of this token.
- The token automatically expires after the validity period. Therefore, you do not need to manually revoke the access permissions of a token.

 **Note** For more information about how to use STS, see [Use a temporary credential provided by STS to access OSS](#) in OSS User Guide. You can call the [AssumeRole](#) operation or use STS SDKs for various programming languages to obtain temporary access credentials. For more information, see [STS SDK overview](#). The temporary access credentials consist of an AccessKey pair and a security token. The AccessKey pair contains an AccessKey ID and an AccessKey secret. The minimum validity period of temporary access credentials is 900 seconds. The maximum validity period of temporary access credentials is the maximum session duration specified for the current role. For more information, see [Specify the maximum session duration for a RAM role](#).

```
#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize the information about the account that is used to access Object Storage S
ervice (OSS). */
    /* Specify a temporary AccessKey pair obtained from STS. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    /* Specify a security token obtained from STS. */
    std::string SecurityToken = "securityToken";
    /* Specify the endpoint of the region in which the bucket is located. For example, if t
he bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-han
gzhou.aliyuncs.com. */
    std::string Endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
    /* Initialize resources such as networks. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, SecurityToken, conf);
    /* Release resources such as networks. */
    ShutdownSdk();
    return 0;
}
```

Use a signed URL to authorize temporary access

 **Notice** A validity period must be set for an STS temporary access credential and a signed URL. When you use an STS temporary account credential to generate a signed URL that is used to perform operations such as object upload and download, the minimum validity period takes precedence. For example, you can set the validity period of your STS temporary access credential to 1,200 seconds and the validity period of your signed URL generated by using this credential to 3,600 seconds. In this case, the signed URL cannot be used to upload objects after the STS temporary access credential expires, even if the signed URL is within its validity period.

You can generate a signed URL and provide the URL to a visitor for temporary access. When you generate a signed URL, you can specify the validity period of the URL to limit the period of access from visitors. By default, the validity period of a signed URL is 3,600 seconds. The maximum validity period of a signed URL is 32,400 seconds.

You can add signature information to a URL and provide the URL to a third-party user for authorized access. For more information, see [Add signatures to URLs](#).

- Use a signed URL to upload an object

The following code provides an example on how to use a signed URL to upload an object:

```
#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize the information about the account that is used to access OSS. */
    /* Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account
    to access OSS because the account has permissions on all API operations. We recommend tha
    t you use a RAM user to call API operations or perform routine O&M. To create a RAM user,
    log on to the RAM console. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    /* Set yourEndpoint to the endpoint of the region in which the bucket is located. For
    example, if the bucket is located in the China (Hangzhou) region, set the endpoint to htt
    ps://oss-cn-hangzhou.aliyuncs.com. */
    std::string Endpoint = "yourEndpoint";
    /* Specify the bucket name. Example: examplebucket. */
    std::string BucketName = "examplebucket";
    /* Specify the full path of the object. The full path cannot contain the bucket name.
    Example: exampledir/exampleobject.txt. */
    std::string PutobjectUrlName = "exampledir/exampleobject.txt";
    /* Initialize resources such as networks. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    /* Specify the validity period of a signed URL. */
    std::time_t t = std::time(nullptr) + 1200;
    /* Generate a signed URL. */
    auto genOutcome = client.GeneratePresignedUrl(BucketName, PutobjectUrlName, t, Http::
    Put);
    if (genOutcome.isSuccess()) {
        std::cout << "GeneratePresignedUrl success, Gen url:" << genOutcome.result().c_st
        r() << std::endl;
    }
}
```

```

    }
    else {
        /* Handle exceptions. */
        std::cout << "GeneratePresignedUrl fail" <<
            ",code:" << genOutcome.error().Code() <<
            ",message:" << genOutcome.error().Message() <<
            ",requestId:" << genOutcome.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }
    std::shared_ptr<std::iostream> content = std::make_shared<std::stringstream>();
    *content << "test cpp sdk";
    /* Use the signed URL to upload the object. */
    auto outcome = client.PutObjectByUrl(genOutcome.result(), content);
    if (!outcome.isSuccess()) {
        /* Handle exceptions. */
        std::cout << "PutObjectByUrl fail" <<
            ",code:" << outcome.error().Code() <<
            ",message:" << outcome.error().Message() <<
            ",requestId:" << outcome.error().RequestId() << std::endl;
        ShutdownSdk();
        return -1;
    }
    /* Release resources such as networks. */
    ShutdownSdk();
    return 0;
}

```

- Use a signed URL to download an object

The following code provides an example on how to use a signed URL to download an object:

```

#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize the information about the account that is used to access OSS. */
    /* Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account
    to access OSS because the account has permissions on all API operations. We recommend tha
    t you use a RAM user to call API operations or perform routine O&M. To create a RAM user,
    log on to the RAM console. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    /* Set yourEndpoint to the endpoint of the region in which the bucket is located. For
    example, if the bucket is located in the China (Hangzhou) region, set the endpoint to htt
    ps://oss-cn-hangzhou.aliyuncs.com. */
    std::string Endpoint = "yourEndpoint";
    /* Specify the bucket name. Example: examplebucket. */
    std::string BucketName = "examplebucket";
    /* Specify the full path of the object. The full path cannot contain the bucket name.
    Example: exampledir/exampleobject.txt. */
    std::string GetobjectUrlName = "exampledir/exampleobject.txt";
    /* Initialize resources such as networks. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(GetobjectUrlName, AccessKeyId, AccessKeySecret, Endpoint);
}

```

```

OssClient client(endpoint, AccessKeyId, AccessKeySecret, conf);
/* Specify the validity period of a signed URL. */
std::time_t t = std::time(nullptr) + 1200;
/* Generate a signed URL. */
auto genOutcome = client.GeneratePresignedUrl(BucketName, GetobjectUrlName, t, Http::
Get);
if (genOutcome.isSuccess()) {
    std::cout << "GeneratePresignedUrl success, Gen url:" << genOutcome.result().c_str() << std::endl;
}
else {
    /* Handle exceptions. */
    std::cout << "GeneratePresignedUrl fail" <<
    ",code:" << genOutcome.error().Code() <<
    ",message:" << genOutcome.error().Message() <<
    ",requestId:" << genOutcome.error().RequestId() << std::endl;
    ShutdownSdk();
    return -1;
}
/* Use the signed URL to download the object. */
auto outcome = client.GetObjectByUrl(genOutcome.result());
if (!outcome.isSuccess()) {
    /* Handle exceptions. */
    std::cout << "GetObjectByUrl fail" <<
    ",code:" << outcome.error().Code() <<
    ",message:" << outcome.error().Message() <<
    ",requestId:" << outcome.error().RequestId() << std::endl;
    ShutdownSdk();
    return -1;
}
/* Release resources such as networks. */
ShutdownSdk();
return 0;
}

```

11.15. Image processing

Image Processing (IMG) provided by Object Storage Service (OSS) is a secure, cost-effective, and highly reliable image processing service that can process large numbers of images. After you upload source images to OSS, you can call RESTful API operations to process the images on a device that is connected over the Internet anytime, anywhere.

For more information about IMG parameters, see [Parameters](#).

Use IMG parameters to process images

- Use a single IMG parameter to process an image and save the image to your local computer

```
#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize OSS account information. */
    std::string AccessKeyId = "<yourAccessKeyId>";
    std::string AccessKeySecret = "<yourAccessKeySecret>";
    std::string Endpoint = "<yourEndpoint>";
    std::string BucketName = "<yourBucketName>";
    std::string ObjectName = "<yourObjectName>";
    /* Initialize resources such as network resources. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    /* Resize the image to a height and width of 100 pixels and save the image to your local computer. */
    std::string Process = "image/resize,m_fixed,w_100,h_100";
    GetObjectRequest request(BucketName, ObjectName);
    request.setProcess(Process);
    auto outcome = client.GetObject(request);
    /* Release resources such as network resources. */
    ShutdownSdk();
    return 0;
}
```

- Use multiple IMG parameters to process an image at a time and save the image to your local computer

The following code provides an example on how to use multiple IMG parameters to process an image at a time. Separate the parameters with forward slashes (/).

```
#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize OSS account information. */
    /* The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    /* Specify the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com. */
    std::string Endpoint = "yourEndpoint";
    /* Specify the name of the bucket in which the source image is stored. Example: examplebucket. */
    std::string BucketName = "examplebucket";
    /* Specify the name of the source image. If the image is not stored in the root directory of the bucket, you must specify the full path of the image. Example: exampledir/example.jpg. */
    std::string ObjectName = "exampledir/example.jpg";
    /* Initialize resources such as network resources. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    /* Resize the image to a height and width of 100 pixels, rotate the image 90 degrees, and save the image to your local computer. */
    std::string Process = "image/resize,m_fixed,w_100,h_100/rotate,90";
    GetObjectRequest request(BucketName, ObjectName);
    request.setProcess(Process);
    auto outcome = client.GetObject(request);
    /* Release resources such as network resources. */
    ShutdownSdk();
    return 0;
}
```

Use an image style to process an image

You can encapsulate multiple IMG parameters within a style, and then use the style to process an image. For more information, see [Image styles](#). The following code provides an example on how to use an image style to process an image:

```
#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize OSS account information. */
    /* The AccessKey pair of an Alibaba Cloud account has permissions on all API operations
     . Using these credentials to perform operations in OSS is a high-risk operation. We recomme
     nd that you use a RAM user to call API operations or perform routine O&M. To create a RAM u
     ser, log on to the RAM console. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    /* Specify the endpoint of the region in which the bucket is located. For example, if t
     he bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-han
     gzhou.aliyuncs.com. */
    std::string Endpoint = "yourEndpoint";
    /* Specify the name of the bucket in which the source image is stored. Example: example
     bucket. */
    std::string BucketName = "examplebucket";
    /* Specify the name of the source image. If the image is not stored in the root directo
     ry of the bucket, you must specify the full path of the image. Example: exampleDir/example.
     jpg. */
    std::string ObjectName = "exampleDir/example.jpg";
    /* Initialize resources such as network resources. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf) ;
    /* Use the image style to process the image. In this example, replace yourCustomStyleNa
     me with the name of the image style that you create in the OSS console. */
    std::string Process = "style/yourCustomStyleName";
    GetObjectRequest request(BucketName, ObjectName);
    request.setProcess(Process);
    auto outcome = client.GetObject(request);
    /* Release resources such as network resources. */
    ShutdownSdk();
    return 0;
}
```

Save processed images

You can call the `ImgSaveAs` operation to save the processed images to the bucket in which the source images are stored. The following code provides an example on how to save a processed image:


```

#include <alibabacloud/oss/OssClient.h>
#include <sstream>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize OSS account information. */
    /* The AccessKey pair of an Alibaba Cloud account has permissions on all API operations
     . Using these credentials to perform operations in OSS is a high-risk operation. We recomme
     nd that you use a RAM user to call API operations or perform routine O&M. To create a RAM u
     ser, log on to the RAM console. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    /* Specify the endpoint of the region in which the bucket is located. For example, if t
     he bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-han
     gzhou.aliyuncs.com. */
    std::string Endpoint = "yourEndpoint";
    /* Specify the name of the bucket. Example: examplebucket. */
    std::string BucketName = "examplebucket";
    /* Specify the name of the source image. If the image is not stored in the root directo
     ry of the bucket, you must specify the full path of the image. Example: example/example.jpg
     . */
    std::string SourceObjectName = "example/example.jpg";
    /* Specify the name of the processed image. If the image is not stored in the root dire
     ctory of the bucket, you must specify the full path of the image. Example: exampledir/examp
     le.jpg. */
    std::string TargetObjectName = "exampledir/example.jpg";
    /* Initialize resources such as network resources. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    /* Resize the image to a height and width of 100 pixels and save the image to the bucke
     t in which the source image is stored. */
    std::string Process = "image/resize,m_fixed,w_100,h_100";
    std::stringstream ss;
    ss << Process
    << "|sys/saveas"
    << ",o_" << Base64EncodeUrlSafe(TargetObjectName)
    << ",b_" << Base64EncodeUrlSafe(BucketName);
    ProcessObjectRequest request(BucketName, SourceObjectName, ss.str());
    auto outcome = client.ProcessObject(request);
    /* Release resources such as network resources. */
    ShutdownSdk();
    return 0;
}

```

Generate a signed URL that includes IMG parameters for an object

URLs of private objects must be signed. IMG parameters cannot be added to a signed URL. If you want to process a private object, add IMG parameters to the signature. The following code provides an example on how to add IMG parameters to the signature:

```
#include <alibabacloud/oss/OssClient.h>
using namespace AlibabaCloud::OSS;
int main(void)
{
    /* Initialize OSS account information. */
    /* The AccessKey pair of an Alibaba Cloud account has permissions on all API operations
     . Using these credentials to perform operations in OSS is a high-risk operation. We recomme
     nd that you use a RAM user to call API operations or perform routine O&M. To create a RAM u
     ser, log on to the RAM console. */
    std::string AccessKeyId = "yourAccessKeyId";
    std::string AccessKeySecret = "yourAccessKeySecret";
    /* Specify the endpoint of the region in which the bucket is located. For example, if t
     he bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-han
     gzhou.aliyuncs.com. */
    std::string Endpoint = "yourEndpoint";
    /* Specify the name of the bucket in which the source image is located. Example: exampl
     ebucket. */
    std::string BucketName = "examplebucket";
    /* Specify the name of the source image. If the image is not stored in the root directo
     ry of the bucket, you must specify the full path of the image. Example: example/
     example.jpg.
    std::string ObjectName = "exampledir/example.jpg";
    /* Initialize resources such as network resources. */
    InitializeSdk();
    ClientConfiguration conf;
    OssClient client(Endpoint, AccessKeyId, AccessKeySecret, conf);
    /* Generate a signed URL that includes IMG parameters for an object */
    std::string Process = "image/resize,m_fixed,w_100,h_100";
    GeneratePresignedUrlRequest request(BucketName, ObjectName, Http::Get);
    request.setProcess(Process);
    auto outcome = client.GeneratePresignedUrl(request);
    /* Release resources such as network resources. */
    ShutdownSdk();
    return 0;
}
```

12.C

12.1. Preface

This topic is written for OSS SDK for C V3.9.2.

Compatibility

Version compatibility of OSS SDK for C:

- For SDKs V3.X.X: compatible.
- For SDKs V2.X.X:
 - Compatible with Windows
 - Compatible with Linux APIs and incompatible with the following linked list (`aos_list_t`) traversal APIs:
 - `aos_list_for_each_entry`
 - `aos_list_for_each_entry_reverse`
 - `aos_list_for_each_entry_safe`
 - `aos_list_for_each_entry_safe_reverse`
- For SDKs V1.0.0: compatible except the following structures and API operations:
 - `oss_request_options_t`
 - `oss_get_object_to_buffer`
 - `oss_get_object_to_file`
 - `oss_get_object_to_buffer_by_url`
 - `oss_get_object_to_file_by_url`
 - `oss_init_multipart_upload`
 - `oss_complete_multipart_upload`

- For SDKs V0.0.X: incompatible.

Sample code

OSS SDK for C provides a variety of sample code for your reference or use. The following table describes the sample code.

Sample file	Content
<code>oss_put_object_sample</code>	Upload objects
<code>oss_get_object_sample.c</code>	Download objects
<code>oss_append_object_sample.c</code>	Append upload
<code>oss_multipart_upload_sample.c</code>	Multipart upload
<code>oss_resumable_sample.c</code>	Resumable upload and Resumable download

Sample file	Content
oss_head_object_sample.c	Manage object metadata
oss_list_object_sample.c	List objects
oss_delete_object_sample.c	Delete objects
oss_callback_sample.c	Upload callback
oss_progress_sample.c	Upload progress bars and Download progress bars
oss_crc_sample.c	CRC for object upload and download
oss_image_sample.c	Image Processing

 **Note** For more information about source code of OSS SDK for C, visit [GitHub](#).

12.2. Installation

This topic describes how to install OSS C SDK.

Install OSS C SDK in Linux

- Install CMake and third-party libraries

When installing OSS C SDK in Linux, you must install CMake and the following third-party libraries: curl, apr, apr-util and minixml.

The following parameters are required to install the environment :

Parameter	Description	Version requirement
CMake	Build and installation tool	Version 2.6.0 and later
curl	Focuses on network problems.	Version 7.32.0 and later
apr-util	Focuses on memory management and cross-platform operations.	Version 1.5.2 and later
minixml	Parses XML returned by a request.	We recommend you use version 2.9.

Select the operation system that you need to install CMake and third-party libraries.

- Ubuntu/Debian

- Install CMake

```
sudo apt-get install cmake
```

- Install third-party libraries

```
sudo apt-get install libcurl4-openssl-dev libapr1-dev libaprutil1-dev libmxml-dev
```

If the mxml installation package is not included in the yum source, you can install the libraries using RPM. Download the rpm installation package based on your operation system:

- 64-bit: [mxml-2.9-1.x86_64.rpm](#)

- 32-bit: [mxml-2.9-1.i386.rpm](#)

Run the following code to install mxml with the rpm installation package:

```
rpm -ivh mxml-2.9-1.x86_64.rpm
```

- RedHat/Aliyun/CentOS

- Install CMake

```
sudo yum install cmake
```

- Install third-party libraries

```
sudo yum install curl-devel apr-devel apr-util-devel mxml mxml-devel
```

- SuSE

- Install CMake

```
zypper install cmake
```

- Install third-party libraries

```
zypper install libcurl-devel libapr1-devel libapr-util-devel mxml-devel
```

- Other Linux

- Install CMake: [Download link](#)

The common installation method is as follows:

```
./configure
make
make install
```

 **Note** When running `./configure`, the default installation path is `/usr/local/`. If you want to specify another path, use the `./configure --prefix` option.

- Install libcurl: [Download link](#)

The common installation method is as follows:

```
./configure
make
make install
```

- Install apr: [Download link](#)

The common installation method is as follows:

```
./configure
make
make install
```

- Install apr-util: [Download link](#)

The common installation method is as follows:

```
// You must specify the --with-apr option in the installation.
./configure --with-apr=/your/apr/install/path
make
make install
```

- Install minixml: [Download link](#)

 **Note** We recommend you use minixml version 2.xx.

The common installation method is as follows:

```
./configure
make
sudo make install
```

- Download SDK

- [Download SDK directly](#)

- [Download SDK from GitHub](#)
- [Download historical SDK](#)

- Install SDK

Use cmake. If third-party libraries, such as url, apr, apr-util and mxml are not installed in the default method during the compiling process, you must specify the path of their header files and library files.

Install the SDK as follows:

```
cmake .
make
make install
```

An installation example in Cent OS is as follows:

```
cmake -f CMakeLists.txt
// The compilation type is Release. Common compilation types include: Debug, Release, Rel
WithDebInfo, and MinSizeRel, in which Debug is the default type.
-DCMAKE_BUILD_TYPE=Release
// Customize the installation directory.
-DCMAKE_INSTALL_PREFIX=/usr/local/
// Specify the path where the header files and library files of third party libraries, suc
h as curl, apr, apr-util, and xml, are located,
-DCURL_INCLUDE_DIR=/usr/include/curl
-DCURL_LIBRARY=/usr/lib64/libcurl.so
-DAPR_INCLUDE_DIR=/usr/include/apr-1
-DAPR_LIBRARY=/usr/lib64/libapr-1.so
-DAPR_UTIL_INCLUDE_DIR=/usr/include/apr-1
-DAPR_UTIL_LIBRARY=/usr/lib64/libaprutil-1.so
-DMINIXML_INCLUDE_DIR=/usr/include
-DMINIXML_LIBRARY=/usr/lib64/libmxml.so
// If the "Could not find apr-config/apr-1-config" error occurs during the compiling proc
ess, it indicates that the apr-1-config file cannot be found in the default path. Add thi
s option.
-DAPR_CONFIG_BIN=/path/to/bin/apr-1-config
// If the "Could not find apu-config/apu-1-config" error occurs, add this option.
-DAPU_CONFIG_BIN=/path/to/bin/apu-1-config
```

Install OSS C SDK in Windows

- Download SDK
 - [Download SDK directly](#)
 - [Download SDK from GitHub](#)
 - [Download historical SDK](#)

- Install SDK

For detailed steps and common problems when using Visual Studio to compile OSS C SDK, see [Compile and Use Alibaba Cloud OSS C SDK in Windows](#).

 **Note** If you use Visual Studio 2012 and later versions, it prompts you to upgrade the project to the latest compiler and libraries. We recommended you to determine whether to upgrade based on your project. You can upgrade if your project uses the latest compiler and libraries.

12.3. Initialization

To use OSS SDK for C, you must initialize request options (`oss_request_options_t`) and specify an endpoint.

For more information about endpoints, see [Regions and endpoints](#) and [Bind custom domain names](#).

Use an OSS endpoint to initialize request options

The following code provides an example on how to initialize request options by using the OSS domain name:


```
#include "oss_api.h"
#include "aos_http_io.h"
const char *endpoint = "<yourEndpoint>";
const char *access_key_id = "<yourAccessKeyId>";
const char *access_key_secret = "<yourAccessKeySecret>";
void init_options(oss_request_options_t *options)
{
    options->config = oss_config_create(options->pool);
    /* Use a char* string to initialize the aos_string_t type. */
    aos_str_set(&options->config->endpoint, endpoint);
    aos_str_set(&options->config->access_key_id, access_key_id);
    aos_str_set(&options->config->access_key_secret, access_key_secret);
    /* Specify whether to use CNAME to access OSS. A value of 0 indicates that CNAME is not
    used. */
    options->config->is_cname = 0;
    /* Specify network parameters. The second parameter specifies the ownership of ctl. The
    default value is 0. */
    options->ctl = aos_http_controller_create(options->pool, 0);
}
int main() {
    aos_pool_t *p;
    oss_request_options_t *options;
    /* Initialize global variables. */
    if (aos_http_io_initialize(NULL, 0) != AOSE_OK) {
        return -1;
    }
    /* Initialize the memory pool and options. */
    aos_pool_create(&p, NULL);
    options = oss_request_options_create(p);
    init_options(options);
    /* Business logic code, which is not listed here. */
    /* Release the memory pool, which releases memory resources allocated for the request.
    */
    aos_pool_destroy(p);
    /* Release the allocated global resources. */
    aos_http_io_deinitialize();
    return 0;
}
```

Use a custom domain name to initialize request options

The following code provides an example on how to initialize request options by using a custom domain name:

```
#include "oss_api.h"
#include "aos_http_io.h"
const char *endpoint = "<yourCustomEndpoint>";
const char *access_key_id = "<yourAccessKeyId>";
const char *access_key_secret = "<yourAccessKeySecret>";
void init_options(oss_request_options_t *options) {
    options->config = oss_config_create(options->pool);
    /* Use a char* string to initialize the aos_string_t type. */
    aos_str_set(&options->config->endpoint, endpoint);
    aos_str_set(&options->config->access_key_id, access_key_id);
    aos_str_set(&options->config->access_key_secret, access_key_secret);
    /* Enable CNAME and bind the custom domain name to the bucket. */
    options->config->is_cname = 1;
    options->ctl = aos_http_controller_create(options->pool, 0);
}
int main() {
    aos_pool_t *p;
    oss_request_options_t *options;
    /* Initialize global variables. */
    if (aos_http_io_initialize(NULL, 0) != AOSE_OK) {
        return -1;
    }
    /* Initialize the memory pool and options. */
    aos_pool_create(&p, NULL);
    options = oss_request_options_create(p);
    init_options(options);
    /* Business logic code, which is not listed here. */
    /* Release the memory pool, which releases memory resources allocated for the request. */
    /*
    aos_pool_destroy(p);
    /* Release the allocated global resources. */
    aos_http_io_deinitialize();
    return 0;
    */
}
```

Use STS to initialize request options

The following code provides an example on how to initialize request options by using STS:

```
#include "oss_api.h"
#include "aos_http_io.h"
const char *endpoint = "<yourEndpoint>";
const char *access_key_id = "<yourAccessKeyId>";
const char *access_key_secret = "<yourAccessKeySecret>";
const char *sts_token = "<yourSecurityToken>";
void init_options(oss_request_options_t *options)
{
    options->config = oss_config_create(options->pool);
    /* Use a char* string to initialize the aos_string_t type. */
    aos_str_set(&options->config->endpoint, endpoint);
    aos_str_set(&options->config->access_key_id, access_key_id);
    aos_str_set(&options->config->access_key_secret, access_key_secret);
    /* Configure STS. */
    aos_str_set(&options->config->sts_token, sts_token);
    /* Specify whether to use CNAME to access OSS. A value of 0 indicates that CNAME is not
    used. */
    options->config->is_cname = 0;
    options->ctl = aos_http_controller_create(options->pool, 0);
}
int main() {
    aos_pool_t *p;
    oss_request_options_t *options;
    /* Initialize global variables. */
    if (aos_http_io_initialize(NULL, 0) != AOSE_OK) {
        return -1;
    }
    /* Initialize the memory pool and options. */
    aos_pool_create(&p, NULL);
    options = oss_request_options_create(p);
    init_options(options);
    /* Business logic code, which is not listed here. */
    /* Release the memory pool, which releases memory resources allocated for the request.
    */
    aos_pool_destroy(p);
    /* Release the allocated global resources. */
    aos_http_io_deinitialize();
    return 0;
}
```

Configure a timeout period

The following code provides an example on how to configure a timeout period:

```

#include "oss_api.h"
#include "aos_http_io.h"
const char *endpoint = "<yourEndpoint>";
const char *access_key_id = "<yourAccessKeyId>";
const char *access_key_secret = "<yourAccessKeySecret>";
void init_options(oss_request_options_t *options)
{
    options->config = oss_config_create(options->pool);
    /* Use a char* string to initialize the aos_string_t type. */
    aos_str_set(&options->config->endpoint, endpoint);
    aos_str_set(&options->config->access_key_id, access_key_id);
    aos_str_set(&options->config->access_key_secret, access_key_secret);
    /* Specify whether to use CNAME to access OSS. A value of 0 indicates that CNAME is not
    used. */
    options->config->is_cname = 0;
    /* Specify network parameters. The second parameter specifies the ownership of ctl. The
    default value is 0. */
    options->ctl = aos_http_controller_create(options->pool, 0);
    /* Set the value of the connection timeout , which is 10 seconds by default. */
    options->ctl->options->connect_timeout = 10;
    /* Set the value of the DNS timeout , which is 60 seconds by default. */
    options->ctl->options->dns_cache_timeout = 60;
    /*
    Set the value of the request timeout:
    Set the value of speed_limit to specify the minimum acceptable speed. The default value
    is 1024, which indicates 1 KB/s.
    Set the value of speed_time to specify the maximum time period during which the minimum
    speed is acceptable, which is 15 seconds by default.
    The following code indicates that if the transmission speed is lower than 1 KB/s for 15
    consecutive seconds, the request times out.
    */
    options->ctl->options->speed_limit = 1024;
    options->ctl->options->speed_time = 15;
}

```

Configure SSL-based authentication

By default, SSL-based authentication is enabled for OSS SDK for C V3.9.2 or later. If SSL-based authentication fails, you need to set the correct path of the SSL certificate, or disable SSL-based authentication.

The following code provides an example on how to configure SSL-based authentication:

```
#include "oss_api.h"
#include "aos_http_io.h"
const char *endpoint = "<yourEndpoint>";
const char *access_key_id = "<yourAccessKeyId>";
const char *access_key_secret = "<yourAccessKeySecret>";
void init_options(oss_request_options_t *options)
{
    options->config = oss_config_create(options->pool);
    /* Use a char* string to initialize the aos_string_t type. */
    aos_str_set(&options->config->endpoint, endpoint);
    aos_str_set(&options->config->access_key_id, access_key_id);
    aos_str_set(&options->config->access_key_secret, access_key_secret);
    /* Specify whether to use CNAME. A value of 0 indicates that CNAME is not used. */
    options->config->is_cname = 0;
    /* Specify network parameters. The second parameter specifies the ownership of ctl. The
    default value is 0. */
    options->ctl = aos_http_controller_create(options->pool, 0);
    /* Configure SSL-based authentication.
    Set verify_ssl to specify whether to enable SSL-based authentication. Valid values: 0 and 1. Default value: 1, which enables SSL-based authentication.
    Set ca_path to specify the root path of the CA certificate. When verify_ssl is set to 1, this parameter is valid. This parameter is empty by default.
    Set ca_file to specify the path of the CA certificate. When verify_ssl is set to 1, this parameter is valid. This parameter is empty by default. */
    // Enable SSL-based authentication. Set the path of the CA certificate.
    //options->ctl->options->verify_ssl = 1;
    //options->ctl->options->ca_path = "/etc/ssl/certs/";
    //options->ctl->options->ca_file = "/etc/ssl/certs/ca-certificates.crt";
    // Disable SSL-based authentication.
    //options->ctl->options->verify_ssl = 0;
}
```

12.4. Quick start

This topic describes how to use OSS C SDK to complete routine operations such as bucket creation, objects uploads, and object downloads.

Demo

- Linux demo

Linux-based demo: [aliyun-oss-c-sdk-demo.tar.gz](#).

- Download and extract the demo project.
- You must specify the installation directory for oss-c-sdk-demo-specified-installation: `/home/your/oss/csdk`. You do not need to specify installation directories for other demo projects because they are automatically installed based on the OSS C SDK and other dependent third-party libraries.
- Replace the `OSS_ENDPOINT`, `ACCESS_KEY_ID`, `ACCESS_KEY_SECRET`, and `BUCKET_NAME` in the demo with valid values. Use `LD_LIBRARY_PATH` to specify the directory of the OSS C SDK and the dynamic libraries of the dependent libraries if they are not in the system directory.

- iv. Enter the directory of the project (oss-c-sdk-demo-xxx), run the `make` command to compile the demo project. Then, run the `./main` command to run the executable program. If the project needs to be re-compiled, run the `make clean` command first before compiling.

- Windows demo

Windows-based demo: [aliyun-oss-c-sdk-sample.zip](#). You can download more demo projects from [GitHub](#).

- i. Download and extract the demo project.
- ii. Use Visual Studio to open the demo project, and then replace the following parameters in the `os_s_config.c` with valid values: `OSS_ENDPOINT`, `ACCESS_KEY_ID`, `ACCESS_KEY_SECRET`, `BUCKET_NAME`, `OBJECT_NAME`, `MULTIPART_UPLOAD_FILE_PATH`, and `DIR_NAME`. Compile and run the project. You can develop your own project based on the demo project.

Create a bucket

A bucket is a global namespace on OSS used to store objects. It is equivalent to a data container. You can run the following code to create a bucket:

```
#include "oss_api.h"
#include "aos_http_io.h"
const char *endpoint = "<yourEndpoint>";
const char *access_key_id = "<yourAccessKeyId>";
const char *access_key_secret = "<yourAccessKeySecret>";
const char *bucket_name = "<yourBucketName>";
void init_options(oss_request_options_t *options)
{
    options->config = oss_config_create(options->pool);
    /* Use a char* string to initialize aos_string_t. */
    aos_str_set(&options->config->endpoint, endpoint);
    aos_str_set(&options->config->access_key_id, access_key_id);
    aos_str_set(&options->config->access_key_secret, access_key_secret);
    /* Determine whether the CNAME is used. 0 indicates the CNAME is not used. */
    options->config->is_cname = 0;
    /* Used to configure network parameters, such as timeout */
    options->ctl = aos_http_controller_create(options->pool, 0);
}
int main(int argc, char *argv[])
{
    /* Call the aos_http_io_initialize method in main() to initialize global resources, such as the network and memory. */
    if (aos_http_io_initialize(NULL, 0) != AOSE_OK) {
        exit(1);
    }
    /* Initialize a memory pool used to manage memories, which is equivalent to apr_pool_t. The implementation code is included in the apr library. */
    aos_pool_t *pool;
    /* Re-create a new memory pool. The second parameter is NULL, indicating that it does not inherit from any other memory pools. */
    aos_pool_create(&pool, NULL);
    /* Create and initialize options. This parameter mainly includes global configuration information, such as endpoint, access_key_id, access_key_secret, is_cname, and curl. */
    oss_request_options_t *oss_client_options;
    /* Allocate memories in the memory pool to options. */
```

```

oss_client_options = oss_request_options_create(pool);
/* Use oss_client_options to initialize client options */
init_options(oss_client_options);
/* Create the bucket. */
aos_status_t *resp_status;
aos_table_t *resp_headers;
oss_acl_e oss_acl = OSS_ACL_PRIVATE;
aos_string_t bucket;
/* Assign a char* to the bucket. */
aos_str_set(&bucket, bucket_name);
resp_status = oss_create_bucket(oss_client_options, &bucket, oss_acl, &resp_headers);
/* Identify whether the request succeeds */
if (aos_status_is_ok(resp_status)) {
    printf("create bucket succeeded\n");
} else {
    printf("create bucket failed\n");
}
/* Release the memory pool, that is, memories allocated to resources during the request
. */
aos_pool_destroy(pool);
/* Release allocated global resources. */
aos_http_io_deinitialize();
return 0;
}

```

For more information about endpoints, see [Regions and endpoints](#).

Upload an object

You can run the following code to upload an object to OSS:

```

#include "oss_api.h"
#include "aos_http_io.h"
const char *endpoint = "<yourEndpoint>";
const char *access_key_id = "<yourAccessKeyId>";
const char *access_key_secret = "<yourAccessKeySecret>";
const char *bucket_name = "<yourBucketName>";
const char *object_name = "<yourObjectName>";
void init_options(oss_request_options_t *options)
{
    options->config = oss_config_create(options->pool);
    /* Use a char* string to initialize aos_string_t. */
    aos_str_set(&options->config->endpoint, endpoint);
    aos_str_set(&options->config->access_key_id, access_key_id);
    aos_str_set(&options->config->access_key_secret, access_key_secret);
    /* Determine whether the CNAME is used. 0 indicates the CNAME is not used. */
    options->config->is_cname = 0;
    /* Used to configure network parameters, such as timeout */
    options->ctl = aos_http_controller_create(options->pool, 0);
}
int main(int argc, char *argv[])
{
    /* Call the aos_http_io_initialize method in main() to initialize global resources, suc
h as networks and memories. */

```

```

    if (aos_http_io_initialize(NULL, 0) != AOSE_OK) { = AOSE_OK) {
        exit(1);
    }
    /* Initialize a memory pool used to manage memories, which is equivalent to apr_pool_t.
    The implementation code is included in the apr library. */
    aos_pool_t *pool;
    /* Re-create a new memory pool. The second parameter is NULL, indicating that it does not
    inherit from any other memory pool. */
    aos_pool_create(&pool, NULL);
    /* Create and initialize options. This parameter mainly includes global configuration information,
    such as endpoint, access_key_id, access_key_secret, is_cname, and curl. */
    oss_request_options_t *oss_client_options;
    /* Allocate memory in the memory pool to options. */
    oss_client_options = oss_request_options_create(pool);
    /* Use oss_client_options to initialize client options */
    init_options(oss_client_options);
    /* Initialization parameters */
    aos_string_t bucket;
    aos_string_t object;
    aos_status_t *resp_status;
    aos_table_t *headers;
    aos_table_t *resp_headers;
    aos_list_t buffer;
    aos_buf_t *content = NULL;
    const char *data = "More than just cloud.";
    aos_str_set(&bucket, bucket_name);
    aos_str_set(&object, object_name);
    headers = aos_table_make(pool, 0);
    /* Convert the char* type data to the aos_list_t type data. */
    aos_list_init(&buffer);
    content = aos_buf_pack(oss_client_options->pool, data, strlen(data));
    aos_list_add_tail(&content->node, &buffer);
    /* Upload an object. */
    resp_status = oss_put_object_from_buffer(oss_client_options, &bucket, &object, &buffer,
    headers, &resp_headers);
    /* Determine whether the upload request is successful. */
    if (aos_status_is_ok(resp_status)) {
        printf("put file succeeded\n");
    } else {
        printf("put file failed\n");
    }
    /* Release the memory pool, that is, memories allocated to resources during the request
    . */
    aos_pool_destroy(pool);
    /* Release allocated global resources. */
    aos_http_io_deinitialize();
    return 0;
}

```

Download an object

You can run the following code to download a specified object:

```
#include "oss_api.h"
```



```

#include "aos_http_io.h"
const char *endpoint = "<yourEndpoint>";
const char *access_key_id = "<yourAccessKeyId>";
const char *access_key_secret = "<yourAccessKeySecret>";
const char *bucket_name = "<yourBucketName>";
const char *object_name = "<yourObjectName>";
const char *download_filename = "<yourDownloadedFilename>";
void init_options(oss_request_options_t *options)
{
    options->config = oss_config_create(options->pool);
    /* Use a char* string to initialize aos_string_t. */
    aos_str_set(&options->config->endpoint, endpoint);
    aos_str_set(&options->config->access_key_id, access_key_id);
    aos_str_set(&options->config->access_key_secret, access_key_secret);
    /* Determine whether the CNAME is used. 0 indicates the CNAME is not used. */
    options->config->is_cname = 0;
    /* Used to configure network parameters, such as timeout */
    options->ctl = aos_http_controller_create(options->pool, 0);
}
int main(int argc, char *argv[])
{
    /* Call the aos_http_io_initialize method in main() to initialize global resources, such as networks and memories. */
    if (aos_http_io_initialize(NULL, 0) != AOSE_OK) {
        exit(1);
    }
    /* Initialize a memory pool used to manage memories, which is equivalent to apr_pool_t. The implementation code is included in the apr library. */
    aos_pool_t *pool;
    /* Re-create a new memory pool. The second parameter is NULL, indicating that it does not inherit from any other memory pools. */
    aos_pool_create(&pool, NULL);
    /* Create and initialize options. This parameter mainly includes global configuration information, such as endpoint, access_key_id, access_key_secret, is_cname, and curl. */
    oss_request_options_t *oss_client_options;
    /* Allocate memories in the memory pool to options. */
    oss_client_options = oss_request_options_create(pool);
    /* Use oss_client_options to initialize client options */
    init_options(oss_client_options);
    /* Initialization parameters */
    aos_string_t bucket;
    aos_string_t object;
    aos_string_t file;
    aos_status_t *resp_status;
    aos_table_t *headers = NULL;
    aos_table_t *params = NULL;
    aos_table_t *resp_headers = NULL;
    aos_str_set(&bucket, bucket_name);
    aos_str_set(&object, object_name);
    aos_str_set(&file, download_filename);
    headers = aos_table_make(pool, 0);
    /* Download an object. */
    resp_status = oss_get_object_to_file(oss_client_options, &bucket, &object, headers, params, &file, &resp_headers);
}

```

```

/* Determine whether the download request is successful. */
if (aos_status_is_ok(resp_status)) {
    printf("get object to local file succeeded\n");
} else {
    printf("get object to local file failed\n");
}
/* Release the memory pool, that is, memories allocated to resources during the request
. */
aos_pool_destroy(pool);
/* Release allocated global resources. */
aos_http_io_deinitialize();
return 0;
}

```

List objects in a bucket

You can run the following code to list objects stored in a specified bucket.

```

#include "oss_api.h"
#include "aos_http_io.h"
const char *endpoint = "<yourEndpoint>";
const char *access_key_id = "<yourAccessKeyId>";
const char *access_key_secret = "<yourAccessKeySecret>";
const char *bucket_name = "<yourBucketName>";
void init_options(oss_request_options_t *options)
{
    options->config = oss_config_create(options->pool);
    /* Use a char* string to initialize aos_string_t. */
    aos_str_set(&options->config->endpoint, endpoint);
    aos_str_set(&options->config->access_key_id, access_key_id);
    aos_str_set(&options->config->access_key_secret, access_key_secret);
    /* Determine whether the CNAME is used. 0 indicates the CNAME is not used. */
    options->config->is_cname = 0;
    /* Used to configure network parameters, such as timeout */
    options->ctl = aos_http_controller_create(options->pool, 0);
}
int main(int argc, char *argv[])
{
    /* Call the aos_http_io_initialize method in main() to initialize global resources, suc
h as networks and memories. */
    if (aos_http_io_initialize(NULL, 0) != AOSE_OK) {
        exit(1);
    }
    /* Initialize a memory pool used to manage memories, which is equivalent to apr_pool_t.
The implementation code is included in the apr library. */
    aos_pool_t *pool = NULL;
    /* Re-create a new memory pool. The second parameter is NULL, indicating that it does n
ot inherit from any other memory pool. */
    aos_pool_create(&pool, NULL);
    /* Create and initialize options. This parameter mainly includes global configuration i
nformation, such as endpoint, access_key_id, acces_key_secret, is_cname, and curl. */
    oss_request_options_t *oss_client_options = NULL;
    /* Allocate memory in the memory pool to options. */
    oss_client_options = oss_request_options_create(pool);
}

```

```

/* Use oss_client_options to initialize client options */
init_options(oss_client_options);
/* Initialization parameters */
aos_string_t bucket;
aos_status_t *resp_status = NULL;
oss_list_object_params_t *params = NULL;
oss_list_object_content_t *content = NULL;
int size = 0;
char *line = NULL;
aos_str_set(&bucket, bucket_name);
params = oss_create_list_object_params(pool);
/* List objects */
resp_status = oss_list_object(oss_client_options, &bucket, params, NULL);
/* Determine whether the list request is successful. */
if (!aos_status_is_ok(resp_status))
{
    printf("list object failed\n");
}
else{
    /* Print objects. */
    printf("Object\tSize\tLastModified\n");
    aos_list_for_each_entry(oss_list_object_content_t, content, &params->object_list, n
ode) {
        ++size;
        line = apr_psprintf(pool, "%.s\t%.s\t%.s\n", content->key.len, content->key.
data,
            content->size.len, content->size.data,
            content->last_modified.len, content->last_modified.data);
        printf("%s", line);
    }
    printf("Total %d\n", size);
}
/* Release the memory pool, that is, memories allocated to resources during the request
. */
aos_pool_destroy(pool);
/* Release allocated global resources. */
aos_http_io_deinitialize();
return 0;
}

```

Delete an object.

You can run the following code to delete an object:

```

#include "oss_api.h"
#include "aos_http_io.h"
const char *endpoint = "<yourEndpoint>";
const char *access_key_id = "<yourAccessKeyId>";
const char *access_key_secret = "<yourAccessKeySecret>";
const char *bucket_name = "<yourBucketName>";
const char *object_name = "<yourObjectName>";
void init_options(oss_request_options_t *options)
{
    options->config = oss_config_create(options->pool);
}

```

```

/* Use a char* string to initialize aos_string_t. */
aos_str_set(&options->config->endpoint, endpoint);
aos_str_set(&options->config->access_key_id, access_key_id);
aos_str_set(&options->config->access_key_secret, access_key_secret);
/* Determine whether the CNAME is used. 0 indicates the CNAME is not used. */
options->config->is_cname = 0;
/* Used to configure network parameters, such as timeout */
options->ctl = aos_http_controller_create(options->pool, 0);
}

int main(int argc, char *argv[])
{
    /* Call the aos_http_io_initialize method in main() to initialize global resources, such as networks and memories. */
    if (aos_http_io_initialize(NULL, 0) != AOSE_OK) {
        exit(1);
    }
    /* Initialize a memory pool used to manage memories, which is equivalent to apr_pool_t. The implementation code is included in the apr library. */
    aos_pool_t *pool = NULL;
    /* Re-create a new memory pool. The second parameter is NULL, indicating that it does not inherit from any other memory pool. */
    aos_pool_create(&pool, NULL);
    /* Create and initialize options. This parameter mainly includes global configuration information, such as endpoint, access_key_id, access_key_secret, is_cname, and curl. */
    oss_request_options_t *oss_client_options = NULL;
    /* Allocate memory in the memory pool to options. */
    oss_client_options = oss_request_options_create(pool);
    /* Use oss_client_options to initialize client options */
    init_options(oss_client_options);
    /* Initialization parameters */
    aos_string_t bucket;
    aos_string_t object;
    aos_status_t *resp_status = NULL;
    aos_table_t *resp_headers = NULL;
    aos_str_set(&bucket, bucket_name);
    aos_str_set(&object, object_name);
    /* Delete an object. */
    resp_status = oss_delete_object(oss_client_options, &bucket, &object, &resp_headers);
    /* Determine whether the delete request is successful. */
    if (aos_status_is_ok(resp_status)) {
        printf("delete object succeeded\n");
    } else {
        printf("delete object failed\n");
    }
    /* Release the memory pool, that is, memories allocated to resources during the request. */
    aos_pool_destroy(pool);
    /* Release allocated global resources. */
    aos_http_io_deinitialize();
    return 0;
}

```

12.5. Buckets

12.5.1. Create buckets

A bucket is a container used to store objects in Object Storage Service (OSS). Each object is contained in a bucket. This topic describes how to create a bucket.

```
#include "oss_api.h"
#include "aos_http_io.h"
const char *endpoint = "<yourEndpoint>";
const char *access_key_id = "<yourAccessKeyId>";
const char *access_key_secret = "<yourAccessKeySecret>";
const char *bucket_name = "<yourBucketName>";
void init_options(oss_request_options_t *options)
{
    options->config = oss_config_create(options->pool);
    /* Use a char* string to initialize the aos_string_t data type. */
    aos_str_set(&options->config->endpoint, endpoint);
    aos_str_set(&options->config->access_key_id, access_key_id);
    aos_str_set(&options->config->access_key_secret, access_key_secret);
    /* Specify whether to use CNAME. 0 indicates that the CNAME is not used. */
    options->config->is_cname = 0;
    /* Configure network parameters, such as timeout. */
    options->ctl = aos_http_controller_create(options->pool, 0);
}
int main(int argc, char *argv[])
{
    /* Call aos_http_io_initialize in main() to initialize global resources such as the net
work and memory. */
    if (aos_http_io_initialize(NULL, 0) != AOSE_OK) {
        exit(1);
    }
    /* Create a memory pool to manage memory. The implementation code is included in the AP
R library. aos_pool_t is equivalent to apr_pool_t. */
    aos_pool_t *pool;
    /* Create a new memory pool. The second parameter is NULL. This value indicates that th
e pool does not inherit any other memory pool. */
    aos_pool_create(&pool, NULL);
    /* Create and initialize options. This parameter includes global configuration informat
ion such as endpoint, access_key_id, access_key_secret, is_cname, and curl. */
    oss_request_options_t *oss_client_options;
    /* Allocate a memory chunk in the memory pool to options. */
    oss_client_options = oss_request_options_create(pool);
    /* Initialize oss_client_options. */
    init_options(oss_client_options);
    /* Initialize parameters. */
    aos_string_t bucket;
    oss_acl_t oss_acl = OSS_ACL_PRIVATE;
    aos_table_t *resp_headers = NULL;
    aos_status_t *resp_status = NULL;
    /* Assign char* data to the aos_string_t bucket. */
    aos_str_set(&bucket, bucket_name);
    // Create the bucket.
}
```

```

resp_status = oss_create_bucket(oss_client_options, &bucket, oss_acl, &resp_headers);
/* Determine whether the bucket is created. */
if (aos_status_is_ok(resp_status)) {
    printf("create bucket succeeded\n");
} else {
    printf("create bucket failed\n");
}
/* Release the memory pool. The memory allocated to various resources used for the request is released. */
aos_pool_destroy(pool);
/* Release the allocated global resources. */
aos_http_io_deinitialize();
return 0;
}

```

The following code provides an example on how to set the ACL of a bucket when creating the bucket:

```

/* Create a bucket and set the ACL of the bucket to public read (the default ACL is private). */
resp_status = oss_create_bucket(oss_client_options, &bucket, OSS_ACL_PUBLIC_READ, &resp_headers);
if (aos_status_is_ok(resp_status)) {
    printf("create bucket succeeded\n");
} else {
    printf("create bucket failed\n");
}

```

The following code provides an example on how to create a bucket of the Infrequent Access (IA) storage class.

To create a bucket of the Archive storage class, replace `OSS_STORAGE_CLASS_IA` in the following example with `OSS_STORAGE_CLASS_ARCHIVE`.

```

resp_status = oss_create_bucket_with_storage_class(oss_client_options, &bucket, OSS_ACL_PRIVATE, OSS_STORAGE_CLASS_IA);
/* Determine whether the bucket of the specified storage class is created. */
if (aos_status_is_ok(resp_status)) {
    printf("create bucket succeeded\n");
} else {
    printf("create bucket failed\n");
}

```

12.5.2. List buckets

```

#include "oss_api.h"
#include "aos_http_io.h"
const char *endpoint = "<yourEndpoint>";
const char *access_key_id = "<yourAccessKeyId>";
const char *access_key_secret = "<yourAccessKeySecret>";
void init_options(oss_request_options_t *options)
{
    options->config = oss_config_create(options->pool);
}

```

```

/* Use a char* string to initialize the aos_string_t data type. */
aos_str_set(&options->config->endpoint, endpoint);
aos_str_set(&options->config->access_key_id, access_key_id);
aos_str_set(&options->config->access_key_secret, access_key_secret);
/* Specify whether to use CNAME. 0 indicates that the CNAME is not used. */
options->config->is_cname = 0;
/* Configure network parameters such as timeout. */
options->ctl = aos_http_controller_create(options->pool, 0);
}

int main(int argc, char *argv[])
{
    /* Call aos_http_io_initialize in main() to initialize global resources, such as the ne
twork and memory. */
    if (aos_http_io_initialize(NULL, 0) != AOSE_OK) {
        exit(1);
    }

    /* Create a memory pool to manage memory. The implementation code is included in the AP
R library. aos_pool_t is equivalent to apr_pool_t. */
    aos_pool_t *pool;
    /* Create a new memory pool. The second parameter value is NULL. This value indicates t
hat the pool does not inherit any other memory pool. */
    aos_pool_create(&pool, NULL);
    /* Create and initialize options. This parameter includes global configuration informat
ion such as endpoint, access_key_id, access_key_secret, is_cname, and curl. */
    oss_request_options_t *oss_client_options;
    /* Allocate a memory chunk in the memory pool to options. */
    oss_client_options = oss_request_options_create(pool);
    /* Initialize oss_client_options. */
    init_options(oss_client_options);
    /* Initialize parameters. */
    aos_table_t *resp_headers = NULL;
    aos_status_t *resp_status = NULL;
    oss_list_buckets_params_t *params = NULL;
    oss_list_bucket_content_t *content = NULL;
    int size = 0;
    params = oss_create_list_buckets_params(pool);
    /*List the buckets.*/
    resp_status = oss_list_bucket(oss_client_options, params, &resp_headers);
    if (aos_status_is_ok(resp_status)) {
        printf("list buckets succeeded\n");
    } else {
        printf("list buckets failed\n");
    }
    // Display the list of buckets.
    aos_list_for_each_entry(oss_list_bucket_content_t, content, &params->bucket_list, node)
    {
        printf("BucketName: %s\n", content->name.data);
        ++size;
    }
    /* Release the memory pool. The memory allocated to various resources used for the requ
est is released. */
    aos_pool_destroy(pool);
    /* Release the allocated global resources. */
    aos_http_io_deinitialize();
}

```

```
    return 0;
}
```

12.5.3. Query the regions of buckets

A bucket is a container for objects stored in Object Storage Service (OSS). Every object is contained in a bucket. This topic describes how to query the region of a bucket.

```
#include "oss_api.h"
#include "aos_http_io.h"
const char *endpoint = "<yourEndpoint>";
const char *access_key_id = "<yourAccessKeyId>";
const char *access_key_secret = "<yourAccessKeySecret>";
const char *bucket_name = "<yourBucketName>";
void init_options(oss_request_options_t *options)
{
    options->config = oss_config_create(options->pool);
    /* Use a char* string to initialize the aos_string_t data type. */
    aos_str_set(&options->config->endpoint, endpoint);
    aos_str_set(&options->config->access_key_id, access_key_id);
    aos_str_set(&options->config->access_key_secret, access_key_secret);
    /* Specify whether to use CNAME. 0 indicates that the CNAME is not used. */
    options->config->is_cname = 0;
    /* Configure network parameters such as timeout. */
    options->ctl = aos_http_controller_create(options->pool, 0);
}
int main(int argc, char *argv[])
{
    /* Call aos_http_io_initialize in main() to initialize global resources such as the net
work and memory. */
    if (aos_http_io_initialize(NULL, 0) != AOSE_OK) {
        exit(1);
    }
    /* Create a memory pool to manage memory. The implementation code is included in the AP
R library. aos_pool_t is equivalent to apr_pool_t. */
    aos_pool_t *pool;
    /* Create a new memory pool. The second parameter value is NULL. This value indicates t
hat the pool does not inherit any other memory pool. */
    aos_pool_create(&pool, NULL);
    /* Create and initialize options. This parameter includes global configuration informat
ion such as endpoint, access_key_id, access_key_secret, is_cname, and curl. */
    oss_request_options_t *oss_client_options;
    /* Allocate a memory chunk in the memory pool to options. */
    oss_client_options = oss_request_options_create(pool);
    /* Call oss_client_options to initialize the client options. */
    init_options(oss_client_options);
    /* Initialize the parameters. */
    aos_string_t bucket;
    aos_string_t oss_location;
    aos_table_t *resp_headers = NULL;
    aos_status_t *resp_status = NULL;
    /* Assign char* data to the aos_string_t bucket. */
    aos_str_set(&bucket, bucket_name);
```



```

    // Query the region of the bucket. */
    resp_status = oss_get_bucket_location(oss_client_options, &bucket, &oss_location, &resp_headers);
    if (aos_status_is_ok(resp_status)) {
        printf("get bucket location succeeded : %s \n", oss_location.data);
    } else {
        printf("get bucket location failed\n");
    }
    /* Release the memory pool. The memory allocated to various resources used for the request is released. */
    aos_pool_destroy(pool);
    /* Release the allocated global resources. */
    aos_http_io_deinitialize();
    return 0;
}

```

12.5.4. Query bucket information

A bucket is a container for objects stored in Object Storage Service (OSS). Every object is contained in a bucket. This topic describes how to query the information about a bucket.

The following code provides an example on how to query the information about a bucket, including the region and creation date:

```

#include "oss_api.h"
#include "aos_http_io.h"
const char *endpoint = "<yourEndpoint>";
const char *access_key_id = "<yourAccessKeyId>";
const char *access_key_secret = "<yourAccessKeySecret>";
const char *bucket_name = "<yourBucketName>";
void init_options(oss_request_options_t *options)
{
    options->config = oss_config_create(options->pool);
    /* Use a char* string to initialize the aos_string_t data type. */
    aos_str_set(&options->config->endpoint, endpoint);
    aos_str_set(&options->config->access_key_id, access_key_id);
    aos_str_set(&options->config->access_key_secret, access_key_secret);
    /* Specify whether to use CNAME. 0 indicates that the CNAME is not used. */
    options->config->is_cname = 0;
    /* Configure network parameters such as timeout. */
    options->ctl = aos_http_controller_create(options->pool, 0);
}
int main(int argc, char *argv[])
{
    /* Call aos_http_io_initialize in main() to initialize global resources, such as the network and memory. */
    if (aos_http_io_initialize(NULL, 0) != AOSE_OK) {
        exit(1);
    }
    /* Create a memory pool to manage memory. The implementation code is included in the APR library. aos_pool_t is equivalent to apr_pool_t. */
    aos_pool_t *pool;
    /* Create a new memory pool. The second parameter value is NULL. This value indicates t

```

```

    hat the pool does not inherit any other memory pool. */
    aos_pool_create(&pool, NULL);
    /* Create and initialize options. This parameter includes global configuration informat
ion such as endpoint, access_key_id, access_key_secret, is_cname, and curl. */
    oss_request_options_t *oss_client_options;
    /* Allocate a memory chunk in the memory pool to options. */
    oss_client_options = oss_request_options_create(pool);
    /* Call oss_client_options to initialize the client options. */
    init_options(oss_client_options);
    /* Initialize the parameters. */
    aos_string_t bucket;
    oss_bucket_info_t bucket_info;
    aos_table_t *resp_headers = NULL;
    aos_status_t *resp_status = NULL;
    /* Assign char* data to the aos_string_t bucket. */
    aos_str_set(&bucket, bucket_name);
    // Query bucket information. */
    resp_status = oss_get_bucket_info(oss_client_options, &bucket, &bucket_info, &resp_head
ers);
    if (aos_status_is_ok(resp_status)) {
        printf("get bucket info succeeded\n");
        printf("location: %s\n", bucket_info.location.data);
        printf("owner_id: %s\n", bucket_info.owner_id.data);
        printf("owner_name: %s\n", bucket_info.owner_name.data);
        printf("created_date: %s\n", bucket_info.created_date.data);
        printf("location: %s\n", bucket_info.location.data);
    } else {
        printf("get bucket info failed\n");
    }
    /* Release the memory pool. The memory allocated to various resources used for the requ
est is released. */
    aos_pool_destroy(pool);
    /* Release the allocated global resources. */
    aos_http_io_deinitialize();
    return 0;
}

```

For more information about how to query bucket information, see [GetBucketInfo](#).

12.5.5. Manage bucket ACLs

A bucket is a container used to store objects in Object Storage Service (OSS). Every object is contained in a bucket. This topic describes how to configure and query the access control list (ACL) of a bucket.

Configure the ACL of a bucket

The following table describes the bucket ACLs.

ACL	Description	Value
-----	-------------	-------

ACL	Description	Value
Private	Only the owner or authorized users of this bucket can read and write objects in the bucket. Other users, including anonymous users cannot access the objects in the bucket without authorization.	OSS_ACL_PRIVATE
Public read	Only the owner or authorized users of this bucket can write objects in the bucket. Other users, including anonymous users can only read objects in the bucket. We recommend that you exercise with caution when you set the bucket ACL to public read.	OSS_ACL_PUBLIC_READ
Public read/write	Any users, including anonymous users can read and write objects in the bucket. We recommend that you exercise with caution when you set the bucket ACL to public read/write.	OSS_ACL_PUBLIC_READ_WRITE

The following code provides an example on how to configure the ACL of a bucket:

```
#include "oss_api.h"
#include "aos_http_io.h"
const char *endpoint = "<yourEndpoint>";
const char *access_key_id = "<yourAccessKeyId>";
const char *access_key_secret = "<yourAccessKeySecret>";
const char *bucket_name = "<yourBucketName>";
void init_options(oss_request_options_t *options)
{
    options->config = oss_config_create(options->pool);
    /* Use a char* string to initialize the aos_string_t data type. */
    aos_str_set(&options->config->endpoint, endpoint);
    aos_str_set(&options->config->access_key_id, access_key_id);
    aos_str_set(&options->config->access_key_secret, access_key_secret);
    /* Specify whether to use CNAME. 0 indicates that the CNAME is not used. */
    options->config->is_cname = 0;
    /* Configure network parameters such as timeout. */
    options->ctl = aos_http_controller_create(options->pool, 0);
}
int main(int argc, char *argv[])
{
    /* Call aos_http_io_initialize in main() to initialize global resources such as the net
work and memory. */
    if (aos_http_io_initialize(NULL, 0) != AOSE_OK) {
        exit(1);
    }
    /* Create a memory pool to manage memory. The implementation code is included in the AP
R library. aos_pool_t is equivalent to apr_pool_t. */
    aos_pool_t *pool;
```

```

aos_pool_t pool;
/* Create a new memory pool. The second parameter value is NULL. This value indicates t
hat the pool does not inherit any other memory pool. */
aos_pool_create(&pool, NULL);
/* Create and initialize options. This parameter includes global configuration informat
ion such as endpoint, access_key_id, access_key_secret, is_cname, and curl. */
oss_request_options_t *oss_client_options;
/* Allocate a memory chunk in the memory pool to options. */
oss_client_options = oss_request_options_create(pool);
/* Call oss_client_options to initialize the client options. */
init_options(oss_client_options);
/* Initialize the parameters. */
aos_string_t bucket;
aos_table_t *resp_headers = NULL;
aos_status_t *resp_status = NULL;
/* Assign char* data to the aos_string_t bucket. */
aos_str_set(&bucket, bucket_name);
/* Set the ACL of the bucket to public read (OSS_ACL_PUBLIC_READ).
resp_status = oss_put_bucket_acl(oss_client_options, &bucket, OSS_ACL_PUBLIC_READ, &res
p_headers);
if (aos_status_is_ok(resp_status)) {
    printf("set bucket acl succeeded\n");
} else {
    printf("set bucket acl failed\n");
}
/* Release the memory pool. The memory allocated to various resources used for the requ
est is released. */
aos_pool_destroy(pool);
/* Release the allocated global resources. */
aos_http_io_deinitialize();
return 0;
}

```

Obtain the ACL of a bucket

The following code provides an example on how to query the ACL of a bucket:

```

#include "oss_api.h"
#include "aos_http_io.h"
const char *endpoint = "<yourEndpoint>";
const char *access_key_id = "<yourAccessKeyId>";
const char *access_key_secret = "<yourAccessKeySecret>";
const char *bucket_name = "<yourBucketName>";
void init_options(oss_request_options_t *options)
{
    options->config = oss_config_create(options->pool);
    /* Use a char* string to initialize the aos_string_t data type. */
    aos_str_set(&options->config->endpoint, endpoint);
    aos_str_set(&options->config->access_key_id, access_key_id);
    aos_str_set(&options->config->access_key_secret, access_key_secret);
    /* Specify whether to use CNAME. 0 indicates that the CNAME is not used. */
    options->config->is_cname = 0;
    /* Configure network parameters, such as timeout. */
    options->ctl = aos_http_controller_create(options->pool, 0);
}

```

```

}
int main(int argc, char *argv[])
{
    /* Call aos_http_io_initialize in main() to initialize global resources such as the net
work and memory. */
    if (aos_http_io_initialize(NULL, 0) != AOSE_OK) {
        exit(1);
    }
    /* Create a memory pool to manage memory. The implementation code is included in the AP
R library. aos_pool_t is equivalent to apr_pool_t. */
    aos_pool_t *pool;
    /* Create a new memory pool. The second parameter value is NULL. This value indicates t
hat the pool does not inherit any other memory pool. */
    aos_pool_create(&pool, NULL);
    /* Create and initialize options. This parameter includes global configuration informat
ion such as endpoint, access_key_id, access_key_secret, is_cname, and curl. */
    oss_request_options_t *oss_client_options;
    /* Allocate a memory chunk in the memory pool to options. */
    oss_client_options = oss_request_options_create(pool);
    /* Call oss_client_options to initialize the client options. */
    init_options(oss_client_options);
    /* Initialize the parameters. */
    aos_string_t bucket;
    aos_string_t oss_acl;
    aos_table_t *resp_headers = NULL;
    aos_status_t *resp_status = NULL;
    /* Assign char* data to the aos_string_t bucket. */
    aos_str_set(&bucket, bucket_name);
    // Query the ACL of the bucket. */
    resp_status = oss_get_bucket_acl(oss_client_options, &bucket, &oss_acl, &resp_headers);
    if (aos_status_is_ok(resp_status)) {
        printf("get bucket acl succeeded : %s \n", oss_acl.data);
    } else {
        printf("get bucket acl failed\n");
    }
    /* Release the memory pool. The memory allocated to various resources used for the requ
est is released. */
    aos_pool_destroy(pool);
    /* Release the allocated global resources. */
    aos_http_io_deinitialize();
    return 0;
}

```

12.5.6. Delete buckets

Note

- To delete parts generated in multipart uploads, call `oss_list_upload_part` to list the parts. Then, call `oss_abort_multipart_upload` to delete the parts. For more information, see [Multipart upload](#).

```

#include "oss_api.h"
#include "aos_http_io.h"
const char *endpoint = "<yourEndpoint>";
const char *access_key_id = "<yourAccessKeyId>";
const char *access_key_secret = "<yourAccessKeySecret>";
const char *bucket_name = "<yourBucketName>";
void init_options(oss_request_options_t *options)
{
    options->config = oss_config_create(options->pool);
    /* Use a char* string to initialize the aos_string_t data type. */
    aos_str_set(&options->config->endpoint, endpoint);
    aos_str_set(&options->config->access_key_id, access_key_id);
    aos_str_set(&options->config->access_key_secret, access_key_secret);
    /* Specify whether to use CNAME. 0 indicates that the CNAME is not used. */
    options->config->is_cname = 0;
    /* Configure network parameters, such as timeout. */
    options->ctl = aos_http_controller_create(options->pool, 0);
}
int main(int argc, char *argv[])
{
    /* Call aos_http_io_initialize in main() to initialize global resources, such as the ne
twork and memory. */
    if (aos_http_io_initialize(NULL, 0) != AOSE_OK) {
        exit(1);
    }
    /* Create a memory pool to manage memory. The implementation code is included in the AP
R library. aos_pool_t is equivalent to apr_pool_t. */
    aos_pool_t *pool;
    /* Create a new memory pool. The second parameter value is NULL. This value indicates t
hat the pool does not inherit any other memory pool. */
    aos_pool_create(&pool, NULL);
    /* Create and initialize options. This parameter includes global configuration informat
ion such as endpoint, access_key_id, access_key_secret, is_cname, and curl. */
    oss_request_options_t *oss_client_options;
    /* Allocate a memory chunk in the memory pool to options. */
    oss_client_options = oss_request_options_create(pool);
    /* Call oss_client_options to initialize the client options. */
    init_options(oss_client_options);
    /* Initialize the parameters. */
    aos_string_t bucket;
    aos_table_t *resp_headers = NULL;
    aos_status_t *resp_status = NULL;
    /* Assign char* data to the aos_string_t bucket. */
    aos_str_set(&bucket, bucket_name);
    /* Delete the bucket. */
    resp_status = oss_delete_bucket (oss_client_options, &bucket, &resp_headers);
    if (aos_status_is_ok(resp_status)) {
        printf("delete bucket succeeded\n");
    } else {
        printf("delete bucket failed\n");
    }
    /* Release the memory pool. The memory allocated to various resources used for the requ
est is released. */
    aos_pool_destroy(pool);
}

```

```
/* Release the allocated global resources. */
aos_http_io_deinitialize();
return 0;
}
```

12.5.7. Manage lifecycle rules

Object Storage Service (OSS) allows you to configure lifecycle rules to delete expired objects and parts or convert the storage class of expired objects to Infrequent Access (IA) or Archive to reduce your storage costs. This topic describes how to manage lifecycle rules for buckets.

Background information

Each lifecycle rule contains the following information:

- Rule ID. It uniquely identifies a rule in a bucket.
- Policy. You can use one of the following methods to configure a policy. Only a single method can be configured for each bucket.
 - Match by prefix: You can create multiple rules to configure different prefixes. Each prefix must be unique.
 - Match by bucket: If you select this method, only a single rule can be created.
- Expiration time. You can use one of the following configuration methods to configure the expiration time:
 - Specify a validity period for which to retain objects after they are last modified.
 - Specify an expiration date that objects that are last modified before this date expire.
- Status.

Configure lifecycle rules

```
#include "oss_api.h"
#include "aos_http_io.h"
const char *endpoint = "<yourEndpoint>";
const char *access_key_id = "<yourAccessKeyId>";
const char *access_key_secret = "<yourAccessKeySecret>";
const char *bucket_name = "<yourBucketName>";
void init_options(oss_request_options_t *options)
{
    options->config = oss_config_create(options->pool);
    /* Use a char* string to initialize the aos_string_t data type. */
    aos_str_set(&options->config->endpoint, endpoint);
    aos_str_set(&options->config->access_key_id, access_key_id);
    aos_str_set(&options->config->access_key_secret, access_key_secret);
    /* Specify whether to use CNAME to access OSS. 0 indicates that CNAME is not used. */
    options->config->is_cname = 0;
    /* Configure network parameters such as the timeout periods. */
    options->ctl = aos_http_controller_create(options->pool, 0);
}
int main(int argc, char *argv[])
{
    /* Call aos_http_io_initialize in main() to initialize global resources such as the net
```

```

work and memory. */
if (aos_http_io_initialize(NULL, 0) != AOSE_OK) {
    exit(1);
}
/* Create a memory pool to manage memory. aos_pool_t is equivalent to apr_pool_t. The code to create a memory pool is included in the APR library. */
aos_pool_t *pool;
/* Create a memory pool. The second parameter is NULL. This parameter indicates that the pool does not inherit any other memory pool. */
aos_pool_create(&pool, NULL);
/* Create and initialize options. This parameter includes global configuration information such as endpoint, access_key_id, access_key_secret, is_cname, and curl. */
oss_request_options_t *oss_client_options;
/* Allocate a memory chunk in the memory pool to options. */
oss_client_options = oss_request_options_create(pool);
/* Call oss_client_options to initialize the client options. */
init_options(oss_client_options);
/* Initialize the parameters. */
aos_string_t bucket;
aos_table_t *resp_headers = NULL;
aos_status_t *resp_status = NULL;
aos_str_set(&bucket, bucket_name);
aos_list_t lifecycle_rule_list;
/* Create lifecycle rules for the bucket. */
aos_str_set(&bucket, bucket_name);
aos_list_init(&lifecycle_rule_list);
/* Specify the validity period. */
oss_lifecycle_rule_content_t *rule_content_days = oss_create_lifecycle_rule_content(pool);
aos_str_set(&rule_content_days->id, "rule-1");
aos_str_set(&rule_content_days->prefix, "obsoleted");
aos_str_set(&rule_content_days->status, "Enabled");
rule_content_days->days = 3;
aos_list_add_tail(&rule_content_days->node, &lifecycle_rule_list);
/* Specify the expiration date. */
oss_lifecycle_rule_content_t *rule_content_date = oss_create_lifecycle_rule_content(pool);
aos_str_set(&rule_content_date->id, "rule-2");
aos_str_set(&rule_content_date->prefix, "delete");
aos_str_set(&rule_content_date->status, "Enabled");
aos_str_set(&rule_content_date->date, "2022-10-11T00:00:00.000Z");
aos_list_add_tail(&rule_content_date->node, &lifecycle_rule_list);
/* Configure the lifecycle rules. */
resp_status = oss_put_bucket_lifecycle(oss_client_options, &bucket, &lifecycle_rule_list, &resp_headers);
if (aos_status_is_ok(resp_status)) {
    printf("put bucket lifecycle succeeded\n");
} else {
    printf("put bucket lifecycle failed, code:%d, error_code:%s, error_msg:%s, request_id:%s\n",
        resp_status->code, resp_status->error_code, resp_status->error_msg, resp_status->req_id);
}
/* Release the memory pool. The memory allocated to various resources used for the request

```



```

est is released. */
aos_pool_destroy(pool);
/* Release the allocated global resources. */
aos_http_io_deinitialize();
return 0;
}

```

View lifecycle rules

The following code provides an example on how to query the lifecycle rules configured for the bucket named `examplebucket`:

```

#include "oss_api.h"
#include "aos_http_io.h"
const char *endpoint = "<yourEndpoint>";
const char *access_key_id = "<yourAccessKeyId>";
const char *access_key_secret = "<yourAccessKeySecret>";
const char *bucket_name = "<yourBucketName>";
void init_options(oss_request_options_t *options)
{
    options->config = oss_config_create(options->pool);
    /* Use a char* string to initialize the aos_string_t data type. */
    aos_str_set(&options->config->endpoint, endpoint);
    aos_str_set(&options->config->access_key_id, access_key_id);
    aos_str_set(&options->config->access_key_secret, access_key_secret);
    /* Specify whether to use CNAME to access OSS. 0 indicates that CNAME is not used. */
    options->config->is_cname = 0;
    /* Configure network parameters such as the timeout periods. */
    options->ctl = aos_http_controller_create(options->pool, 0);
}
int main(int argc, char *argv[])
{
    /* Call aos_http_io_initialize in main() to initialize global resources such as the net
work and memory. */
    if (aos_http_io_initialize(NULL, 0) != AOSE_OK) {
        exit(1);
    }
    /* Create a memory pool to manage memory. aos_pool_t is equivalent to apr_pool_t. The c
ode to create a memory pool is included in the APR library. */
    aos_pool_t *pool;
    /* Create a memory pool. The second parameter is NULL. This parameter indicates that th
e pool does not inherit any other memory pool. */
    aos_pool_create(&pool, NULL);
    /* Create and initialize options. This parameter includes global configuration informat
ion such as endpoint, access_key_id, access_key_secret, is_cname, and curl. */
    oss_request_options_t *oss_client_options;
    /* Allocate a memory chunk in the memory pool to options. */
    oss_client_options = oss_request_options_create(pool);
    /* Call oss_client_options to initialize the client options. */
    init_options(oss_client_options);
    /* Initialize the parameters. */
    aos_string_t bucket;
    aos_table_t *resp_headers = NULL;
    aos_status_t *resp_status = NULL;

```

```

aos_list_t lifecycle_rule_list;
oss_lifecycle_rule_content_t *rule_content;
char *rule_id;
char *prefix;
char *status;
int days = INT_MAX;
char* date = "";
aos_str_set(&bucket, bucket_name);
/* Query and display the lifecycle rules configured for the bucket. */
aos_list_init(&lifecycle_rule_list);
resp_status = oss_get_bucket_lifecycle(oss_client_options, &bucket, &lifecycle_rule_list, &resp_headers);
if (aos_status_is_ok(resp_status)) {
    printf("get bucket lifecycle succeeded\n");
    aos_list_for_each_entry(oss_lifecycle_rule_content_t, rule_content, &lifecycle_rule_list, node) {
        rule_id = apr_psprintf(pool, "%.s", rule_content->id.len, rule_content->id.data);
        prefix = apr_psprintf(pool, "%.s", rule_content->prefix.len, rule_content->prefix.data);
        status = apr_psprintf(pool, "%.s", rule_content->status.len, rule_content->status.data);
        date = apr_psprintf(pool, "%.s", rule_content->date.len, rule_content->date.data);
        days = rule_content->days;
    }
    printf("rule_id: %s \n", rule_id);
    printf("prefix: %s \n", prefix);
    printf("status: %s \n", status);
    printf("date: %s \n", date);
    printf("days: %d \n", days);
} else {
    printf("get bucket lifecycle failed, code:%d, error_code:%s, error_msg:%s, request_id:%s\n",
        resp_status->code, resp_status->error_code, resp_status->error_msg, resp_status->req_id);
}
/* Release the memory pool. The memory allocated to various resources used for the request is released. */
aos_pool_destroy(pool);
/* Release the allocated global resources. */
aos_http_io_deinitialize();
return 0;
}

```

Delete lifecycle rules

The following code provides an example on how to delete lifecycle rules configured for the bucket named examplebucket:

```

#include "oss_api.h"
#include "aos_http_io.h"
const char *endpoint = "<yourEndpoint>";
const char *access_key_id = "<yourAccessKeyId>";

```

```

const char *access_key_secret = "<yourAccessKeySecret>";
const char *bucket_name = "<yourBucketName>";
void init_options(oss_request_options_t *options)
{
    options->config = oss_config_create(options->pool);
    /* Use a char* string to initialize the aos_string_t data type. */
    aos_str_set(&options->config->endpoint, endpoint);
    aos_str_set(&options->config->access_key_id, access_key_id);
    aos_str_set(&options->config->access_key_secret, access_key_secret);
    /* Specify whether to use CNAME to access OSS. 0 indicates that CNAME is not used. */
    options->config->is_cname = 0;
    /* Configure network parameters such as the timeout periods. */
    options->ctl = aos_http_controller_create(options->pool, 0);
}

int main(int argc, char *argv[])
{
    /* Call aos_http_io_initialize in main() to initialize global resources such as the net
work and memory. */
    if (aos_http_io_initialize(NULL, 0) != AOSE_OK) {
        exit(1);
    }

    /* Create a memory pool to manage memory. aos_pool_t is equivalent to apr_pool_t. The c
ode to create a memory pool is included in the APR library. */
    aos_pool_t *pool;
    /* Create a memory pool. The second parameter is NULL. This parameter indicates that th
e pool does not inherit any other memory pool. */
    aos_pool_create(&pool, NULL);
    /* Create and initialize options. This parameter includes global configuration informat
ion such as endpoint, access_key_id, access_key_secret, is_cname, and curl. */
    oss_request_options_t *oss_client_options;
    /* Allocate a memory chunk in the memory pool to options. */
    oss_client_options = oss_request_options_create(pool);
    /* Call oss_client_options to initialize the client options. */
    init_options(oss_client_options);
    /* Initialize the parameters. */
    aos_string_t bucket;
    aos_table_t *resp_headers = NULL;
    aos_status_t *resp_status = NULL;
    aos_str_set(&bucket, bucket_name);
    /* Delete the lifecycle rules configured for the bucket. */
    aos_str_set(&bucket, bucket_name);
    resp_status = oss_delete_bucket_lifecycle(oss_client_options, &bucket, &resp_headers);
    if (aos_status_is_ok(resp_status)) {
        printf("delete bucket lifecycle succeeded\n");
    } else {
        printf("delete bucket lifecycle failed, code:%d, error_code:%s, error_msg:%s, reques
t_id:%s\n",
            resp_status->code, resp_status->error_code, resp_status->error_msg, resp_status
->req_id);
    }
    /* Release the memory pool. The memory allocated to various resources used for the requ
est is released. */
    aos_pool_destroy(pool);
    /* Release the allocated global resources. */
}

```

```

aos_http_io_deinitialize();
return 0;
}

```

12.5.8. CORS

Cross-origin resource sharing (CORS) allows web applications to access resources that belong to another region. OSS provides CORS APIs for convenient cross-origin access control.

For more information, see [Cross-origin resource sharing](#) and [PutBucketcors](#) in OSS Developer Guide.

Configure CORS rules

Run the following code to configure CORS rules for a specified bucket:

```

#include "oss_api.h"
#include "aos_http_io.h"
const char *endpoint = "<yourEndpoint>";
const char *access_key_id = "<yourAccessKeyId>";
const char *access_key_secret = "<yourAccessKeySecret>";
const char *bucket_name = "<yourBucketName>";
void init_options(oss_request_options_t *options)
{
    options->config = oss_config_create(options->pool);
    /* Use a char* string to initialize aos_string_t. */
    aos_str_set(&options->config->endpoint, endpoint);
    aos_str_set(&options->config->access_key_id, access_key_id);
    aos_str_set(&options->config->access_key_secret, access_key_secret);
    /* Determine whether the CNAME is used. 0 indicates that the CNAME is not used. */
    options->config->is_cname = 0;
    /* Configure network parameters, such as timeout. */
    options->ctl = aos_http_controller_create(options->pool, 0);
}
int main(int argc, char *argv[])
{
    /* Call the aos_http_io_initialize method in main() to initialize global resources, such as networks and memories. */
    if (aos_http_io_initialize(NULL, 0) != AOSE_OK) {
        exit(1);
    }
    /* Memory pool used to manage memories, which is equivalent to apr_pool_t. The implementation code is included in the apr library. */
    aos_pool_t *pool;
    /* Re-create a new memory pool. The second parameter is NULL, indicating that it does not inherit from any other memory pools. */
    aos_pool_create(&pool, NULL);
    /* Create and initialize options. This parameter mainly includes global configuration information, such as endpoint, access_key_id, access_key_secret, is_cname, and curl. */
    oss_request_options_t *oss_client_options;
    /* Allocate memories in the memory pool to options. */
    oss_client_options = oss_request_options_create(pool);
    /* Initialize oss_client_options. */
    init_options(oss_client_options);
    /* Initialization parameters. */
}

```

```

aos_string_t bucket;
aos_table_t *resp_headers = NULL;
aos_status_t *resp_status = NULL;
aos_list_t cors_rule_list;
oss_cors_rule_t *cors_rule1 = NULL, *cors_rule2 = NULL;
aos_str_set(&bucket, bucket_name);
aos_list_init(&cors_rule_list);
cors_rule1 = oss_create_cors_rule(pool);
aos_list_add_tail(&cors_rule1->node, &cors_rule_list);
oss_create_sub_cors_rule(pool, &cors_rule1->allowed_origin_list, "allowed_origin_1_1");
oss_create_sub_cors_rule(pool, &cors_rule1->allowed_origin_list, "allowed_origin_1_1");
oss_create_sub_cors_rule(pool, &cors_rule1->allowed_method_list, "PUT");
oss_create_sub_cors_rule(pool, &cors_rule1->allowed_method_list, "GET");
oss_create_sub_cors_rule(pool, &cors_rule1->allowed_head_list, "Authorization");
oss_create_sub_cors_rule(pool, &cors_rule1->expose_head_list, "expose_head_1_1");
oss_create_sub_cors_rule(pool, &cors_rule1->expose_head_list, "expose_head_1_1");
cors_rule2 = oss_create_cors_rule(pool);
aos_list_add_tail(&cors_rule2->node, &cors_rule_list);
oss_create_sub_cors_rule(pool, &cors_rule2->allowed_origin_list, "allowed_origin_2_1");
oss_create_sub_cors_rule(pool, &cors_rule2->allowed_origin_list, "allowed_origin_2_2");
oss_create_sub_cors_rule(pool, &cors_rule2->allowed_method_list, "PUT");
oss_create_sub_cors_rule(pool, &cors_rule2->allowed_method_list, "GET");
oss_create_sub_cors_rule(pool, &cors_rule2->allowed_head_list, "Authorization");
oss_create_sub_cors_rule(pool, &cors_rule2->expose_head_list, "expose_head_2_1");
oss_create_sub_cors_rule(pool, &cors_rule2->expose_head_list, "expose_head_2_2");
/* Configure the CORS rule. */
resp_status = oss_put_bucket_cors(oss_client_options, &bucket, &cors_rule_list, &resp_h
eaders);
if (aos_status_is_ok(resp_status)) {
    printf("put bucket cors succeeded\n");
} else {
    printf("put bucket cors failed\n");
}
/* Release the memory pool, that is, memories allocated to resources during the request
. */
aos_pool_destroy(pool);
/* Release the allocated global resources. */
aos_http_io_deinitialize();
return 0;
}

```

Obtain CORS rules

Run the following code to obtain CORS rules:

```

#include "oss_api.h"
#include "aos_http_io.h"
const char *endpoint = "<yourEndpoint>";
const char *access_key_id = "<yourAccessKeyId>";
const char *access_key_secret = "<yourAccessKeySecret>";
const char *bucket_name = "<yourBucketName>";
void init_options(oss_request_options_t *options)
{
    options->config = oss_config_create(options->pool);
}

```

```

/* Use a char* string to initialize aos_string_t. */
aos_str_set(&options->config->endpoint, endpoint);
aos_str_set(&options->config->access_key_id, access_key_id);
aos_str_set(&options->config->access_key_secret, access_key_secret);
/* Determine whether the CNAME is used. 0 indicates that the CNAME is not used. */
options->config->is_cname = 0;
/* Configure network parameters, such as timeout. */
options->ctl = aos_http_controller_create(options->pool, 0);
}

int main(int argc, char *argv[])
{
    /* Call the aos_http_io_initialize method in main() to initialize global resources, such as networks and memories. */
    if (aos_http_io_initialize(NULL, 0) != AOSE_OK) {
        exit(1);
    }

    /* Memory pool used to manage memories, which is equivalent to apr_pool_t. The implementation code is included in the apr library. */
    aos_pool_t *pool;
    /* Re-create a new memory pool. The second parameter is NULL, indicating that it does not inherit from any other memory pools. */
    aos_pool_create(&pool, NULL);
    /* Create and initialize options. This parameter mainly includes global configuration information, such as endpoint, access_key_id, access_key_secret, is_cname, and curl. */
    oss_request_options_t *oss_client_options;
    /* Allocate memories in the memory pool to options. */
    oss_client_options = oss_request_options_create(pool);
    /* Initialize oss_client_options. */
    init_options(oss_client_options);
    /* Initialization parameters. */
    aos_string_t bucket;
    aos_table_t *resp_headers = NULL;
    aos_status_t *resp_status = NULL;
    aos_list_t cors_rule_list;
    oss_cors_rule_t *cors_rule = NULL;
    oss_sub_cors_rule_t *sub_cors_rule = NULL;
    aos_str_set(&bucket, bucket_name);
    /* Obtain CORS rules. */
    aos_list_init(&cors_rule_list);
    resp_status = oss_get_bucket_cors(oss_client_options, &bucket, &cors_rule_list, &resp_headers);

    if (aos_status_is_ok(resp_status)) {
        printf("get bucket cors succeeded\n");
        aos_list_for_each_entry(oss_cors_rule_t, cors_rule, &cors_rule_list, node) {
            printf("max_age_seconds: %d\n", cors_rule->max_age_seconds);
            aos_list_for_each_entry(oss_sub_cors_rule_t, sub_cors_rule, &cors_rule->allowed_origin_list, node) {
                printf("allowed_origin_list: %s \n", sub_cors_rule->rule.data);
            }
            aos_list_for_each_entry(oss_sub_cors_rule_t, sub_cors_rule, &cors_rule->allowed_method_list, node) {
                printf("allowed_method_list: %s \n", sub_cors_rule->rule.data);
            }
            aos_list_for_each_entry(oss_sub_cors_rule_t, sub_cors_rule, &cors_rule->allowed

```

```

_head_list, node) {
    printf("allowed_head_list: %s \n", sub_cors_rule->rule.data);
}
aos_list_for_each_entry(oss_sub_cors_rule_t, sub_cors_rule, &cors_rule->expose_
head_list, node) {
    printf("expose_head_list: %s \n", sub_cors_rule->rule.data);
}
}
} else {
    printf("get bucket cors failed\n");
}
/* Release the memory pool, that is, memories allocated to resources during the request
. */
aos_pool_destroy(pool);
/* Release the allocated global resources. */
aos_http_io_deinitialize();
return 0;
}

```

Delete CORS rules

Run the following code to delete all CORS rules for the specified bucket:

```

#include "oss_api.h"
#include "aos_http_io.h"
const char *endpoint = "<yourEndpoint>";
const char *access_key_id = "<yourAccessKeyId>";
const char *access_key_secret = "<yourAccessKeySecret>";
const char *bucket_name = "<yourBucketName>";
void init_options(oss_request_options_t *options)
{
    options->config = oss_config_create(options->pool);
    /* Use a char* string to initialize aos_string_t. */
    aos_str_set(&options->config->endpoint, endpoint);
    aos_str_set(&options->config->access_key_id, access_key_id);
    aos_str_set(&options->config->access_key_secret, access_key_secret);
    /* Determine whether the CNAME is used. 0 indicates that the CNAME is not used. */
    options->config->is_cname = 0;
    /* Configure network parameters, such as timeout. */
    options->ctl = aos_http_controller_create(options->pool, 0);
}
int main(int argc, char *argv[])
{
    /* Call the aos_http_io_initialize method in main() to initialize global resources, suc
h as networks and memories. */
    if (aos_http_io_initialize(NULL, 0) != AOSE_OK) {
        exit(1);
    }
    /* Memory pool used to manage memories, which is equivalent to apr_pool_t. The implemen
tation code is included in the apr library. */
    aos_pool_t *pool;
    /* Re-create a new memory pool. The second parameter is NULL, indicating that it does n
ot inherit from any other memory pools. */
    aos_pool_create(&pool, NULL);
}

```

```

/* Create and initialize options. This parameter mainly includes global configuration i
nformation, such as endpoint, access_key_id, acces_key_secret, is_cname, and curl. */
oss_request_options_t *oss_client_options;
/* Allocate memories in the memory pool to options. */
oss_client_options = oss_request_options_create(pool);
/* Initialize oss_client_options. */
init_options(oss_client_options);
/* Initialization parameters. */
aos_string_t bucket;
aos_table_t *resp_headers = NULL;
aos_status_t *resp_status = NULL;
aos_str_set(&bucket, bucket_name);
/* Delete all CORS rules. */
resp_status = oss_delete_bucket_cors(oss_client_options, &bucket, &resp_headers);
if (aos_status_is_ok(resp_status)) {
    printf("get bucket cors succeeded\n");
} else {
    printf("get bucket cors failed\n");
}
/* Release the memory pool, that is, memories allocated to resources during the request
. */
aos_pool_destroy(pool);
/* Release the allocated global resources. */
aos_http_io_deinitialize();
return 0;
}

```

12.5.9. Set logging

You can enable access logs to record bucket access to log files, which are stored in a specified bucket.

The log file format is as follows:

```
<TargetPrefix><SourceBucket>YYYY-mm-DD-HH-MM-SS-UniqueString
```

For more information about access log files, see [Set access logging](#).

Enable access logging

Run the following code to enable bucket access logging:

```

#include "oss_api.h"
#include "aos_http_io.h"
const char *endpoint = "<yourEndpoint>";
const char *access_key_id = "<yourAccessKeyId>";
const char *access_key_secret = "<yourAccessKeySecret>";
const char *bucket_name = "<yourBucketName>";
const char *target_bucket_name = "<yourTargetBucketName>";
const char *target_logging_prefix = "<yourTargetPrefix>";
void init_options(oss_request_options_t *options)
{
    options->config = oss_config_create(options->pool);
    /* Use a char* string to initialize aos_string_t. */
    aos_str_set(&options->config->endpoint, endpoint);
    aos_str_set(&options->config->access_key_id, access_key_id);
}

```



```

    aos_str_set(&options->config->access_key_secret, access_key_secret);
    /* Determine whether the CNAME is used. The value 0 indicates that the CNAME is not use
d. */
    options->config->is_cname = 0;
    /* Configure network parameters, such as timeout. */
    options->ctl = aos_http_controller_create(options->pool, 0);
}
int main(int argc, char *argv[])
{
    /* Call the maid method at the program portal to initialize global resources such as ne
twork, memory, and so on. */
    if (aos_http_io_initialize(NULL, 0) != AOSE_OK) {
        exit(1);
    }
    /* Memory pool used to manage memories, equivalent to apr_pool_t. The implementation co
de is included in the apr library. */
    aos_pool_t *pool;
    /* Re-create a memory pool. The second parameter is NULL, indicating that the new pool
does not inherit any other memory pool. */
    aos_pool_create(&pool, NULL);
    /* Create and initialize options. This parameter includes global configuration informat
ion, such as endpoint, access_key_id, acces_key_secret, is_cname, and curl. */
    oss_request_options_t *oss_client_options;
    /* Allocate memories in the memory pool to options. */
    oss_client_options = oss_request_options_create(pool);
    /* Initialize oss_client_options. */
    init_options(oss_client_options);
    /* Initialize parameters. */
    aos_string_t bucket;
    oss_logging_config_content_t *content;
    aos_table_t *resp_headers = NULL;
    aos_status_t *resp_status = NULL;
    aos_str_set(&bucket, bucket_name);
    content = oss_create_logging_rule_content(pool);
    aos_str_set(&content->target_bucket, target_bucket_name);
    aos_str_set(&content->prefix, target_logging_prefix);
    /* Enable bucket access logging. */
    resp_status = oss_put_bucket_logging(oss_client_options, &bucket, content, &resp_header
s);
    if (aos_status_is_ok(resp_status)) {
        printf("put symlink succeeded\n");
    } else {
        printf("put symlink failed, code:%d, error_code:%s, error_msg:%s, request_id:%s\n",
            resp_status->code, resp_status->error_code, resp_status->error_msg, resp_status
->req_id);
    }
    /* Release the memory pool, that is, the memories allocated to resources during the req
uest. */
    aos_pool_destroy(pool);
    /* Release the allocated global resources. */
    aos_http_io_deinitialize();
    return 0;
}

```

View access logging configurations

Run the following code to view the access logging configurations for a bucket:

```
#include "oss_api.h"
#include "aos_http_io.h"
const char *endpoint = "<yourEndpoint>";
const char *access_key_id = "<yourAccessKeyId>";
const char *access_key_secret = "<yourAccessKeySecret>";
const char *bucket_name = "<yourBucketName>";
void init_options(oss_request_options_t *options)
{
    options->config = oss_config_create(options->pool);
    /* Use a char* string to initialize aos_string_t. */
    aos_str_set(&options->config->endpoint, endpoint);
    aos_str_set(&options->config->access_key_id, access_key_id);
    aos_str_set(&options->config->access_key_secret, access_key_secret);
    /* Determine whether the CNAME is used. The value 0 indicates that the CNAME is not used. */
    options->config->is_cname = 0;
    /* Configure network parameters, such as timeout. */
    options->ctl = aos_http_controller_create(options->pool, 0);
}
int main(int argc, char *argv[])
{
    /* Call the aos_http_io_initialize method in main() to initialize global resources, such as networks and memories. */
    if (aos_http_io_initialize(NULL, 0) != AOSE_OK) {
        exit(1);
    }
    /* Memory pool used to manage memories, equivalent to apr_pool_t. The implementation code is included in the apr library. */
    aos_pool_t *pool;
    /* Re-create a memory pool. The second parameter is NULL, indicating that the new pool does not inherit any other memory pool. */
    aos_pool_create(&pool, NULL);
    /* Create and initialize options. This parameter includes global configuration information, such as endpoint, access_key_id, access_key_secret, is_cname, and curl. */
    oss_request_options_t *oss_client_options;
    /* Allocate memories in the memory pool to options. */
    oss_client_options = oss_request_options_create(pool);
    /* Initialize oss_client_options. */
    init_options(oss_client_options);
    /* Initialization parameters. */
    aos_string_t bucket;
    oss_logging_config_content_t *content;
    aos_table_t *resp_headers = NULL;
    aos_status_t *resp_status = NULL;
    aos_str_set(&bucket, bucket_name);
    content = oss_create_logging_rule_content(pool);
    /* View bucket access logging configurations. */
    resp_status = oss_get_bucket_logging(oss_client_options, &bucket, content, &resp_headers);
    if (aos_status_is_ok(resp_status)) {
        printf("Get bucket logging succeeded!\n");
    }
}
```

```

        printf("get bucket logging succeeded\n");
    } else {
        printf("get bucket logging failed, code:%d, error_code:%s, error_msg:%s, request_id:%s\n",
            resp_status->code, resp_status->error_code, resp_status->error_msg, resp_status->req_id);
    }
    printf("bucket:%s, prefix:%s\n", content->target_bucket.data, content->prefix.data);
    /* Release the memory pool, that is, the memories allocated to resources during the request. */
    aos_pool_destroy(pool);
    /* Release the allocated global resources. */
    aos_http_io_deinitialize();
    return 0;
}

```

Disable access logging

Run the following code to disable access logging for a bucket:

```

#include "oss_api.h"
#include "aos_http_io.h"
const char *endpoint = "<yourEndpoint>";
const char *access_key_id = "<yourAccessKeyId>";
const char *access_key_secret = "<yourAccessKeySecret>";
const char *bucket_name = "<yourBucketName>";
void init_options(oss_request_options_t *options)
{
    options->config = oss_config_create(options->pool);
    /* Use a char* string to initialize aos_string_t. */
    aos_str_set(&options->config->endpoint, endpoint);
    aos_str_set(&options->config->access_key_id, access_key_id);
    aos_str_set(&options->config->access_key_secret, access_key_secret);
    /* Determine whether the CNAME is used. The value 0 indicates that the CNAME is not used. */
    options->config->is_cname = 0;
    /* Configure network parameters, such as timeout. */
    options->ctl = aos_http_controller_create(options->pool, 0);
}
int main(int argc, char *argv[])
{
    /* Call the aos_http_io_initialize method in main() to initialize global resources, such as networks and memories. */
    if (aos_http_io_initialize(NULL, 0) != AOSE_OK) {
        exit(1);
    }
    /* Memory pool used to manage memories, equivalent to apr_pool_t. The implementation code is included in the apr library. */
    aos_pool_t *pool;
    /* Re-create a memory pool. The second parameter is NULL, indicating that the new pool does not inherit any other memory pool. */
    aos_pool_create(&pool, NULL);
    /* Create and initialize options. This parameter includes global configuration information, such as endpoint, access_key_id, access_key_secret, is_cname, and curl. */

```

```

    oss_request_options_t *oss_client_options;
    /* Allocate memories in the memory pool to options. */
    oss_client_options = oss_request_options_create(pool);
    /* Initialize oss_client_options. */
    init_options(oss_client_options);
    /* Initialization parameters. */
    aos_string_t bucket;
    aos_table_t *resp_headers = NULL;
    aos_status_t *resp_status = NULL;
    aos_str_set(&bucket, bucket_name);
    /* Disable access logging. */
    resp_status = oss_delete_bucket_logging(oss_client_options, &bucket, &resp_headers);
    if (aos_status_is_ok(resp_status)) {
        printf("delete bucket logging succeeded\n");
    } else {
        printf("delete bucket logging failed, code:%d, error_code:%s, error_msg:%s, request
_id:%s\n",
            resp_status->code, resp_status->error_code, resp_status->error_msg, resp_status
->req_id);
    }
    /* Release the memory pool, that is, the memories allocated to resources during the req
uest. */
    aos_pool_destroy(pool);
    /* Release the allocated global resources. */
    aos_http_io_deinitialize();
    return 0;
}

```

12.5.10. Hotlink protection

This topic describes how to use hotlink protection.

For more information about hotlink protection, see [Hotlink protection](#).

Configure hotlink protection for a bucket

The following code provides an example on how to configure a Referer whitelist for a bucket:

```

#include "oss_api.h"
#include "aos_http_io.h"
const char *endpoint = "<yourEndpoint>";
const char *access_key_id = "<yourAccessKeyId>";
const char *access_key_secret = "<yourAccessKeySecret>";
const char *bucket_name = "<yourBucketName>";
void init_options(oss_request_options_t *options)
{
    options->config = oss_config_create(options->pool);
    /* Use a char* string to initialize the aos_string_t data type. */
    aos_str_set(&options->config->endpoint, endpoint);
    aos_str_set(&options->config->access_key_id, access_key_id);
    aos_str_set(&options->config->access_key_secret, access_key_secret);
    /* Specify whether to use CNAME. The value of 0 indicates that CNAME is not used. */
    options->config->is_cname = 0;
    /* Configure network parameters such as the timeout period. */
}

```

```

    options->ctl = aos_http_controller_create(options->pool, 0);
}
int main(int argc, char *argv[])
{
    /* Call the aos_http_io_initialize method in main() to initialize global resources such
    as networks and memory. */
    if (aos_http_io_initialize(NULL, 0) != AOSE_OK) {
        exit(1);
    }
    /* Create a memory pool to manage memory. aos_pool_t is equivalent to apr_pool_t. The c
    ode used to create a memory pool is included in the APR library. */
    aos_pool_t *pool;
    /* Create a memory pool. The second parameter is set to NULL. This value indicates that
    the pool does not inherit other memory pools. */
    aos_pool_create(&pool, NULL);
    /* Create and initialize options. This parameter includes global configuration informat
    ion such as endpoint, access_key_id, access_key_secret, is_cname, and curl. */
    oss_request_options_t *oss_client_options;
    /* Allocate the memory resources in the memory pool to the options. */
    oss_client_options = oss_request_options_create(pool);
    /* Initialize oss_client_options. */
    init_options(oss_client_options);
    /* Initialize the parameters. */
    aos_string_t bucket;
    aos_table_t *resp_headers = NULL;
    aos_status_t *resp_status = NULL;
    oss_referer_config_t referer_config;
    aos_str_set(&bucket, bucket_name);
    aos_list_init(&referer_config.referer_list);
    oss_create_and_add_refer(pool, &referer_config, "http://www.aliyun.com");
    oss_create_and_add_refer(pool, &referer_config, "https://www.aliyun.com");
    referer_config.allow_empty_referer = 0;
    /* Add a Referer whitelist. You can use an asterisk (*) or a question mark (?) as a wil
    dcard character to set the Referer parameter. */
    resp_status = oss_put_bucket_referer(oss_client_options, &bucket, &referer_config, &res
    p_headers);
    if (aos_status_is_ok(resp_status)) {
        printf("put bucket referer succeeded\n");
    } else {
        printf("put bucket referer failed\n");
    }
    /* Release the memory pool. This operation releases the memory resources allocated for
    the request. */
    aos_pool_destroy(pool);
    /* Release the allocated global resources. */
    aos_http_io_deinitialize();
    return 0;
}

```

Query the hotlink protection configurations of the bucket

The following code provides an example on how to query a Referer whitelist of a bucket:

```
#include "oss_api.h"
```

```

#include "aos_http_io.h"
const char *endpoint = "<yourEndpoint>";
const char *access_key_id = "<yourAccessKeyId>";
const char *access_key_secret = "<yourAccessKeySecret>";
const char *bucket_name = "<yourBucketName>";
void init_options(oss_request_options_t *options)
{
    options->config = oss_config_create(options->pool);
    /* Use a char* string to initialize the aos_string_t data type. */
    aos_str_set(&options->config->endpoint, endpoint);
    aos_str_set(&options->config->access_key_id, access_key_id);
    aos_str_set(&options->config->access_key_secret, access_key_secret);
    /* Specify whether to use CNAME. The value of 0 indicates that CNAME is not used. */
    options->config->is_cname = 0;
    /* Configure network parameters such as the timeout period. */
    options->ctl = aos_http_controller_create(options->pool, 0);
}
int main(int argc, char *argv[])
{
    /* Call the aos_http_io_initialize method in main() to initialize global resources such
    as networks and memory. */
    if (aos_http_io_initialize(NULL, 0) != AOSE_OK) {
        exit(1);
    }
    /* Create a memory pool to manage memory. aos_pool_t is equivalent to apr_pool_t. The c
    ode used to create a memory pool is included in the APR library. */
    aos_pool_t *pool;
    /* Create a memory pool. The second parameter is set to NULL. This value indicates that
    the pool does not inherit other memory pools. */
    aos_pool_create(&pool, NULL);
    /* Create and initialize options. This parameter includes global configuration informat
    ion such as endpoint, access_key_id, access_key_secret, is_cname, and curl. */
    oss_request_options_t *oss_client_options;
    /* Allocate the memory resources in the memory pool to the options. */
    oss_client_options = oss_request_options_create(pool);
    /* Initialize oss_client_options. */
    init_options(oss_client_options);
    /* Initialize the parameters. */
    aos_string_t bucket;
    aos_table_t *resp_headers = NULL;
    aos_status_t *resp_status = NULL;
    oss_referer_config_t referer_config;
    oss_referer_t *referer;
    aos_str_set(&bucket, bucket_name);
    aos_list_init(&referer_config.referer_list);
    /* Query the Referer whitelist of the bucket. */
    resp_status = oss_get_bucket_referer(oss_client_options, &bucket, &referer_config, &res
    p_headers);
    if (aos_status_is_ok(resp_status)) {
        printf("get bucket referer succeeded\n");
        aos_list_for_each_entry(oss_referer_t, referer, &referer_config.referer_list, node)
    {
        printf("get referer %s\n", referer->referer.data);
    }
}

```

```

    } else {
        printf("get bucket referer failed\n");
    }
    /* Release the memory pool. This operation releases the memory resources allocated for
the request. */
    aos_pool_destroy(pool);
    /* Release the allocated global resources. */
    aos_http_io_deinitialize();
    return 0;
}

```

Clear the hotlink protection configurations of the bucket

The following code provides an example on how to clear a Referer whitelist of a bucket:

```

#include "oss_api.h"
#include "aos_http_io.h"
const char *endpoint = "<yourEndpoint>";
const char *access_key_id = "<yourAccessKeyId>";
const char *access_key_secret = "<yourAccessKeySecret>";
const char *bucket_name = "<yourBucketName>";
void init_options(oss_request_options_t *options)
{
    options->config = oss_config_create(options->pool);
    /* Use a char* string to initialize the aos_string_t data type. */
    aos_str_set(&options->config->endpoint, endpoint);
    aos_str_set(&options->config->access_key_id, access_key_id);
    aos_str_set(&options->config->access_key_secret, access_key_secret);
    /* Specify whether to use CNAME. The value of 0 indicates that CNAME is not used. */
    options->config->is_cname = 0;
    /* Configure network parameters such as the timeout period. */
    options->ctl = aos_http_controller_create(options->pool, 0);
}
int main(int argc, char *argv[])
{
    /* Call the aos_http_io_initialize method in main() to initialize global resources such
as networks and memory. */
    if (aos_http_io_initialize(NULL, 0) != AOSE_OK) {
        exit(1);
    }
    /* Create a memory pool to manage memory. aos_pool_t is equivalent to apr_pool_t. The c
ode used to create a memory pool is included in the APR library. */
    aos_pool_t *pool;
    /* Create a memory pool. The second parameter is set to NULL. This value indicates that
the pool does not inherit other memory pools. */
    aos_pool_create(&pool, NULL);
    /* Create and initialize options. This parameter includes global configuration informat
ion such as endpoint, access_key_id, access_key_secret, is_cname, and curl. */
    oss_request_options_t *oss_client_options;
    /* Allocate the memory resources in the memory pool to the options. */
    oss_client_options = oss_request_options_create(pool);
    /* Initialize oss_client_options. */
    init_options(oss_client_options);
    /* Initialize the parameters. */
}

```

```

aos_string_t bucket;
aos_table_t *resp_headers = NULL;
aos_status_t *resp_status = NULL;
oss_referer_config_t referer_config;
aos_str_set(&bucket, bucket_name);
aos_list_init(&referer_config.referer_list);
referer_config.allow_empty_referer = 1;
/* Hotlink protection configurations cannot be cleared directly. To clear hotlink protection configurations, you must create a rule that allows an empty Referer field to replace the existing rule. */
resp_status = oss_put_bucket_referer(oss_client_options, &bucket, &referer_config, &resp_headers);
if (aos_status_is_ok(resp_status)) {
    printf("delete bucket referer succeeded\n");
} else {
    printf("delete bucket referer failed\n");
}
/* Release the memory pool. This operation releases the memory resources allocated for the request. */
aos_pool_destroy(pool);
/* Release the allocated global resources. */
aos_http_io_deinitialize();
return 0;
}

```

12.5.11. Static website hosting

You can set your bucket to static website hosting mode. After the configuration takes effect, you can access this static website with the bucket domain and be redirected to a specified index page or error page.

For more information about static website hosting, see [Configure static website hosting](#).

Configure static website hosting

Run the following code to configure static website hosting:

```

#include "oss_api.h"
#include "aos_http_io.h"
const char *endpoint = "<yourEndpoint>";
const char *access_key_id = "<yourAccessKeyId>";
const char *access_key_secret = "<yourAccessKeySecret>";
const char *bucket_name = "<yourBucketName>";
void init_options(oss_request_options_t *options)
{
    options->config = oss_config_create(options->pool);
    /* Use a char* string to initialize aos_string_t. */
    aos_str_set(&options->config->endpoint, endpoint);
    aos_str_set(&options->config->access_key_id, access_key_id);
    aos_str_set(&options->config->access_key_secret, access_key_secret);
    /* Determine whether the CNAME is used. The value 0 indicates that the CNAME is not used. */
    options->config->is_cname = 0;
    /* Configure network parameters such as timeout. */
}

```



```

/* Configure network parameters, such as timeout. */
options->ctl = aos_http_controller_create(options->pool, 0);
}

int main(int argc, char *argv[])
{
    /* Call the aos_http_io_initialize method in main() to initialize global resources, such as networks and memories. */
    if (aos_http_io_initialize(NULL, 0) != AOSE_OK) {
        exit(1);
    }

    /* Memory pool used to manage memories, equivalent to apr_pool_t. The implementation code is included in the apr library. */
    aos_pool_t *pool;

    /* Re-create a memory pool. The second parameter is NULL, indicating that the new pool does not inherit any other memory pool. */
    aos_pool_create(&p, NULL);

    /* Create and initialize options. This parameter includes global configuration information, such as endpoint, access_key_id, access_key_secret, is_cname, and curl. */
    oss_request_options_t *oss_client_options;
    /* Allocate memories in the memory pool to options. */
    oss_client_options = oss_request_options_create(pool);
    /* Initialize oss_client_options. */
    init_options(oss_client_options);
    /* Initialize parameters. */
    aos_string_t bucket;
    aos_table_t *resp_headers = NULL;
    aos_status_t *resp_status = NULL;
    oss_website_config_t website_config;
    aos_str_set(&bucket, bucket_name);
    aos_str_set(&website_config.suffix_str, "index.html");
    aos_str_set(&website_config.key_str, "error.html");
    /* Configure static web hosting: The index page is index.html and the error page is error.html. */
    resp_status = oss_put_bucket_website(oss_client_options, &bucket, &website_config, &resp_headers);
    if (aos_status_is_ok(resp_status)) {
        printf("put bucket website succeeded\n");
    } else {
        printf("put bucket website failed\n");
    }
    /* Release the memory pool, that is, the memories allocated to resources during the request. */
    aos_pool_destroy(pool);
    /* Release allocated global resources. */
    aos_http_io_deinitialize();
    return 0;
}

```

View static website hosting configurations

Run the following code to view static website hosting configurations:

```

#include "oss_api.h"
#include "aos_http_io.h"

```

```

const char *endpoint = "<yourEndpoint>";
const char *access_key_id = "<yourAccessKeyId>";
const char *access_key_secret = "<yourAccessKeySecret>";
const char *bucket_name = "<yourBucketName>";
void init_options(oss_request_options_t *options)
{
    options->config = oss_config_create(options->pool);
    /* Use a char* string to initialize aos_string_t. */
    aos_str_set(&options->config->endpoint, endpoint);
    aos_str_set(&options->config->access_key_id, access_key_id);
    aos_str_set(&options->config->access_key_secret, access_key_secret);
    /* Determine whether the CNAME is used. The value 0 indicates that the CNAME is not use
d. */
    options->config->is_cname = 0;
    /* Configure network parameters, such as timeout. */
    options->ctl = aos_http_controller_create(options->pool, 0);
}
int main(int argc, char *argv[])
{
    /* Call the aos_http_io_initialize method in main() to initialize global resources, suc
h as networks and memories. */
    if (aos_http_io_initialize(NULL, 0) != AOSE_OK) {
        exit(1);
    }
    /* Memory pool used to manage memories, equivalent to apr_pool_t. The implementation co
de is included in the apr library. */
    aos_pool_t *pool;
    /* Re-create a memory pool. The second parameter is NULL, indicating that the new pool
does not inherit any other memory pool. */
    aos_pool_create(&pool, NULL);
    /* Create and initialize options. This parameter includes global configuration informat
ion, such as endpoint, access_key_id, acces_key_secret, is_cname, and curl. */
    oss_request_options_t *oss_client_options;
    /* Allocate memories in the memory pool to options. */
    oss_client_options = oss_request_options_create(pool);
    /* Initialize oss_client_options. */
    init_options(oss_client_options);
    /* Initialization parameters. */
    aos_string_t bucket;
    aos_table_t *resp_headers = NULL;
    aos_status_t *resp_status = NULL;
    oss_website_config_t website_config;
    aos_str_set(&bucket, bucket_name);
    /* View static web hosting configurations. */
    resp_status = oss_get_bucket_website(oss_client_options, &bucket, &website_config, &res
p_headers);
    if (aos_status_is_ok(resp_status)) {
        printf("get bucket website succeeded\n");
        printf("website_config: %s %s\n", website_config.suffix_str.data, website_config.
key_str.data);
    } else {
        printf("get bucket website failed\n");
    }
    /* Release the memory pool, that is, the memories allocated to resources during the req
uest. */
}

```

```

aos_pool_destroy(pool);
/* Release the allocated global resources. */
aos_http_io_deinitialize();
return 0;
}

```

Delete static website hosting configurations

Run the following code to delete static website hosting configurations:

```

#include "oss_api.h"
#include "aos_http_io.h"
const char *endpoint = "<yourEndpoint>";
const char *access_key_id = "<yourAccessKeyId>";
const char *access_key_secret = "<yourAccessKeySecret>";
const char *bucket_name = "<yourBucketName>";
void init_options(oss_request_options_t *options)
{
    options->config = oss_config_create(options->pool);
    /* Use a char* string to initialize aos_string_t. */
    aos_str_set(&options->config->endpoint, endpoint);
    aos_str_set(&options->config->access_key_id, access_key_id);
    aos_str_set(&options->config->access_key_secret, access_key_secret);
    /* Determine whether the CNAME is used. The value 0 indicates that the CNAME is not use
d. */
    options->config->is_cname = 0;
    /* Configure network parameters, such as timeout. */
    options->ctl = aos_http_controller_create(options->pool, 0);
}
int main(int argc, char *argv[])
{
    /* Call the aos_http_io_initialize method in main() to initialize global resources, suc
h as networks and memories. */
    if (aos_http_io_initialize(NULL, 0) != AOSE_OK) {
        exit(1);
    }
    /* Memory pool used to manage memories, equivalent to apr_pool_t. The implementation co
de is included in the apr library. */
    aos_pool_t *pool;
    /* Re-create a memory pool. The second parameter is NULL, indicating that the new pool
does not inherit any other memory pool. */
    aos_pool_create(&pool, NULL);
    /* Create and initialize options. This parameter includes global configuration informat
ion, such as endpoint, access_key_id, acces_key_secret, is_cname, and curl. */
    oss_request_options_t *oss_client_options;
    /* Allocate the memories in the memory pool to options. */
    oss_client_options = oss_request_options_create(pool);
    /* Initialize oss_client_options. */
    init_options(oss_client_options);
    /* Initialize parameters. */
    aos_string_t bucket;
    aos_table_t *resp_headers = NULL;
    aos_status_t *resp_status = NULL;
    aos_str_set(&bucket, bucket_name);

```

```
aos_str_set(&bucket, bucket_name);
/* Delete static website hosting configurations. */
resp_status = oss_delete_bucket_website(oss_client_options, &bucket, &resp_headers);
if (aos_status_is_ok(resp_status)) {
    printf("delete bucket website succeeded\n");
} else {
    printf("delete bucket website failed\n");
}
/* Release the memory pool, that is, the memories allocated to resources during the request. */
aos_pool_destroy(pool);
/* Release the allocated global resources. */
aos_http_io_deinitialize();
return 0;
}
```

12.6. Upload objects

12.6.1. Overview

In OSS, an object is the basic unit for data operations. OSS C SDK provides the following file upload methods:

- **Simple upload**: supports the upload of a file up to 5 GB in size. It includes streaming upload and file upload.
- **Append upload**: supports the upload of a file up to 5 GB in size.
- **Resumable upload**: supports the upload of a file up to 48.8 TB in size. It applies to the upload of large files. It also supports concurrent upload and you can define the size of each part.
- **Multipart upload**: supports the upload of a file up to 48.8 TB in size. It applies to the upload of large files.

 **Note** For more information about the usage scenarios of each upload method, see “Upload files” in OSS Developer Guide.

During the file upload, you can set **Object Meta**, and check the upload progress in **upload progress bar**. After you complete the file upload, you can perform **upload callback**.

12.6.2. Simple upload

This topic describes how to use simple upload.

For the complete simple upload code, visit [GitHub](#).

Upload data from memory

The following code provides an example on how to upload data from memory:

```
#include "oss_api.h"
#include "aos_http_io.h"
const char *endpoint = "<yourEndpoint>";
const char *access_key_id = "<yourAccessKeyId>";
const char *access_key_secret = "<yourAccessKeySecret>";
```

```

const char *bucket_name = "<yourBucketName>";
const char *object_name = "<yourObjectName>";
const char *object_content = "More than just cloud.";
void init_options(oss_request_options_t *options)
{
    options->config = oss_config_create(options->pool);
    /* Use a char* string to initialize aos_string_t. */
    aos_str_set(&options->config->endpoint, endpoint);
    aos_str_set(&options->config->access_key_id, access_key_id);
    aos_str_set(&options->config->access_key_secret, access_key_secret);
    /* Specify whether to use CNAME. A value of 0 indicates that CNAME is not used. */
    options->config->is_cname = 0;
    /* Configure network parameters such as timeout periods. */
    options->ctl = aos_http_controller_create(options->pool, 0);
}
int main(int argc, char *argv[])
{
    /* Call the aos_http_io_initialize method in main() to initialize global resources such
as networks and memory. */
    if (aos_http_io_initialize(NULL, 0) != AOSE_OK) {
        exit(1);
    }

    /* Create a memory pool to manage memory. aos_pool_t is equivalent to apr_pool_t. The c
ode used to create a memory pool is included in the APR library. */
    aos_pool_t *pool;
    /* Create a memory pool. The second parameter is NULL. This value indicates that the po
ol does not inherit other memory pool. */
    aos_pool_create(&pool, NULL);
    /* Create and initialize options. This parameter includes global configuration informat
ion such as endpoint, access_key_id, access_key_secret, is_cname, and curl. */
    oss_request_options_t *oss_client_options;
    /* Allocate the memory resources in the memory pool to the options. */
    oss_client_options = oss_request_options_create(pool);
    /* Initialize oss_client_options. */
    init_options(oss_client_options);
    /* Initialize the parameters. */
    aos_string_t bucket;
    aos_string_t object;
    aos_list_t buffer;
    aos_buf_t *content = NULL;
    aos_table_t *headers = NULL;
    aos_table_t *resp_headers = NULL;
    aos_status_t *resp_status = NULL;
    aos_str_set(&bucket, bucket_name);
    aos_str_set(&object, object_name);
    aos_list_init(&buffer);
    content = aos_buf_pack(oss_client_options->pool, object_content, strlen(object_content)
);
    aos_list_add_tail(&content->node, &buffer);
    /* Upload the object. */
    resp_status = oss_put_object_from_buffer(oss_client_options, &bucket, &object, &buffer,
headers, &resp_headers);
    /* Determine whether the object is uploaded. */
    if (aos_status_is_ok(resp_status)) {

```

```

        printf("put object from buffer succeeded\n");
    } else {
        printf("put object from buffer failed\n");
    }
    /* Release the memory pool, which releases memory resources allocated for the request.
    */
    aos_pool_destroy(pool);
    /* Release the allocated global resources. */
    aos_http_io_deinitialize();
    return 0;
}

```

Encrypt objects to upload from a file

The following code provides an example on how to upload a file to OSS:

```

#include "oss_api.h"
#include "aos_http_io.h"
const char *endpoint = "<yourEndpoint>";
const char *access_key_id = "<yourAccessKeyId>";
const char *access_key_secret = "<yourAccessKeySecret>";
const char *bucket_name = "<yourBucketName>";
const char *object_name = "<yourObjectName>";
const char *local_filename = "<yourLocalFilename>";
void init_options(oss_request_options_t *options)
{
    options->config = oss_config_create(options->pool);
    /* Use a char* string to initialize aos_string_t. */
    aos_str_set(&options->config->endpoint, endpoint);
    aos_str_set(&options->config->access_key_id, access_key_id);
    aos_str_set(&options->config->access_key_secret, access_key_secret);
    /* Specify whether to use CNAME. A value of 0 indicates that CNAME is not used. */
    options->config->is_cname = 0;
    /* Configure network parameters such as timeout periods. */
    options->ctl = aos_http_controller_create(options->pool, 0);
}
int main(int argc, char *argv[])
{
    /* Call the aos_http_io_initialize method in main() to initialize global resources such
    as networks and memory. */
    if (aos_http_io_initialize(NULL, 0) != AOSE_OK) {
        exit(1);
    }
    /* Create a memory pool to manage memory. aos_pool_t is equivalent to apr_pool_t. The c
    ode used to create a memory pool is included in the APR library. */
    aos_pool_t *pool;
    /* Create a memory pool. The second parameter is NULL. This value indicates that the po
    ol does not inherit other memory pool. */
    aos_pool_create(&pool, NULL);
    /* Create and initialize options. This parameter includes global configuration informat
    ion such as endpoint, access_key_id, access_key_secret, is_cname, and curl. */
    oss_request_options_t *oss_client_options;
    /* Allocate the memory resources in the memory pool to the options. */
    oss_client_options = oss_request_options_create(pool);
}

```

```

/* Initialize oss_client_options. */
init_options(oss_client_options);
/* Initialize the parameters. */
aos_string_t bucket;
aos_string_t object;
aos_string_t file;
aos_table_t *headers = NULL;
aos_table_t *resp_headers = NULL;
aos_status_t *resp_status = NULL;
aos_str_set(&bucket, bucket_name);
aos_str_set(&object, object_name);
aos_str_set(&file, local_filename);
/* Upload the object. */
resp_status = oss_put_object_from_file(oss_client_options, &bucket, &object, &file, headers, &resp_headers);
/* Determine whether the object is uploaded. */
if (aos_status_is_ok(resp_status)) {
    printf("put object from file succeeded\n");
} else {
    printf("put object from file failed\n");
}
/* Release the memory pool, which releases memory resources allocated for the request. */
aos_pool_destroy(pool);
/* Release the allocated global resources. */
aos_http_io_deinitialize();
return 0;
}

```

FAQ: How can I upload objects whose names contain Chinese characters?

On Windows systems, you can use UTF-8 to encode the object names that contain letters and upload the object. The following code provides an example on how to upload objects whose names contain Chinese characters:

```

#include "oss_api.h"
#include "aos_http_io.h"
#include <wchar.h>
const char *endpoint = "<yourEndpoint>";
const char *access_key_id = "<yourAccessKeyId>";
const char *access_key_secret = "<yourAccessKeySecret>";
const char *bucket_name = "<yourBucketName>";
const char *object_name = "<yourObjectName>";
const char *local_filename_ch = "c:\\test.txt";
void init_options(oss_request_options_t *options)
{
    options->config = oss_config_create(options->pool);
    /* Use a char* string to initialize aos_string_t. */
    aos_str_set(&options->config->endpoint, endpoint);
    aos_str_set(&options->config->access_key_id, access_key_id);
    aos_str_set(&options->config->access_key_secret, access_key_secret);
    /* Specify whether to use CNAME. A value of 0 indicates that CNAME is not used. */
    options->config->is_cname = 0;
}

```

```

options->config->is_cname = 0;
/* Configure network parameters such as timeout periods. */
options->ctl = aos_http_controller_create(options->pool, 0);
}
char* multibyte_to_utf8(const char * ch, char *str, int size)
{
    if (!ch || !str || size <= 0)
        return NULL;
    int chlen = strlen(ch);
    int multi_byte_cnt = 0;
    for (int i = 0; i < chlen - 1; i++) {
        if ((ch[i] & 0x80) && (ch[i + 1] & 0x80)) {
            i++;
            multi_byte_cnt++;
        }
    }
    if (multi_byte_cnt == 0)
        return ch;
    int len = MultiByteToWideChar(CP_ACP, 0, ch, -1, NULL, 0);
    if (len <= 0)
        return NULL;
    wchar_t *wch = malloc(sizeof(wchar_t) * (len + 1));
    if (!wch)
        return NULL;
    wmemset(wch, 0, len + 1);
    MultiByteToWideChar(CP_ACP, 0, ch, -1, wch, len);
    len = WideCharToMultiByte(CP_UTF8, 0, wch, -1, NULL, 0, NULL, NULL);
    if ((len <= 0) || (size < len + 1)) {
        free(wch);
        return NULL;
    }
    memset(str, 0, len + 1);
    WideCharToMultiByte(CP_UTF8, 0, wch, -1, str, len, NULL, NULL);
    free(wch);
    return str;
}
int main(int argc, char argv[])
{
    /* Call the aos_http_io_initialize method in main() to initialize global resources such
    as networks and memory. */
    if (aos_http_io_initialize(NULL, 0) != AOSE_OK) {
        exit(1);
    }
    /* Create a memory pool to manage memory. aos_pool_t is equivalent to apr_pool_t. The co
    de used to create a memory pool is included in the APR library. */
    aos_pool_t pool;
    /* Create a memory pool. The second parameter is NULL. This value indicates that the poo
    l does not inherit other memory pool. */
    aos_pool_create(&pool, NULL);
    /* Create and initialize options. This parameter includes global configuration informat
    ion such as endpoint, access_key_id, access_key_secret, is_cname, and curl. */
    oss_request_options_t oss_client_options;
    /* Allocate the memory resources in the memory pool to the options. */
    oss_client_options = oss_request_options_create(pool);
    /* Initialize oss client options. */

```



```
init_options(oss_client_options);
/* Initialize the parameters. */
aos_string_t bucket;
aos_string_t object;
aos_string_t file;
aos_table_t *headers = NULL;
aos_table_t *resp_headers = NULL;
aos_status_t *resp_status = NULL;
aos_str_set(&bucket, bucket_name);
aos_str_set(&object, object_name);
/* Use UTF-8 to encode the object names that contain multibyte Chinese characters. */
char local_filename_buf[1024];
char * local_filename = multibyte_to_utf8(local_filename_ch, local_filename_buf, 1024);
aos_str_set(&file, local_filename);
/* Upload the object. */
resp_status = oss_put_object_from_file(oss_client_options, &bucket, &object, &file, headers, &resp_headers);
/* Determine whether the object is uploaded. */
if (aos_status_is_ok(resp_status)) {
    printf("put object from file succeeded\n");
} else {
    printf("put object from file failed\n");
}
/* Release the memory pool, which releases memory resources allocated for the request. */
aos_pool_destroy(pool);
/* Release the allocated global resources. */
aos_http_io_deinitialize();
return 0;
}
```

12.6.3. Append upload

You can call the AppendObject operation to append content to existing appendable objects.

Usage notes

- If the object to which you want to append content does not exist, an appendable object is created when you call the AppendObject operation.
- When you call the AppendObject operation to append content to an object, you may encounter a few situations: If the object is an appendable object, and the specified position from which the append operation starts is equal to the current object length, the content is appended to the end of the object. If the specified position from which the append operation starts is not equal to the current object length, a `PositionNotEqualToLength` exception is thrown. If the object is not an appendable object, an `ObjectNotAppendable` exception is thrown.
- The CopyObject operation cannot be performed on appendable objects.

For more information about how to perform append upload, see [AppendObject](#).

Upload data from the local memory by using append upload

The following code provides an example on how to upload data from the local memory to a specified bucket by using append upload:

```
#include "oss_api.h"
#include "aos_http_io.h"
/* The endpoint of the China (Hangzhou) region is used in this example. Specify the actual endpoint. */
const char *endpoint = "<yourEndpoint>";
/* Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log on to OSS because the account has permissions on all API operations. We recommend that you use your RAM user's credentials to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console. */
const char *access_key_id = "<yourAccessKeyId>";
const char *access_key_secret = "<yourAccessKeySecret>";
/* Configure the bucket name. */
const char *bucket_name = "<yourBucketName>";
/* Configure the object name. The object name indicates the complete path of the object excluding the bucket name. Example: example/folder/abc.jpg. */
const char *object_name = "<yourObjectName>";
const char *object_content = "More than just cloud.";
void init_options(oss_request_options_t *options)
{
    options->config = oss_config_create(options->pool);
    /* Use a char* string to initialize aos_string_t. */
    aos_str_set(&options->config->endpoint, endpoint);
    aos_str_set(&options->config->access_key_id, access_key_id);
    aos_str_set(&options->config->access_key_secret, access_key_secret);
    /* Specify whether to use CNAME. A value of 0 indicates that CNAME is not used. */
    options->config->is_cname = 0;
    /* Configure network parameters such as timeout periods. */
    options->ctl = aos_http_controller_create(options->pool, 0);
}
int main(int argc, char *argv[])
{
    /* Call the aos_http_io_initialize method in main() to initialize global resources such as networks and memory. */
    if (aos_http_io_initialize(NULL, 0) != AOSE_OK) {
        exit(1);
    }
    /* Create a memory pool to manage memory. aos_pool_t is equivalent to apr_pool_t. The code used to create a memory pool is included in the APR library. */
    aos_pool_t *pool;
    /* Create a memory pool. The second parameter is NULL. This value indicates that the pool does not inherit any other memory pool. */
    aos_pool_create(&pool, NULL);
    /* Create and initialize options. This parameter includes global configuration information such as endpoint, access_key_id, access_key_secret, is_cname, and curl. */
    oss_request_options_t *oss_client_options;
    /* Allocate the memory resources in the memory pool to the options. */
    oss_client_options = oss_request_options_create(pool);
    /* Initialize oss_client_options. */
    init_options(oss_client_options);
    /* Initialize the parameters. */
    aos_string_t bucket;
```

```

aos_string_t object;
aos_list_t buffer;
int64_t position = 0;
char *next_append_position = NULL;
aos_buf_t *content = NULL;
aos_table_t *headers1 = NULL;
aos_table_t *headers2 = NULL;
aos_table_t *resp_headers = NULL;
aos_status_t *resp_status = NULL;
aos_str_set(&bucket, bucket_name);
aos_str_set(&object, object_name);
headers1 = aos_table_make(pool, 0);
/* Obtain the position from which the first append operation starts. */
resp_status = oss_head_object(oss_client_options, &bucket, &object, headers1, &resp_headers);
if (aos_status_is_ok(resp_status)) {
    next_append_position = (char*)(apr_table_get(resp_headers, "x-oss-next-append-position"));
    position = atoi(next_append_position);
}
/* Perform the append operation. */
headers2 = aos_table_make(pool, 0);
aos_list_init(&buffer);
content = aos_buf_pack(pool, object_content, strlen(object_content));
aos_list_add_tail(&content->node, &buffer);
resp_status = oss_append_object_from_buffer(oss_client_options, &bucket, &object, position, &buffer, headers2, &resp_headers);
if (aos_status_is_ok(resp_status)) {
    printf("append object from buffer succeeded\n");
} else {
    printf("append object from buffer failed\n");
}
/* Release the memory pool, which releases memory resources allocated for the request. */
aos_pool_destroy(pool);
/* Release the allocated global resources. */
aos_http_io_deinitialize();
return 0;
}

```

Upload a local file by using append upload

The following code provides an example on how to upload a local file to a specified bucket by using append upload:

```

#include "oss_api.h"
#include "aos_http_io.h"
/* The endpoint of the China (Hangzhou) region is used in this example. Specify the actual endpoint. */
const char *endpoint = "<yourEndpoint>";
/* Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to log on to OSS because the account has permissions on all API operations. We recommend that you use your RAM user's credentials to call API operations or perform routine operations and maintenance. To create a RAM user, log on to the RAM console. */

```

```

const char *access_key_id = "<yourAccessKeyId>";
const char *access_key_secret = "<yourAccessKeySecret>";
/* Configure the bucket name. */
const char *bucket_name = "<yourBucketName>";
/* Configure the object name. The object name indicates the complete path of the object excluding the bucket name. Example: example/folder/abc.jpg. */
const char *object_name = "<yourObjectName>";
/* Configure the name of the local file you want to upload. Specify the complete path of the local file. Example: /users/local/abc.jpg. */
const char *local_filename = "<yourLocalFilename>";
void init_options(oss_request_options_t *options)
{
    options->config = oss_config_create(options->pool);
    /* Use a char* string to initialize aos_string_t. */
    aos_str_set(&options->config->endpoint, endpoint);
    aos_str_set(&options->config->access_key_id, access_key_id);
    aos_str_set(&options->config->access_key_secret, access_key_secret);
    /* Specify whether to use CNAME. A value of 0 indicates that CNAME is not used. */
    options->config->is_cname = 0;
    /* Configure network parameters such as timeout periods. */
    options->ctl = aos_http_controller_create(options->pool, 0);
}
int main(int argc, char *argv[])
{
    /* Call the aos_http_io_initialize method in main() to initialize global resources such as networks and memory. */
    if (aos_http_io_initialize(NULL, 0) != AOSE_OK) {
        exit(1);
    }
    /* Create a memory pool to manage memory. aos_pool_t is equivalent to apr_pool_t. The code used to create a memory pool is included in the APR library. */
    aos_pool_t *pool;
    /* Create a memory pool. The second parameter is NULL. This value indicates that the pool does not inherit any other memory pool. */
    aos_pool_create(&pool, NULL);
    /* Create and initialize options. This parameter includes global configuration information such as endpoint, access_key_id, access_key_secret, is_cname, and curl. */
    oss_request_options_t *oss_client_options;
    /* Allocate the memory resources in the memory pool to the options. */
    oss_client_options = oss_request_options_create(pool);
    /* Initialize oss_client_options. */
    init_options(oss_client_options);
    /* Initialize the parameters. */
    aos_string_t bucket;
    aos_string_t object;
    aos_string_t file;
    int64_t position = 0;
    char *next_append_position = NULL;
    aos_table_t *headers1 = NULL;
    aos_table_t *headers2 = NULL;
    aos_table_t *resp_headers = NULL;
    aos_status_t *resp_status = NULL;
    aos_str_set(&bucket, bucket_name);
    aos_str_set(&object, object_name);

```

```
aos_str_set(&file, local_filename);
/* Obtain the position from which the first append operation starts. */
resp_status = oss_head_object(oss_client_options, &bucket, &object, headers1, &resp_headers);
if(aos_status_is_ok(resp_status)) {
    next_append_position = (char*)(apr_table_get(resp_headers, "x-oss-next-append-position"));
    position = atoi(next_append_position);
}
/* Perform the append operation. */
resp_status = oss_append_object_from_file(oss_client_options, &bucket, &object, position, &file, headers2, &resp_headers);
/* Determine whether the append operation is successful. */
if (aos_status_is_ok(resp_status)) {
    printf("append object from file succeeded\n");
} else {
    printf("append object from file failed\n");
}
/* Release the memory pool, which releases memory resources allocated for the request. */
aos_pool_destroy(pool);
/* Release the allocated global resources. */
aos_http_io_deinitialize();
return 0;
}
```

12.6.4. Resumable upload

In resumable upload, the object is divided into multiple parts, which are then uploaded separately. After all the parts are uploaded, they are merged into a complete object to finish the upload.

Current upload progress is recorded into a checkpoint file during the upload. If a part fails to be uploaded during the process, the object is uploaded from the part recorded in the checkpoint file when the upload restarts, that is, the resumable upload. After the upload is complete, the checkpoint is deleted.

You can use the `oss_resumable_clt_params_t` method to achieve resumable upload. Common parameters used in this method are listed as follows:

Parameter	Description
thread_num	The number of concurrent threads, which is 1 by default.
enable_checkpoint	Whether to enable resumable upload, which is disabled by default.
partSize	The part size, ranging from 100 KB to 5 GB.
checkpoint_path	The path that the checkpoint file is stored, which is {upload_file_path}.cp by default.

You can run the following code to perform resumable upload:

```

#include "oss_api.h"
#include "aos_http_io.h"
const char *endpoint = "<yourEndpoint>";
const char *access_key_id = "<yourAccessKeyId>";
const char *access_key_secret = "<yourAccessKeySecret>";
const char *bucket_name = "<yourBucketName>";
const char *object_name = "<yourObjectName>";
const char *local_filename = "<yourLocalFilename>";
void init_options(oss_request_options_t *options)
{
    options->config = oss_config_create(options->pool);
    /* Use a char* string to initialize the aos_string_t type. */
    aos_str_set(&options->config->endpoint, endpoint);
    aos_str_set(&options->config->access_key_id, access_key_id);
    aos_str_set(&options->config->access_key_secret, access_key_secret);
    /* Determine whether the CNAME is used. 0 indicates that it is not used. */
    options->config->is_cname = 0;
    /* Used to configure network parameters, such as timeout. */
    options->ctl = aos_http_controller_create(options->pool, 0);
}
int main(int argc, char *argv[])
{
    /* Call the aos_http_io_initialize method in main() to initialize global resources, such as networks and memories. */
    if (aos_http_io_initialize(NULL, 0) != AOSE_OK) {
        exit(1);
    }
    /* Memory pool used to manage memories, which is equivalent to apr_pool_t. The implementation code is included in the apr library. */
    aos_pool_t *pool;
    /* Re-create a new memory pool. The second parameter is NULL, indicating that it does not inherit from any other memory pools. */
    aos_pool_create(&pool, NULL);
    /* Create and initialize options. This parameter mainly includes global configuration information, such as endpoint, access_key_id, access_key_secret, is_cname, and curl. */
    oss_request_options_t *oss_client_options;
    /* Allocate memories in the memory pool to options. */
    oss_client_options = oss_request_options_create(pool);
    /* Use oss_client_options to initialize client options */
    init_options(oss_client_options);
    /* Initialization parameters. */
    aos_string_t bucket;
    aos_string_t object;
    aos_string_t file;
    aos_list_t resp_body;
    aos_table_t *headers = NULL;
    aos_table_t *resp_headers = NULL;
    aos_status_t *resp_status = NULL;
    oss_resumable_clt_params_t *clt_params;
    aos_str_set(&bucket, bucket_name);
    aos_str_set(&object, object_name);
    aos_str_set(&file, local_filename);
    aos_list_init(&resp_body);
    /* Resumable upload. */

```

```
    clt_params = oss_create_resumable_clt_params_content(pool, 1024 * 100, 3, AOS_TRUE, NULL);  
    resp_status = oss_resumable_upload_file(oss_client_options, &bucket, &object, &file, headers, NULL, clt_params, NULL, &resp_headers, &resp_body);  
    if (aos_status_is_ok(resp_status)) {  
        printf("resumable upload succeeded\n");  
    } else {  
        printf("resumable upload failed\n");  
    }  
    /* Release the memory pool, that is, memories allocated to resources during the request . */  
    aos_pool_destroy(pool);  
    /* Release allocated global resources. */  
    aos_http_io_deinitialize();  
    return 0;  
}
```

12.6.5. Multipart upload

By using the multipart upload feature provided by OSS, you can split a large object into multiple parts and upload them separately. After all parts are uploaded, call the CompleteMultipartUpload operation to combine these parts into a single object to implement resumable upload.

Multipart upload process

To perform multipart upload, follow these steps:

1. Initialize a multipart upload task.

Call the `oss_init_multipart_upload` method to create a globally unique upload ID in OSS.

2. Start multipart upload.

Call the `oss_upload_part_from_file` method to upload parts.

Note

- For parts that share an upload ID, `partNumber` identifies the relative position of a part in the entire object. If you have uploaded a part and used its `partNumber` again to upload another part, the latter part overwrites the former part.
- OSS places the MD5 value of part data in the `ETag` header and returns the MD5 value to the user.
- OSS calculates the MD5 value of uploaded data and compares it with the MD5 value calculated by the SDK. If the two values are different, the `InvalidDigest` error code is returned.

3. Complete the multipart upload task.

After you upload all parts, you can call the `oss_complete_multipart_upload` method to combine the parts into a complete object.

Complete sample code

The following code provides a complete example that describes the process of multipart upload:

```

#include "oss_api.h"
#include "aos_http_io.h"
#include <sys/stat.h>
const char *endpoint = "<yourEndpoint>";
const char *access_key_id = "<yourAccessKeyId>";
const char *access_key_secret = "<yourAccessKeySecret>";
const char *bucket_name = "<yourBucketName>";
const char *object_name = "<yourObjectName>";
const char *local_filename = "<yourLocalFilename>";
void init_options(oss_request_options_t *options)
{
    options->config = oss_config_create(options->pool);
    /* Use a char* string to initialize the aos_string_t data type. */
    aos_str_set(&options->config->endpoint, endpoint);
    aos_str_set(&options->config->access_key_id, access_key_id);
    aos_str_set(&options->config->access_key_secret, access_key_secret);
    /* Determine whether CNAME is used. A value of 0 indicates that CNAME is not used. */
    options->config->is_cname = 0;
    /* Configure network parameters such as timeout. */
    options->ctl = aos_http_controller_create(options->pool, 0);
}
int64_t get_file_size(const char *file_path)
{
    int64_t filesize = -1;
    struct stat statbuff;
    if(stat(file_path, &statbuff) < 0){
        return filesize;
    } else {
        filesize = statbuff.st_size;
    }
    return filesize;
}
int main(int argc, char *argv[])
{
    /* Call aos_http_io_initialize in main() to initialize global resources such as the net
work and memory. */
    if (aos_http_io_initialize(NULL, 0) != AOSE_OK) {
        exit(1);
    }
    /* Create a memory pool to manage memory resources. The implementation code is included
in the APR library. aos_pool_t is equivalent to apr_pool_t. */
    aos_pool_t *pool;
    /* Create a new memory pool. The second parameter value is NULL. This value indicates t
hat the pool does not inherit any other memory pool. */
    aos_pool_create(&pool, NULL);
    /* Create and initialize options. This parameter includes global configuration informat
ion such as endpoint, access_key_id, access_key_secret, is_cname, and curl. */
    oss_request_options_t *oss_client_options;
    /* Allocate memory resources in the memory pool to options. */
    oss_client_options = oss_request_options_create(pool);
    /* Call oss_client_options to initialize the client options. */
    init_options(oss_client_options);
    /* Initialize parameters. */
    aos_string_t bucket;

```



```

aos_string_t object;
oss_upload_file_t *upload_file = NULL;
aos_string_t upload_id;
aos_table_t *headers = NULL;
aos_table_t *complete_headers = NULL;
aos_table_t *resp_headers = NULL;
aos_status_t *resp_status = NULL;
aos_str_set(&bucket, bucket_name);
aos_str_set(&object, object_name);
aos_str_null(&upload_id);
headers = aos_table_make(pool, 1);
complete_headers = aos_table_make(pool, 1);
int part_num = 1;
/* Initialize a multipart upload task and obtain an upload ID. */
resp_status = oss_init_multipart_upload(oss_client_options, &bucket, &object, &upload_id, headers, &resp_headers);
/* Determine whether the multipart upload task is initialized. */
if (aos_status_is_ok(resp_status)) {
    printf("Init multipart upload succeeded, upload_id: %s\n",
           upload_id.len, upload_id.data);
} else {
    printf("Init multipart upload failed, upload_id: %s\n",
           upload_id.len, upload_id.data);
}
/* Upload the parts. */
int64_t file_length = 0;
int64_t pos = 0;
aos_list_t complete_part_list;
oss_complete_part_content_t* complete_content = NULL;
char* part_num_str = NULL;
char* etag = NULL;
aos_list_init(&complete_part_list);
file_length = get_file_size(local_filename);
while(pos < file_length) {
    upload_file = oss_create_upload_file(pool);
    aos_str_set(&upload_file->filename, local_filename);
    upload_file->file_pos = pos;
    pos += 100 * 1024;
    upload_file->file_last = pos < file_length ? pos : file_length;
    resp_status = oss_upload_part_from_file(oss_client_options, &bucket, &object, &upload_id, part_num++, upload_file, &resp_headers);
    /* Save the part numbers and ETags. */
    complete_content = oss_create_complete_part_content(pool);
    part_num_str = apr_psprintf(pool, "%d", part_num-1);
    aos_str_set(&complete_content->part_number, part_num_str);
    etag = apr_pstrdup(pool,
                       (char*)apr_table_get(resp_headers, "ETag"));
    aos_str_set(&complete_content->etag, etag);
    aos_list_add_tail(&complete_content->node, &complete_part_list);
    if (aos_status_is_ok(resp_status)) {
        printf("Multipart upload part from file succeeded\n");
    } else {
        printf("Multipart upload part from file failed\n");
    }
}

```

```

    }
    /* Complete the multipart upload task. */
    resp_status = oss_complete_multipart_upload(oss_client_options, &bucket, &object, &upload_id,
        &complete_part_list, complete_headers, &resp_headers);
    /* Determine whether the multipart upload is complete. */
    if (aos_status_is_ok(resp_status)) {
        printf("Complete multipart upload from file succeeded, upload_id:%.s\n",
            upload_id.len, upload_id.data);
    } else {
        printf("Complete multipart upload from file failed\n");
    }
    /* Release the memory pool. The memory allocated to various resources used for the request is released. */
    aos_pool_destroy(pool);
    /* Release the allocated global resources. */
    aos_http_io_deinitialize();
    return 0;
}

```

Obtain the uploaded parts. You must provide the ETag value of each part to call `oss_complete_multipart_upload`. You can use one of the following methods to obtain the ETag value:

- The ETag of a part is included in the returned result when the part is uploaded. You can save and use this ETag value.
- You can call `oss_list_upload_part` to obtain the ETag of an uploaded part. In this example, the first method is used.

Cancel a multipart upload task

You can use the `ossClient.abortMultipartUpload` method to cancel a multipart upload task. If a multipart upload task is canceled, the upload ID can no longer be used to perform further operations. The parts uploaded with that upload ID are also deleted.

The following code provides an example on how to cancel a multipart upload task:

```

#include "oss_api.h"
#include "aos_http_io.h"
const char *endpoint = "<yourEndpoint>";
const char *access_key_id = "<yourAccessKeyId>";
const char *access_key_secret = "<yourAccessKeySecret>";
const char *bucket_name = "<yourBucketName>";
const char *object_name = "<yourObjectName>";
void init_options(oss_request_options_t *options)
{
    options->config = oss_config_create(options->pool);
    /* Use a char* string to initialize the aos_string_t data type. */
    aos_str_set(&options->config->endpoint, endpoint);
    aos_str_set(&options->config->access_key_id, access_key_id);
    aos_str_set(&options->config->access_key_secret, access_key_secret);
    /* Determine whether CNAME is used. A value of 0 indicates that CNAME is not used. */
    options->config->is_cname = 0;
    /* Configure network parameters such as timeout. */
    options->ctl = aos_http_controller_create(options->pool, 0);
}

```

```

}
int main(int argc, char *argv[])
{
    /* Call aos_http_io_initialize in main() to initialize global resources such as the net
work and memory. */
    if (aos_http_io_initialize(NULL, 0) != AOSE_OK) {
        exit(1);
    }
    /* Create a memory pool to manage memory resources. The implementation code is included
in the APR library. aos_pool_t is equivalent to apr_pool_t. */
    aos_pool_t *pool;
    /* Create a new memory pool. The second parameter value is NULL. This value indicates t
hat the pool does not inherit any other memory pool. */
    aos_pool_create(&pool, NULL);
    /* Create and initialize options. This parameter includes global configuration informat
ion such as endpoint, access_key_id, access_key_secret, is_cname, and curl. */
    oss_request_options_t *oss_client_options;
    /* Allocate memory resources in the memory pool to options. */
    oss_client_options = oss_request_options_create(pool);
    /* Call oss_client_options to initialize the client options. */
    init_options(oss_client_options);
    /* Initialize parameters. */
    aos_string_t bucket;
    aos_string_t object;
    aos_string_t upload_id;
    aos_table_t *headers = NULL;
    aos_table_t *resp_headers = NULL;
    aos_status_t *resp_status = NULL;
    aos_str_set(&bucket, bucket_name);
    aos_str_set(&object, object_name);
    aos_str_null(&upload_id);
    /* Initialize a multipart upload task and obtain an upload ID. */
    resp_status = oss_init_multipart_upload(oss_client_options, &bucket, &object, &upload_i
d, headers, &resp_headers);
    /* Determine whether the multipart upload task is initialized. */
    if (aos_status_is_ok(resp_status)) {
        printf("Init multipart upload succeeded, upload_id: %s\n",
            upload_id.len, upload_id.data);
    } else {
        printf("Init multipart upload failed, upload_id: %s\n",
            upload_id.len, upload_id.data);
    }
    /* Cancel the multipart upload task. */
    resp_status = oss_abort_multipart_upload(oss_client_options, &bucket, &object, &upload_
id, &resp_headers);
    /* Determine whether the multipart upload task is canceled. */
    if (aos_status_is_ok(resp_status)) {
        printf("Abort multipart upload succeeded, upload_id: %s\n",
            upload_id.len, upload_id.data);
    } else {
        printf("Abort multipart upload failed\n");
    }
    /* Release the memory pool. The memory allocated to various resources used for the requ
est is released. */
    aos_pool_destroy(pool);
}

```

```

    /* Release the allocated global resources. */
    aos_http_io_deinitialize();
    return 0;
}

```

12.6.6. Upload progress bars

You can use progress bars to indicate the upload or download progress.

For the complete code of upload progress bar, see [GitHub](#).

The following code is used as an example to describe how to view progress information with PutObject:

```

#include "oss_api.h"
#include "aos_http_io.h"
const char *endpoint = "<yourEndpoint>";
const char *access_key_id = "<yourAccessKeyId>";
const char *access_key_secret = "<yourAccessKeySecret>";
const char *bucket_name = "<yourBucketName>";
const char *object_name = "<yourObjectName>";
const char *local_filename = "<yourLocalFilename>";
void init_options(oss_request_options_t *options)
{
    options->config = oss_config_create(options->pool);
    /* Use a char* string to initialize aos_string_t. */
    aos_str_set(&options->config->endpoint, endpoint);
    aos_str_set(&options->config->access_key_id, access_key_id);
    aos_str_set(&options->config->access_key_secret, access_key_secret);
    /* Determine whether the CNAME is used. 0 indicates that the CNAME is not used. */
    options->config->is_cname = 0;
    /* Used to configure network parameters, such as timeout. */
    options->ctl = aos_http_controller_create(options->pool, 0);
}
void percentage(int64_t consumed_bytes, int64_t total_bytes)
{
    assert(total_bytes >= consumed_bytes);
    printf("%%%" APR_INT64_T_FMT "\n", consumed_bytes * 100 / total_bytes);
}
int main(int argc, char *argv[])
{
    /* Call the aos_http_io_initialize method in main() to initialize global resources, such as networks and memories. */
    if (aos_http_io_initialize(NULL, 0) != AOSE_OK) {
        exit(1);
    }

    /* Memory pool used to manage memories, which is equivalent to apr_pool_t. The implementation code is included in the apr library. */
    aos_pool_t *pool;
    /* Re-create a new memory pool. The second parameter is NULL, indicating that it does not inherit from any other memory pools. */
    aos_pool_create(&pool, NULL);
    /* Create and initialize options. This parameter mainly includes global configuration information, such as endpoint, access key id, access key secret, is cname, and curl. */

```

```

information, such as endpoint, access_key_id, access_key_secret, is_chname, and curl. */
    oss_request_options_t *oss_client_options;
    /* Allocate memories in the memory pool to options. */
    oss_client_options = oss_request_options_create(pool);
    /* Use oss_client_options to initialize client options. */
    init_options(oss_client_options);
    /* Initialization parameters. */
    aos_string_t bucket;
    aos_string_t object;
    aos_string_t file;
    aos_list_t resp_body;
    aos_table_t *resp_headers = NULL;
    aos_status_t *resp_status = NULL;
    aos_str_set(&bucket, bucket_name);
    aos_str_set(&object, object_name);
    aos_str_set(&file, local_filename);
    /* Upload with a progress bar displayed. */
    resp_status = oss_do_put_object_from_file(oss_client_options, &bucket, &object, &file,
    NULL, NULL, percentage, &resp_headers, &resp_body);
    if (aos_status_is_ok(resp_status)) {
        printf("put object from file succeeded\n");
    } else {
        printf("put object from file failed\n");
    }
    /* Release the memory pool, that is, memories allocated to resources during the request
    . */
    aos_pool_destroy(pool);
    /* Release allocated global resources. */
    aos_http_io_deinitialize();
    return 0;
}

```

`oss_put_object_from_file` and `oss_put_object_from_buffer` do not support progress bars.

12.6.7. Upload callback

This topic describes how to use upload callback.

For the complete code used to implement upload callback, visit [GitHub](#).

The following code provides an example on how to use upload callback:

```

#include "oss_api.h"
#include "aos_http_io.h"
const char *endpoint = "<yourEndpoint>";
const char *access_key_id = "<yourAccessKeyId>";
const char *access_key_secret = "<yourAccessKeySecret>";
const char *bucket_name = "<yourBucketName>";
const char *object_name = "<yourObjectName>";
const char *object_content = "More than just cloud.";
void init_options(oss_request_options_t *options)
{
    options->config = oss_config_create(options->pool);
    /* Use a char* string to initialize the aos_string_t data type. */
    aos_str_set(&options->config->endpoint, endpoint);
}

```

```

aos_str_set(&options->config->access_key_id, access_key_id);
aos_str_set(&options->config->access_key_secret, access_key_secret);
/* Specify whether to use CNAME. The value 0 indicates that the CNAME is not used. */
options->config->is_cname = 0;
/* Configure network parameters such as timeout. */
options->ctl = aos_http_controller_create(options->pool, 0);
}
int main(int argc, char *argv[])
{
    aos_pool_t *p = NULL;
    aos_status_t *s = NULL;
    aos_string_t bucket;
    aos_string_t object;
    aos_table_t *headers = NULL;
    oss_request_options_t *options = NULL;
    aos_table_t *resp_headers = NULL;
    aos_list_t resp_body;
    aos_list_t buffer;
    aos_buf_t *content;
    char *buf = NULL;
    int64_t len = 0;
    int64_t size = 0;
    int64_t pos = 0;
    char b64_buf[1024];
    int b64_len;
    /* Use the JSON format. */
    /* (Optional) Configure the value of the host field carried in the callback request header. */
    char *callback = "{\n"
        "\"callbackUrl\": \"http://oss-demo.aliyuncs.com:23450\",\"\n"
        "\"callbackHost\": \"yourCallbackHost\",\"\n"
        "\"callbackBody\": \"bucket=${bucket}&object=${object}&size=${size}&mimeType=${mimeType}\",\"\n"
        "\"callbackBodyType\": \"application/x-www-form-urlencoded\"\n"
        "\"}";
    /* Call aos_http_io_initialize in main() to initialize global resources, such as the network and memory. */
    if (aos_http_io_initialize(NULL, 0) != AOSE_OK) {
        exit(1);
    }
    /* Initialize the parameters. */
    aos_pool_create(&p, NULL);
    options = oss_request_options_create(p);
    init_options(options);
    aos_str_set(&bucket, bucket_name);
    aos_str_set(&object, object_name);
    aos_list_init(&resp_body);
    aos_list_init(&buffer);
    content = aos_buf_pack(options->pool, object_content, strlen(object_content));
    aos_list_add_tail(&content->node, &buffer);
    /* Add callback to the header. */
    b64_len = aos_base64_encode((unsigned char*)callback, strlen(callback), b64_buf);
    b64_buf[b64_len] = '\\0';
    headers = aos_table_make(p, 1);

```

```
apr_table_set(headers, OSS_CALLBACK, b64_buf);
/* Configure upload callback. */
s = oss_do_put_object_from_buffer(options, &bucket, &object, &buffer,
    headers, NULL, NULL, &resp_headers, &resp_body);
if (aos_status_is_ok(s)) {
    printf("put object from buffer succeeded\n");
} else {
    printf("put object from buffer failed\n");
}
/* Obtain the buffer length. */
len = aos_buf_list_len(&resp_body);
buf = (char *)aos_palloc(p, (apr_size_t)(len + 1));
buf[len] = '\0';
/* Copy the content in the buffer to the memory. */
aos_list_for_each_entry(aos_buf_t, content, &resp_body, node) {
    size = aos_buf_size(content);
    memcpy(buf + pos, content->pos, (size_t)size);
    pos += size;
}
/* Release the memory pool. The memory allocated to various resources used for the request is released. */
aos_pool_destroy(p);
/* Release the allocated global resources. */
aos_http_io_deinitialize();
return 0;
}
```

For more information about upload callback, see [Upload callback](#). For more information about debugging methods and troubleshooting, see [Upload callback errors and troubleshooting](#).

12.7. Download objects

12.7.1. Overview

For the complete code of object download, see [GitHub](#).

OSS C SDK provides the following download methods:

- [Download to your local file](#)
- [Download to local memory](#)
- [Range download](#)
- [Resumable download](#)

You can view the download progress shown in the [download progress bar](#).

12.7.2. Download objects to local computers

This topic describes how to download an object to the local computer.

For the complete code of object download, visit [GitHub](#).

The following code provides an example on how to download a specified object to a local file:

```
#include "oss_api.h"
```

```

#include "aos_http_io.h"
const char *endpoint = "<yourEndpoint>";
const char *access_key_id = "<yourAccessKeyId>";
const char *access_key_secret = "<yourAccessKeySecret>";
const char *bucket_name = "<yourBucketName>";
const char *object_name = "<yourObjectName>";
const char *local_filename = "<yourLocalFilename>";
void init_options(oss_request_options_t *options)
{
    options->config = oss_config_create(options->pool);
    /* Use a char* string to initialize the aos_string_t data type. */
    aos_str_set(&options->config->endpoint, endpoint);
    aos_str_set(&options->config->access_key_id, access_key_id);
    aos_str_set(&options->config->access_key_secret, access_key_secret);
    /* Specify whether to use CNAME. The value of 0 indicates that CNAME is not used. */
    options->config->is_cname = 0;
    /* Configure network parameters such as the timeout period. */
    options->ctl = aos_http_controller_create(options->pool, 0);
}
int main(int argc, char *argv[])
{
    /* Call the aos_http_io_initialize method in main() to initialize global resources such
    as networks and memory. */
    if (aos_http_io_initialize(NULL, 0) != AOSE_OK) {
        exit(1);
    }
    /* Create a memory pool to manage memory. aos_pool_t is equivalent to apr_pool_t. The c
    ode that is used to create a memory pool is included in the APR library. */
    aos_pool_t *pool;
    /* Create a memory pool. The value of the second parameter is NULL. This value specifie
    s that the pool does not inherit other memory pools. */
    aos_pool_create(&pool, NULL);
    /* Create and initialize options. This parameter includes global configuration informat
    ion such as endpoint, access_key_id, access_key_secret, is_cname, and curl. */
    oss_request_options_t *oss_client_options;
    /* Allocate memory resources in the memory pool to the options. */
    oss_client_options = oss_request_options_create(pool);
    /* Initialize oss_client_options. */
    init_options(oss_client_options);
    /* Initialize the parameters. */
    aos_string_t bucket;
    aos_string_t object;
    aos_string_t file;
    aos_table_t *params;
    aos_table_t *headers = NULL;
    aos_table_t *resp_headers = NULL;
    aos_status_t *resp_status = NULL;
    aos_str_set(&bucket, bucket_name);
    aos_str_set(&object, object_name);
    aos_str_set(&file, local_filename);
    params = aos_table_make(pool, 0);
    /* Download the object to your local file. If a file with the same name already exists,
    the downloaded object overwrites the file. Otherwise, a new file is created. */
    resp_status = oss_get_object_to_file(oss_client_options, &bucket, &object, headers, par

```



```

ams, &file, &resp_headers);
    if (aos_status_is_ok(resp_status)) {
        printf("Get object from file succeeded\n");
    } else {
        printf("Get object from file failed\n");
    }
    /* Release the memory pool. This operation releases memory resources allocated for the
    request. */
    aos_pool_destroy(pool);
    /* Release the allocated global resources. */
    aos_http_io_deinitialize();
    return 0;
}

```

 **Note** Compared with OSS SDK for C V1.0.0, this version adds the `params` parameter to the `oss_get_object_to_file` operation, and the values of the headers and `params` parameters can be `NULL`.

12.7.3. Download to your local memory

This topic describes how to download an object to your local memory.

For the complete code of object download, see [GitHub](#).

Run the following code to download a specified object to local memory:

```

#include "oss_api.h"
#include "aos_http_io.h"
const char *endpoint = "<yourEndpoint>";
const char *access_key_id = "<yourAccessKeyId>";
const char *access_key_secret = "<yourAccessKeySecret>";
const char *bucket_name = "<yourBucketName>";
const char *object_name = "<yourObjectName>";
void init_options(oss_request_options_t *options)
{
    options->config = oss_config_create(options->pool);
    /* Use a char* string to initialize aos_string_t. */
    aos_str_set(&options->config->endpoint, endpoint);
    aos_str_set(&options->config->access_key_id, access_key_id);
    aos_str_set(&options->config->access_key_secret, access_key_secret);
    /* Determine whether the CNAME is used. 0 indicates that the CNAME is not used. */
    options->config->is_cname = 0;
    /* Used to configure network parameters, such as timeout. */
    options->ctl = aos_http_controller_create(options->pool, 0);
}
int main(int argc, char *argv[])
{
    /* Call the aos_http_io_initialize method in main() to initialize global resources, suc
    h as networks and memories. */
    if (aos_http_io_initialize(NULL, 0) != AOSE_OK) {
        exit(1);
    }
    /* Memory pool used to manage memories, which is equivalent to apr pool t. The implemen

```

```

tation code is included in the apr library. */
aos_pool_t *pool;
/* Re-create a new memory pool. The second parameter is NULL, indicating that it does not
inherit from any other memory pools. */
aos_pool_create(&pool, NULL);
/* Create and initialize options. This parameter mainly includes global configuration information,
such as endpoint, access_key_id, access_key_secret, is_cname, and curl. */
oss_request_options_t *oss_client_options;
/* Allocate memories in the memory pool to options. */
oss_client_options = oss_request_options_create(pool);
/* Use oss_client_options to initialize client options. */
init_options(oss_client_options);
/* Initialization parameters. */
aos_string_t bucket;
aos_string_t object;
aos_list_t buffer;
aos_buf_t *content = NULL;
aos_table_t *params = NULL;
aos_table_t *headers = NULL;
aos_table_t *resp_headers = NULL;
aos_status_t *resp_status = NULL;
char *buf = NULL;
int64_t len = 0;
int64_t size = 0;
int64_t pos = 0;
aos_str_set(&bucket, bucket_name);
aos_str_set(&object, object_name);
aos_list_init(&buffer);
/* Download an object to local memory. */
resp_status = oss_get_object_to_buffer(oss_client_options, &bucket, &object,
                                     headers, params, &buffer, &resp_headers);
if (aos_status_is_ok(resp_status)) {
    printf("get object to buffer succeeded\n");
    /* Copy the downloaded content to the buffer. */
    len = aos_buf_list_len(&buffer);
    buf = aos_palloc(pool, len + 1);
    buf[len] = '\0';
    aos_list_for_each_entry(aos_buf_t, content, &buffer, node) {
        size = aos_buf_size(content);
        memcpy(buf + pos, content->pos, size);
        pos += size;
    }
}
else {
    printf("get object to buffer failed\n");
}
/* Release the memory pool, that is, memories allocated to resources during the request. */
aos_pool_destroy(pool);
/* Release the allocated global resources. */
aos_http_io_deinitialize();
return 0;
}

```

12.7.4. Range download

If you only need part of the data in an object, you can use range downloads to download specified content.

Run the following code for range download:

```
#include "oss_api.h"
#include "aos_http_io.h"
const char *endpoint = "<yourEndpoint>";
const char *access_key_id = "<yourAccessKeyId>";
const char *access_key_secret = "<yourAccessKeySecret>";
const char *bucket_name = "<yourBucketName>";
const char *object_name = "<yourObjectName>";
void init_options(oss_request_options_t *options)
{
    options->config = oss_config_create(options->pool);
    /* Use a char* string to initialize the aos_string_t type. */
    aos_str_set(&options->config->endpoint, endpoint);
    aos_str_set(&options->config->access_key_id, access_key_id);
    aos_str_set(&options->config->access_key_secret, access_key_secret);
    /* Determine whether the CNAME is used. 0 indicates that the CNAME is not used. */
    options->config->is_cname = 0;
    /* Used to configure network parameters, such as timeout. */
    options->ctl = aos_http_controller_create(options->pool, 0);
}
int main(int argc, char *argv[])
{
    /* Call the aos_http_io_initialize method in main() to initialize global resources, such as networks and memories. */
    if (aos_http_io_initialize(NULL, 0) != AOSE_OK) {
        exit(1);
    }
    /* Memory pool used to manage memories, which is equivalent to apr_pool_t. The implementation code is included in the apr library. */
    aos_pool_t *pool;
    /* Re-create a new memory pool. The second parameter is NULL, indicating that it does not inherit from any other memory pools. */
    aos_pool_create(&pool, NULL);
    /* Create and initialize options. This parameter mainly includes global configuration information, such as endpoint, access_key_id, access_key_secret, is_cname, and curl. */
    oss_request_options_t *oss_client_options;
    /* Allocate memories in the memory pool to options. */
    oss_client_options = oss_request_options_create(pool);
    /* Use oss_client_options to initialize client options. */
    init_options(oss_client_options);
    /* Initialization parameters. */
    aos_string_t bucket;
    aos_string_t object;
    aos_list_t buffer;
    aos_buf_t *content = NULL;
    aos_table_t *params = NULL;
    aos_table_t *headers = NULL;
    aos_table_t *resp_headers = NULL;
```

```

aos_status_t *resp_status = NULL;
char *buf = NULL;
int64_t len = 0;
int64_t size = 0;
int64_t pos = 0;
aos_str_set(&bucket, bucket_name);
aos_str_set(&object, object_name);
aos_list_init(&buffer);
headers = aos_table_make(pool, 1);
/* Set the download range, which ranges from the 20th byte to the 100th byte. */
apr_table_set(headers, "Range", "bytes=20-100");
/* Download the object to memory. */
resp_status = oss_get_object_to_buffer(oss_client_options, &bucket, &object, headers, p
arams, &buffer, &resp_headers);
if (aos_status_is_ok(resp_status)) {
    printf("get object to buffer succeeded\n");
    /* Obtain the buffer length. */
    aos_list_for_each_entry(aos_buf_t, content, &buffer, node) {
        len += aos_buf_size(content);
    }
    buf = aos_palloc(pool, (apr_size_t)(len + 1));
    buf[len] = '\0';
    /* Copy the content in the buffer to memory. */
    aos_list_for_each_entry(aos_buf_t, content, &buffer, node) {
        size = aos_buf_size(content);
        memcpy(buf + pos, content->pos, (size_t)size);
        pos += size;
    }
}
else {
    printf("get object to buffer failed\n");
}
/* Release the memory pool, that is, memories allocated to resources during the request
. */
aos_pool_destroy(pool);
/* Release allocated global resources. */
aos_http_io_deinitialize();
return 0;
}

```

12.7.5. Resumable download

Large-sized objects may fail to be downloaded because of unstable network or other exceptions, and the entire object needs to be downloaded again. However, the object may still fail to be downloaded after multiple attempts. Therefore, OSS provides resumable download.

To enable resumable download, you need to split the object you want to download into multiple parts and download them separately. After you have downloaded all parts, these parts will be combined into a complete object.

You can use `oss_resumable_clt_params_t` to enable resumable download.

`oss_resumable_clt_params_t` contains the following parameters:

Parameter	Description
thread_num	The number of concurrent threads, which is 1 by default.
enable_checkpoint	Determines whether to enable resumable download, which is disabled by default.
partSize	Specifies the size of a part. The value can be between 1 byte to 5 GB and the unit is byte.
checkpoint_path	The path where the checkpoint file is located, which is under {upload_file_path}.cp by default.

This `checkpoint` file stores information about the download progress. If you fail to download a part, the download will continue based on the progress recorded in the `checkpoint` file. After the object is downloaded, the `checkpoint` file will be deleted.

Run the following code for resumable download:

```
#include "oss_api.h"
#include "aos_http_io.h"
const char *endpoint = "<yourEndpoint>";
const char *access_key_id = "<yourAccessKeyId>";
const char *access_key_secret = "<yourAccessKeySecret>";
const char *bucket_name = "<yourBucketName>";
const char *object_name = "<yourObjectName>";
const char *local_filename = "<yourLocalFilename>";
void init_options(oss_request_options_t *options)
{
    options->config = oss_config_create(options->pool);
    /* Use a char* string to initialize the aos_string_t type. */
    aos_str_set(&options->config->endpoint, endpoint);
    aos_str_set(&options->config->access_key_id, access_key_id);
    aos_str_set(&options->config->access_key_secret, access_key_secret);
    /* Determine whether the CNAME is used. 0 indicates that the CNAME is not used. */
    options->config->is_cname = 0;
    /* Used to configure network parameters, such as timeout. */
    options->ctl = aos_http_controller_create(options->pool, 0);
}
int main(int argc, char *argv[])
{
    /* Call the aos_http_io_initialize method in main() to initialize global resources, such as networks and memories. */
    if (aos_http_io_initialize(NULL, 0) != AOSE_OK) {
        exit(1);
    }
    /* Memory pool used to manage memories, which is equivalent to apr_pool_t. The implementation code is included in the apr library. */
    aos_pool_t *pool;
    /* Re-create a new memory pool. The second parameter is NULL, indicating that it does not inherit from any other memory pools. */
    aos_pool_create(&pool, NULL);
    /* Create and initialize options. This parameter mainly includes global configuration i
```

```

nformation, such as endpoint, access_key_id, access_key_secret, is_cname, and curl. */
    oss_request_options_t *oss_client_options;
    /* Allocate memories in the memory pool to options. */
    oss_client_options = oss_request_options_create(pool);
    /* Use oss_client_options to initialize client options. */
    init_options(oss_client_options);
    /* Initialization parameters. */
    aos_string_t bucket;
    aos_string_t object;
    aos_string_t file;
    aos_table_t *headers = NULL;
    aos_table_t *resp_headers = NULL;
    aos_status_t *resp_status = NULL;
    oss_resumable_clt_params_t *clt_params;
    aos_str_set(&bucket, bucket_name);
    aos_str_set(&object, object_name);
    aos_str_set(&file, local_filename);
    /* Download the object using resumable download. */
    clt_params = oss_create_resumable_clt_params_content(pool, 1024 * 100, 3, AOS_TRUE, NULL);
    resp_status = oss_resumable_download_file(oss_client_options, &bucket, &object, &file,
    headers, NULL, clt_params, NULL, &resp_headers);
    if (aos_status_is_ok(resp_status)) {
        printf("download succeeded\n");
    } else {
        printf("download failed\n");
    }
    /* Release the memory pool, that is, memories allocated to resources during the request. */
    aos_pool_destroy(pool);
    /* Release allocated global resources. */
    aos_http_io_deinitialize();
    return 0;
}

```

12.7.6. Download progress bars

You can use progress bars to indicate the upload or download progress.

For the complete code of download progress bar, see [GitHub](#).

The following code is used as an example to describe how to use the download progress bar.

```

#include "oss_api.h"
#include "aos_http_io.h"
const char *endpoint = "<yourEndpoint>";
const char *access_key_id = "<yourAccessKeyId>";
const char *access_key_secret = "<yourAccessKeySecret>";
const char *bucket_name = "<yourBucketName>";
const char *object_name = "<yourObjectName>";
const char *local_filename = "<yourLocalFilename>";
const char *download_filename = "<yourDownloadFilename>";
void init_options(oss_request_options_t *options)
{

```

```

options->config = oss_config_create(options->pool);
/* Use a char* string to initialize aos_string_t. */
aos_str_set(&options->config->endpoint, endpoint);
aos_str_set(&options->config->access_key_id, access_key_id);
aos_str_set(&options->config->access_key_secret, access_key_secret);
/* Determine whether the CNAME is used. 0 indicates that the CNAME is not used. */
options->config->is_cname = 0;
/* Used to configure network parameters, such as timeout. */
options->ctl = aos_http_controller_create(options->pool, 0);
}
void percentage(int64_t consumed_bytes, int64_t total_bytes)
{
    assert(total_bytes >= consumed_bytes);
    printf("%%% APR_INT64_T_FMT "\n", consumed_bytes * 100 / total_bytes);
}
int main(int argc, char *argv[])
{
    /* Call the aos_http_io_initialize method in main() to initialize global resources, such as networks and memories. */
    if (aos_http_io_initialize(NULL, 0) != AOSE_OK) {
        exit(1);
    }
    /* Memory pool used to manage memories, which is equivalent to apr_pool_t. The implementation code is included in the apr library. */
    aos_pool_t *pool;
    /* Re-create a new memory pool. The second parameter is NULL, indicating that it does not inherit from any other memory pools. */
    aos_pool_create(&pool, NULL);
    /* Create and initialize options. This parameter mainly includes global configuration information, such as endpoint, access_key_id, access_key_secret, is_cname, and curl. */
    oss_request_options_t *oss_client_options;
    /* Allocate memories in the memory pool to options. */
    oss_client_options = oss_request_options_create(pool);
    /* Use oss_client_options to initialize client options. */
    init_options(oss_client_options);
    /* Initialization parameters. */
    aos_string_t bucket;
    aos_string_t object;
    aos_string_t file;
    aos_table_t *resp_headers = NULL;
    aos_list_t resp_body;
    aos_status_t *resp_status = NULL;
    aos_str_set(&bucket, bucket_name);
    aos_str_set(&object, object_name);
    aos_str_set(&file, local_filename);
    /* Upload with a progress bar displayed. */
    resp_status = oss_do_put_object_from_file(oss_client_options, &bucket, &object, &file, NULL, NULL, percentage, &resp_headers, &resp_body);
    if (aos_status_is_ok(resp_status)) {
        printf("put object from file succeeded\n");
    } else {
        printf("put object from file failed\n");
    }
    /* Download with a progress bar displayed. */

```

```
aos_pool_create(&pool, NULL);
oss_client_options = oss_request_options_create(pool);
init_options(oss_client_options);
aos_str_set(&file, download_filename);
resp_status = oss_do_get_object_to_file(oss_client_options, &bucket, &object, NULL, NULL, &file, percentage, NULL);
if (aos_status_is_ok(resp_status)) {
    printf("get object to file succeeded\n");
} else {
    printf("get object to file failed\n");
}
/* Release the memory pool, that is, memories allocated to resources during the request . */
aos_pool_destroy(pool);
/* Release allocated global resources. */
aos_http_io_deinitialize();
return 0;
}
```

`oss_get_object_from_file` and `oss_get_object_from_buffer` do not support progress bars.

12.8. Manage objects

12.8.1. Overview

You can perform the following operations on objects in a bucket with APIs:

- [Manage ACL for an object](#)
- [Manage Object Meta](#)
- [List objects](#)
- [Copy objects](#)
- [Delete objects](#)
- [Restore an archive object](#)
- [Manage a symbolic link](#)

12.8.2. Manage object ACLs

This topic describes how to manage the access control lists (ACLs) of objects.

The following table describes the ACLs that you can configure for an object.

Permission	Description	Value
Inherited from the bucket	The ACL of an object is the same as the ACL of the bucket in which the object is stored.	OSS_ACL_DEFAULT

Permission	Description	Value
Private	Only the object owner and authorized users have read and write permissions on the object. Other users cannot access the object.	OSS_ACL_PRIVATE
Public read	Only the object owner and authorized users have read and write permissions on the object. Other users have only read permissions on the object. Exercise caution when you use this permission.	OSS_ACL_PUBLIC_READ
Public read/write	All users have read and write permissions on the object. Exercise caution when you use this permission.	OSS_ACL_PUBLIC_READ_WRITE

The ACL of an object takes precedence over the ACL of the bucket in which the object is stored. For example, if the ACL of an object in a private bucket is public read/write, all users have read and write permissions on the object. If you do not configure the ACL of an object, the ACL of the object is the same as the ACL of the bucket in which the object is stored.

You can use the following complete code to configure and query the ACL of an object:

```
#include "oss_api.h"
#include "aos_http_io.h"
const char *endpoint = "<yourEndpoint>";
const char *access_key_id = "<yourAccessKeyId>";
const char *access_key_secret = "<yourAccessKeySecret>";
const char *bucket_name = "<yourBucketName>";
const char *object_name = "<yourObjectName>";
void init_options(oss_request_options_t *options)
{
    options->config = oss_config_create(options->pool);
    /* Use a char* string to initialize the aos_string_t data type. */
    aos_str_set(&options->config->endpoint, endpoint);
    aos_str_set(&options->config->access_key_id, access_key_id);
    aos_str_set(&options->config->access_key_secret, access_key_secret);
    /* Specify whether to use CNAME. The value of 0 indicates that CNAME is not used. */
    options->config->is_cname = 0;
    /* Configure network parameters such as the timeout period. */
    options->ctl = aos_http_controller_create(options->pool, 0);
}
int main(int argc, char *argv[])
{
    /* Call the aos_http_io_initialize method in main() to initialize global resources such
    as networks and memory. */
    if (aos_http_io_initialize(NULL, 0) != AOSE_OK) {
        exit(1);
    }
    /* Create a memory pool to manage memory. aos_pool_t is equivalent to apr_pool_t. The c
```

```

/* Create a memory pool to manage memory. aos_pool_t is equivalent to apr_pool_t. The code that is used to create a memory pool is included in the APR library. */
aos_pool_t *pool;
/* Create a memory pool. The value of the second parameter is NULL. This value specifies that the pool does not inherit other memory pools. */
aos_pool_create(&pool, NULL);
/* Create and initialize options. This parameter includes global configuration information such as endpoint, access_key_id, access_key_secret, is_cname, and curl. */
oss_request_options_t *oss_client_options;
/* Allocate the memory resources in the memory pool to the options. */
oss_client_options = oss_request_options_create(pool);
/* Initialize oss_client_options. */
init_options(oss_client_options);
/* Initialize the parameters. */
aos_string_t bucket;
aos_string_t object;
aos_table_t *resp_headers = NULL;
aos_status_t *resp_status = NULL;
aos_str_set(&bucket, bucket_name);
aos_str_set(&object, object_name);
oss_acl_e oss_acl = OSS_ACL_PUBLIC_READ;
/* Set the ACL of the object. */
resp_status = oss_put_object_acl(oss_client_options, &bucket, &object, oss_acl, &resp_headers);
if (aos_status_is_ok(resp_status)) {
    printf("put object acl success!\n");
} else {
    printf("put object acl failed!\n");
}
/* Obtain the ACL of the object. */
aos_string_t oss_acl_string;
resp_status = oss_get_object_acl(oss_client_options, &bucket, &object, &oss_acl_string, &resp_headers);
if (aos_status_is_ok(resp_status)) {
    printf("get object acl success!\n");
    printf("acl: %s \n", oss_acl_string.data);
} else {
    printf("get object acl failed!\n");
}
/* Release the memory pool. This operation releases memory resources that are allocated for the request. */
aos_pool_destroy(pool);
/* Release the allocated global resources. */
aos_http_io_deinitialize();
return 0;
}

```

12.8.3. Manage object metadata

Object metadata includes HTTP headers and user metadata.

Examples

After you upload an object to OSS or before you read an object from OSS, you can call the `oss_get_object_meta` operation to query the metadata of the object such as the length and type of the object. The following code provides an example on how to configure and query the metadata of an object:

```
#include "oss_api.h"
#include "aos_http_io.h"
/* Specify the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com. */
const char *endpoint = "yourEndpoint";
/* The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in OSS is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console. */
const char *access_key_id = "yourAccessKeyId";
const char *access_key_secret = "yourAccessKeySecret";
/* Specify the name of the bucket. Example: examplebucket. */
const char *bucket_name = "examplebucket";
/* Specify the full path of the object. The path cannot contain the bucket name. Example: exampledir/exampleobject.txt. */
const char *object_name = "exampledir/exampleobject.txt";
void init_options(oss_request_options_t *options)
{
    options->config = oss_config_create(options->pool);
    /* Use a char* string to initialize the aos_string_t data type. */
    aos_str_set(&options->config->endpoint, endpoint);
    aos_str_set(&options->config->access_key_id, access_key_id);
    aos_str_set(&options->config->access_key_secret, access_key_secret);
    /* Specify whether to use CNAME. The value of 0 indicates that CNAME is not used. */
    options->config->is_cname = 0;
    /* Configure network parameters such as the timeout period. */
    options->ctl = aos_http_controller_create(options->pool, 0);
}
int main(int argc, char *argv[])
{
    /* Call the aos_http_io_initialize method in main() to initialize global resources such as networks and memory. */
    if (aos_http_io_initialize(NULL, 0) != AOSE_OK) {
        exit(1);
    }
    /* Create a memory pool to manage memory. aos_pool_t is equivalent to apr_pool_t. The code used to create a memory pool is included in the APR library. */
    aos_pool_t *pool;
    /* Create a memory pool. The value of the second parameter is NULL. This value specifies that the pool does not inherit other memory pools. */
    aos_pool_create(&pool, NULL);
    /* Create and initialize options. This parameter includes global configuration information such as endpoint, access_key_id, access_key_secret, is_cname, and curl. */
    oss_request_options_t *oss_client_options;
    /* Allocate memory resources in the memory pool to the options parameter. */
    oss_client_options = oss_request_options_create(pool);
    /* Initialize oss_client_options. */
    init_options(oss_client_options);
```

```

/* Initialize the parameters. */
aos_string_t bucket;
aos_string_t object;
aos_table_t *headers;
aos_list_t buffer;
aos_table_t *resp_headers = NULL;
aos_status_t *resp_status = NULL;
aos_buf_t *content = NULL;
char *content_length_str = NULL;
char *object_type = NULL;
char *object_author = NULL;
int64_t content_length = 0;
aos_str_set(&bucket, bucket_name);
aos_str_set(&object, object_name);
headers = aos_table_make(pool, 2);
/* Configure user metadata. */
apr_table_set(headers, "Expires", "Fri, 28 Feb 2032 05:38:42 GMT");
apr_table_set(headers, "x-oss-meta-author", "oss");
aos_list_init(&buffer);
content = aos_buf_pack(oss_client_options->pool, object_content, strlen(object_content)
);
aos_list_add_tail(&content->node, &buffer);
/* Upload the object from the buffer. */
resp_status = oss_put_object_from_buffer(oss_client_options, &bucket, &object,
&buffer, headers, &resp_headers);
if (aos_status_is_ok(resp_status)) {
    printf("put object from buffer with md5 succeeded\n");
} else {
    printf("put object from buffer with md5 failed\n");
}
// Query the metadata of the object. */
resp_status = oss_get_object_meta(oss_client_options, &bucket, &object, &resp_headers);
if (aos_status_is_ok(resp_status)) {
    content_length_str = (char*)apr_table_get(resp_headers, OSS_CONTENT_LENGTH);
    if (content_length_str != NULL) {
        content_length = atol(content_length_str);
    }
    object_author = (char*)apr_table_get(resp_headers, OSS_AUTHORIZATION);
    object_type = (char*)apr_table_get(resp_headers, OSS_OBJECT_TYPE);
    printf("get object meta succeeded, object author:%s, object type:%s, content_length
:%ld\n", object_author, object_type, content_length);
} else {
    printf("req:%s, get object meta failed\n", resp_status->req_id);
}
/* Release the memory pool, which releases memory resources allocated for the request.
*/
aos_pool_destroy(pool);
/* Release the allocated global resources. */
aos_http_io_deinitialize();
return 0;
}

```

References

- For more information about the complete sample code that is used to query object metadata, visit [Git Hub](#).
- For more information about the API operation that you can call to query object metadata, see [Get Object Metadata](#).

12.8.4. List objects

This topic describes how to list the objects stored in a bucket.

For the complete code used to list objects, visit [GitHub](#).

Objects are listed in alphabetical order. You can use `oss_list_object` to list objects in a bucket. The following table describes common parameters.

Parameter	Description
delimiter	Groups objects by name. CommonPrefixes specifies a set of substrings of object names. The substrings start with the prefix and end with the next occurrence of the specified delimiter.
prefix	Specifies the prefix that returned object names must contain.
max_ret	Specifies the maximum number of objects that can be listed at a time. The default value is 100. The maximum value is 1000.
marker	Specifies the name of the object after which the listing begins.

List all objects

The following code provides an example on how to list all objects in a bucket:

```
#include "oss_api.h"
#include "aos_http_io.h"

const char *endpoint = "<yourEndpoint>";
const char *access_key_id = "<yourAccessKeyId>";
const char *access_key_secret = "<yourAccessKeySecret>";
const char *bucket_name = "<yourBucketName>";

void init_options(oss_request_options_t *options)
{
    options->config = oss_config_create(options->pool);
    /* Use a char* string to initialize the aos_string_t data type. */
    aos_str_set(&options->config->endpoint, endpoint);
    aos_str_set(&options->config->access_key_id, access_key_id);
    aos_str_set(&options->config->access_key_secret, access_key_secret);
    /* Specify whether to use CNAME. 0 indicates that the CNAME is not used. */
    options->config->is_cname = 0;
    /* Configure network parameters, such as timeout. */
    options->ctl = aos_http_controller_create(options->pool, 0);
}

int main(int argc, char *argv[])
{
    /*
```

```

    /* Call aos_http_io_initialize in main() to initialize global resources, such as the ne
network and memory. */
    if (aos_http_io_initialize(NULL, 0) != AOSE_OK) {
        exit(1);
    }

    /* Create a memory pool to manage memory. The implementation code is included in the AP
R library. aos_pool_t is equivalent to apr_pool_t. */
    aos_pool_t *pool;

    /* Create a new memory pool. The second parameter value is NULL, indicating that the po
ol does not inherit any other memory pool. */
    aos_pool_create(&pool, NULL);

    /* Create and initialize options. This parameter includes global configuration informat
ion such as endpoint, access_key_id, access_key_secret, is_cname, and curl. */
    oss_request_options_t *oss_client_options;
    /* Allocate a memory chunk in the memory pool to options. */
    oss_client_options = oss_request_options_create(pool);
    /* Initialize oss_client_options. */
    init_options(oss_client_options);
    /* Initialize parameters. */
    aos_string_t bucket;
    aos_status_t *resp_status = NULL;
    oss_list_object_params_t *params = NULL;
    oss_list_object_content_t *content = NULL;
    int size = 0;
    char *line = NULL;
    char *prefix = "";
    char *nextMarker = "";
    aos_str_set(&bucket, bucket_name);
    params = oss_create_list_object_params(pool);
    params->max_ret = 100;
    aos_str_set(&params->prefix, prefix);
    aos_str_set(&params->marker, nextMarker);
    printf("Object\tSize\tLastModified\n");
    /* List all objects. */
    do {
        resp_status = oss_list_object(oss_client_options, &bucket, params, NULL);
        if (! aos_status_is_ok(resp_status))
        {
            printf("list object failed\n");
            break;
        }
        aos_list_for_each_entry(oss_list_object_content_t, content, &params->object_list, n
ode) {
            ++size;
            line = apr_psprintf(pool, "%.s\t%.s\t%.s\n", content->key.len, content->key.
data,
                                content->size.len, content->size.data,
                                content->last_modified.len, content->last_modified.data);
            printf("%s", line);
        }
        nextMarker = apr_psprintf(pool, "%.s", params->next_marker.len, params->next_marke
r.data);
        aos_str_set(&params->marker, nextMarker);
        aos_list_init(&params->object_list);

```

```

    aos_list_init(&params->common_prefix_list);
} while (params->truncated == AOS_TRUE);
printf("Total %d\n", size);
/* Release the memory pool. The memory allocated to various resources used for the request is released. */
aos_pool_destroy(pool);
/* Release the allocated global resources. */
aos_http_io_deinitialize();
return 0;
}

```

 **Note** If a bucket contains more than 1,000 objects, only the first 1,000 objects are returned and the truncated parameter in the returned results is true. Additionally, the next_marker parameter is returned to serve as the start point for the next read. To list more objects at a time, you can modify the max_ret parameter. You can also use the marker parameter to perform multiple read operations.

List a specified number of objects

```

#include "oss_api.h"
#include "aos_http_io.h"
const char *endpoint = "<yourEndpoint>";
const char *access_key_id = "<yourAccessKeyId>";
const char *access_key_secret = "<yourAccessKeySecret>";
const char *bucket_name = "<yourBucketName>";
void init_options(oss_request_options_t *options)
{
    options->config = oss_config_create(options->pool);
    /* Use a char* string to initialize the aos_string_t data type. */
    aos_str_set(&options->config->endpoint, endpoint);
    aos_str_set(&options->config->access_key_id, access_key_id);
    aos_str_set(&options->config->access_key_secret, access_key_secret);
    /* Specify whether to use CNAME. */
    options->config->is_cname = 0;
    /* Configure network parameters, such as timeout. */
    options->ctl = aos_http_controller_create(options->pool, 0);
}
int main(int argc, char *argv[])
{
    /* Call aos_http_io_initialize in main() to initialize global resources, such as the network and memory. */
    if (aos_http_io_initialize(NULL, 0) != AOSE_OK) {
        exit(1);
    }
    /* Create a memory pool to manage memory. The implementation code is included in the APR library. aos_pool_t is equivalent to apr_pool_t. */
    aos_pool_t *pool;
    /* Create a new memory pool. The second parameter value is NULL, indicating that the pool does not inherit any other memory pool. */
    aos_pool_create(&pool, NULL);
    /* Create and initialize options. This parameter includes global configuration information such as endpoint, access_key_id, access_key_secret, is_cname, and curl. */

```

```

    oss_request_options_t *oss_client_options;
    /* The memory consumed by options is allocated by the memory pool and is released with
the pool. */
    oss_client_options = oss_request_options_create(pool);
    /* Initialize oss_client_options. */
    init_options(oss_client_options);
    /* Initialize parameters. */
    aos_string_t bucket;
    aos_status_t *resp_status = NULL;
    oss_list_object_params_t *params = NULL;
    oss_list_object_content_t *content = NULL;
    int size = 0;
    char *line = NULL;
    char *prefix = "";
    char *nextMarker = "";
    aos_str_set(&bucket, bucket_name);
    params = oss_create_list_object_params(pool);
    /* Set the maximum number of returned objects to 200. */
    params->max_ret = 200;
    aos_str_set(&params->prefix, prefix);
    aos_str_set(&params->marker, nextMarker);
    printf("Object\tSize\tLastModified\n");
    /* List all objects. */
    resp_status = oss_list_object(oss_client_options, &bucket, params, NULL);
    if (! aos_status_is_ok(resp_status))
    {
        printf("list object failed\n");
    }
    aos_list_for_each_entry(oss_list_object_content_t, content, &params->object_list, node)
    {
        ++size;
        line = apr_psprintf(pool, "%.s\t%.s\t%.s\n", content->key.len, content->key.data
,
        content->size.len, content->size.data,
        content->last_modified.len, content->last_modified.data);
        printf("%s", line);
    }
    printf("Total %d\n", size);
    /* Release the memory pool after the request is complete. In this case, the memory allo
cated to various resources used for the request is released. */
    aos_pool_destroy(pool);
    /* Before you end the program, call aos_http_io_deinitialize to release the allocated g
lobal resources. */
    aos_http_io_deinitialize();
    return 0;
}

```

List objects whose names contain the specified prefix

```

#include "oss_api.h"
#include "aos_http_io.h"
const char *endpoint = "<yourEndpoint>";
const char *access_key_id = "<yourAccessKeyId>";

```



```

const char *access_key_secret = "<yourAccessKeySecret>";
const char *bucket_name = "<yourBucketName>";
void init_options(oss_request_options_t *options)
{
    options->config = oss_config_create(options->pool);
    /* Use a char* string to initialize the aos_string_t data type. */
    aos_str_set(&options->config->endpoint, endpoint);
    aos_str_set(&options->config->access_key_id, access_key_id);
    aos_str_set(&options->config->access_key_secret, access_key_secret);
    /* Specify whether to use CNAME. */
    options->config->is_cname = 0;
    /* Configure network parameters, such as timeout. */
    options->ctl = aos_http_controller_create(options->pool, 0);
}
int main(int argc, char *argv[])
{
    /* Call aos_http_io_initialize in main() to initialize global resources, such as the ne
twork and memory.*/
    if (aos_http_io_initialize(NULL, 0) != AOSE_OK) {
        exit(1);
    }
    /* Create a memory pool to manage memory. The implementation code is included in the AP
R library. aos_pool_t is equivalent to apr_pool_t. */
    aos_pool_t *pool;
    /* Create a new memory pool. The second parameter value is NULL, indicating that the po
ol does not inherit any other memory pool. */
    aos_pool_create(&pool, NULL);
    /* Create and initialize options. This parameter includes global configuration informat
ion such as endpoint, access_key_id, access_key_secret, is_cname, and curl. */
    oss_request_options_t *oss_client_options;
    /* The memory consumed by options is allocated by the memory pool and is released with
the pool. */
    oss_client_options = oss_request_options_create(pool);
    /* Initialize oss_client_options. */
    init_options(oss_client_options);
    /* Initialize parameters. */
    aos_string_t bucket;
    aos_status_t *resp_status = NULL;
    oss_list_object_params_t *params = NULL;
    oss_list_object_content_t *content = NULL;
    int size = 0;
    char *line = NULL;
    /* List the objects whose names contain the specified prefix. */
    char *prefix = "<yourObjectPefix>";
    char *nextMarker = "";
    aos_str_set(&bucket, bucket_name);
    params = oss_create_list_object_params(pool);
    params->max_ret = 100;
    aos_str_set(&params->prefix, prefix);
    aos_str_set(&params->marker, nextMarker);
    printf("Object\tSize\tLastModified\n");
    /* List all objects. */
    do {
        resp_status = oss_list_object(oss_client_options, &bucket, params, NULL);
        if (resp_status != NULL) {
            printf("%s\t%ld\t%s\n", bucket_name, params->max_ret, resp_status->message);
        }
    } while (resp_status != NULL);
}

```

```

    if (! aos_status_is_ok(resp_status))
    {
        printf("list object failed\n");
        break;
    }
    aos_list_for_each_entry(oss_list_object_content_t, content, &params->object_list, n
ode) {
        ++size;
        line = apr_psprintf(pool, "%.s\t%.s\t%.s\n", content->key.len, content->key.
data,
            content->size.len, content->size.data,
            content->last_modified.len, content->last_modified.data);
        printf("%s", line);
    }
    nextMarker = apr_psprintf(pool, "%.s", params->next_marker.len, params->next_marke
r.data);
    aos_str_set(&params->marker, nextMarker);
    aos_list_init(&params->object_list);
    aos_list_init(&params->common_prefix_list);
} while (params->truncated == AOS_TRUE);
printf("Total %d\n", size);
/* Release the memory pool after the request is complete. In this case, the memory allo
cated to various resources used for the request is released. */
aos_pool_destroy(pool);
/* Before you end the program, call aos_http_io_deinitialize to release the allocated g
lobal resources. */
aos_http_io_deinitialize();
return 0;
}

```

List objects that follow a specified marker

```

#include "oss_api.h"
#include "aos_http_io.h"
const char *endpoint = "<yourEndpoint>";
const char *access_key_id = "<yourAccessKeyId>";
const char *access_key_secret = "<yourAccessKeySecret>";
const char *bucket_name = "<yourBucketName>";
void init_options(oss_request_options_t *options)
{
    options->config = oss_config_create(options->pool);
    /* Use a char* string to initialize the aos_string_t data type. */
    aos_str_set(&options->config->endpoint, endpoint);
    aos_str_set(&options->config->access_key_id, access_key_id);
    aos_str_set(&options->config->access_key_secret, access_key_secret);
    /* Specify whether to use CNAME. */
    options->config->is_cname = 0;
    /* Configure network parameters, such as timeout. */
    options->ctl = aos_http_controller_create(options->pool, 0);
}
int main(int argc, char *argv[])
{
    /* Call aos_http_io_initialize in main() to initialize global resources, such as the ne
twork and memory.*/

```

```

    if (aos_http_io_initialize(NULL, 0) != AOSE_OK) {
        exit(1);
    }

    /* Create a memory pool to manage memory. The implementation code is included in the AP
R library. aos_pool_t is equivalent to apr_pool_t. */
    aos_pool_t *pool;
    /* Create a new memory pool. The second parameter value is NULL, indicating that the po
ol does not inherit any other memory pool. */
    aos_pool_create(&pool, NULL);
    /* Create and initialize options. This parameter includes global configuration informat
ion such as endpoint, access_key_id, access_key_secret, is_cname, and curl. */
    oss_request_options_t *oss_client_options;
    /* The memory consumed by options is allocated by the memory pool and is released with
the pool. */
    oss_client_options = oss_request_options_create(pool);
    /* Initialize oss_client_options. */
    init_options(oss_client_options);
    /* Initialize parameters. */
    aos_string_t bucket;
    aos_status_t *resp_status = NULL;
    oss_list_object_params_t *params = NULL;
    oss_list_object_content_t *content = NULL;
    int size = 0;
    char *line = NULL;
    char *prefix = "";
    /* List objects that follow a specified marker. */
    char *nextMarker = "<yourObjectMarker>";
    aos_str_set(&bucket, bucket_name);
    params = oss_create_list_object_params(pool);
    params->max_ret = 100;
    aos_str_set(&params->prefix, prefix);
    aos_str_set(&params->marker, nextMarker);
    printf("Object\tSize\tLastModified\n");
    /* List all objects. */
    do {
        resp_status = oss_list_object(oss_client_options, &bucket, params, NULL);
        if (! aos_status_is_ok(resp_status))
        {
            printf("list object failed\n");
            break;
        }
        aos_list_for_each_entry(oss_list_object_content_t, content, &params->object_list, n
ode) {
            ++size;
            line = apr_psprintf(pool, "%.s\t%.s\t%.s\n", content->key.len, content->key.
data,
                                content->size.len, content->size.data,
                                content->last_modified.len, content->last_modified.data);
            printf("%s", line);
        }
        nextMarker = apr_psprintf(pool, "%.s", params->next_marker.len, params->next_marke
r.data);
        aos_str_set(&params->marker, nextMarker);
        aos_list_init(&params->object_list);

```

```

    aos_list_init(&params->common_prefix_list);
} while (params->truncated == AOS_TRUE);
printf("Total %d\n", size);
/* Release the memory pool after the request is complete. In this case, the memory allo
cated to various resources used for the request is released. */
aos_pool_destroy(pool);
/* Before you end the program, call aos_http_io_deinitialize to release the allocated g
lobal resources. */
aos_http_io_deinitialize();
return 0;
}

```

Folder simulation

For more information about folders, see [List objects](#). For the complete code used to create a folder, visit [GitHub](#).

- List all objects in a bucket

```

#include "oss_api.h"
#include "aos_http_io.h"
const char *endpoint = "<yourEndpoint>";
const char *access_key_id = "<yourAccessKeyId>";
const char *access_key_secret = "<yourAccessKeySecret>";
const char *bucket_name = "<yourBucketName>";
void init_options(oss_request_options_t *options)
{
    options->config = oss_config_create(options->pool);
    /* Use a char* string to initialize the aos_string_t data type. */
    aos_str_set(&options->config->endpoint, endpoint);
    aos_str_set(&options->config->access_key_id, access_key_id);
    aos_str_set(&options->config->access_key_secret, access_key_secret);
    /* Specify whether to use CNAME. 0 indicates that the CNAME is not used. */
    options->config->is_cname = 0;
    /* Configure network parameters, such as timeout. */
    options->ctl = aos_http_controller_create(options->pool, 0);
}
int main(int argc, char *argv[])
{
    /* Call aos_http_io_initialize in main() to initialize global resources, such as the
network and memory. */
    if (aos_http_io_initialize(NULL, 0) != AOSE_OK) {
        exit(1);
    }
    /* Create a memory pool to manage memory. The implementation code is included in the
APR library. aos_pool_t is equivalent to apr_pool_t. */
    aos_pool_t *pool;
    /* Create a new memory pool. The second parameter value is NULL, indicating that the
pool does not inherit any other memory pool. */
    aos_pool_create(&pool, NULL);
    /* Create and initialize options. This parameter includes global configuration inform
ation such as endpoint, access_key_id, access_key_secret, is_cname, and curl. */
    oss_request_options_t *oss_client_options;
    /* Allocate a memory chunk in the memory pool to options. */

```

```

oss_client_options = oss_request_options_create(pool);
/* Initialize oss_client_options. */
init_options(oss_client_options);
/* Initialize parameters. */
aos_string_t bucket;
aos_status_t *resp_status = NULL;
oss_list_object_params_t *params = NULL;
oss_list_object_content_t *content = NULL;
int size = 0;
char *line = NULL;
char *nextMarker = "";
aos_str_set(&bucket, bucket_name);
params = oss_create_list_object_params(pool);
params->max_ret = 100;
aos_str_set(&params->marker, nextMarker);
printf("Object\tSize\tLastModified\n");
/* List all objects. */
do {
    resp_status = oss_list_object(oss_client_options, &bucket, params, NULL);
    if (! aos_status_is_ok(resp_status))
    {
        printf("list object failed\n");
        break;
    }
    aos_list_for_each_entry(oss_list_object_content_t, content, &params->object_list,
node) {
        ++size;
        line = apr_psprintf(pool, "%.s\t%.s\t%.s\n", content->key.len, content->key.data,
            content->size.len, content->size.data,
            content->last_modified.len, content->last_modified.data);
        printf("%s", line);
    }
    nextMarker = apr_psprintf(pool, "%.s", params->next_marker.len, params->next_mar
ker.data);
    aos_str_set(&params->marker, nextMarker);
    aos_list_init(&params->object_list);
    aos_list_init(&params->common_prefix_list);
} while (params->truncated == AOS_TRUE);
printf("Total %d\n", size);
/* Release the memory pool. The memory allocated to various resources used for the re
quest is released. */
aos_pool_destroy(pool);
/* Release the allocated global resources. */
aos_http_io_deinitialize();
return 0;
}

```

- List all objects in a specified folder

```

#include "oss_api.h"
#include "aos_http_io.h"
const char *endpoint = "<yourEndpoint>";
const char *access_key_id = "<yourAccessKeyId>";

```

```

const char *access_key_id = "<yourAccessKeyId>";
const char *access_key_secret = "<yourAccessKeySecret>";
const char *bucket_name = "<yourBucketName>";
void init_options(oss_request_options_t *options)
{
    options->config = oss_config_create(options->pool);
    /* Use a char* string to initialize the aos_string_t data type. */
    aos_str_set(&options->config->endpoint, endpoint);
    aos_str_set(&options->config->access_key_id, access_key_id);
    aos_str_set(&options->config->access_key_secret, access_key_secret);
    /* Specify whether to use CNAME. */
    options->config->is_cname = 0;
    /* Configure network parameters, such as timeout. */
    options->ctl = aos_http_controller_create(options->pool, 0);
}
int main(int argc, char *argv[])
{
    /* Call aos_http_io_initialize in main() to initialize global resources, such as the
    network and memory. */
    if (aos_http_io_initialize(NULL, 0) != AOSE_OK) {
        exit(1);
    }
    /* Create a memory pool to manage memory. The implementation code is included in the
    APR library. aos_pool_t is equivalent to apr_pool_t. */
    aos_pool_t *pool;
    /* Create a new memory pool. The second parameter value is NULL, indicating that the
    pool does not inherit any other memory pool. */
    aos_pool_create(&pool, NULL);
    /* Create and initialize options. This parameter includes global configuration inform
    ation such as endpoint, access_key_id, access_key_secret, is_cname, and curl. */
    oss_request_options_t *oss_client_options;
    /* The memory consumed by options is allocated by the memory pool and is released wit
    h the pool. */
    oss_client_options = oss_request_options_create(pool);
    /* Initialize oss_client_options. */
    init_options(oss_client_options);
    /* Initialize parameters. */
    aos_string_t bucket;
    aos_status_t *resp_status = NULL;
    oss_list_object_params_t *params = NULL;
    oss_list_object_content_t *content = NULL;
    int size = 0;
    char *line = NULL;
    /* List the objects whose names contain the specified prefix. */
    char *prefix = "fun/";
    char *nextMarker = "";
    aos_str_set(&bucket, bucket_name);
    params = oss_create_list_object_params(pool);
    params->max_ret = 100;
    aos_str_set(&params->prefix, prefix);
    aos_str_set(&params->marker, nextMarker);
    printf("Object\tSize\tLastModified\n");
    /* List all objects. */
    do {
        resp_status = oss_list_object(oss_client_options, &bucket, params, NULL);
    }

```

```

        if (! aos_status_is_ok(resp_status))
        {
            printf("list object failed\n");
            break;
        }
        aos_list_for_each_entry(oss_list_object_content_t, content, &params->object_list,
node) {
            ++size;
            line = apr_psprintf(pool, "%.s\t%.s\t%.s\n", content->key.len, content->ke
y.data,
                content->size.len, content->size.data,
                content->last_modified.len, content->last_modified.data);
            printf("%s", line);
        }
        nextMarker = apr_psprintf(pool, "%.s", params->next_marker.len, params->next_mar
ker.data);
        aos_str_set(&params->marker, nextMarker);
        aos_list_init(&params->object_list);
        aos_list_init(&params->common_prefix_list);
    } while (params->truncated == AOS_TRUE);
    printf("Total %d\n", size);
    /* Release the memory pool after the request is complete. In this case, the memory al
located to various resources used for the request is released. */
    aos_pool_destroy(pool);
    /* Before you end the program, call aos_http_io_deinitialize to release the allocated
global resources. */
    aos_http_io_deinitialize();
    return 0;
}

```

- List the objects and subfolders in a folder

```

#include "oss_api.h"
#include "aos_http_io.h"
const char *endpoint = "<yourEndpoint>";
const char *access_key_id = "<yourAccessKeyId>";
const char *access_key_secret = "<yourAccessKeySecret>";
const char *bucket_name = "<yourBucketName>";
void init_options(oss_request_options_t *options)
{
    options->config = oss_config_create(options->pool);
    /* Use a char* string to initialize the aos_string_t data type. */
    aos_str_set(&options->config->endpoint, endpoint);
    aos_str_set(&options->config->access_key_id, access_key_id);
    aos_str_set(&options->config->access_key_secret, access_key_secret);
    /* Specify whether to use CNAME. */
    options->config->is_cname = 0;
    /* Configure network parameters, such as timeout. */
    options->ctl = aos_http_controller_create(options->pool, 0);
}
int main(int argc, char *argv[])
{
    /* Call aos_http_io_initialize in main() to initialize global resources, such as the

```

```

network and memory. */
    if (aos_http_io_initialize(NULL, 0) != AOSE_OK) {
        exit(1);
    }
    /* Create a memory pool to manage memory. The implementation code is included in the
    APR library. aos_pool_t is equivalent to apr_pool_t. */
    aos_pool_t *pool;
    /* Create a new memory pool. The second parameter value is NULL, indicating that the
    pool does not inherit any other memory pool. */
    aos_pool_create(&pool, NULL);
    /* Create and initialize options. This parameter includes global configuration inform
    ation such as endpoint, access_key_id, access_key_secret, is_cname, and curl. */
    oss_request_options_t *oss_client_options;
    /* The memory consumed by options is allocated by the memory pool and is released wit
    h the pool. */
    oss_client_options = oss_request_options_create(pool);
    /* Initialize oss_client_options. */
    init_options(oss_client_options);
    /* Initialize parameters. */
    aos_string_t bucket;
    aos_status_t *resp_status = NULL;
    oss_list_object_params_t *params = NULL;
    oss_list_object_content_t *content = NULL;
    oss_list_object_common_prefix_t *commonPrefix = NULL;
    int size = 0;
    char *line = NULL;
    /* List the objects whose names contain the specified prefix. */
    char *prefix = "fun/";
    char *nextMarker = "";
    /* Set the delimiter to forward slashes (/). */
    char *delimiter = "/";
    aos_str_set(&bucket, bucket_name);
    params = oss_create_list_object_params(pool);
    params->max_ret = 100;
    aos_str_set(&params->prefix, prefix);
    aos_str_set(&params->marker, nextMarker);
    aos_str_set(&params->delimiter, delimiter);
    printf("Object\tSize\tLastModified\n");
    /* List all objects. */
    do {
        resp_status = oss_list_object(oss_client_options, &bucket, params, NULL);
        if (! aos_status_is_ok(resp_status))
        {
            printf("list object failed\n");
            break;
        }
        aos_list_for_each_entry(oss_list_object_content_t, content, &params->object_list,
node) {
            ++size;
            line = apr_psprintf(pool, "%.s\t%.s\t%.s\n", content->key.len, content->ke
y.data,
                                content->size.len, content->size.data,
                                content->last_modified.len, content->last_modified.data);
            printf("%s", line);

```



```

    }
    aos_list_for_each_entry(oss_list_object_common_prefix_t, commonPrefix, &params->common_prefix_list, node) {
        line = apr_psprintf(pool, "%.s", commonPrefix->prefix.len, commonPrefix->prefix.data);
        printf("%s", line);
    }
    nextMarker = apr_psprintf(pool, "%.s", params->next_marker.len, params->next_marker.data);
    aos_str_set(&params->marker, nextMarker);
    aos_list_init(&params->object_list);
    aos_list_init(&params->common_prefix_list);
} while (params->truncated == AOS_TRUE);
printf("Total %d\n", size);
/* Release the memory pool after the request is complete. In this case, the memory allocated to various resources used for the request is released. */
aos_pool_destroy(pool);
/* Before you end the program, call aos_http_io_deinitialize to release the allocated global resources. */
aos_http_io_deinitialize();
return 0;
}

```

- List sizes of objects in a specified folder

The following code provides an example on how to query sizes of objects in a specified folder:

```

#include "oss_api.h"
#include "aos_http_io.h"
const char *endpoint = "<yourEndpoint>";
const char *access_key_id = "<yourAccessKeyId>";
const char *access_key_secret = "<yourAccessKeySecret>";
const char *bucket_name = "<yourBucketName>";
void init_options(oss_request_options_t *options)
{
    options->config = oss_config_create(options->pool);
    /* Use a char* string to initialize the aos_string_t data type. */
    aos_str_set(&options->config->endpoint, endpoint);
    aos_str_set(&options->config->access_key_id, access_key_id);
    aos_str_set(&options->config->access_key_secret, access_key_secret);
    /* Specify whether to use CNAME. */
    options->config->is_cname = 0;
    /* Configure network parameters, such as timeout. */
    options->ctl = aos_http_controller_create(options->pool, 0);
}
int64_t calculateFolderLength (aos_string_t bucketName, char* folder)
{
    int64_t size = 0;
    oss_list_object_params_t *params = NULL;
    char *nextMarker = "";
    aos_status_t *resp_status = NULL;
    aos_pool_t *p = NULL;
    oss_request_options_t *options = NULL;
    oss_list_object_content_t* content = NULL;

```

```

aos_pool_create(&p, NULL);
options = oss_request_options_create(p);
params = oss_create_list_object_params(p);
params->max_ret = 10;
aos_str_set(&params->prefix, folder);
aos_str_set(&params->marker, nextMarker);
/* List all objects. */
do {
    resp_status = oss_list_object(options, &bucketName, params, NULL);
    if (! aos_status_is_ok(resp_status))
    {
        printf("list object failed\n");
        break;
    }
    aos_list_for_each_entry(oss_list_object_content_t, content, &params->object_list,
node) {
        size += content->size.len;
    }
    nextMarker = apr_psprintf(p, "%.s", params->next_marker.len, params->next_marker
.data);
    aos_str_set(&params->marker, nextMarker);
    aos_list_init(&params->object_list);
    aos_list_init(&params->common_prefix_list);
} while (params->truncated == AOS_TRUE);
/* Release the memory pool after the request is complete. In this case, the memory al
located to various resources used for the request is released. */
aos_pool_destroy(p);
return size;
}
int main(int argc, char *argv[])
{
    /* Call aos_http_io_initialize in main() to initialize global resources, such as the
network and memory. */
    if (aos_http_io_initialize(NULL, 0) != AOSE_OK) {
        exit(1);
    }
    /* Create a memory pool to manage memory. The implementation code is included in the
APR library. aos_pool_t is equivalent to apr_pool_t. */
    aos_pool_t *pool;
    /* Create a new memory pool. The second parameter value is NULL, indicating that the
pool does not inherit any other memory pool. */
    aos_pool_create(&pool, NULL);
    /* Create and initialize options. This parameter includes global configuration inform
ation such as endpoint, access_key_id, access_key_secret, is_cname, and curl. */
    oss_request_options_t *oss_client_options;
    /* The memory consumed by options is allocated by the memory pool and is released wit
h the pool. */
    oss_client_options = oss_request_options_create(pool);
    /* Initialize oss_client_options. */
    init_options(oss_client_options);
    /* Initialize parameters. */
    aos_string_t bucket;
    aos_status_t *resp_status = NULL;
    oss_list_object_params_t *params = NULL;

```

```

    oss_list_object_content_t *content = NULL;
    oss_list_object_common_prefix_t *commonPrefix = NULL;
    int size = 0;
    char *line = NULL;
    /* List the objects whose names contain the specified prefix. */
    char *prefix = "fun/";
    char *nextMarker = "";
    /* Set the delimiter to forward slashes (/). */
    char *delimiter = "/";
    aos_str_set(&bucket, bucket_name);
    params = oss_create_list_object_params(pool);
    params->max_ret = 100;
    aos_str_set(&params->prefix, prefix);
    aos_str_set(&params->marker, nextMarker);
    aos_str_set(&params->delimiter, delimiter);
    printf("Object\tSize\tLastModified\n");
    /* List all objects. */
    do {
        resp_status = oss_list_object(oss_client_options, &bucket, params, NULL);
        if (! aos_status_is_ok(resp_status))
        {
            printf("list object failed\n");
            break;
        }
        aos_list_for_each_entry(oss_list_object_content_t, content, &params->object_list,
node) {
            ++size;
            line = apr_psprintf(pool, "%.s\t%.s\t%.s\n", content->key.len, content->key.data,
                content->size.len, content->size.data,
                content->last_modified.len, content->last_modified.data);
            printf("%s", line);
        }
        aos_list_for_each_entry(oss_list_object_common_prefix_t, commonPrefix, &params->common_prefix_list, node) {
            line = apr_psprintf(pool, "%.s", commonPrefix->prefix.len, commonPrefix->prefix.data);
            int64_t size = calculateFolderLength(bucket, commonPrefix->prefix.data);
            printf("Total %d\n", size);
        }
        nextMarker = apr_psprintf(pool, "%.s", params->next_marker.len, params->next_marker.data);
        aos_str_set(&params->marker, nextMarker);
        aos_list_init(&params->object_list);
        aos_list_init(&params->common_prefix_list);
    } while (params->truncated == AOS_TRUE);
    printf("Total %d\n", size);
    /* Release the memory pool after the request is complete. In this case, the memory allocated to various resources used for the request is released. */
    aos_pool_destroy(pool);
    /* Before you end the program, call aos_http_io_deinitialize to release the allocated global resources. */
    aos_http_io_deinitialize();
    return 0;

```

}

12.8.5. Delete objects

You can delete a single object, multiple specified objects, objects whose names contain a specified prefix, or a specified directory and all objects in the directory.

 **Warning** Deleted objects cannot be recovered. Exercise caution when you delete objects.

Delete a single object

The following code provides an example on how to delete an object named *exampleobject.txt* from a bucket named *examplebucket*:

```
#include "oss_api.h"
#include "aos_http_io.h"
/* Specify the endpoint of the region in which the bucket is located. For example, if the bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com. */
const char *endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
/* The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Using these credentials to perform operations in Object Storage Service (OSS) is a high-risk operation. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console. */
const char *access_key_id = "yourAccessKeyId";
const char *access_key_secret = "yourAccessKeySecret";
/* Specify the bucket name. Example: examplebucket. */
const char *bucket_name = "examplebucket";
/* Specify the full path of the object that you want to delete. The full path of the object cannot contain the bucket name. */
const char *object_name = "exampleobject.jpg";
void init_options(oss_request_options_t *options)
{
    options->config = oss_config_create(options->pool);
    /* Use a char* string to initialize the aos_string_t data type. */
    aos_str_set(&options->config->endpoint, endpoint);
    aos_str_set(&options->config->access_key_id, access_key_id);
    aos_str_set(&options->config->access_key_secret, access_key_secret);
    /* Specify whether to use CNAME. The value 0 indicates that CNAME is not used. */
    options->config->is_cname = 0;
    /* Configure network parameters such as the timeout period. */
    options->ctl = aos_http_controller_create(options->pool, 0);
}
int main(int argc, char *argv[])
{
    /* Call the aos_http_io_initialize method in main() to initialize global resources such as networks and memory. */
    if (aos_http_io_initialize(NULL, 0) != AOSE_OK) {
        exit(1);
    }
    /* Create a memory pool to manage memory. aos_pool_t is equivalent to apr_pool_t. The code used to create a memory pool is included in the APR library. */
    aos_pool_t *pool;
```

```

/* Create a memory pool. The second parameter is set to NULL. This value indicates that
the pool does not inherit other memory pools. */
aos_pool_create(&pool, NULL);
/* Create and initialize options. This parameter includes global configuration informat
ion such as endpoint, access_key_id, access_key_secret, is_cname, and curl. */
oss_request_options_t *oss_client_options;
/* Allocate the memory resources in the memory pool to the options. */
oss_client_options = oss_request_options_create(pool);
/* Initialize oss_client_options. */
init_options(oss_client_options);
/* Initialize the parameters. */
aos_string_t bucket;
aos_string_t object;
aos_table_t *resp_headers = NULL;
aos_status_t *resp_status = NULL;
/* Assign char* data to a bucket of the aos_string_t type. */
aos_str_set(&bucket, bucket_name);
aos_str_set(&object, object_name);
/* Delete the object. */
resp_status = oss_delete_object(oss_client_options, &bucket, &object, &resp_headers);
/* Determine whether the object is deleted. */
if (aos_status_is_ok(resp_status)) {
    printf("delete object succeed\n");
} else {
    printf("delete object failed\n");
}
/* Release the memory pool. This operation releases the memory resources allocated for
the request. */
aos_pool_destroy(pool);
/* Release the allocated global resources. */
aos_http_io_deinitialize();
return 0;
}

```

Delete multiple objects

You can delete up to 1,000 objects at a time. You can delete multiple specified objects, objects whose names contain a specified prefix, or a specified directory and all objects in the directory.

You can also configure lifecycle rules to automatically delete objects. For more information, see [Lifecycle rules based on the last modified time](#).

Delete multiple specified objects

The following code provides an example on how to delete multiple specified objects:

```

#include "oss_api.h"
#include "aos_http_io.h"
/* Specify the endpoint of the region in which the bucket is located. For example, if the b
ucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzho
u.aliyuncs.com. */
const char *endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
/* The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Us
ing these credentials to perform operations in OSS is a high-risk operation. We recommend t
hat you use a RAM user to call API operations or perform routine O&M. To create a RAM user,

```

```

log on to the RAM console. */
const char *access_key_id = "yourAccessKeyId";
const char *access_key_secret = "yourAccessKeySecret";
/* Specify the bucket name. Example: examplebucket. */
const char *bucket_name = "examplebucket";
/* Specify the full paths of the objects that you want to delete. The full paths of the objects cannot contain the bucket name. */
const char *object_name1 = "exampleobject1.jpg";
const char *object_name2 = "testobject2.png";
void init_options(oss_request_options_t *options)
{
    options->config = oss_config_create(options->pool);
    /* Use a char* string to initialize the aos_string_t data type. */
    aos_str_set(&options->config->endpoint, endpoint);
    aos_str_set(&options->config->access_key_id, access_key_id);
    aos_str_set(&options->config->access_key_secret, access_key_secret);
    /* Specify whether to use CNAME. The value 0 indicates that CNAME is not used. */
    options->config->is_cname = 0;
    /* Configure network parameters such as the timeout period. */
    options->ctl = aos_http_controller_create(options->pool, 0);
}
int main(int argc, char *argv[])
{
    /* Call the aos_http_io_initialize method in main() to initialize global resources such as networks and memory. */
    if (aos_http_io_initialize(NULL, 0) != AOSE_OK) {
        exit(1);
    }
    /* Create a memory pool to manage memory. aos_pool_t is equivalent to apr_pool_t. The code used to create a memory pool is included in the APR library. */
    aos_pool_t *pool;
    /* Create a memory pool. The second parameter is set to NULL. This value indicates that the pool does not inherit other memory pools. */
    aos_pool_create(&pool, NULL);
    /* Create and initialize options. This parameter includes global configuration information such as endpoint, access_key_id, access_key_secret, is_cname, and curl. */
    oss_request_options_t *oss_client_options;
    /* Allocate the memory resources in the memory pool to the options. */
    oss_client_options = oss_request_options_create(pool);
    /* Initialize oss_client_options. */
    init_options(oss_client_options);
    /* Initialize the parameters. */
    aos_string_t bucket;
    aos_string_t object1;
    aos_string_t object2;
    int is_quiet = 1;
    aos_table_t *resp_headers = NULL;
    aos_status_t *resp_status = NULL;
    aos_str_set(&bucket, bucket_name);
    aos_str_set(&object1, object_name1);
    aos_str_set(&object2, object_name2);
    /* Create a list of objects that you want to delete. */
    aos_list_t object_list;
    aos_list_t deleted_object_list;

```

```

    oss_object_key_t *content1;
    oss_object_key_t *content2;
    aos_list_init(&object_list);
    aos_list_init(&deleted_object_list);
    content1 = oss_create_oss_object_key(pool);
    aos_str_set(&content1->key, object_name1);
    aos_list_add_tail(&content1->node, &object_list);
    content2 = oss_create_oss_object_key(pool);
    aos_str_set(&content2->key, object_name2);
    aos_list_add_tail(&content2->node, &object_list);
    /* Delete the objects in the list. `is_quiet` specifies whether to return the deletion
    result. */
    resp_status = oss_delete_objects(oss_client_options, &bucket, &object_list, is_quiet, &
    resp_headers, &deleted_object_list);
    if (aos_status_is_ok(resp_status)) {
        printf("delete objects succeeded\n");
    } else {
        printf("delete objects failed\n");
    }
    /* Release the memory pool. This operation releases the memory resources allocated for
    the request. */
    aos_pool_destroy(pool);
    /* Release the allocated global resources. */
    aos_http_io_deinitialize();
    return 0;
}

```

Delete multiple objects with a specified prefix or in a specified directory

The following code provides an example on how to delete multiple objects with a specified prefix and how to delete a specified directory and all objects in the directory.

 **Warning** If the value of `object_prefix` is an empty string or set to `NULL` in the following code, all objects in the bucket are deleted. Exercise caution when you configure this value.

```

#include "oss_api.h"
#include "aos_http_io.h"
/* Specify the endpoint of the region in which the bucket is located. For example, if the b
ucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzho
u.aliyuncs.com. */
const char *endpoint = "https://oss-cn-hangzhou.aliyuncs.com";
/* The AccessKey pair of an Alibaba Cloud account has permissions on all API operations. Us
ing these credentials to perform operations in OSS is a high-risk operation. We recommend t
hat you use a RAM user to call API operations or perform routine O&M. To create a RAM user,
log on to the RAM console. */
const char *access_key_id = "yourAccessKeyId";
const char *access_key_secret = "yourAccessKeySecret";
/* Specify the bucket name. Example: examplebucket. */
const char *bucket_name = "examplebucket";
/* Specify the prefix of the objects that you want to delete. If you want to delete all obj
ects whose names contain the src prefix, set the prefix to src. After you set the prefix to
src, all non-directory objects whose names contain the src prefix, the src directory, and a

```

```

    /* If you want to delete only the src directory and all objects in the directory, set the p
    refix to src/. */
    /* const char *object_prefix = "src/"; */
    void init_options(oss_request_options_t *options)
    {
        options->config = oss_config_create(options->pool);
        /* Use a char* string to initialize the aos_string_t data type. */
        aos_str_set(&options->config->endpoint, endpoint);
        aos_str_set(&options->config->access_key_id, access_key_id);
        aos_str_set(&options->config->access_key_secret, access_key_secret);
        /* Specify whether to use CNAME. The value 0 indicates that CNAME is not used. */
        options->config->is_cname = 0;
        /* Configure network parameters such as the timeout period. */
        options->ctl = aos_http_controller_create(options->pool, 0);
    }
    int main(int argc, char *argv[])
    {
        /* Call the aos_http_io_initialize method in main() to initialize global resources such
        as networks and memory. */
        if (aos_http_io_initialize(NULL, 0) != AOSE_OK) {
            exit(1);
        }
        /* Create a memory pool to manage memory. aos_pool_t is equivalent to apr_pool_t. The c
        ode used to create a memory pool is included in the APR library. */
        aos_pool_t *pool;
        /* Create a memory pool. The second parameter is set to NULL. This value indicates that
        the pool does not inherit other memory pools. */
        aos_pool_create(&pool, NULL);
        /* Create and initialize options. This parameter includes global configuration informat
        ion such as endpoint, access_key_id, access_key_secret, is_cname, and curl. */
        oss_request_options_t *oss_client_options;
        /* Allocate the memory resources in the memory pool to the options. */
        oss_client_options = oss_request_options_create(pool);
        /* Initialize oss_client_options. */
        init_options(oss_client_options);
        /* Initialize the parameters. */
        aos_string_t bucket;
        aos_string_t prefix;
        aos_status_t *resp_status = NULL;
        aos_str_set(&bucket, bucket_name);
        aos_str_set(&prefix, object_prefix);
        /* Delete the objects whose names contain the specified prefix. */
        resp_status = oss_delete_objects_by_prefix(oss_client_options, &bucket, &prefix);
        if (aos_status_is_ok(resp_status)) {
            printf("delete objects by prefix succeeded\n");
        } else {
            printf("delete objects by prefix failed\n");
        }
        /* Release the memory pool. This operation releases the memory resources allocated for
        the request. */
        aos_pool_destroy(pool);
        /* Release the allocated global resources. */
    }

```



```
aos_http_io_deinitialize();
return 0;
}
```

References

- For the complete sample code that is used to delete a single object or multiple objects, visit [GitHub](#).
- For more information about the API operation that you can call to delete a single object, see [DeleteObject](#).
- For more information about the API operation that you can call to delete multiple objects, see [DeleteMultipleObjects](#).

12.8.6. Copy objects

This topic describes how to copy an object from a source bucket to a destination bucket within the same region. The source bucket and the destination bucket can be the same or different buckets.

Copy a small-sized object

You must note the following limits before copy a small-sized object:

- The user must have the read and write permission on the source object.
- You cannot copy an object from a region to another region. For example, you cannot copy an object from a bucket in Hangzhou region to a bucket in Qingdao.
- The object size must be smaller than 1 GB.

Run the following code to copy a small-sized object:

```
#include "oss_api.h"
#include "aos_http_io.h"
const char *endpoint = "<yourEndpoint>";
const char *access_key_id = "<yourAccessKeyId>";
const char *access_key_secret = "<yourAccessKeySecret>";
const char *bucket_name = "<yourBucketName>";
const char *object_name = "<yourObjectName>";
const char *source_bucket_name = "<yourSourceBucketName>";
const char *source_object_name = "<yourSourceObjectName>";
const char *dest_bucket_name = "<yourDestBucketName>";
const char *dest_object_name = "<yourDestObjectName>";
void init_options(oss_request_options_t *options)
{
    options->config = oss_config_create(options->pool);
    /* Use a char* string to initialize aos_string_t. */
    aos_str_set(&options->config->endpoint, endpoint);
    aos_str_set(&options->config->access_key_id, access_key_id);
    aos_str_set(&options->config->access_key_secret, access_key_secret);
    /* Determine whether the CNAME is used. 0 indicates that the CNAME is not used. */
    options->config->is_cname = 0;
    /* Used to configure network parameters, such as timeout. */
    options->ctl = aos_http_controller_create(options->pool, 0);
}
int main(int argc, char *argv[])
{

```

```

/* Call the aos_http_io_initialize method in main() to initialize global resources, such as networks and memories. */
if (aos_http_io_initialize(NULL, 0) != AOSE_OK) {
    exit(1);
}

/* Memory pool used to manage memories, which is equivalent to apr_pool_t. The implementation code is included in the apr library. */
aos_pool_t *pool;
/* Re-create a new memory pool. The second parameter is NULL, indicating that it does not inherit from any other memory pools. */
aos_pool_create(&pool, NULL);
/* Create and initialize options. This parameter mainly includes global configuration information, such as endpoint, access_key_id, access_key_secret, is_cname, and curl. */
oss_request_options_t *oss_client_options;
/* Allocate memories in the memory pool to options. */
oss_client_options = oss_request_options_create(pool);
/* Use oss_client_options to initialize client options. */
init_options(oss_client_options);
/* Initialization parameters. */
aos_string_t source_bucket;
aos_string_t source_object;
aos_string_t dest_bucket;
aos_string_t dest_object;
aos_table_t *headers = NULL;
aos_table_t *resp_headers = NULL;
aos_status_t *resp_status = NULL;
aos_str_set(&source_bucket, source_bucket_name);
aos_str_set(&source_object, source_object_name);
aos_str_set(&dest_bucket, dest_bucket_name);
aos_str_set(&dest_object, dest_object_name);
headers = aos_table_make(pool, 0);
/* Copy an object. */
resp_status = oss_copy_object(oss_client_options, &source_bucket, &source_object, &dest_bucket, &dest_object, headers, &resp_headers);
if (aos_status_is_ok(resp_status)) {
    printf("copy object succeeded\n");
} else {
    printf("copy object failed\n");
}
/* Release the memory pool, that is, memories allocated to resources during the request. */
aos_pool_destroy(pool);
/* Release the allocated global resources. */
aos_http_io_deinitialize();
return 0;
}

```

Copy a large-sized object

For objects larger than 1 GB, we recommend that you should use `oss_upload_part_copy` to perform multipart copy (UploadPartCopy). Run the following code for multipart copy:

```

#include "oss_api.h"
#include "aos_http_io.h"

```

```

#include <sys/stat.h>
const char *endpoint = "<yourEndpoint>";
const char *access_key_id = "<yourAccessKeyId>";
const char *access_key_secret = "<yourAccessKeySecret>";
const char *source_bucket_name = "<yourSourceBucketName>";
const char *source_object_name = "<yourSourceObjectName>";
const char *dest_bucket_name = "<yourDestBucketName>";
const char *dest_object_name = "<yourDestObjectName>";
const char *local_filename = "<yourLocalFilename>";
void init_options(oss_request_options_t *options)
{
    options->config = oss_config_create(options->pool);
    /* Use a char* string to initialize aos_string_t. */
    aos_str_set(&options->config->endpoint, endpoint);
    aos_str_set(&options->config->access_key_id, access_key_id);
    aos_str_set(&options->config->access_key_secret, access_key_secret);
    /* Determine whether the CNAME is used. 0 indicates that the CNAME is not used. */
    options->config->is_cname = 0;
    /* Used to configure network parameters, such as timeout. */
    options->ctl = aos_http_controller_create(options->pool, 0);
}
int64_t get_file_size(const char *file_path)
{
    int64_t filesize = -1;
    struct stat statbuff;
    if(stat(file_path, &statbuff) < 0){
        return filesize;
    } else {
        filesize = statbuff.st_size;
    }
    return filesize;
}
int main(int argc, char *argv[])
{
    /* Call the aos_http_io_initialize method in main() to initialize global resources, such as networks and memories. */
    if (aos_http_io_initialize(NULL, 0) != AOSE_OK) {
        exit(1);
    }

    /* Memory pool used to manage memories, which is equivalent to apr_pool_t. The implementation code is included in the apr library. */
    aos_pool_t *pool;
    /* Re-create a new memory pool. The second parameter is NULL, indicating that it does not inherit from any other memory pools. */
    aos_pool_create(&pool, NULL);
    /* Create and initialize options. This parameter mainly includes global configuration information, such as endpoint, access_key_id, access_key_secret, is_cname, and curl. */
    oss_request_options_t *oss_client_options;
    /* Allocate memories in the memory pool to options. */
    oss_client_options = oss_request_options_create(pool);
    /* Use oss_client_options to initialize client options. */
    init_options(oss_client_options);
    /* Initialization parameters. */
    aos_string_t dest_bucket;

```

```

aos_string_t dest_object;
aos_string_t upload_id;
aos_table_t *headers = NULL;
aos_table_t *resp_headers = NULL;
aos_status_t *resp_status = NULL;
oss_list_upload_part_params_t *list_upload_part_params;
oss_upload_part_copy_params_t *upload_part_copy_params1;
oss_upload_part_copy_params_t *upload_part_copy_params2;
oss_list_part_content_t *part_content;
aos_list_t complete_part_list;
oss_complete_part_content_t *complete_content;
aos_table_t *list_part_resp_headers = NULL;
aos_table_t *complete_resp_headers = NULL;
int part1 = 1;
int part2 = 2;
int64_t range_start1 = 0;
int64_t range_end1 = 6000000; //not less than 5MB
int64_t range_start2 = 6000001;
int64_t range_end2;
int max_ret = 1000;
headers = aos_table_make(pool, 0);
aos_str_set(&dest_bucket, dest_bucket_name);
aos_str_set(&dest_object, dest_object_name);
resp_status = oss_init_multipart_upload(oss_client_options, &dest_bucket, &dest_object,
&upload_id, headers, &resp_headers);
if (aos_status_is_ok(resp_status)) {
    printf("init multipart upload succeeded, upload_id is %s\n", upload_id.data);
} else {
    printf("init multipart upload failed\n");
}
/* Copy the first part of the object. */
upload_part_copy_params1 = oss_create_upload_part_copy_params(pool);
aos_str_set(&upload_part_copy_params1->source_bucket, source_bucket_name);
aos_str_set(&upload_part_copy_params1->source_object, source_object_name);
aos_str_set(&upload_part_copy_params1->dest_bucket, dest_bucket_name);
aos_str_set(&upload_part_copy_params1->dest_object, dest_object_name);
aos_str_set(&upload_part_copy_params1->upload_id, upload_id.data);
upload_part_copy_params1->part_num = part1;
upload_part_copy_params1->range_start = range_start1;
upload_part_copy_params1->range_end = range_end1;
headers = aos_table_make(pool, 0);
resp_status = oss_upload_part_copy(oss_client_options, upload_part_copy_params1, headers,
&resp_headers);
if (aos_status_is_ok(resp_status)) {
    printf("upload part 1 copy succeeded\n");
} else {
    printf("upload part 1 copy failed\n");
}
/* Copy the second part of the object. */
range_end2 = get_file_size(local_filename) - 1;
upload_part_copy_params2 = oss_create_upload_part_copy_params(pool);
aos_str_set(&upload_part_copy_params2->source_bucket, source_bucket_name);
aos_str_set(&upload_part_copy_params2->source_object, source_object_name);
aos_str_set(&upload_part_copy_params2->dest_bucket, dest_bucket_name);

```

```

aos_str_set(&upload_part_copy_params2->dest_object, dest_object_name);
aos_str_set(&upload_part_copy_params2->upload_id, upload_id.data);
upload_part_copy_params2->part_num = part2;
upload_part_copy_params2->range_start = range_start2;
upload_part_copy_params2->range_end = range_end2;
headers = aos_table_make(pool, 0);
resp_status = oss_upload_part_copy(oss_client_options, upload_part_copy_params2, headers, &resp_headers);
if (aos_status_is_ok(resp_status)) {
    printf("upload part 2 copy succeeded\n");
} else {
    printf("upload part 2 copy failed\n");
}
/* Lists the parts. */
headers = aos_table_make(pool, 0);
list_upload_part_params = oss_create_list_upload_part_params(pool);
list_upload_part_params->max_ret = max_ret;
aos_list_init(&complete_part_list);
resp_status = oss_list_upload_part(oss_client_options, &dest_bucket, &dest_object, &upload_id, list_upload_part_params, &list_part_resp_headers);
aos_list_for_each_entry(oss_list_part_content_t, part_content, &list_upload_part_params->part_list, node) {
    complete_content = oss_create_complete_part_content(pool);
    aos_str_set(&complete_content->part_number, part_content->part_number.data);
    aos_str_set(&complete_content->etag, part_content->etag.data);
    aos_list_add_tail(&complete_content->node, &complete_part_list);
}
/* Complete the multipart copy. */
resp_status = oss_complete_multipart_upload(oss_client_options, &dest_bucket, &dest_object, &upload_id, &complete_part_list, headers, &complete_resp_headers);
if (aos_status_is_ok(resp_status)) {
    printf("complete multipart upload succeeded\n");
} else {
    printf("complete multipart upload failed\n");
}
/* Release the memory pool, that is, memories allocated to resources during the request. */
aos_pool_destroy(pool);
/* Release the allocated global resources. */
aos_http_io_deinitialize();
return 0;
}

```

12.8.7. Restore Archive objects

You must restore an Archive object before you can read it. This topic describes how to restore Archive objects.

The following section describes the status of an Archive object throughout the restoration process:

- By default, the Archive object is in the frozen state.
- After you submit a request to restore the object, the object enters the restoring state. It takes about one minute to complete the restoration task.

- After the Object Storage Service (OSS) server completes the restoration task, the object enters the restored state and can be read. By default, the restored state lasts 24 hours and can be extended by another 24 hours after you call `RestoreObject` again. During one restoration process, `RestoreObject` can be called up to seven times on an object. In other words, an object can remain in the restored state for up to seven days.
- After the restored state expires, the object returns to the frozen state.

The following code provides an example on how to restore an Archive object:

```
#include "oss_api.h"
#include "aos_http_io.h"
const char *endpoint = "<yourEndpoint>";
const char *access_key_id = "<yourAccessKeyId>";
const char *access_key_secret = "<yourAccessKeySecret>";
const char *bucket_name = "<yourBucketName>";
const char *object_name = "<yourObjectName>";
const char *object_content = "More than just cloud.";
void init_options(oss_request_options_t *options)
{
    options->config = oss_config_create(options->pool);
    /* Use a char* string to initialize the aos_string_t data type. */
    aos_str_set(&options->config->endpoint, endpoint);
    aos_str_set(&options->config->access_key_id, access_key_id);
    aos_str_set(&options->config->access_key_secret, access_key_secret);
    /* Specify whether to use CNAME. The value of 0 indicates that CNAME is not used. */
    options->config->is_cname = 0;
    /* Configure network parameters such as the timeout period. */
    options->ctl = aos_http_controller_create(options->pool, 0);
}
int main(int argc, char *argv[])
{
    /* Call the aos_http_io_initialize method in main() to initialize global resources such
    as networks and memory. */
    if (aos_http_io_initialize(NULL, 0) != AOSE_OK) {
        exit(1);
    }
    /* Create a memory pool to manage memory. aos_pool_t is equivalent to apr_pool_t. The c
    ode that is used to create a memory pool is included in the APR library. */
    aos_pool_t *pool;
    /* Create a memory pool. The value of the second parameter is NULL. This value specifie
    s that the pool does not inherit other memory pools. */
    aos_pool_create(&pool, NULL);
    /* Create and initialize options. This parameter includes global configuration informat
    ion such as endpoint, access_key_id, access_key_secret, is_cname, and curl. */
    oss_request_options_t *oss_client_options;
    /* Allocate memory resources in the memory pool to the options. */
    oss_client_options = oss_request_options_create(pool);
    /* Initialize oss_client_options. */
    init_options(oss_client_options);
    /* Initialize the parameters. */
    aos_string_t bucket;
    aos_string_t object;
    aos_list_t buffer;
    oss_acl_t oss_acl = OSS_ACL_PRIVATE;
```

```
aos_buf_t *content = NULL;
aos_table_t *headers = NULL;
aos_table_t *resp_headers = NULL;
aos_status_t *resp_status = NULL;
aos_str_set(&bucket, bucket_name);
aos_str_set(&object, object_name);
headers = aos_table_make(pool, 0);
/* Create an Archive bucket. */
resp_status = oss_create_bucket_with_storage_class(oss_client_options, &bucket, oss_acl
, OSS_STORAGE_CLASS_ARCHIVE, &resp_headers);
if (aos_status_is_ok(resp_status)) {
    printf("create bucket succeeded\n");
} else {
    printf("create bucket failed\n");
}
aos_list_init(&buffer);
content = aos_buf_pack(oss_client_options->pool, object_content, strlen(object_content)
);
aos_list_add_tail(&content->node, &buffer);
/* Upload an object. */
resp_status = oss_put_object_from_buffer(oss_client_options, &bucket, &object, &buffer,
headers, &resp_headers);
if (aos_status_is_ok(resp_status)) {
    printf("put object from buffer succeeded\n");
} else {
    printf("put object from buffer failed\n");
}
/* Restore the uploaded object. */
do {
    headers = aos_table_make(pool, 0);
    resp_status = oss_restore_object(oss_client_options, &bucket, &object, headers, &re
sp_headers);
    printf("restore object resp_status->code: %d \n", resp_status->code);
    if (resp_status->code != 409) {
        break;
    } else {
        printf("restore object is already in progress, resp_status->code: %d \n", resp_
status->code);
        apr_sleep(5000);
    }
} while (1);
/* Release the memory pool. This operation releases memory resources that are allocated
for the request. */
aos_pool_destroy(pool);
/* Release the allocated global resources. */
aos_http_io_deinitialize();
return 0;
}
```

For more information about the Archive storage class, see [Overview](#). For more information about related status codes, see [RestoreObject](#) in API Reference.

12.8.8. Manage symbolic links

This topic describes how to create a symbolic link and obtain the name of the object to which the symbolic link points.

Create a symbolic link

A symbolic link is a special object that maps to an object similar to a shortcut used in Windows. You can configure user-defined Object Meta for symbolic links.

Run the following code to create a symbolic link:

```
#include "oss_api.h"
#include "aos_http_io.h"
const char *endpoint = "<yourEndpoint>";
const char *access_key_id = "<yourAccessKeyId>";
const char *access_key_secret = "<yourAccessKeySecret>";
const char *bucket_name = "<yourBucketName>";
const char *object_name = "<yourObjectName>";
const char *link_object_name = "<yourLinkObjectName>";
void init_options(oss_request_options_t *options)
{
    options->config = oss_config_create(options->pool);
    /* Use a char* string to initialize the aos_string_t type. */
    aos_str_set(&options->config->endpoint, endpoint);
    aos_str_set(&options->config->access_key_id, access_key_id);
    aos_str_set(&options->config->access_key_secret, access_key_secret);
    /* Determine whether the CNAME is used. 0 indicates that the CNAME is not used. */
    options->config->is_cname = 0;
    /* Configure network parameters, such as timeout periods. */
    options->ctl = aos_http_controller_create(options->pool, 0);
}
int main(int argc, char *argv[])
{
    /* Use the aos_http_io_initialize method in main() to initialize global resources such
    as the network and memory. */
    if (aos_http_io_initialize(NULL, 0) != AOSE_OK) {
        exit(1);
    }
    /* Create a memory pool to manage memory resources. aos_pool_t *pool is equivalent to a
    pr_pool_t. The implementation code is included in the APR library. */
    aos_pool_t *pool;
    /* Create a new memory pool. The second parameter is NULL. This indicates that the pool
    does not inherit any other memory pool. */
    aos_pool_create(&pool, NULL);
    /* Create and initialize options. This parameter includes global configuration informat
    ion such as endpoint, access_key_id, access_key_secret, is_cname, and curl. */
    oss_request_options_t *oss_client_options;
    /* Allocate the memory resources in the memory pool to the options. */
    oss_client_options = oss_request_options_create(pool);
    /* Use oss_client_options to initialize the client options. */
    init_options(oss_client_options);
    /* Initialize the parameters. */
    aos_string_t bucket;
```



```

aos_string_t object;
aos_string_t sym_object;
aos_table_t *resp_headers = NULL;
aos_status_t *resp_status = NULL;
aos_str_set(&bucket, bucket_name);
aos_str_set(&object, object_name);
aos_str_set(&sym_object, link_object_name);
resp_status = oss_put_symlink(oss_client_options, &bucket, &sym_object, &object, &resp_
headers);
if (aos_status_is_ok(resp_status)) {
    printf("put symlink succeeded\n");
} else {
    printf("put symlink failed\n");
}
/* Release the memory pool. This releases the memory resources that are allocated durin
g the request. */
aos_pool_destroy(pool);
/* Release the allocated global resources. */
aos_http_io_deinitialize();
return 0;
}

```

Obtain the name of the object to which the symbolic link points

To obtain a symbolic link, you must have the read permissions on it. The following code provides an example on how to obtain the name of the object to which the symbolic link points:

```

#include "oss_api.h"
#include "aos_http_io.h"
const char *endpoint = "<yourEndpoint>";
const char *access_key_id = "<yourAccessKeyId>";
const char *access_key_secret = "<yourAccessKeySecret>";
const char *bucket_name = "<yourBucketName>";
const char *link_object_name = "<yourLinkObjectName>";
void init_options(oss_request_options_t *options)
{
    options->config = oss_config_create(options->pool);
    /* Use a char* string to initialize the aos_string_t type. */
    aos_str_set(&options->config->endpoint, endpoint);
    aos_str_set(&options->config->access_key_id, access_key_id);
    aos_str_set(&options->config->access_key_secret, access_key_secret);
    /* Determine whether the CNAME is used. 0 indicates that the CNAME is not used. */
    options->config->is_cname = 0;
    /* Configure network parameters, such as timeout periods. */
    options->ctl = aos_http_controller_create(options->pool, 0);
}
int main(int argc, char *argv[])
{
    /* Use the aos_http_io_initialize method in main() to initialize global resources such
as the network and memory. */
    if (aos_http_io_initialize(NULL, 0) != AOSE_OK) {
        exit(1);
    }
    /* Create a memory pool to manage memory resources. aos_pool_t *pool is equivalent to a

```

```

pr_pool_t. The implementation code is included in the APR library. */
aos_pool_t *pool;
/* Create a new memory pool. The second parameter is NULL. This indicates that the pool
does not inherit any other memory pool. */
aos_pool_create(&pool, NULL);
/* Create and initialize options. This parameter includes global configuration informat
ion such as endpoint, access_key_id, access_key_secret, is_cname, and curl. */
oss_request_options_t *oss_client_options;
/* Allocate the memory resources in the memory pool to the options. */
oss_client_options = oss_request_options_create(pool);
/* Use oss_client_options to initialize the client options. */
init_options(oss_client_options);
/* Initialize the parameters. */
aos_string_t bucket;
aos_string_t link_object;
aos_table_t *resp_headers = NULL;
aos_status_t *resp_status = NULL;
char *target_object_name = NULL;
aos_str_set(&bucket, bucket_name);
aos_str_set(&link_object, link_object_name);
resp_status = oss_get_symlink(oss_client_options, &bucket, &link_object, &resp_headers)
;
if (aos_status_is_ok(resp_status)) {
    printf("get symlink succeeded\n");
    target_object_name = (char*)(apr_table_get(resp_headers, OSS_CANNONICALIZED_HEADER_
SYMLINK));
    printf("target_object_name: %s \n", target_object_name);
} else {
    printf("get symlink failed\n");
}
/* Release the memory pool. This releases the memory resources that are allocated durin
g the request. */
aos_pool_destroy(pool);
/* Release the allocated global resources. */
aos_http_io_deinitialize();
return 0;
}

```

12.9. Authorized access

This topic describes how to authorize temporary access to Object Storage Service (OSS) by using Security Token Service (STS) or a signed URL.

Use STS to authorize temporary access

You can use Alibaba Cloud STS to authorize temporary access to OSS. STS is a web service that provides temporary access tokens for users. You can use STS to grant a set of temporary access credentials that have a custom validity period and custom permissions to a third-party application or a RAM user managed by you. For more information about STS, see [What is STS?](#)

STS provides the following benefits:

- You need only to generate an access token and send the access token to a third-party application,

instead of exposing your AccessKey pair to the third-party application. You can define the access permissions and validity period of this token.

- The token automatically expires after the validity period. Therefore, you do not need to manually revoke the access permissions of a token.

```
#include "oss_api.h"
#include "aos_http_io.h"
const char *endpoint = "<yourEndpoint>";
const char *access_key_id = "<yourAccessKeyId>";
const char *access_key_secret = "<yourAccessKeySecret>";
const char *sts_token = "<yourStsToken>";
const char *bucket_name = "<yourBucketName>";
const char *object_name = "<yourObjectName>";
const char *object_content = "More than just cloud.";
void init_options(oss_request_options_t *options)
{
    options->config = oss_config_create(options->pool);
    /* Use a char* string to initialize the aos_string_t data type. */
    aos_str_set(&options->config->endpoint, endpoint);
    aos_str_set(&options->config->access_key_id, access_key_id);
    aos_str_set(&options->config->access_key_secret, access_key_secret);
    aos_str_set(&options->config->sts_token, sts_token);
    /* Specify whether to use CNAME to access OSS. A value of 0 indicates that CNAME is not
    used. */
    options->config->is_cname = 0;
    /* Configure network parameters such as the timeout period. */
    options->ctl = aos_http_controller_create(options->pool, 0);
}
int main(int argc, char *argv[])
{
    /* Call aos_http_io_initialize in main() to initialize global resources such as the net
    work and memory. */
    if (aos_http_io_initialize(NULL, 0) != AOSE_OK) {
        exit(1);
    }
    /* Create a memory pool to manage memory. aos_pool_t is equivalent to apr_pool_t. The c
    ode to create a memory pool is included in the APR library. */
    aos_pool_t *pool;
    /* Create a memory pool. The second parameter is NULL. This parameter indicates that th
    e pool does not inherit any other memory pool. */
    aos_pool_create(&pool, NULL);
    /* Create and initialize options. This parameter includes global configuration informat
    ion such as endpoint, access_key_id, access_key_secret, is_cname, and curl. */
    oss_request_options_t *oss_client_options;
    /* Allocate a memory chunk in the memory pool to options. */
    oss_client_options = oss_request_options_create(pool);
    /* Call oss_client_options to initialize the client options. */
    init_options(oss_client_options);
    /* Initialize the parameters. */
    aos_string_t bucket;
    aos_string_t object;
    aos_list_t buffer;
    aos_buf_t *content = NULL;
    aos_table_t *headers = NULL;
```

```

aos_table_t *resp_headers = NULL;
aos_status_t *resp_status = NULL;
/* Assign the char* data to the bucket. */
aos_str_set(&bucket, bucket_name);
aos_str_set(&object, object_name);
aos_list_init(&buffer);
content = aos_buf_pack(oss_client_options->pool, object_content, strlen(object_content)
);
aos_list_add_tail(&content->node, &buffer);
/* Upload the object. */
resp_status = oss_put_object_from_buffer(oss_client_options, &bucket, &object, &buffer,
headers, &resp_headers);
/* Determine whether the object is uploaded. */
if (aos_status_is_ok(resp_status)) {
    printf("put object from buffer succeeded\n");
} else {
    printf("put object from buffer failed\n");
}
/* Release the memory pool. The memory allocated to various resources used for the requ
est is released. */
aos_pool_destroy(pool);
/* Release the allocated global resources. */
aos_http_io_deinitialize();
return 0;
}

```

Note

Use a signed URL to authorize temporary access

You can generate a signed URL and provide the URL to a visitor for temporary access. When you generate a signed URL, you can specify the validity period of the URL to limit the period of access from visitors. By default, the validity period of a signed URL is 3,600 seconds. The maximum validity period of a signed URL is 32,400 seconds.

This section provides examples on how to use a signed URL to authorize temporary access.

- Use a signed URL to upload an object

The following code provides an example on how to upload an object by using a signed URL:

```

#include "oss_api.h"
#include "aos_http_io.h"
const char *endpoint = "<yourEndpoint>";
const char *access_key_id = "<yourAccessKeyId>";
const char *access_key_secret = "<yourAccessKeySecret>";
const char *bucket_name = "<yourBucketName>";
const char *object_name = "<yourObjectName>";
const char *local_filename = "<yourLocalFilename>";
void init_options(oss_request_options_t *options)
{
    options->config = oss_config_create(options->pool);
    /* Use a char* string to initialize the aos string t data type. */
}

```

```

aos_str_set(&options->config->endpoint, endpoint);
aos_str_set(&options->config->access_key_id, access_key_id);
aos_str_set(&options->config->access_key_secret, access_key_secret);
/* Specify whether to use CNAME to access OSS. A value of 0 indicates that CNAME is not used. */
options->config->is_cname = 0;
/* Configure network parameters such as the timeout period. */
options->ctl = aos_http_controller_create(options->pool, 0);
}
int main(int argc, char *argv[])
{
    /* Call aos_http_io_initialize in main() to initialize global resources such as the network and memory. */
    if (aos_http_io_initialize(NULL, 0) != AOSE_OK) {
        exit(1);
    }
    /* Create a memory pool to manage memory. aos_pool_t is equivalent to apr_pool_t. The code to create a memory pool is included in the APR library. */
    aos_pool_t *pool;
    /* Create a memory pool. The second parameter is NULL. This parameter indicates that the pool does not inherit any other memory pool. */
    aos_pool_create(&pool, NULL);
    /* Create and initialize options. This parameter includes global configuration information such as endpoint, access_key_id, access_key_secret, is_cname, and curl. */
    oss_request_options_t *oss_client_options;
    /* Allocate a memory chunk in the memory pool to options. */
    oss_client_options = oss_request_options_create(pool);
    /* Call oss_client_options to initialize the client options. */
    init_options(oss_client_options);
    /* Initialize the parameters. */
    aos_string_t bucket;
    aos_string_t object;
    aos_string_t file;
    aos_table_t *headers = NULL;
    aos_table_t *resp_headers = NULL;
    aos_status_t *resp_status = NULL;
    aos_http_request_t *req;
    apr_time_t now;
    char *url_str;
    aos_string_t url;
    int64_t expire_time;
    int one_hour = 3600;
    aos_str_set(&bucket, bucket_name);
    aos_str_set(&object, object_name);
    aos_str_set(&file, local_filename);
    expire_time = now / 1000000 + one_hour;
    headers = aos_table_make(pool, 0);
    req = aos_http_request_create(pool);
    req->method = HTTP_PUT;
    now = apr_time_now(); /* Unit: microseconds */
    expire_time = now / 1000000 + one_hour;
    /* Generate a signed URL. */
    url_str = oss_gen_signed_url(oss_client_options, &bucket, &object, expire_time, req);
    aos_str_set(&url, url_str);

```

```

printf("Temporary upload URL: %s\n", url_str);
/* Use the signed URL to upload an object. */
resp_status = oss_put_object_from_file_by_url(oss_client_options, &url, &file, headers, &resp_headers);
if (aos_status_is_ok(resp_status)) {
    printf("put objects by signed url succeeded\n");
} else {
    printf("put objects by signed url failed\n");
}
/* Release the memory pool. The memory allocated to various resources used for the request is released. */
aos_pool_destroy(pool);
/* Release the allocated global resources. */
aos_http_io_deinitialize();
return 0;
}

```

- Use a signed URL to download an object

The following code provides an example on how to download a specified object by using a signed URL:

```

#include "oss_api.h"
#include "aos_http_io.h"
const char *endpoint = "<yourEndpoint>";
const char *access_key_id = "<yourAccessKeyId>";
const char *access_key_secret = "<yourAccessKeySecret>";
const char *bucket_name = "<yourBucketName>";
const char *object_name = "<yourObjectName>";
const char *local_filename = "<yourLocalFilename>";
void init_options(oss_request_options_t *options)
{
    options->config = oss_config_create(options->pool);
    /* Use a char* string to initialize the aos_string_t data type. */
    aos_str_set(&options->config->endpoint, endpoint);
    aos_str_set(&options->config->access_key_id, access_key_id);
    aos_str_set(&options->config->access_key_secret, access_key_secret);
    /* Specify whether to use CNAME to access OSS. A value of 0 indicates that CNAME is not used. */
    options->config->is_cname = 0;
    /* Configure network parameters such as the timeout period. */
    options->ctl = aos_http_controller_create(options->pool, 0);
}
int main(int argc, char *argv[])
{
    /* Call aos_http_io_initialize in main() to initialize global resources such as the network and memory. */
    if (aos_http_io_initialize(NULL, 0) != AOSE_OK) {
        exit(1);
    }
    /* Create a memory pool to manage memory. aos_pool_t is equivalent to apr_pool_t. The code to create a memory pool is included in the APR library. */
    aos_pool_t *pool;
    /* Create a memory pool. The second parameter is NULL. This parameter indicates that the pool does not inherit any other memory pool. */

```

```

aos_pool_create(&pool, NULL);
/* Create and initialize options. This parameter includes global configuration information such as endpoint, access_key_id, access_key_secret, is_cname, and curl. */
oss_request_options_t *oss_client_options;
/* Allocate a memory chunk in the memory pool to options. */
oss_client_options = oss_request_options_create(pool);
/* Call oss_client_options to initialize the client options. */
init_options(oss_client_options);
/* Initialize the parameters. */
aos_string_t bucket;
aos_string_t object;
aos_string_t file;
aos_table_t *headers = NULL;
aos_table_t *params = NULL;
aos_table_t *resp_headers = NULL;
aos_status_t *resp_status = NULL;
aos_http_request_t *req;
apr_time_t now;
char *url_str;
aos_string_t url;
int64_t expire_time;
int one_hour = 3600;
aos_str_set(&bucket, bucket_name);
aos_str_set(&object, object_name);
aos_str_set(&file, local_filename);
expire_time = now / 1000000 + one_hour;
headers = aos_table_make(pool, 0);
params = aos_table_make(pool, 0);
req = aos_http_request_create(pool);
req->method = HTTP_GET;
now = apr_time_now(); /* Unit: microseconds */
expire_time = now / 1000000 + one_hour;
/* Generate a signed URL. */
url_str = oss_gen_signed_url(oss_client_options, &bucket, &object, expire_time, req);
aos_str_set(&url, url_str);
/* Use the signed URL to download the object. */
resp_status = oss_get_object_to_file_by_url(oss_client_options, &url, headers, params, &file, &resp_headers);
if (aos_status_is_ok(resp_status)) {
    printf("get object succeeded\n");
} else {
    printf("get object failed\n");
}
/* Release the memory pool. The memory allocated to various resources used for the request is released. */
aos_pool_destroy(pool);
/* Release the allocated global resources. */
aos_http_io_deinitialize();
return 0;
}

```

12.10. IMG

Image Processing (IMG) provided by OSS is a secure, cost-effective, and highly reliable image processing service that processes large amounts of data. After you upload source images to OSS, you can call RESTful APIs to process the images anytime, anywhere, and on any Internet device.

For the complete IMG code, visit [GitHub](#).

Use the IMG parameters to process images

- Use a single IMG parameter to process an image and save the image as the local image

```
#include "oss_api.h"
#include "aos_http_io.h"
const char *endpoint = "<yourEndpoint>";
const char *access_key_id = "<yourAccessKeyId>";
const char *access_key_secret = "<yourAccessKeySecret>";
const char *bucket_name = "<yourBucketName>";
const char *object_name = "<yourObjectName>";
void init_options(oss_request_options_t *options)
{
    options->config = oss_config_create(options->pool);
    /* Use a char* string to initialize aos_string_t. */
    aos_str_set(&options->config->endpoint, endpoint);
    aos_str_set(&options->config->access_key_id, access_key_id);
    aos_str_set(&options->config->access_key_secret, access_key_secret);
    /* Specify whether to use CNAME. A value of 0 indicates that CNAME is not used. */
    options->config->is_cname = 0;
    /* Configure network parameters such as timeout periods. */
    options->ctl = aos_http_controller_create(options->pool, 0);
}
int main(int argc, char *argv[])
{
    /* Call the aos_http_io_initialize method in main() to initialize global resources such as networks and memory. */
    if (aos_http_io_initialize(NULL, 0) != AOSE_OK) {
        exit(1);
    }
    /* Create a memory pool to manage memory. aos_pool_t is equivalent to apr_pool_t. The code to create a memory pool is included in the APR library. */
    aos_pool_t *pool;
    /* Create a memory pool. The second parameter is NULL. This value indicates that the pool does not inherit any other memory pool. */
    aos_pool_create(&pool, NULL);
    /* Create and initialize options. This parameter specifies global configuration information such as endpoint, access_key_id, access_key_secret, is_cname, and curl. */
    oss_request_options_t *oss_client_options;
    /* Allocate the memory resources in the memory pool to the options. */
    oss_client_options = oss_request_options_create(pool);
    /* Initialize oss_client_options. */
    init_options(oss_client_options);
    /* Initialize the parameters. */
    aos_string_t bucket;
    aos_string_t object;
    aos_string_t file;
    aos_table_t *headers = NULL;
    aos_table_t *params = NULL;
```



```

aos_table_t *resp_headers = NULL;
aos_status_t *resp_status = NULL;
aos_str_set(&bucket, bucket_name);
aos_str_set(&object, object_name);
/* Resize the image to a height and width of 100 pixels. */
params = aos_table_make(pool, 1);
apr_table_set(params, OSS_PROCESS, "image/resize,m_fixed,w_100,h_100");
/* Download the processed image to your local computer. */
aos_str_set(&file, "<LocalFileName>");
resp_status = oss_get_object_to_file(oss_client_options, &bucket, &object, headers, p
arams, &file, &resp_headers);
if (aos_status_is_ok(resp_status)) {
    printf("get object to file succeeded\n");
} else {
    printf("get object to file failed\n");
}
/* Release the memory pool, which releases memory resources allocated for the request
. */
aos_pool_destroy(pool);
/* Release the allocated global resources. */
aos_http_io_deinitialize();
return 0;
}

```

- Use multiple IMG parameters to process an image and save the image as the local image

When you use multiple IMG parameters to process the image, separate these parameters with forward slashes (/).

```

#include "oss_api.h"
#include "aos_http_io.h"
const char *endpoint = "<yourEndpoint>";
const char *access_key_id = "<yourAccessKeyId>";
const char *access_key_secret = "<yourAccessKeySecret>";
const char *bucket_name = "<yourBucketName>";
const char *object_name = "<yourObjectName>";
void init_options(oss_request_options_t *options)
{
    options->config = oss_config_create(options->pool);
    /* Use a char* string to initialize aos_string_t. */
    aos_str_set(&options->config->endpoint, endpoint);
    aos_str_set(&options->config->access_key_id, access_key_id);
    aos_str_set(&options->config->access_key_secret, access_key_secret);
    /* Specify whether to use CNAME. A value of 0 indicates that CNAME is not used. */
    options->config->is_cname = 0;
    /* Configure network parameters such as timeout periods. */
    options->ctl = aos_http_controller_create(options->pool, 0);
}
int main(int argc, char *argv[])
{
    /* Call the aos_http_io_initialize method in main() to initialize global resources su
ch as networks and memory. */
    if (aos_http_io_initialize(NULL, 0) != AOSE_OK) {
        exit(1);
    }
}

```

```

/* Create a memory pool to manage memory. aos_pool_t is equivalent to apr_pool_t. The
code to create a memory pool is included in the APR library. */
aos_pool_t *pool;
/* Create a memory pool. The second parameter is NULL. This value indicates that the
pool does not inherit any other memory pool. */
aos_pool_create(&pool, NULL);
/* Create and initialize options. This parameter specifies global configuration infor
mation such as endpoint, access_key_id, acces_key_secret, is_cname, and curl. */
oss_request_options_t *oss_client_options;
/* Allocate the memory resources in the memory pool to the options. */
oss_client_options = oss_request_options_create(pool);
/* Initialize oss_client_options. */
init_options(oss_client_options);
/* Initialize the parameters. */
aos_string_t bucket;
aos_string_t object;
aos_string_t file;
aos_table_t *headers = NULL;
aos_table_t *params = NULL;
aos_table_t *resp_headers = NULL;
aos_status_t *resp_status = NULL;
aos_str_set(&bucket, bucket_name);
aos_str_set(&object, object_name);
/* After you resize the image to a height and width of 100 pixels, rotate the image 9
0°. */
params = aos_table_make(pool, 1);
apr_table_set(params, OSS_PROCESS, "image/resize,m_fixed,w_100,h_100/rotate,90");
/* Download the processed image to your local computer. */
aos_str_set(&file, "<LocalFileName>");
resp_status = oss_get_object_to_file(oss_client_options, &bucket, &object, headers, p
arams, &file, &resp_headers);
if (aos_status_is_ok(resp_status)) {
    printf("get object to file succeeded\n");
} else {
    printf("get object to file failed\n");
}
/* Release the memory pool, which releases memory resources allocated for the request
. */
aos_pool_destroy(pool);
/* Release the allocated global resources. */
aos_http_io_deinitialize();
return 0;
}

```

Use image style to process images

You can encapsulate multiple IMG parameters in a style, and use the style to process multiple images at a time. For more information, see [Image styles](#). The following code provides an example on how to use image style to process an image:

```

#include "oss_api.h"
#include "aos_http_io.h"
const char *endpoint = "<yourEndpoint>";
const char *access_key_id = "<yourAccessKeyId>";

```

```

const char *access_key_secret = "<yourAccessKeySecret>";
const char *bucket_name = "<yourBucketName>";
const char *object_name = "<yourObjectName>";
void init_options(oss_request_options_t *options)
{
    options->config = oss_config_create(options->pool);
    /* Use a char* string to initialize aos_string_t. */
    aos_str_set(&options->config->endpoint, endpoint);
    aos_str_set(&options->config->access_key_id, access_key_id);
    aos_str_set(&options->config->access_key_secret, access_key_secret);
    /* Specify whether to use CNAME. A value of 0 indicates that CNAME is not used. */
    options->config->is_cname = 0;
    /* Configure network parameters such as timeout periods. */
    options->ctl = aos_http_controller_create(options->pool, 0);
}
int main(int argc, char *argv[])
{
    /* Call the aos_http_io_initialize method in main() to initialize global resources such
    as networks and memory. */
    if (aos_http_io_initialize(NULL, 0) != AOSE_OK) {
        exit(1);
    }
    /* Create a memory pool to manage memory. aos_pool_t is equivalent to apr_pool_t. The c
    ode to create a memory pool is included in the APR library. */
    aos_pool_t *pool;
    /* Create a memory pool. The second parameter is NULL. This value indicates that the po
    ol does not inherit any other memory pool. */
    aos_pool_create(&pool, NULL);
    /* Create and initialize options. This parameter specifies global configuration informa
    tion such as endpoint, access_key_id, acces_key_secret, is_cname, and curl. */
    oss_request_options_t *oss_client_options;
    /* Allocate the memory resources in the memory pool to the options. */
    oss_client_options = oss_request_options_create(pool);
    /* Initialize oss_client_options. */
    init_options(oss_client_options);
    /* Initialize the parameters. */
    aos_string_t bucket;
    aos_string_t object;
    aos_string_t file;
    aos_table_t *headers = NULL;
    aos_table_t *params = NULL;
    aos_table_t *resp_headers = NULL;
    aos_status_t *resp_status = NULL;
    aos_str_set(&bucket, bucket_name);
    aos_str_set(&object, object_name);
    /* Specify the image style. */
    params = aos_table_make(pool, 1);
    apr_table_set(params, OSS_PROCESS, "style/<yourCustomStyleName>");
    /* Download the processed image to your local computer. */
    aos_str_set(&file, "<LocalFileName>");
    resp_status = oss_get_object_to_file(oss_client_options, &bucket, &object, headers, par
    ams, &file, &resp_headers);
    if (aos_status_is_ok(resp_status)) {
        printf("get object to file succeeded\n");
    } else {

```

```

    , else {
        printf("get object to file failed\n");
    }
    /* Release the memory pool, which releases memory resources allocated for the request.
    */
    aos_pool_destroy(pool);
    /* Release the allocated global resources. */
    aos_http_io_deinitialize();
    return 0;
}

```

Generate a signed object URL that includes IMG parameters

URLs of private objects must be signed. OSS does not allow you to add IMG parameters to a signed URL. If you want to process private objects, add IMG parameters to the signature. The following code provides an example on how to add IMG parameters to the signature:

```
#include "oss_api.h"
#include "aos_http_io.h"

const char *endpoint = "<yourEndpoint>";
const char *access_key_id = "<yourAccessKeyId>";
const char *access_key_secret = "<yourAccessKeySecret>";
const char *bucket_name = "<yourBucketName>";
const char *object_name = "<yourObjectName>";

void init_options(oss_request_options_t *options)
{
    options->config = oss_config_create(options->pool);
    /* Use a char* string to initialize aos_string_t. */
    aos_str_set(&options->config->endpoint, endpoint);
    aos_str_set(&options->config->access_key_id, access_key_id);
    aos_str_set(&options->config->access_key_secret, access_key_secret);
    /* Specify whether to use CNAME. A value of 0 indicates that CNAME is not used. */
    options->config->is_cname = 0;
    /* Configure network parameters such as timeout periods. */
    options->ctl = aos_http_controller_create(options->pool, 0);
}

int main(int argc, char *argv[])
{
    /* Call the aos_http_io_initialize method in main() to initialize global resources such
as networks and memory. */
    if (aos_http_io_initialize(NULL, 0) != AOSE_OK) {
        exit(1);
    }

    /* Create a memory pool to manage memory. aos_pool_t is equivalent to apr_pool_t. The code
to create a memory pool is included in the APR library. */
    aos_pool_t *pool;

    /* Create a memory pool. The second parameter is NULL. This value indicates that the pool
does not inherit any other memory pool. */
    aos_pool_create(&pool, NULL);

    /* Create and initialize options. This parameter specifies global configuration information
such as endpoint, access_key_id, access_key_secret, is_cname, and curl. */
    oss_request_options_t *oss_client_options;

    /* Allocate the memory resources in the memory pool to the options. */
    oss_client_options = oss_request_options_create(pool);
```

```
/* Initialize oss_client_options. */
init_options(oss_client_options);
/* Initialize the parameters. */
aos_string_t bucket;
aos_string_t object;
aos_table_t *params = NULL;
aos_http_request_t *req;
char *url_str;
apr_time_t now;
int64_t expire_time;
aos_str_set(&bucket, bucket_name);
aos_str_set(&object, object_name);
/* Resize the image to a height and width of 100 pixels. */
params = aos_table_make(pool, 1);
apr_table_set(params, OSS_PROCESS, "image/resize,m_fixed,w_100,h_100");
req = aos_http_request_create(pool);
req->method = HTTP_GET;
req->query_params = params;
/* Specify the validity period (expire_time) in seconds. */
now = apr_time_now();
expire_time = now / 1000000 + 10 * 60;
/* Generate a signed URL. */
url_str = oss_gen_signed_url(oss_client_options, &bucket, &object, expire_time, req);
printf("url: %s\n", url_str);
/* Release the memory pool, which releases memory resources allocated for the request.
*/
aos_pool_destroy(pool);
/* Release the allocated global resources. */
aos_http_io_deinitialize();
return 0;
}
```

12.11. Error handling

Exception handling

If a request error occurs in the use of OSS C SDK, the corresponding error message is output in `aos_status_s`. The attributes of `aos_status_s` include:

Error code	Description	Type
code	The HTTP status code of the error request	Integer
error_code	Error code returned by OSS	Character string
error_msg	Error message returned by OSS	Character string
req_id	The UUID that identifies the request. You can provide the req_id to OSS development engineers for further help.	Character string

Timeout handling

If the code in the returned `aos_status_t` is not "2XX", and the `error_code` is -992 or -995, it indicates a connection timeout error or a request timeout error occurs. You can try again.

You can use the `aos_should_retry(aos_status_t *)` in the `aos_status.h` to determine whether the returned error code requires a retry. If the returned value is 1, you must try again.

Common error codes

Error code	Description	HTTP status code
AccessDenied	Access denied	403
BucketAlreadyExists	The bucket already exists.	409
BucketNotEmpty	The bucket is not empty.	409
EntityTooLarge	The entity size exceeds the maximum limit.	400
EntityTooSmall	The entity size is below the minimum limit.	400
FileGroupTooLarge	The total file group size exceeds the maximum limit.	400
FilePartNotExist	The file part does not exist.	400
FilePartStale	The file part has expired.	400
InvalidArgument	The parameter format is invalid.	400
InvalidAccessKeyId	The AccessKeyId does not exist.	403
InvalidBucketName	The bucket name is invalid.	400
InvalidDigest	The digest is invalid.	400
InvalidObjectName	The object name is invalid.	400
InvalidPart	The part is invalid.	400
InvalidPartOrder	The part order is invalid.	400
InvalidTargetBucketForLogging	The buckets that store log files are invalid.	400
InternalError	Internal OSS error	500
MalformedXML	The XML format is invalid.	400
MethodNotAllowed	The method is not supported.	405
MissingArgument	Parameters are not configured.	411

Error code	Description	HTTP status code
MissingContentLength	The content length is not configured.	411
NoSuchBucket	The bucket does not exist.	404
NoSuchKey	The object does not exist	404
NoSuchUpload	The multipart upload ID does not exist.	404
NotImplemented	The method cannot be processed.	501
PreconditionFailed	Pre-processing error.	412
RequestTimeTooSkewed	The local time set for OSSClient is deviated from the time set for the OSS server by over 15 minutes.	403
RequestTimeout	Request timeout	400
SignatureDoesNotMatch	Signature mismatch	403
TooManyBuckets	The number of buckets exceed the limits,.	400

13. Ruby

13.1. Installation

This topic describes how to install Object Storage Service (OSS) SDK for Ruby.

Prerequisites

- Alibaba Cloud OSS is activated.

To activate OSS or obtain more information about OSS, go to the [OSS product page](#).

- An AccessKey pair is created.

For more information about how to create an AccessKey pair for a Resource Access Management (RAM) user, see [Create an AccessKey pair for a RAM user](#).

Install OSS SDK for Ruby

You can use one of the following methods to install OSS SDK for Ruby:

- Method 1

Run the `gem install aliyun-sdk` command to install OSS SDK for Ruby.

 **Note** If you cannot access <https://rubygems.org>, we recommend that you use the mirror that is maintained by Ruby China.

```
gem install aliyun-sdk --clear-sources --source https://gems.ruby-china.com
```

- Method 2

Use [bundler](#) to install OSS SDK for Ruby.

- Add `gem 'aliyun-sdk', '~> 0.6.0'` to the `Gemfile` file of your application.
- Use a mirror to install OSS SDK for Ruby.

 **Note** <https://gems.ruby-china.com> is a mirror of <https://rubygems.org> and is maintained by Ruby China. The mirror is automatically synchronized and updated. If you cannot access rubygems.org, you can use the mirror to install OSS SDK for Ruby.

```
bundle config mirror.https://rubygems.org https://gems.ruby-china.com
bundle install
```

Dependencies

- Ruby 1.9.3 or later is required.
- A Linux, Windows, or macOS operating system is required.

 Notice

- Some dependency files are gem native extensions. Therefore, you must install Ruby Devkit to compile the files.
- The nokogiri gem dependency that OSS SDK for Ruby uses to process XML files must include the zlib library.

In Linux, Windows, and macOS operating systems, perform the following steps to install the dependencies.

- **Linux**

Run the following command to install the preceding dependencies. In this example, the Ubuntu operating system is used, which is a Linux distribution.

```
sudo apt-get install ruby ruby-dev zlib1g-dev
```

Each Linux distribution has its package manager. You can install a package manager in a similar way.

- **Windows**

- i. Visit [RubyInstaller](#) to download an installer. Double-click to download an installer and select **Add Ruby executables to your PATH** to install the installer.

 **Note** Download Ruby 2.1 or earlier. Ruby 2.2 cannot be installed. For more information, visit [GitHub](#).

- ii. Visit [RubyInstaller](#) to download the corresponding development kit. Create a folder named `RubyDev` in the root directory of the C drive. Then, decompress the development kit to this folder.
- iii. Press `Windows + R` and enter `cmd`. Then, press the Enter key. In the Command Prompt, enter the following command:

```
cd C:\RubyDev
ruby dk.rb init
ruby dk.rb install
```

If a config.yml configuration error occurs, we recommend that you open `C:\RubyDev\config.yml` by using the text editor and change the content to the following content:

```
---
- C:/Ruby21
```

Save the config.yml file. Then, run the `ruby dk.rb install` command to complete the installation. In the preceding example, Ruby21 indicates the installation directory of Ruby. You can specify your actual directory. After the development kit is installed, close the Command Prompt.

- iv. Enter the `gem install aliyun-sdk` command.

After OSS SDK for Ruby is installed, enter `irb` to open the interactive command-line prompt of Ruby. Enter `require 'aliyun/oss'` in the interactive command-line prompt. If true is displayed, OSS SDK for Ruby is installed.

- macOS

By default, the macOS operating system has Ruby installed. For your convenience, we recommend that you install another version of Ruby for your development.

- i. Enter `xcode-select --install` in Terminal to install Xcode Command Line Tools. If the tools fail to be installed, we recommend that you manually download the tools and install them.

 **Note** You can use your Apple ID to log on to [Apple Developer](#) and download Xcode Command Line Tools. Select your macOS version. After the tools are downloaded, double-click the dmg file. In the window that appears, double-click the installation program. During this process, you must enter your Apple password.

- ii. Run the following command in Terminal to install brew:

```
ruby -e "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/master/install)"
```

- iii. Run the following command in Terminal to install Ruby:

```
brew install ruby
exec $SHELL -l
```

- iv. Run the following command in Terminal to install OSS SDK for Ruby:

```
gem install aliyun-sdk
```

- v. Run the following command in Terminal to check whether OSS SDK for Ruby is installed. If true is displayed, OSS SDK for Ruby is installed.

```
irb
> require 'aliyun/oss'
=> true
```

References

- For more information about OSS SDK for Ruby, visit [GitHub](#).
- For more information about associated API operations, see [Alibaba Cloud OSS SDK for Ruby](#).
- [ChangeLog](#)

13.2. Quick start

This topic describes how to use the OSS Ruby SDK to access OSS services to view the bucket list and object list, to upload, download, and delete the objects.

For usage convenience, the following operations are performed in `irb` which is Ruby's interactive command-line interface (CLI).

Initialize the client

Input the following content on the CLI and press the Enter:

```
irb
```

Enter the Ruby interactive CLI mode. Introduce the SDK package with `require` :

```
> require 'aliyun/oss'
=> true
```

 **Note** In the following demonstration, the `>` symbol introduces the command that the user enters, and the `=>` symbol introduces the content returned by a program.

Run the following command to create an OSSClient:

```
> client = Aliyun::OSS::Client.new(
>   endpoint: 'endpoint',
>   access_key_id: 'AccessKeyId',
>   access_key_secret: 'AccessKeySecret')
=> #<Aliyun::OSS::Client...
```

Replace the parameters with the actual endpoint, AccessKeyID, and AccessKeySecret.

View the bucket list

Run the following command to view the bucket list:

```
> buckets = client.list_buckets
=> #<Enumerator...
> buckets.each { |b| puts b.name }
=> bucket-1
=> bucket-2
=> ...
```

If the bucket list is empty, run the following command to create a bucket:

```
> client.create_bucket('my-bucket')
=> true
```

 **Note**

- For bucket naming conventions, see [Basic OSS Concepts](#).
- The bucket name cannot be the same with existing bucket names of other users in OSS. To create a bucket successfully, select a unique bucket name.

View the object list

You can run the following command to view the objects in a bucket:

```
> bucket = client.get_bucket('my-bucket')
=> #<Aliyun::OSS::Bucket...
> objects = bucket.list_objects
=> #<Enumerator...
> objects.each { |obj| puts obj.key }
=> object-1
=> object-2
=> ...
```

Upload an object

You can run the following command to upload a file to a bucket:

```
> bucket.put_object('my-object', :file => 'local-file')
=> true
```

Specifically, `local-file` indicates the path to the local file to be uploaded. When the file is uploaded, you can view it using `list_objects`:

```
> objects = bucket.list_objects
=> #<Enumerator...
> objects.each { |obj| puts obj.key }
=> my-object
=> ...
```

Download an object

You can run the following command to download an object from a bucket:

```
> bucket.get_object('my-object', :file => 'local-file')
=> #<Aliyun::OSS::Object...
```

Specifically, `local-file` is the path to the downloaded object. After the object is downloaded, you can open the object to view its content.

Delete an object

You can run the following command to delete an object from a bucket:

```
> bucket.delete_object('my-object')
=> true
```

After deletion, you can run `list_objects` to check whether the object has been successfully deleted from the bucket.

```
> objects = bucket.list_objects
=> #<Enumerator...
> objects.each { |obj| puts obj.key }
=> object-1
=> ...
```

Learn more

- [Manage a bucket](#)
- [Upload objects](#)
- [Download objects](#)
- [Manage objects](#)
- [Integration with Rails](#)
- [CNAME](#)
- [Use STS to access OSS](#)
- [Manage lifecycle rules](#)
- [Set logging](#)
- [Static website hosting](#)
- [Anti-leech](#)
- [CORS](#)
- [Exception handling](#)

13.3. Buckets

13.3.1. Create buckets

A bucket is a container used to store objects in Object Storage Service (OSS). Each object is contained in a bucket. This topic describes how to create a bucket.

The following code provides an example on how to call `Client#create_bucket` to create a bucket with a specified name:

```
require 'aliyun/oss'
client = Aliyun::OSS::Client.new(
  endpoint: 'endpoint',
  access_key_id: 'AccessKeyId', access_key_secret: 'AccessKeySecret')
client.create_bucket('my-bucket')
```

Note

- For more information about the bucket naming conventions, see [bucket](#).
- The name of a bucket must be globally unique in OSS and cannot be changed after the bucket is created.
- For more information about endpoints, see [Regions and endpoints](#).
- For more information about how to create a bucket, see [Put Bucket](#).

13.3.2. List buckets

The following code provides an example on how to call `Client#list_buckets` to list all buckets of the current user and specify `:prefix` to list buckets whose names contain a specified prefix:

```
require 'aliyun/oss'
client = Aliyun::OSS::Client.new(
  endpoint: 'endpoint',
  access_key_id: 'AccessKeyId', access_key_secret: 'AccessKeySecret')
buckets = client.list_buckets
buckets.each { |b| puts b.name }
buckets = client.list_buckets(:prefix => 'my-')
buckets.each { |b| puts b.name }
```

13.3.3. Determine whether buckets exist

A bucket is a container for objects stored in OSS. Every object is contained in a bucket. This topic describes how to determine whether a bucket exists.

The following code provides an example on how to call `Client#bucket_exists?` to determine whether a bucket exists:

```
require 'aliyun/oss'
client = Aliyun::OSS::Client.new(
  endpoint: 'endpoint',
  access_key_id: 'AccessKeyId', access_key_secret: 'AccessKeySecret')
puts client.bucket_exists?('my-bucket')
```

13.3.4. Manage bucket ACLs

A bucket is a container used to store objects in Object Storage Service (OSS). Every object is contained in a bucket. This topic describes how to configure and query the access control list (ACL) of a bucket.

The following table describes the bucket ACLs.

ACL	Description	Value
-----	-------------	-------

ACL	Description	Value
Private	Only the owner or authorized users of this bucket can read and write objects in the bucket. Other users, including anonymous users cannot access objects in the bucket without authorization.	Aliyun::OSS::ACL::PRIVATE
Public read	Only the owner or authorized users of this bucket can write objects in the bucket. Other users, including anonymous users can only read objects from the bucket. We recommend that you exercise with caution when you set the bucket ACL to public read.	Aliyun::OSS::ACL::PUBLIC_READ
Public read/write	Any users, including anonymous users can read and write objects in the bucket. We recommend that you exercise caution when you set the bucket ACL to public read/write.	Aliyun::OSS::ACL::PUBLIC_READ_WRITE

Configure the ACL of a bucket

The following code provides an example on how to call `Bucket#acl=` to configure the ACL of a bucket:

```
require 'aliyun/oss'
client = Aliyun::OSS::Client.new(
  endpoint: 'endpoint',
  access_key_id: 'AccessKeyId', access_key_secret: 'AccessKeySecret')
bucket = client.get_bucket('my-bucket')
bucket.acl = Aliyun::OSS::ACL::PUBLIC_READ
puts bucket.acl
```

Query the ACL of a bucket

The following code provides an example on how to call `Bucket#acl` to query the ACL of a bucket:

```
require 'aliyun/oss'
client = Aliyun::OSS::Client.new(
  endpoint: 'endpoint',
  access_key_id: 'AccessKeyId', access_key_secret: 'AccessKeySecret')
bucket = client.get_bucket('my-bucket')
puts bucket.acl
```

13.3.5. Delete buckets

 Note

- To delete parts generated in incomplete multipart uploads, call `list_upload` to list all the parts, and then call `abort_upload` to delete the parts.

The following code provides an example on how to call `Client#delete_bucket` to delete a specified bucket:

```
require 'aliyun/oss'
client = Aliyun::OSS::Client.new(
  endpoint: 'endpoint',
  access_key_id: 'AccessKeyId', access_key_secret: 'AccessKeySecret')
client.delete_bucket('my-bucket')
```

13.3.6. Specify ACLs

You can specify access control lists (ACLs) for buckets and objects to control access to your resources.

For more information about access control, see [Overview](#).

Bucket ACLs

You can specify one of the following ACLs for a bucket:

- **Public read/write:** All users, including anonymous users, can perform read and write operations on objects in the bucket.
- **Public read:** Only the bucket owner can perform write operations on objects in the bucket. Other users, including anonymous users, can perform only read operations on objects in the bucket.
- **Private:** Only the owner and authorized users of this bucket can perform read and write operations on objects in the bucket. Other users, including anonymous users, cannot access the objects in the bucket without authorization.

By default, the ACL of a bucket is set to private when the bucket is created. The following code provides an example on how to configure the ACL of a bucket by using `puts bucket.acl`:

```
require 'aliyun/oss'
client = Aliyun::OSS::Client.new(
  endpoint: 'endpoint',
  access_key_id: 'AccessKeyId', access_key_secret: 'AccessKeySecret')
bucket = client.get_bucket('my-bucket')
puts bucket.acl
```

Object ACLs

You can specify one of the following ACLs for an object:

- **Default:** The object inherits the ACL of the bucket to which the object belongs. The ACL of the object is the same as the ACL of the bucket in which the object is stored.
- **Public read/write:** All users, including anonymous users, can perform read and write operations on the

object.

- Public read: Only the bucket owner can perform write operations on the object. Other users, including anonymous users, can perform only read operations on the object.
- Private: Only the owner and authorized users of the bucket in which the object is stored can perform read and write operations on the object. Other users, including anonymous users, cannot access the object without authorization.

By default, the ACL of an object is set to default when the object is created. The following code provides an example on how to configure the ACL of an object by using `bucket.set_object_acl` :

```
require 'aliyun/oss'
client = Aliyun::OSS::Client.new(
  endpoint: 'endpoint',
  access_key_id: 'AccessKeyId', access_key_secret: 'AccessKeySecret')
bucket = client.get_bucket('my-bucket')
acl = bucket.get_object_acl('my-object')
puts acl # default
bucket.set_object_acl('my-object', Aliyun::OSS::ACL::PUBLIC_READ)
acl = bucket.get_object_acl('my-object')
puts acl # public-read
```

Notice

- If you do not set the object ACL, the object inherits the ACL of the bucket in which the object is stored.
- If the ACL of an object is not set to default, the object ACL takes precedence over the ACL of the bucket in which the object is stored when the object is accessed.
- If the ACL of an object is set to public read or public read/write, you can directly access the object in a browser by using the object URL such as `http://bucket-name.oss-cn-hangzhou.aliyuncs.com/object.jpg`.
- You can create an anonymous client to access a bucket or an object whose ACL is public read or public read/write.

```
require 'aliyun/oss'
# If you do not specify access_key_id and access_key_secret, an anonymous client is created. The client can only access a bucket or an object whose ACL is public.
client = Aliyun::OSS::Client.new(endpoint: 'endpoint')
bucket = client.get_bucket('my-bucket')
bucket.get_object('my-object', :file => 'local_file')
```

13.3.7. Manage lifecycle rules

Object Storage Service (OSS) allows you to configure lifecycle rules to delete expired objects and parts or convert the storage class of expired objects to Infrequent Access (IA) or Archive to reduce your storage costs. This topic describes how to manage lifecycle rules for buckets.

Background information

Each lifecycle rule contains the following information:

- Rule ID. It uniquely identifies a rule in a bucket.
- Policy. You can use one of the following methods to configure a policy. Only a single method can be configured for each bucket.
 - Match by prefix: You can create multiple rules to configure different prefixes. Each prefix must be unique.
 - Match by bucket: If you select this method, only a single rule can be created.
- Expiration time. You can use one of the following configuration methods to configure the expiration time:
 - Specify a validity period for which to retain objects after they are last modified.
 - Specify an expiration date that objects that are last modified before this date expire.
- Status.

Configure lifecycle rules

The following code provides an example on how to configure lifecycle rules:

```
require 'aliyun/oss'
client = Aliyun::OSS::Client.new(
  endpoint: 'endpoint',
  access_key_id: 'AccessKeyId', access_key_secret: 'AccessKeySecret')
bucket.lifecycle = [
  LifecycleRule.new(
    :id => 'rule1', :enabled => true, :prefix => 'foo/', :expiry => 3),
  LifecycleRule.new(
    :id => 'rule2', :enabled => false, :prefix => 'bar/', :expiry => Date.new(2016, 1, 1))
]
```

View lifecycle rules

The following code provides an example on how to view the lifecycle rules configured for a bucket:

```
require 'aliyun/oss'
client = Aliyun::OSS::Client.new(
  endpoint: 'endpoint',
  access_key_id: 'AccessKeyId', access_key_secret: 'AccessKeySecret')
rules = bucket.lifecycle
puts rules
```

Delete lifecycle rules

The following code provides an example on how to delete the lifecycle rules configured for a bucket:

```
require 'aliyun/oss'
client = Aliyun::OSS::Client.new(
  endpoint: 'endpoint',
  access_key_id: 'AccessKeyId', access_key_secret: 'AccessKeySecret')
bucket.lifecycle = []
```

13.3.8. Hotlink protection

This topic describes how to use hot link protection.

To prevent your data on OSS from being leached, OSS supports hot link protection through the referer field settings in the HTTP header, including the following parameters:

- **Referer whitelist:** Used to allow access only for specified domains to OSS data.
- **Empty referer:** Determines whether the referer can be empty. If it is not allowed, only requests with the referer filed in their HTTP or HTTPS headers can access OSS data.

For more information about hot link protection, see [Hotlink protection](#).

Configure a Referer whitelist for a bucket

The following code provides an example on how to configure a Referer whitelist for a bucket by configuring the `Bucket#referer=` field:

```
require 'aliyun/oss'
client = Aliyun::OSS::Client.new(
  endpoint: 'endpoint',
  access_key_id: 'AccessKeyId', access_key_secret: 'AccessKeySecret')
bucket = client.get_bucket('my-bucket')
bucket.referer = BucketReferer.new(
  allow_empty: true, whitelist: ['example.com', '*.example.com'])
```

Query the Referer whitelist of a bucket

The following code provides an example on how to query the Referer whitelist of a bucket by configuring the `Bucket#referer=` field:

```
require 'aliyun/oss'
client = Aliyun::OSS::Client.new(
  endpoint: 'endpoint',
  access_key_id: 'AccessKeyId', access_key_secret: 'AccessKeySecret')
bucket = client.get_bucket('my-bucket')
ref = bucket.referer
puts ref.to_s
```

Clear the Referer whitelist of a bucket

The following code provides an example on how to clear the Referer whitelist of a bucket by configuring the `Bucket#referer=` field:

```
require 'aliyun/oss'
client = Aliyun::OSS::Client.new(
  endpoint: 'endpoint',
  access_key_id: 'AccessKeyId', access_key_secret: 'AccessKeySecret')
bucket = client.get_bucket('my-bucket')
bucket.referer = BucketReferer.new(allow_empty: true, whitelist: [])
```

13.3.9. Set logging

OSS allows you to set access logging for a bucket. After the configuration, accesses to the bucket are logged.

The logs are then stored in a specified bucket on OSS in the format of:

```
<TargetPrefix><SourceBucket>YYYY-mm-DD-HH-MM-SS-UniqueString
```

You must specify the `TargetPrefix`. A logging rule consists the following:

- `enable`: Whether to enable access logging.
- `target_bucket`: The bucket where logs are stored.
- `target_prefix`: The prefix of the log object

For more information, see [Set access logging](#).

Enable bucket logging

Use `Bucket#logging=` to enable logging:

```
require 'aliyun/oss'
client = Aliyun::OSS::Client.new(
  endpoint: 'endpoint',
  access_key_id: 'AccessKeyId', access_key_secret: 'AccessKeySecret')
bucket = client.get_bucket('my-bucket')
bucket.logging = BucketLogging.new(
  enable: true, target_bucket: 'logging_bucket', target_prefix: 'my-log')
```

View bucket logging settings

Use `Bucket#logging` to view logging settings:

```
require 'aliyun/oss'
client = Aliyun::OSS::Client.new(
  endpoint: 'endpoint',
  access_key_id: 'AccessKeyId', access_key_secret: 'AccessKeySecret')
bucket = client.get_bucket('my-bucket')
log = bucket.logging
puts log.to_s
```

Disable bucket logging

Use `Bucket#logging=` to disable logging.

```
require 'aliyun/oss'
client = Aliyun::OSS::Client.new(
  endpoint: 'endpoint',
  access_key_id: 'AccessKeyId', access_key_secret: 'AccessKeySecret')
bucket = client.get_bucket('my-bucket')
bucket.logging = BucketLogging.new(enable: false)
```

13.3.10. CORS

Cross-origin resource sharing (CORS) allows web applications to access resources that belong to another region. OSS provides CORS APIs for convenient cross-origin access control. This topic describes how to perform CORS.

For more information about CORS, see [Cross-origin resource sharing](#) in OSS Developer Guide.

OSS CORS settings contain one or more CORS rules. Each CORS rule includes the following parameters:

- `allowed_origins`: The origins allowed for cross-origin requests, for example, `example.com`, `*`.
- `allowed_methods`: The HTTP methods (PUT, POST, GET, DELETE, HEAD) allowed for cross-origin requests.
- `allowed_headers`: The headers allowed in a prefetch command (OPTIONS), for example, `x-oss-test`, `*`.
- `expose_headers`: The headers allowed for the user to access in the application.
- `max_age_seconds`: The cache time for the returned result of a browser prefetch (OPTIONS) request to a specific resource.

Configure CORS rules

The following code uses `bucket.cors` to configure a CORS rule:

```
require 'aliyun/oss'
client = Aliyun::OSS::Client.new(
  endpoint: 'endpoint',
  access_key_id: 'AccessKeyId', access_key_secret: 'AccessKeySecret')
bucket = client.get_bucket('my-bucket')
bucket.cors = [
  CORSRule.new(
    :allowed_origins => ['http://example.com', 'http://example.net'],
    :allowed_methods => ['PUT', 'POST', 'GET'],
    :allowed_headers => ['Authorization'],
    :expose_headers => ['x-oss-test'],
    :max_age_seconds => 100)
]
```

View CORS rules

The following code uses `bucket.cors` to display CORS rules:

```
require 'aliyun/oss'
client = Aliyun::OSS::Client.new(
  endpoint: 'endpoint',
  access_key_id: 'AccessKeyId', access_key_secret: 'AccessKeySecret')
bucket = client.get_bucket('my-bucket')
cors = bucket.cors
puts cors.map(&:to_s)
```

Clear CORS rules

The following code uses `bucket.cors` to clear CORS rules:

```
require 'aliyun/oss'
client = Aliyun::OSS::Client.new(
  endpoint: 'endpoint',
  access_key_id: 'AccessKeyId', access_key_secret: 'AccessKeySecret')
bucket = client.get_bucket('my-bucket')
bucket.cors = []
```

13.3.11. Static website hosting

You can set your bucket to the static website hosting mode. After the configuration takes effect, you can access this static website with the bucket domain and be redirected to a specified index page or error page.

In **CNAME**, we mentioned that the OSS allows you to direct your domain name to the OSS service address. In this way, the OSS bucket is accessed when you access your website. You must specify the names of the objects in the bucket corresponding to the homepage (index) and error page (error) of the website respectively.

For more information, see [Static website hosting](#).

Configure website hosting

Use `Bucket#website=` to configure hosted pages:

```
require 'aliyun/oss'
client = Aliyun::OSS::Client.new(
  endpoint: 'endpoint',
  access_key_id: 'AccessKeyId', access_key_secret: 'AccessKeySecret')
bucket = client.get_bucket('my-bucket')
bucket.website = BucketWebsite.new(index: 'index.html', error: 'error.html')
```

View website hosting

Use `Bucket#website` to display hosted pages:

```
require 'aliyun/oss'
client = Aliyun::OSS::Client.new(
  endpoint: 'endpoint',
  access_key_id: 'AccessKeyId', access_key_secret: 'AccessKeySecret')
bucket = client.get_bucket('my-bucket')
web = bucket.website
puts web.to_s
```

Clear website hosting

Use `Bucket#website=` to clear hosted pages:

```
require 'aliyun/oss'
client = Aliyun::OSS::Client.new(
  endpoint: 'endpoint',
  access_key_id: 'AccessKeyId', access_key_secret: 'AccessKeySecret')
bucket = client.get_bucket('my-bucket')
bucket.website = BucketWebsite.new(enable: false)
```

13.3.12. Map custom domain names

This topic describes how to map a custom domain name to a bucket.

Object Storage Service (OSS) allows you to map a custom domain name to an OSS bucket based on a CNAME record. You can use this method to migrate data to OSS. For example, you have the domain name `example.com`, and all of your images can be accessed by using URLs in the following format:

`http://img.example.com/x.jpg`. To access your images by using original URLs after you migrate the images to an OSS bucket, perform the following steps:

1. Activate OSS and create a bucket.
2. Map `img.example.com` to the created bucket.
3. Upload an image to the OSS bucket.
4. Modify the Domain Name System (DNS) configurations of the domain name. Add a CNAME record to map `img.example.com` to an OSS endpoint such as `my-bucket.oss-cn-hangzhou.aliyuncs.com`.

You can use the original URL `http://img.example.com/x.jpg` to access the image that is stored in the OSS bucket. For more information about how to map custom domain names, see [Map custom domain names](#).

To use your custom domain name as the endpoint of a bucket when you use OSS SDK for Ruby, set the value of `:cname` to `true`. Example:

```
require 'aliyun/oss'
include Aliyun::OSS
client = Client.new(
  endpoint: 'ENDPOINT',
  access_key_id: 'ACCESS_KEY_ID',
  access_key_secret: 'ACCESS_KEY_SECRET',
  cname: true)
bucket = client.get_bucket('my-bucket')
```

 **Note** If a custom domain name is mapped to a bucket, the domain name cannot be used to call the `list_buckets` operation.

13.4. Upload objects

Objects are the basic unit for data operations in Object Storage Service (OSS). OSS SDK for Ruby provides a variety of methods to upload objects.

Upload a local file

The following code provides an example on how to call `Bucket#put_object` and set the `:file` parameter to upload a local file to OSS:

```
require 'aliyun/oss'
client = Aliyun::OSS::Client.new(
  endpoint: 'endpoint',
  access_key_id: 'AccessKeyId', access_key_secret: 'AccessKeySecret')
bucket = client.get_bucket('my-bucket')
bucket.put_object('my-object', :file => 'local-file')
```

Streaming upload

If a long period of time is required to upload a large object, you can use streaming upload to upload the object in increments until the entire object is uploaded.

The following code provides an example on how to call `Bucket#put_object` and set the `block` parameter to upload an object to OSS by using streaming upload:

```
require 'aliyun/oss'
client = Aliyun::OSS::Client.new(
  endpoint: 'endpoint',
  access_key_id: 'AccessKeyId', access_key_secret: 'AccessKeySecret')
bucket = client.get_bucket('my-bucket')
bucket.put_object('my-object') do stream
  100.times { i stream << i.to_s }
end
```

Resumable upload

An object can be split into several parts that can be uploaded simultaneously. After all parts are uploaded, they are combined into a complete object.

You can call `Bucket#resumable_upload` to implement resumable upload. You must configure the following parameters in the operation:

- `key`: the name of the object to upload to OSS.
- `file`: the path of the local file to upload.
- `opts`: an optional parameter that contains the following options.
 - `:cpt_file`: the path of the checkpoint file. The default value is `file.cpt` that is in the same

directory as the local file. `file` indicates the name of the local file.

- `:disable_cpt`: If you set this option to true, the upload progress is not recorded during the object upload. As a result, if the object fails to upload, the upload cannot be resumed.
- `:part_size`: the size of each part. Default value: 4 MB.
- `&block`: If you pass in block when you call `Bucket#resumable_upload`, the upload progress is processed by block.

For more information about the related parameters, see [Alibaba Cloud OSS SDK for Ruby](#).

Note If a part fails to upload, the next upload operation of this object continues from the point recorded in the checkpoint file. Next time you call `Bucket#resumable_upload`, you must specify the same checkpoint file as last time. The checkpoint file is deleted after the object is uploaded.

The following code provides an example on how to upload an object by using resumable upload:

```
require 'aliyun/oss'
client = Aliyun::OSS::Client.new(
  endpoint: 'endpoint',
  access_key_id: 'AccessKeyId', access_key_secret: 'AccessKeySecret')
bucket = client.get_bucket('my-bucket')
bucket.resumable_upload('my-object', 'local-file') do p
  puts "Progress: #{p}"
end
bucket.resumable_upload(
  'my-object', 'local-file',
  :part_size => 100 * 1024, :cpt_file => '/tmp/x.cpt') { p
  puts "Progress: #{p}"
}
```

Note

- OSS SDK for Ruby records upload progress in a cpt file. Make sure that you have write permissions on the cpt file.
- The cpt file records the intermediate status information of uploads, and can verify uploaded data. You cannot edit the cpt file. If the cpt file is damaged, the upload is interrupted. After the entire object is uploaded, the cpt file is deleted.
- If changes occur to the local file during upload, the upload fails.

Append upload

You can call `Bucket#append_object` to upload appendable objects. You must specify the position where the append operation starts. The position must be 0 for the first append operation. The position of the subsequent append operations is the current size of the object.

- If the object to append does not exist, an appendable object is created when you call `append_object`.
- If the object exists, new content is appended to the end of the object when you call `append_object`.

```
.  
  
require 'aliyun/oss'  
client = Aliyun::OSS::Client.new(  
  endpoint: 'endpoint',  
  access_key_id: 'AccessKeyId', access_key_secret: 'AccessKeySecret')  
bucket = client.get_bucket('my-bucket')  
# Create an appendable object.  
bucket.append_object('my-object', 0) {}  
# Append content to the end of the object.  
next_pos = bucket.append_object('my-object', 0) do stream  
  100.times { i stream << i.to_s }  
end  
next_pos = bucket.append_object('my-object', next_pos, :file => 'local-file-1')  
next_pos = bucket.append_object('my-object', next_pos, :file => 'local-file-2')
```

Note

- You can append content only to appendable objects.
- Appendable objects cannot be copied.

Upload callback

You can configure upload callback when you upload an object. Then, after the object is uploaded, OSS sends an HTTP POST request to the application server address that you provide. This way, the application server is informed of the upload. For more information about upload callback, see [Upload callback](#).

 **Note** You can call only `put_object` or `resumable_upload` to upload an object and configure upload callback at the same time. For more information about the complete sample code that is used to configure upload callback, visit [GitHub](#).

When you call `put_object` to upload an object and configure upload callback at the same time, you must specify the related bucket and the object information in the body field. If the application server receives the callback request, the object is uploaded to OSS.

```
require 'aliyun/oss'
client = Aliyun::OSS::Client.new(
  endpoint: 'endpoint',
  access_key_id: 'AccessKeyId', access_key_secret: 'AccessKeySecret')
bucket = client.get_bucket('my-bucket')
callback = Aliyun::OSS::Callback.new(
  url: 'http://10.101.168.94:1234/callback',
  query: {user: 'put_object'},
  body: 'bucket=${bucket}&object=${object}'
)
begin
  bucket.put_object('files/hello', file: '/tmp/x', callback: callback)
rescue Aliyun::OSS::CallbackError => e
  puts "Callback failed: #{e.message}"
end
```

You can call `resumable_upload` in a similar way to `put_object` .

```
require 'aliyun/oss'
client = Aliyun::OSS::Client.new(
  endpoint: 'endpoint',
  access_key_id: 'AccessKeyId', access_key_secret: 'AccessKeySecret')
bucket = client.get_bucket('my-bucket')
callback = Aliyun::OSS::Callback.new(
  url: 'http://10.101.168.94:1234/callback',
  query: {user: 'put_object'},
  body: 'bucket=${bucket}&object=${object}'
)
begin
  bucket.resumable_upload('files/hello', '/tmp/x', callback: callback)
rescue Aliyun::OSS::CallbackError => e
  puts "Callback failed: #{e.message}"
end
```

Note

- The callback URL cannot contain query strings. You must specify query strings in the `query` parameter.
- If the object is uploaded but the upload callback failed, the client throws the `CallbackError` or error. If you ignore the error, you must explicitly handle the exception.
- For more information about the application server that serves as the callback server, visit [GitHub](#).

13.5. Download objects

This topic describes the methods provided by OSS SDK for Ruby to download objects.

Download an object to your computer

You can call the `Bucket#get_object` operation and specify the `:file` parameter to download the specified OSS object to your computer.

```
require 'aliyun/oss'
client = Aliyun::OSS::Client.new(
  endpoint: 'endpoint',
  access_key_id: 'AccessKeyId', access_key_secret: 'AccessKeySecret')
bucket = client.get_bucket('my-bucket')
bucket.get_object('my-object', :file => 'local-file')
```

Streaming download

If a long period of time is required to download an object, you can use streaming download to download the object in increments until the entire object is downloaded.

You can call the `Bucket#get_object` operation and specify the `block` parameter to stream the content to download.

```
require 'aliyun/oss'
client = Aliyun::OSS::Client.new(
  endpoint: 'endpoint',
  access_key_id: 'AccessKeyId', access_key_secret: 'AccessKeySecret')
bucket = client.get_bucket('my-bucket')
bucket.get_object('my-object') do |chunk|
  # handle_data(chunk)
  puts "Got a chunk, size: #{chunk.size}."
end
```

Resumable download

You may fail to download a large object if the network is unstable or other exceptions occur. In some cases, you may still fail to download the object even after multiple attempts. To resolve this issue, OSS SDK for Ruby provides the resumable download feature.

In resumable download, the object that you want to download is split into multiple parts and each part is separately downloaded. After all parts are downloaded, these parts are combined into a complete object.

You can call the `Bucket#resumable_download` operation to implement resumable download. You can configure the following parameters in the operation:

- `key`: the name of the object you want to download.
- `file`: the local path to which you want to download the object.
- `opts`: contains the following options:
 - `:cpt_file`: the path of the checkpoint file. If you do not specify this option, `file.cpt` in the local path to which the object is downloaded is used by default. `file` specifies the name of the local file.
 - `:disable_cpt`: If you set this option to true, the download progress is not recorded during the object download. If the object fails to download, the download cannot be resumed.

- `:part_size`: the size of each part. By default, the size of each part is 10 MB.
- `&block`: If you pass in `block` when you call the `Bucket#resumable_download` operation, the download progress is processed by block.

For more information about parameters, see [Download objects](#).

The current download progress information is recorded in the checkpoint file during the object download. A downloaded part is stored as `file.part.N`. `file` indicates the name of the local file after the part is downloaded. If a part fails to download in the download process, the part continues to download based on the record in the checkpoint file next time the download is resumed. Next time you call `Bucket#resumable_download`, you must specify the same checkpoint file as last time. After the object is downloaded, the part and checkpoint file are deleted.

```
require 'aliyun/oss'
client = Aliyun::OSS::Client.new(
  endpoint: 'endpoint',
  access_key_id: 'AccessKeyId', access_key_secret: 'AccessKeySecret')
bucket = client.get_bucket('my-bucket')
bucket.resumable_download('my-object', 'local-file') do |p|
  puts "Progress: #{p}"
end
bucket.resumable_download(
  'my-object', 'local-file',
  :part_size => 100 * 1024, :cpt_file => '/tmp/x.cpt') { |p|
  puts "Progress: #{p}"
}
```

Note

- OSS SDK for Ruby records the download progress in the cpt file. Make sure that you have write permissions on the cpt file.
- OSS SDK for Ruby stores the downloaded parts in the part local file. You must make sure that you have permissions to create files in the local directory that contains `file`.
- The cpt file records the intermediate status information of downloads, and downloaded data can be verified. You cannot edit the cpt file. If the cpt file is damaged, the file cannot be downloaded. After the entire object is downloaded, the cpt file is deleted.
- If the object to download is modified, the ETag value is changed, or the part file is missing or modified, the object fails to download.

HTTP-based download

You can download objects stored in OSS by using OSS SDK for Ruby. You can also directly download the objects over HTTP. To download an object over HTTP, you can use a browser or a command-line tool such as `wget` or `curl`. In this case, you must use OSS SDK for Ruby to generate the URL of the object.

If you use the `Bucket#object_url` method to generate an HTTP address used to download the object, you must configure the following parameters:

- `key`: the name of the object that you want to download.

- **sign**: specifies whether to generate a signed URL.

 Note

- If the ACL of the object is public read or public read/write, you can access the object by using a URL that does not contain a signature.
- If the ACL of the object is private, you must use a signed URL to access the object.

- **expires**: the validity period of the URL. Default value: 60. Unit: seconds.

```
require 'aliyun/oss'
client = Aliyun::OSS::Client.new(
  endpoint: 'endpoint',
  access_key_id: 'AccessKeyId', access_key_secret: 'AccessKeySecret')
bucket = client.get_bucket('my-bucket')
# Generate a URL. By default, the URL includes a signature. The validity period is 60 seconds.
puts bucket.object_url('my-object')
# http://my-bucket.oss-cn-hangzhou.aliyuncs.com/my-object?Expires=1448349966&OSSAccessKeyId=5vxxxxx&Signature=aM2HpBLemQlaec6JCd7BBAKYiwI%3D
# Generate a URL that does not contain a signature.
puts bucket.object_url('my-object', false)
# http://my-bucket.oss-cn-hangzhou.aliyuncs.com/my-object
# Set the validity period of the specified URL to 3600 seconds.
puts bucket.object_url('my-object', true, 3600)
```

13.6. Manage objects

OSS SDK for Ruby provides API operations to manage multiple objects in a bucket.

List all objects in a bucket

You can call the `bucket.list_objects` operation to list all objects in a specified bucket. You can specify the following parameters when you list the objects:

- **:prefix**: lists objects whose names contain the specified prefix.
- **:marker**: lists objects whose names are alphabetically greater than the value of marker.
- **:delimiter**: obtains the common prefix of objects.

The following code provides an example on how to list all objects that meet the specified conditions in a bucket named `examplebucket`:

```
require 'aliyun/oss'
client = Aliyun::OSS::Client.new(
  # Specify the endpoint of the region in which the bucket is located. For example, if the
  bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzh
  ou.aliyuncs.com.
  endpoint: 'https://oss-cn-hangzhou.aliyuncs.com',
  # Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to a
  ccess Object Storage Service (OSS) because the account has permissions on all API operation
  s. We recommend that you use a RAM user to call API operations or perform routine O&M. To c
  reate a RAM user, log on to the RAM console.
  access_key_id: 'AccessKeyId', access_key_secret: 'AccessKeySecret')
# Specify the bucket name. Example: examplebucket.
bucket = client.get_bucket('examplebucket')
# List all objects.
objects = bucket.list_objects
objects.each { |o| puts o.key }
# List all objects whose names contain the 'my-' prefix in the bucket.
objects = bucket.list_objects(:prefix => 'my-')
objects.each { |o| puts o.key }
# List all objects whose names are alphabetically greater than 'my-object' and contain the
'my-' prefix in the bucket.
objects = bucket.list_objects(:prefix => 'my-', :marker => 'my-object')
objects.each { |o| puts o.key }
```

List directories

OSS uses a flat structure for objects instead of a hierarchical structure to store objects. A directory is an object whose size is 0 and whose name ends with a forward slash (/). You can upload and download this object. By default, objects whose names end with a forward slash (/) are displayed as directories in the OSS console.

Assume that the following objects are stored in a specific bucket:

```
foo/x
foo/y
foo/bar/a
foo/bar/b
foo/hello/C/1
foo/hello/C/2
...
foo/hello/C/9999
```

The following code provides an example on how to use the `list_dir` function to list objects and subdirectories in a specific directory:

```
require 'aliyun/oss'
client = Aliyun::OSS::Client.new(
  # Specify the endpoint of the region in which the bucket is located. For example, if the
  bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzh
  ou.aliyuncs.com.
  endpoint: 'https://oss-cn-hangzhou.aliyuncs.com',
  # Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to a
  ccess OSS because the account has permissions on all API operations. We recommend that you
  use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on
  to the RAM console.
  access_key_id: 'AccessKeyId', access_key_secret: 'AccessKeySecret')
# Specify the bucket name. Example: examplebucket.
bucket = client.get_bucket('examplebucket')
def list_dir(dir)
  objects = bucket.list_objects(:prefix => dir, :delimiter => '/')
  objects.each do |obj|
    if obj.is_a?(OSS::Object) # object
      puts "Object: #{obj.key}"
    else # common prefix
      puts "SubDir: #{obj}"
    end
  end
end
end
```

The following result is returned:

```
> list_dir('foo/')
=> SubDir: foo/bar/
    SubDir: foo/hello/
    Object: foo/x
    Object: foo/y
> list_dir('foo/bar/')
=> Object: foo/bar/a
    Object: foo/bar/b
> list_dir('foo/hello/C/')
=> Object: foo/hello/C/1
    Object: foo/hello/C/2
    ...
    Object: foo/hello/C/9999
```

Manage object metadata

When you upload an object to OSS, you can specify some attributes of the object. These attributes are called object metadata. The metadata of an object is uploaded and downloaded together with the object.

In OSS SDK for Ruby, the metadata of an object is represented by a `Hash` that contains multiple object attributes. Other keys and values are of the `String` type. Object metadata is transferred as HTTP headers in requests and responses. According to the HTTP protocol, HTTP headers cannot contain complex characters. Therefore, object metadata can contain only visible ASCII characters. Metadata cannot contain line feeds.

 **Note** The metadata of an object cannot exceed 8 KB in size.

When you use `bucket.put_object`, `bucket.append_object`, and `bucket.resumable_upload`, you can set the `:metas` parameter to specify object metadata.

The following code provides an example on how to specify the object metadata:

```
require 'aliyun/oss'
client = Aliyun::OSS::Client.new(
  # Specify the endpoint of the region in which the bucket is located. For example, if the
  # bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com.
  endpoint: 'https://oss-cn-hangzhou.aliyuncs.com',
  # Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
  access_key_id: 'AccessKeyId', access_key_secret: 'AccessKeySecret')
# Specify the bucket name. Example: examplebucket.
bucket = client.get_bucket('examplebucket')
bucket.put_object(
  'my-object-1',
  :file => 'local-file',
  :metas => {'year' => '2016', 'people' => 'mary'})
bucket.append_object(
  'my-object-2', 0,
  :file => 'local-file',
  :metas => {'year' => '2016', 'people' => 'mary'})
bucket.resumable_upload(
  'my-object',
  'local-file',
  :metas => {'year' => '2016', 'people' => 'mary'})
```

The following code provides an example on how to use `bucket.update_object_metas` to update the object metadata:

```
require 'aliyun/oss'
client = Aliyun::OSS::Client.new(
  # Specify the endpoint of the region in which the bucket is located. For example, if the
  # bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com.
  endpoint: 'https://oss-cn-hangzhou.aliyuncs.com',
  # Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
  access_key_id: 'AccessKeyId', access_key_secret: 'AccessKeySecret')
# Specify the bucket name. Example: examplebucket.
bucket = client.get_bucket('examplebucket')
bucket.update_object_metas('my-object', {'year' => '2017'})
```

Copy an object

You can call `bucket.copy_object` to copy an object and use the `:meta_directive` parameter to specify the object metadata. When you copy an object, you can process the object metadata by using one of the following methods:

- If meta is not specified, the metadata of the source object is copied to the destination object.
- If meta is specified, the specified object metadata is used as the metadata of the destination object.

The following code provides an example on how to copy an object:

```
require 'aliyun/oss'
client = Aliyun::OSS::Client.new(
  # Specify the endpoint of the region in which the bucket is located. For example, if the
  # bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com.
  endpoint: 'https://oss-cn-hangzhou.aliyuncs.com',
  # Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
  access_key_id: 'AccessKeyId', access_key_secret: 'AccessKeySecret')
# Specify the bucket name. Example: examplebucket.
bucket = client.get_bucket('examplebucket')
# Copy the metadata of the source object to the destination object.
bucket.copy_object(
  'my-object', 'copy-object',
  :meta_directive => Aliyun::OSS::MetaDirective::COPY)
# Use the specified metadata as the metadata of the destination object.
bucket.copy_object(
  'my-object', 'copy-object',
  :metas => {'year' => '2017'},
  :meta_directive => Aliyun::OSS::MetaDirective::REPLACE)
```

Delete objects



Warning You cannot recover deleted objects. Exercise caution when you perform this operation.

- Delete a single object

The following code provides an example on how to call `bucket.delete_object` to delete an object:

```
require 'aliyun/oss'
client = Aliyun::OSS::Client.new(
  # Specify the endpoint of the region in which the bucket is located. For example, if the
  # bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com.
  endpoint: 'https://oss-cn-hangzhou.aliyuncs.com',
  # Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to
  # access OSS because the account has permissions on all API operations. We recommend that you
  # use a RAM user to call API operations or perform routine O&M. To create a RAM user, log
  # on to the RAM console.
  access_key_id: 'AccessKeyId', access_key_secret: 'AccessKeySecret')
# Specify the bucket name. Example: examplebucket.
bucket = client.get_bucket('examplebucket')
# Specify the name of the object that you want to delete.
bucket.delete_object('my-object')
```

- Deletes multiple objects

The following code provides an example on how to call `bucket.batch_delete_objects` to delete multiple objects. You can use the `:quiet` parameter to specify whether to return the information about deleted objects.

```
require 'aliyun/oss'
client = Aliyun::OSS::Client.new(
  # Specify the endpoint of the region in which the bucket is located. For example, if the
  # bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com.
  endpoint: 'https://oss-cn-hangzhou.aliyuncs.com',
  # Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to
  # access OSS because the account has permissions on all API operations. We recommend that you
  # use a RAM user to call API operations or perform routine O&M. To create a RAM user, log
  # on to the RAM console.
  access_key_id: 'AccessKeyId', access_key_secret: 'AccessKeySecret')
# Specify the bucket name. Example: examplebucket.
bucket = client.get_bucket('examplebucket')
# Specify the names of the objects that you want to delete.
objs = ['my-object-1', 'my-object-2']
result = bucket.batch_delete_objects(objs)
# By default, the information about deleted objects is returned.
puts result #['my-object-1', 'my-object-2']
objs = ['my-object-3', 'my-object-4']
result = bucket.batch_delete_objects(objs, :quiet => true)
# Specify that the information about deleted objects is not returned.
puts result #[]
```

Configure the ACL of an object

You can set the ACL of an object to one of the following values:

- **default**: The object inherits the ACL of the bucket to which the object belongs. The ACL of the object is the same as that of the bucket in which the object is stored.
- **public-read-write**: All users, including anonymous users, can read and write data from and to the object.
- **public-read**: Only the owner or authorized users of this object can write data to the object. Other

users, including anonymous users, can only read data from the object.

- **private:** Only the owner or authorized users of this object can read and write data from and to the object. Other users, including anonymous users, can access the object only by using a signed URL.

By default, the ACL of an object is set to default when the object is created. You can use

`bucket.set_object_acl` to modify the ACL of the object.

```
require 'aliyun/oss'
client = Aliyun::OSS::Client.new(
  # Specify the endpoint of the region in which the bucket is located. For example, if the
  bucket is located in the China (Hangzhou) region, set the endpoint to https://oss-cn-hangzhou.aliyuncs.com.
  endpoint: 'https://oss-cn-hangzhou.aliyuncs.com',
  # Security risks may arise if you use the AccessKey pair of an Alibaba Cloud account to access OSS because the account has permissions on all API operations. We recommend that you use a RAM user to call API operations or perform routine O&M. To create a RAM user, log on to the RAM console.
  access_key_id: 'AccessKeyId', access_key_secret: 'AccessKeySecret')
# Specify the bucket name. Example: examplebucket.
bucket = client.get_bucket('examplebucket')
acl = bucket.get_object_acl('my-object')
puts acl # default
bucket.set_object_acl('my-object', Aliyun::OSS::ACL::PUBLIC_READ)
acl = bucket.get_object_acl('my-object')
puts acl # public-read
```

- If you do not set the object ACL, the object inherits the ACL of the bucket in which the object is stored.
- If the ACL of an object is not set to default, the object ACL takes precedence over the ACL of the bucket in which the object is stored when the object is accessed.
- If the ACL of an object is set to public read or public read/write, you can directly access the object in a browser by using the URL of the object. Example: `http://bucket-name.oss-cn-hangzhou.aliyuncs.com/object.jpg`.
- You can create an anonymous client to access an object whose ACL is public read or public read/write.

```
require 'aliyun/oss'
# If you do not specify access_key_id and access_key_secret, an anonymous client is created. The client can only access objects whose ACL is public read or public read/write.

client = Aliyun::OSS::Client.new(endpoint: 'endpoint')
bucket.get_object('my-object', :file => 'local_file')
```

13.7. Use STS to authorize temporary access

You can use Alibaba Cloud Security Token Service (STS) to authorize temporary access to OSS.

You can use Alibaba Cloud STS to authorize temporary access to OSS. STS is a web service that provides temporary access tokens for users. You can use STS to grant a set of temporary access credentials that have a custom validity period and custom permissions to a third-party application or a RAM user managed by you. For more information about STS, see [What is STS?](#)

STS provides the following benefits:

- You need only to generate an access token and send the access token to a third-party application, instead of exposing your AccessKey pair to the third-party application. You can define the access permissions and validity period of this token.
- The token automatically expires after the validity period. Therefore, you do not need to manually revoke the access permissions of a token.

To use STS to access OSS, you must set `:sts_token`, as shown in the following sample code:

```
require 'aliyun/sts'
require 'aliyun/oss'
sts = Aliyun::STS::Client.new(
  access_key_id: '<The AccessKey ID of the RAM user>',
  access_key_secret: '<The AccessKey secret of the RAM user>')
token = sts.assume_role('<role-arn>', '<session-name>')
client = Aliyun::OSS::Client.new(
  endpoint: '<endpoint>',
  access_key_id: token.access_key_id,
  access_key_secret: token.access_key_secret,
  sts_token: token.security_token)
bucket = client.get_bucket('my-bucket')
```

```
require 'aliyun/sts'
require 'aliyun/oss'
sts = Aliyun::STS::Client.new(
  access_key_id: '<The AccessKey ID of the RAM user>',
  access_key_secret: '<The AccessKey secret of the RAM user>')
policy = Aliyun::STS::Policy.new
policy.allow(['oss:Get*'], ['acs:oss:*:*:my-bucket/*'])
token = sts.assume_role('<role-arn>', '<session name>', policy, 15 * 60)
client = Aliyun::OSS::Client.new(
  endpoint: 'ENDPOINT',
  access_key_id: token.access_key_id,
  access_key_secret: token.access_key_secret,
  sts_token: token.security_token)
bucket = client.get_bucket('my-bucket')
```

For more information about how to use STS to authorize temporary access and how to set parameters, visit [Alibaba Cloud OSS SDK for Ruby](#).

13.8. Exception handling

When a request error occurs during SDK usage, an exception is thrown, and the error details are logged (saved to the program runtime directory `oss_sdk.log` by default).

The two types of OSS Ruby SDK exceptions are: `ClientError` and `ServerError` and both are the subclasses of `RuntimeError`.

ClientError

`ClientError` is an internal SDK error, such as incorrect parameter setting or object modification during the resumable upload or download.

ServerError

`ServerError` indicates a server error, generated by parsing a server error message. `ServerError` has the following attributes:

- `http_code`: The HTTP status code of the error request.
- `error_code`: OSS error code.
- `message`: OSS error message.
- `request_id`: The UUID that uniquely identifies the request. When you cannot solve the problem, you can seek help from OSS development engineers by providing this `RequestId`.

For more information, see [OSS error response](#).

13.9. Rails application

This topic describes how to use Object Storage Service (OSS) SDK for Ruby in Rails applications to list buckets, upload objects, and download objects.

Preparations

To use OSS SDK for Ruby in Rails applications, you must add the following dependency to `Gemfile` :

```
gem 'aliyun-sdk', '~> 0.3.0'
```

Add the following dependency when you use OSS SDK for Ruby:

```
require 'aliyun/oss'
```

Integration with Rails

You can perform the following steps to integrate OSS SDK for Ruby with Rails.

1. Step 1: Create a project.
 - i. Install Rails and create a Rails application named `oss-manager`.

```
gem install rails
rails new oss-manager
```

 **Note** In this example, the `oss-manager` application is a management tool for OSS objects. It can list all buckets, list objects in a bucket by directory, upload objects, and download objects.

- ii. Use git to manage the project code.

```
cd oss-manager
git init
git add .
git commit -m "init project"
```

2. Step 2: Add the dependency on OSS SDK for Ruby.

- i. Edit the `oss-manager/Gemfile` file to add the dependency on OSS SDK for Ruby.

```
gem 'aliyun-sdk', '~> 0.3.0'
```

- ii. Run the `bundle install` command under `oss-manager/`.

- iii. Save the changes.

```
git add .
git commit -m "add aliyun-sdk dependency"
```

Initialize the OSS client

To avoid initialization each time when you use the OSS client in the project, we recommend that you create an initialization file in the project.

```
# oss-manager/config/initializers/aliyun_oss_init.rb
require 'aliyun/oss'
module OSS
  def self.client
    unless @client
      Aliyun::Common::Logging.set_log_file('./log/oss_sdk.log')
      @client = Aliyun::OSS::Client.new(
        endpoint:
          Rails.application.secrets.aliyun_oss['endpoint'],
        access_key_id:
          Rails.application.secrets.aliyun_oss['access_key_id'],
        access_key_secret:
          Rails.application.secrets.aliyun_oss['access_key_secret']
      )
    end
    @client
  end
end
```

The preceding code can be found in the `rails/` directory of the installation path of OSS SDK for Ruby. You can use the OSS client in the project after initialization.

```
buckets = OSS.client.list_buckets
```

The endpoint, AccessKey ID, and AccessKey secret are saved in the `oss-manager/conf/secrets.yml` file.

```
development:
  secret_key_base: xxxx
  aliyun_oss:
    endpoint: xxxx
    access_key_id: aaaa
    access_key_secret: bbbb
```

Save the changes.

```
git add .
git commit -m "add aliyun-sdk initializer"
```

List all buckets

You can perform the following steps to list all buckets.

1. Step 1: Use Rails to generate the controller to manage buckets.

```
rails g controller buckets index
```

2. Step 2: Generate the following files in the `oss-manager` directory.

- `app/controller/buckets_controller.rb`: contains the logical code related to buckets.
- `app/views/buckets/index.html.erb`: contains the code used to display the content related to buckets.
- `app/helpers/buckets_helper.rb`: defines the auxiliary functions.

3. Step 3: Edit the `buckets_controller.rb` file and call the OSS client to save the result of `list_buckets` to the `@buckets` variable.

```
class BucketsController < ApplicationController
  def index
    @buckets = OSS.client.list_buckets
  end
end
```

4. Step 4: Edit the `views/Buckets/index.html.erb` file to list the bucket list.


```
<h1>Buckets</h1>
<table class="table table-striped">
  <tr>
    <th>Name</th>
    <th>Location</th>
    <th>CreationTime</th>
  </tr>
  <% @buckets.each do |bucket| %>
    <tr>
      <td><%= link_to bucket.name, bucket_objects_path(bucket.name) %></td>
      <td><%= bucket.location %></td>
      <td><%= bucket.creation_time.localtime.to_s %></td>
    </tr>
  <% end %>
</table>
```

5. Step 5: Define the `bucket_objects_path` auxiliary function in the `app/helpers/buckets_helper.rb` file.

```
module BucketsHelper
  def bucket_objects_path(bucket_name)
    "/buckets/#{bucket_name}/objects"
  end
end
```

After you complete the preceding steps, you can list all buckets.

In addition, you must configure the routing of Rails to make sure that the address you enter in the browser can use the correct logic. Edit the `config/routes.rb` file and add the following content:

```
resources :buckets do
  resources :objects
end
```

Enter `rails s` under `oss-manager/` to start the Rails server, and then enter `http://localhost:3000/buckets/` in the browser to view the bucket list.

6. Step 6: Save the changes.

```
git add .
git commit -m "add list buckets feature"
```

List all objects in a bucket by directory

You can perform the following steps to list all objects in a bucket by directory:

1. Step 1: Generate a controller to manage objects.

```
rails g controller objects index
```

2. Step 2: Edit the app/controllers/objects_controller.rb file.

```
class ObjectsController < ApplicationController
  def index
    @bucket_name = params[:bucket_id]
    @prefix = params[:prefix]
    @bucket = OSS.client.get_bucket(@bucket_name)
    @objects = @bucket.list_objects(:prefix => @prefix, :delimiter => '/')
  end
end
```

You can obtain the bucket name from the parameters in the URL. Add a prefix to list objects by directory and call the list_object operation of the OSS client to obtain the object list.

 **Note** In this example, objects whose names contain the specified prefix and a forward slash (/) as a delimiter are obtained. For more information, see [Manage objects](#).

3. Step 3: Edit the app/views/objects/index.html.erb file.

```
<h1>Objects in <%= @bucket_name %></h1>
<p> <%= link_to 'Upload file', new_object_path(@bucket_name, @prefix) %></p>
<table class="table table-striped">
  <tr>
    <th>Key</th>
    <th>Type</th>
    <th>Size</th>
    <th>LastModified</th>
  </tr>
  <tr>
    <td><%= link_to '..', with_prefix(upper_dir(@prefix)) %></td>
    <td>Directory</td>
    <td>N/A</td>
    <td>N/A</td>
  </tr>
  <% @objects.each do |object| %>
  <tr>
    <% if object.is_a?(Aliyun::OSS::Object) %>
    <td><%= link_to remove_prefix(object.key, @prefix),
      @bucket.object_url(object.key) %></td>
    <td><%= object.type %></td>
    <td><%= number_to_human_size(object.size) %></td>
    <td><%= object.last_modified.localtime.to_s %></td>
    <% else %>
    <td><%= link_to remove_prefix(object, @prefix), with_prefix(object) %></td>
    <td>Directory</td>
    <td>N/A</td>
    <td>N/A</td>
    <% end %>
  </tr>
  <% end %>
</table>
```

Objects are listed by directory based on the following logic:

- i. The content before the first forward slash (/) in the object name is considered as the upper-level directory.
- ii. Common prefixes are listed as directories.
- iii. Objects are displayed as files.

The auxiliary functions used in the preceding code, such as `with_prefix` and `remove_prefix` are defined in the `app/helpers/objects_helper.rb` file.

```
module ObjectsHelper
  def with_prefix(prefix)
    "?prefix=#{prefix}"
  end
  def remove_prefix(key, prefix)
    key.sub(/^#{prefix}/, '')
  end
  def upper_dir(dir)
    dir.sub(/^[^/]+\$/, '') if dir
  end
  def new_object_path(bucket_name, prefix = nil)
    "/buckets/#{bucket_name}/objects/new/#{with_prefix(prefix)}"
  end
  def objects_path(bucket_name, prefix = nil)
    "/buckets/#{bucket_name}/objects/#{with_prefix(prefix)}"
  end
end
```

4. Step 4: Run `rails s` and enter `http://localhost:3000/buckets/my-bucket/objects/` in the browser to view the object list.

5. Step 5: Save the changes.

```
git add .
git commit -m "add list objects feature"
```

Download objects

In the steps in the preceding section, URLs are added to the objects. You can use the URL of an object to download the object.

```
<td><%= link_to remove_prefix(object.key, @prefix),
  @bucket.object_url(object.key) %></td>
```

You can also use `bucket.object_url` to generate a temporary URL for the object you want to download. For more information, see [Download objects](#).

Upload objects

In Rails applications on the server, you can use the following methods to upload objects:

- Upload objects in a way similar to normal upload. You must first upload an object to the Rails server. Then, the Rails server uploads the object to OSS.

- Upload objects by using the forms and temporary credentials generated by the server.
 - i. Add the `#new` method to the `app/controllers/objects_controller.rb` file to generate upload forms.

```
def new
  @bucket_name = params[:bucket_id]
  @prefix = params[:prefix]
  @bucket = OSS.client.get_bucket(@bucket_name)
  @options = {
    :prefix => @prefix,
    :redirect => 'http://localhost:3000/buckets/'
  }
end
```

- ii. Edit the `app/views/objects/new.html.erb` file.

```
<h2>Upload object</h2>
<%= upload_form(@bucket, @options) do %>
  <table class="table table-striped">
    <tr>
      <td><label>Bucket:</label></td>
      <td><%= @bucket.name %></td>
    </tr>
    <tr>
      <td><label>Prefix:</label></td>
      <td><%= @prefix %></td>
    </tr>
    <tr>
      <td><label>Select file:</label></td>
      <td><input type="file" name="file" style="display:inline" /></td>
    </tr>
    <tr>
      <td colspan="2">
        <input type="submit" class="btn btn-default" value="Upload" />
        <span>&nbsp;&nbsp;&nbsp;</span>
        <%= link_to 'Back', objects_path(@bucket_name, @prefix) %>
      </td>
    </tr>
  </table>
<% end %>
```

Note `upload_form` is an auxiliary function provided by OSS SDK for Ruby. It is used to generate upload forms and is defined in the `rails/aliyun_oss_helper.rb` file. You must copy the function to the `app/helpers/` directory. Then, you can run `rails s` and access `http://localhost:3000/buckets/my-bucket/objects/new` to upload objects to OSS.

- iii. Save the changes.

```
git add .
git commit -m "add upload object feature"
```

Add styles

You can perform the following steps to add styles to the interface of the Rails application.

1. Step 1: Download **bootstrap**. Decompress the package and then copy the `bootstrap.min.css` file to the `app/assets/stylesheets/` directory.
2. Modify the `app/views/layouts/application.html.erb` file and change the `yield` line to the following content:

```
<div id="main">
  <%= yield %>
</div>
```

3. Add a `<div>` field whose ID is main for each page. Modify the `app/assets/stylesheets/application.css` file and add the following contents:

```
body {
  text-align: center;
}
div#main {
  text-align: left;
  width: 1024px;
  margin: 0 auto;
}
```

For the complete demo code, visit [OSS Rails Demo](#).