ALIBABA CLOUD

# Alibaba Cloud

## Tablestore

## Quick Start

Document Version: 20220120

Alibaba Cloud

# 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**. |
| `Courier font` | Courier font is used for commands | Run the `cd /d C:/window` command to enter the Windows system folder. |
| *Italic* | Italic formatting is used for parameters and variables. | `bae log list --instanceid` *Instance_ID* |
| [] or [a\|b] | This format is used for an optional value, where only one item can be selected. | `ipconfig [-all\|-t]` |
| {} or {a\|b} | This format is used for a required value, where only one item can be selected. | `switch {active\|stand}` |

# Table of Contents

# 1.Use Tablestore

You can access Tablestore by using the Tablestore console, the Tablestore CLI, and Tablestore SDKs. To get started with Tablestore, you can use the Tablestore console or the Tablestore CLI. This topic describes how to use the Wide Column model and TimeSeries model in the Tablestore console and Tablestore CLI.

## Use the Wide Column model

> **Note** For more information about the Wide Column model, see Wide Column model.

The following procedure describes how to use the Wide Column model in the Tablestore console. For more information, see Manage the Wide Column model in the Tablestore console.

1. Activate Tablestore.
2. Create an instance.
3. Create a data table.
4. Read and write data.
5. Use SQL to read data.

The following procedure describes how to use the Wide Column model in the Tablestore CLI. For more information, see Manage the Wide Column model in the Tablestore CLI.

1. Download and start the Tablestore CLI.
2. Activate Tablestore.
3. Create an instance.
4. Configure access information.
5. Create a data table and use the table.
6. Perform data operations.
7. Use SQL to read data.

## Use the TimeSeries model

> **Note** For more information about the TimeSeries model, see Overview.

The following procedure describes how to use the TimeSeries model in the Tablestore console. For more information, see Manage the TimeSeries model in the Tablestore console.

1. Activate Tablestore.
2. Create an instance.
3. Create a time series table.
4. Write data to the table.
5. Retrieve time series.
6. Query time series data.
7. Use SQL to read data.

The following procedure describes how to use the TimeSeries model in the Tablestore CLI. For more information, see Manage the TimeSeries model in the Tablestore CLI.

1. Download and start the Tablestore CLI.
2. Activate Tablestore.
3. Create an instance.
4. Configure access information.
5. Create a time series table and use the table.
6. Perform data operations.
7. Use SQL to read data.

# 2.Wide Column model

## 2.1. Manage the Wide Column model in the Tablestore console

You can access Tablestore in the Tablestore console, in the Tablestore CLI, and by using Tablestore SDKs. You can get started with Tablestore in the Tablestore console. This topic describes how to manage the Wide Column model in the Tablestore console.

### Prerequisites

Before you use Tablestore, make sure that you have an Alibaba Cloud account and you have completed the real-name verification. If you do not have an Alibaba Cloud account, the system prompts you to create an Alibaba Cloud account when you activate Tablestore.

### Step 1: Activate Tablestore

If Tablestore is activated, skip this operation. You must activate Tablestore only once. You are not charged when you activate Tablestore.

1. Log on to the Tablestore product page.
2. Click **Get it Free**.
3. On the **Tablestore (Pay-as-you-go)** page, click **Buy Now**.
4. On the **Confirm Order** page, select **I have read and agree to Tablestore (Pay-as-you-go) Agreement of Service** and click **Activate Now**.

   After you activate Tablestore, click **Management Console** to access the Tablestore console.

   You can also click **Console** in the upper-right corner of the homepage. Click the  icon. In the left-side navigation pane, choose **Products and Services > Table Store** to go to the Tablestore console.

### Step 2: Create an instance

Instances are the basic unit of resource management in Tablestore. Tablestore controls whether applications can access a Tablestore instance and collects statistics about the resources that are used by applications at the instance level. After an instance is created, you can create and manage tables in the instance.

1. Log on to the Tablestore console.
2. In the top navigation bar, select a region, for example, China (Hangzhou) or China (Shenzhen).

   The instance types that are available in the region are displayed in the console.
3. Click **Create Instance**.
4. On the **On-demand** tab in the **Billing Method** dialog box, specify **Instance Name**, select an instance type from the **Instance Type** drop-down list, and specify **Instance Description**.

   For more information about the naming conventions for instances and how to select an instance type, see Instance.

> **Notice**
>
> - You cannot change the instance type after the instance is created.
> - Each Alibaba Cloud account can create up to 10 instances. The name of an instance must be unique within the region where the instance resides.

5. Click **OK**.

## Step 3: Create a data table

You can read and write data in a data table. When you create a data table, you can create a global secondary index to read data.

1. On the **Overview** page, click the name of the instance that you want to manage or click Manage Instance in the **Actions** column of the instance that you want to manage.
2. On the **Instance Details** tab, click **Create Table**.

> **Note** You can create up to 64 data tables in an instance.

3. In the **Create Table** dialog box, specify **Table Name** and **Primary Key**.

| Parameter | Description |
| --- | --- |
| Table Name | The name of the data table, which is used to identify a data table in an instance.<br>The name must be 1 to 255 bytes in length and can contain letters, digits, and underscores (_). The name must start with a letter or an underscore (_). |

| Parameter | Description |
| --- | --- |
| Primary Key | The primary key column that is used to identify a record in a table.<br>Enter a name for the primary key column and select a data type. Click **Add Primary Key Column** to add a primary key column.<br>You can add one to four primary key columns. The first primary key column is the partition key. After you create a data table, you cannot modify the configurations and the order of primary key columns.<br>**Note**<br>◦ In Tablestore, only one primary key column can be used as an auto-increment primary key column for each data table. You cannot use partition keys as auto-increment primary key columns.<br>◦ After you specify a primary key column as an auto-increment primary key column, Tablestore automatically generates a value for the auto-increment primary key column when you write a row of data. The values of auto-increment primary key columns are incremental and unique within the rows that share the same partition key.<br>◦ Naming conventions for primary key columns: The name must be 1 to 255 bytes in length and can contain letters, digits, and underscores (_). The name must start with a letter or an underscore (_).<br>◦ The **STRING**, **INTEGER**, and **BINARY** data types are supported by primary key columns. |

4. (Optional) In the **Create Table** dialog box, turn on Advanced Settings or Global Secondary Index.

   To configure advanced settings such as time to live (TTL), turn on Advanced Settings. To create a global secondary index, turn on Global Secondary Index.

   ◦ Turn on **Advanced Settings** and then configure the advanced parameters. The following table describes the parameters.

| Parameter | Description |
| --- | --- |
| Time to Live | The duration during which the data in the data table can be retained. If the retention period exceeds the TTL value, Tablestore automatically deletes expired data. Unit: seconds.<br>Minimum value: 86400 seconds (one day). A value of -1 specifies that the data never expires. |

| Parameter | Description |
| --- | --- |
| Max Versions | The maximum number of versions that can be retained for data in attribute columns of the data table. If the number of versions of data in attribute columns exceeds the value of this parameter, the system deletes data of earlier versions.<br>The value of this parameter for an attribute column is a positive integer. |
| Max Version Offset | The maximum difference between the current system time and the specified data version. Unit: seconds. The value of the Max Version Offset parameter is a positive integer that can be greater than the number of seconds that have elapsed since 00:00:00 on January 1, 1970 (UTC).<br>The difference between the version number and the time at which the data is written must be less than or equal to the value of Max Version Offset. Otherwise, an error occurs when the data is written.<br>The valid version range of data in an attribute column is calculated by using the following formula: Valid version range = **[Data written time - Max version offset, Data written time + Max version offset)**. |
| Reserved Read Throughput | The reserved read or write throughput for the data table.<br>This parameter is unavailable for capacity instances. Data type: INTEGER. Valid values: 0 to 100000. Unit: capacity unit (CU).<br>■ If you set Reserved Read Throughput or Reserved Write Throughput to a value greater than 0, Tablestore allocates and reserves related resources for the data table. After you create the data table, Tablestore charges you for the reserved throughput resources.<br>■ If you set Reserved Read Throughput or Reserved Write Throughput to 0, Tablestore does not allocate or reserve related resources for the data table. |
| Reserved Write Throughput | |

- Turn on Global Secondary Index and create a global secondary index.
  - Click **+ Add** next to Pre-defined Column. Specify a name for the predefined column and select a data type from the drop-down list.

    Naming conventions for predefined columns: The name must be 1 to 255 bytes in length and can contain letters, digits, and underscores (_). The name must start with a letter or an underscore (_).

    Predefined columns support the STRING, INTEGER, BINARY, FLOAT, and BOOLEAN data types.
  - Click **Add Global Secondary Index**. You must configure the Index Name and Primary Key parameters. You can configure the Pre-defined Column parameter based on your requirements.

5. Click **OK**.

   After a data table is created, you can view the data table in the **Tables** section. If the data table that you created is not displayed in the list of data tables, click the icon to refresh the list of data tables.

## Step 4: Read and write data

You can write, update, delete, and read data in a data table.

1. In the **Tables** section of the **Instance Details** tab, click the name of the required data table, and click the **Query Data** tab. You can also click **Query** in the Actions column of the data table.
2. Insert a row of data.
   i. On the **Query Data** tab, click **Insert**.
   ii. In the **Insert** dialog box, configure the Primary Key Value parameter. Click **Add Column** and configure the **Name**, **Type**, **Value**, and **Version** parameters.

   By default, **System Time** is selected. This value specifies that the current system time is used as the version number of the data. You can clear **System Time** and enter the version number of the data.

   iii. Click **OK**.
3. Update a row of data.
   i. On the **Query Data** tab, select the row of data that you want to update. Click **Update**.
   ii. In the **Update** dialog box, modify the types and values of primary key columns, add or remove attribute columns, or update data in or delete data from attribute columns.

   - You can click **Add Column** to add an attribute column. You can also click the icon to delete an attribute column.
   - In the first Actions column, if you select **Update**, you can modify the data in attribute columns. If you select **Delete**, you can delete the data of the selected version. If you select **Delete All**, you can delete all versions of the data.

   iii. Click **OK**.
4. Read data.

   To query data in a single row, perform the following steps:

   i. On the **Query Data** tab, click **Search**.
   ii. Specify query conditions.
      a. Set Modes to **Get Row** and select a table or secondary index that you want to query from the Table/Secondary Index drop-down list.
      b. By default, the system returns all columns. To return specified attribute columns, turn off **All Columns** and enter the attribute columns that you want to return. Separate multiple attribute columns with commas (,).
      c. Configure the **Primary Key Value** parameter of the row that you want to query.

      The integrity and accuracy of the primary key values affect the query results.

      d. Configure the **Max Versions** parameter to specify the maximum number of versions to return.
   iii. Click **OK**.

   To query data within a specified range, perform the following steps:

   i.

ii. Specify query conditions.

a. Set Modes to **Range Search** and select a table or secondary index that you want to query from the Table/Secondary Index drop-down list.

b. By default, the system returns all columns. To return specified attribute columns, turn off **All Columns** and enter the attribute columns that you want to return. Separate multiple attribute columns with commas (,).

c. By default, the system returns all columns. To return specified attribute columns, turn off **All Columns** and enter the attribute columns that you want to return. Separate multiple attribute columns with commas (,).

d. Specify Start Primary Key Column and End Primary Key Column.

> **Note**
>
> - If you use the range query mode, the values in the first primary key column take priority. If the values of the start primary key column and the end primary key column are the same in the first primary key column, the system uses the values in the second primary key column to perform queries. The query rules for subsequent primary key columns are the same as the query rules for the first two primary key columns.
> - The Custom range is a left-open and right-closed interval.

e. Configure the **Max Versions** parameter to specify the maximum number of versions to return.

f. Set Sequence to **Forward Search** or **Backward Search**.

iii. Click **OK**.

5. Delete data.

i. On the **Query Data** tab, select the row of data that you want to delete. Click **Delete**.

ii. In the **Delete** message, click **OK**.

## Step 5: Use SQL to query data

The SQL query feature is compatible with MySQL query syntax and supports table creation by using Data Definition Language (DDL) statements. For existing data tables, you can execute the CREATE TABLE statement to create mapping tables for the existing data tables. Then, you can use SQL statements to access the data in the existing data tables.

1. On the **Overview** page, click the name of the instance that you want to manage or click Manage Instance in the **Actions** column of the instance that you want to manage.
2. On the **Query by Executing SQL Statement** tab, create a mapping table.

i. Click the + icon.


ii. In the **Create Mapping Table** dialog box, select a table, and click **Generate SQL Statement**.

The system automatically generates the data structures of the mapping table.

> **Notice** Make sure that the field data types in the mapping table match the field data types in the data table. For more information about data type mappings, see Data type mappings.

iii. After you modify the data structures based on your business requirements, hold down the left mouse button to select an SQL statement and click **Execute SQL Statement (F8)**.

After the execution is successful, the execution result is displayed in the **Execution Result** section.

> **Notice**
>
> - Before you execute an SQL statement, you must select the SQL statement that you want to execute. Otherwise, the system executes the first SQL statement by default.
> - You cannot select multiple SQL statements to execute at the same time. If you select multiple SQL statements, the system reports an error.


3. Execute the SELECT statement to query the required data.

# 2.2. Manage the Wide Column model in the Tablestore CLI

You can access Tablestore in the Tablestore console, in the Tablestore CLI, or by using Tablestore SDKs. You can get started with Tablestore in the Tablestore CLI. This topic describes how to manage the Wide Column model in the Tablestore CLI.

## Step 1: Download and start the Tablestore CLI

1. Download the Tablestore CLI package based on your operating system.

| Operating system | Download link |
| --- | --- |
| Windows | Windows 10 |
| Linux | ○ Linux (AMD64)<br>○ Linux (ARM64) |
| macOS | macOS |

2. Decompress the Tablestore CLI package that you downloaded. Go to the root directory of the Tablestore CLI and select a method that is used to start the Tablestore CLI based on your operating system.
   - For Windows, double-click the ts.exe file.
   - For Linux or macOS, run the `./ts` command.

   > **Note** If you do not have execute permissions to run the command in Linux or macOS, run the `chmod 755 ts` command to obtain permissions and start the Tablestore CLI.

   The following code shows a sample Tablestore startup interface:

```
# Welcome to use Command Line Tool for Aliyun Tablestore. Current Version is '2021-11-1
1'.
#    ________      _      _              _
#   |__    __|    | |    | |            | |
#      | |  __ _ | |__  | |  ___   ___ | |_   ___  _ __    ___
#      | | / _' || '_ \ | | / _ \ / __|| __| / _ \ | '__| / _ \
#      | || (_| || |_) || ||  __/ \__ \| |_ | (_) || |   |  __/
#      |_| \__,_||_.__/ |_| \___| |___/ \__| \___/ |_|    \___|
#
# Please visit our product website: https://www.aliyun.com/product/ots
# You can also join our DingTalk Chat Group (ID: 11789671 or 23307953) to discuss and a
sk Tablestore related questions.
#
tablestore>
```

## Step 2: Activate Tablestore

If Tablestore is activated, skip this operation. You must activate Tablestore only once. You are not charged when you activate Tablestore.

1. Run the config command to configure the AccessKey pair information.

   > **Notice** An AccessKey pair, which consists of the AccessKey ID and the AccessKey secret of an Alibaba Cloud account, is required. For information about how to obtain an AccessKey pair, see Obtain an AccessKey pair.

   ```
   config --id NTSVLeBHzgX2iZfcaXXPJ**** --key 7NR2DiotscDbauohSq9kSHX8BDp99bjs7eNpCR7o***
   *
   ```

2. Run the `enable_service` command to activate Tablestore.

## Step 3: Create an instance

Run the create_instance command to create a high-performance instance.

The following sample code shows how to create a high-performance instance named myinstance in the China (Hangzhou) region:

```
create_instance -d "First instance created by CLI." -n myinstance -r cn-hangzhou
```

## Step 4: Configure access information

Run the config command to configure access information.

The following sample code shows how to configure access information for the instance named myinstance:

```
config --endpoint https://myinstance.cn-hangzhou.ots.aliyuncs.com --instance myinstance --i
d NTSVLeBHzgX2iZfcaXXPJ**** --key 7NR2DiotscDbauohSq9kSHX8BDp99bjs7eNpCR7o****
```

## Step 5: Create and use a data table

After a data table is created, you can perform operations on the table or data in the table.

1. Run the following command to create a data table named mytable:

   ```
   create -t mytable --pk '[{"c":"uid", "t":"string"}, {"c":"pid", "t":"integer"}]'
   ```

2. Run the `use -t mytable` command to use the data table named mytable.

## Step 6: Perform data operations

You can insert, update, read, and delete a row of data based on your business requirements.

- Insert a row of data

  Insert a row of data into a data table. The value of the first primary key column in the row is "86". The value of the second primary key column in the row is 6771. The row contains the following two attribute columns: name and country. The name and country columns are of the string type.

  ```
  put --pk '["86", 6771]' --attr '[{"c":"name", "v":"redchen"}, {"c":"country", "v":"china"
  }]'
  ```

- Update a row of data

  Update a row of data in which the value of the first primary key column is "86" and the value of the second primary key column is 6771. Data is inserted regardless of whether the row exists. If the row exists, the inserted data overwrites the existing data.

```
update --pk '["86", 6771]' --attr '[{"c":"name", "v":"redchen"}, {"c":"country", "v":"chi
na"}]'  --condition ignore
```

- Read a row of data

  Read a row of data in which the value of the first primary key column is "86" and the value of the second primary key column is 6771.

```
get --pk '["86",6771]'
```

- Scan data

  The following sample code shows how to scan data in a data table to return up to 10 rows of data:

```
scan --limit 10
```

- Delete a row of data

  Delete a row of data in which the value of the first primary key column is "86" and the value of the second primary key column is 6771.

```
delete --pk '["86", 6771]'
```

## Step 7: Use SQL to query data

The SQL query feature is compatible with MySQL query syntax and supports table creation by using Data Definition Language (DDL) statements. For existing data tables, you can execute the CREATE TABLE statement to create mapping tables for the existing data tables. Then, you can use SQL statements to access the data in the existing data tables.

To use SQL statements to quickly query data in a table, perform the following steps:

1. Run the `sql` command to enter the SQL mode.
2. Run the following command to create a mapping table for the data table named mytable:

   > **Note** For more information about the field type mappings between data tables and mapping tables, see Data type mappings.

```
CREATE TABLE `mytable` (
     `uid` VARCHAR(1024),
     `pid` BIGINT(20),
     `b` DOUBLE,
     `c` BOOL,
     `d` MEDIUMTEXT,
     PRIMARY KEY(`uid`,`pid`)
);
```

3. Run the following command to query all data in the table named mytable:

```
SELECT * FROM mytable;
```

If you want to exit the SQL mode, run the `exit;` command.

# 3.TimeSeries model

## 3.1. Manage the TimeSeries model in the Tablestore console

You can access Tablestore in the Tablestore console, in the Tablestore CLI, or by using Tablestore SDKs. You can get started with Tablestore in the Tablestore console. This topic describes how to manage the TimeSeries model in the Tablestore console.

### Prerequisites

Before you use Tablestore, make sure that you have an Alibaba Cloud account and you have completed the real-name verification. If you do not have an Alibaba Cloud account, the system prompts you to create an Alibaba Cloud account when you activate Tablestore.

### Step 1: Activate Tablestore

If Tablestore is activated, skip this operation. You must activate Tablestore only once. You are not charged when you activate Tablestore.

1. Log on to the Tablestore product page.
2. Click **Get it Free**.
3. On the **Tablestore (Pay-as-you-go)** page, click **Buy Now**.
4. On the **Confirm Order** page, select **I have read and agree to Tablestore (Pay-as-you-go) Agreement of Service** and click **Activate Now**.

   After you activate Tablestore, click **Management Console** to access the Tablestore console.

   You can also click **Console** in the upper-right corner of the homepage. Click the  icon. In the left-side navigation pane, choose **Products and Services > Table Store** to go to the Tablestore console.

### Step 2: Create a public preview instance for the TimeSeries model

1. Log on to the Tablestore console.
2. On the **Overview** page, click **Create Public Preview Instance for TimeSeries Model**.
3. In the **Create Public Preview Instance for TimeSeries Model** dialog box, select a region and specify Instance Name and Instance Description based on your business requirements.

   > **Notice** Each Alibaba Cloud account can create up to 10 instances. The name of an instance must be unique within the region in which the instance resides.

4. Click **OK**.

### Step 3: Create a time series table

1. Log on to the Tablestore console.
2. On the **Overview** page, click the name of the instance in which you want to create a time series

table or click **Manage Instance** in the Actions column that corresponds to the instance.

3. On the **Instance Details** tab, click the **Time Series Tables** tab.
4. On the **Time Series Tables** tab, click **Create Time Series Table**.

   > **Note** You can also click **Generate Demo with One Click** to create a test table with sample data for a quick start. When you create a time series table, the system performs some initialization operations. Therefore, you need to wait for dozens of seconds until the time series are displayed.

5. In the **Create Time Series Table** dialog box, specify **Name** and **Time to Live** as described in the following table.

| Parameter | Description |
| --- | --- |
| Name | The name of the time series table, which is used to identify the time series table in an instance.<br>The name must be 1 to 128 characters in length and can contain letters, digits, and underscores (_). The name must start with a letter or an underscore (_).<br>The name of a time series table cannot be the same as the name of an existing data table. |
| Time to Live | The retention period of the data in the time series table. Unit: seconds. If the system detects that the difference between the current time and the time column that is passed to the table exceeds the specified TTL value, the system automatically deletes the expired data.<br>**Notice** In the time series table, the system determines the time when the data is generated based on the time column that is passed to the table, not the time when the data is written to the table.<br>The value of this parameter must be -1 or a value that is greater than or equal to 86400 seconds (one day). |

6. Click **OK**.

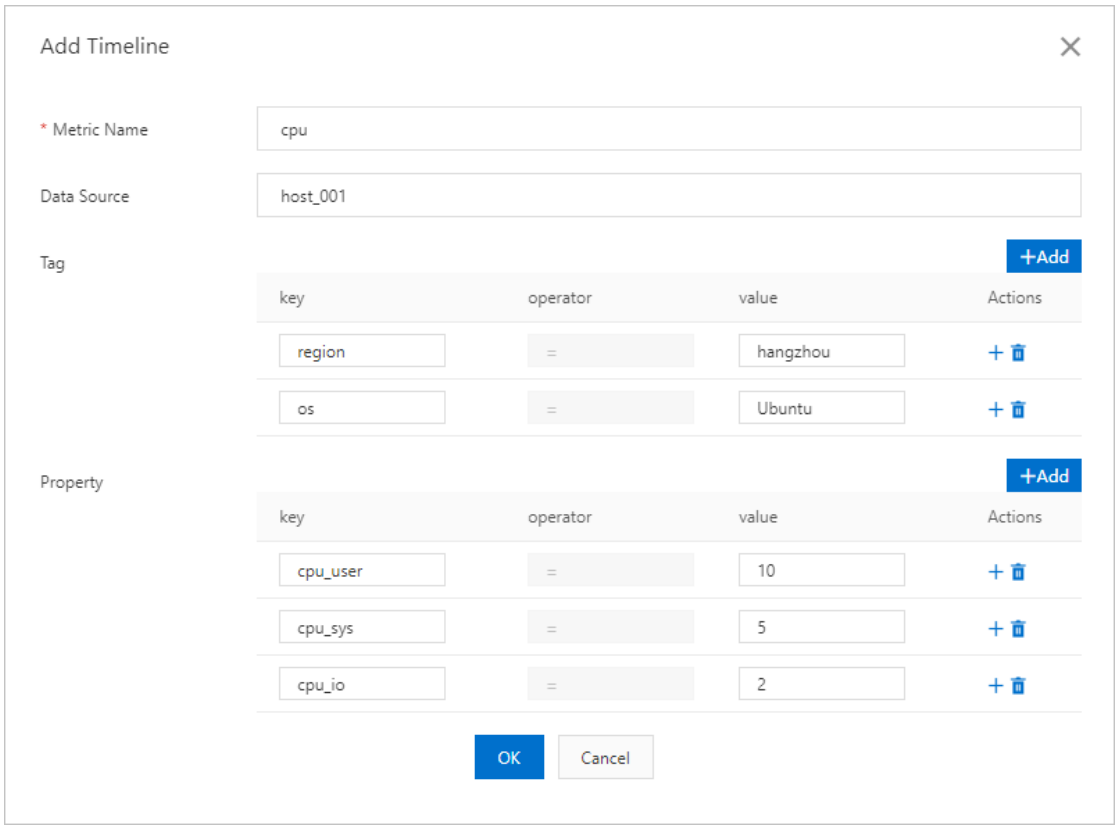
   After the time series table is created, you can view the time series table on the **Time Series Tables** tab. If the time series table is not displayed in the list of time series tables, click the icon to refresh the list of time series tables.

## Step 4: Write data to the time series table

Write time series data to the time series table in the Tablestore console. Time series data consists of metadata and data. If you do not create metadata before you write the time series data, the system automatically extracts the metadata from the written data.

1. On the **Time Series Tables** tab, click the name of the time series table and then click the **Query Data** tab or click **Manage Data** in the Actions column that corresponds to the time series table.

2. (Optional) Create a time series.
   i. On the **Query Data** tab, click **Add Timeline**.
   ii. In the **Add Timeline** dialog box, configure the metadata of the time series.

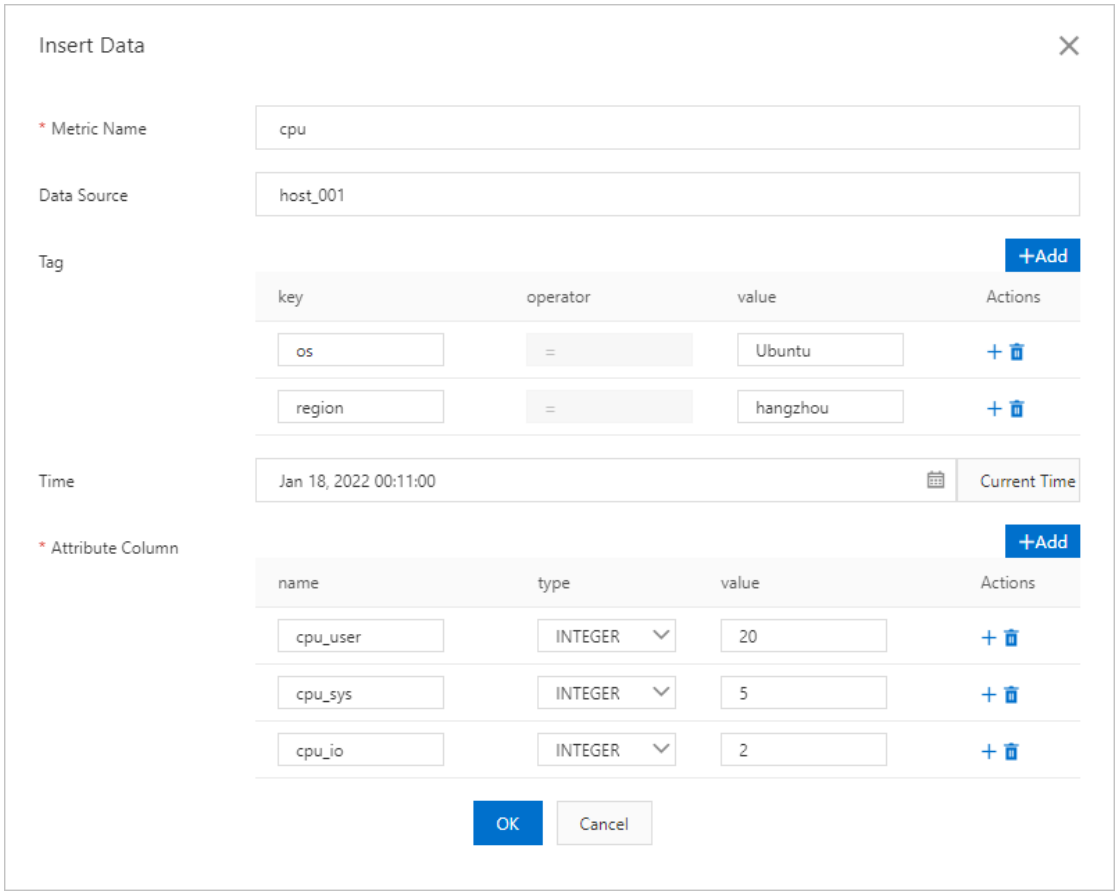



The following table describes the parameters that you can configure to add a time series.

| Parameter | Description |
| --- | --- |
| Metric Name | The name of a physical quantity or metric for the data in the time series, such as cpu or net, which specifies that the CPU usage or network usage is recorded in the time series. |
| Data Source | The identifier of the data source for the time series. This parameter can be empty. |
| Tag | The tag of the time series. You can customize multiple key-value pairs of the string type. |
| Property | The property column of the time series, which is used to record some property information of the time series. |

   iii. Click **OK**.
3. Insert data.
   i. Click **Insert Data**.

ii. In the **Insert Data** dialog box, specify Time and Attribute Column.



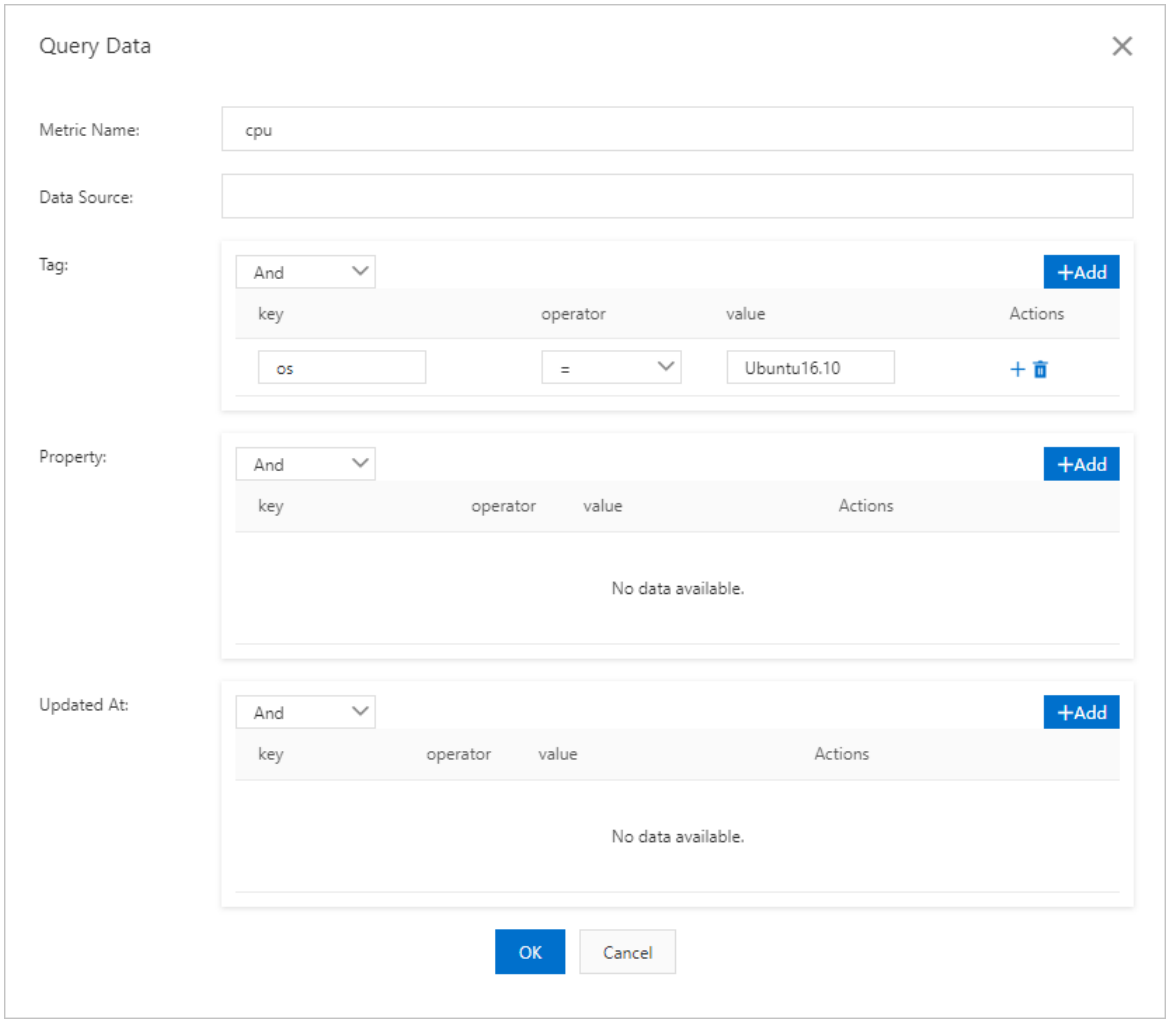

iii. Click **OK**.

## Step 5: Retrieve time series

Retrieves all the time series that meet the specified conditions.

1. On the **Query Data** tab, click **Query Data** in the upper-right corner.
2. In the **Query Data** dialog box, specify Metric Name and Data Source, and click **Add** in the Tag, Property, and Updated At sections to add conditions.

The following figure shows an example on how to query the time series in which the metric name is cpu and the tags contain os=Ubuntu16.10.




3. Click **OK**.

   The time series that meet the conditions are displayed on the **Query Data** tab.

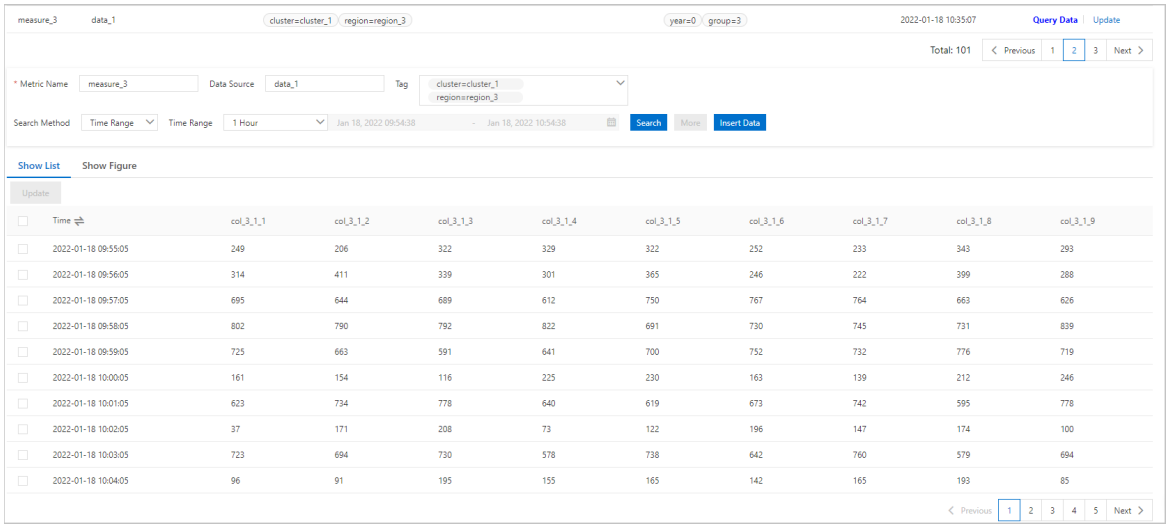
## Step 6: Query time series data

Query the data in a time series within a specific time range.

1. On the **Query Data** tab, click **Query Data** in the **Actions** column that corresponds to the time series whose data you want to query.
2. Select Time Range or Microsecond Timestamp from the drop-down list of Search Method, specify the time, and click **Search**.

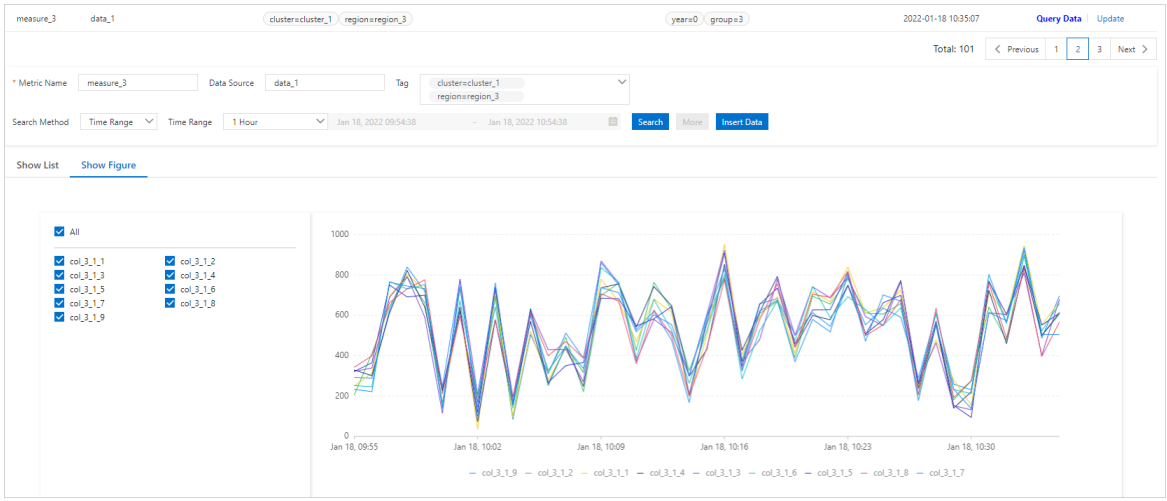
   The data that meets the conditions is displayed on the **Query Data** tab. The query results can be displayed in a list or figure.

The following figure shows an example of the query results in a list.




The following figure shows an example of the query results in a figure.

> **Note** Different colors in the figure represent different data columns. If you move the pointer over the data trend line, the values of the corresponding data columns are displayed. You can also select or clear specific data columns to display the required data columns.




## Step 7: Execute SQL statements to query time series data

SQL queries are compatible with the query syntax of MySQL. You can execute SQL statements to access data in time series tables.

1. On the **Overview** page, click the name of the instance that you want to manage or click Manage Instance in the **Actions** column of the instance that you want to manage.
2. On the **Query by Executing SQL Statement** tab, select a time series table from which you want to query data, and execute the SELECT statement to query data in the table.

# 3.2. Manage the TimeSeries model in the Tablestore CLI

You can access Tablestore in the Tablestore console, in the Tablestore CLI, or by using Tablestore SDKs. You can get started with Tablestore in the Tablestore CLI. This topic describes how to manage the TimeSeries model in the Tablestore CLI.

## Step 1: Download and start the Tablestore CLI

1. Download the Tablestore CLI package based on your operating system.

| Operating system | Download link |
| --- | --- |
| Windows | Windows 10 |
| Linux | ○ Linux (AMD64)<br>○ Linux (ARM64) |
| macOS | macOS |

2. Decompress the Tablestore CLI package that you downloaded. Go to the root directory of the Tablestore CLI and select a method that is used to start the Tablestore CLI based on your operating system.
   - For Windows, double-click the ts.exe file.
   - For Linux or macOS, run the `./ts` command.

   > **Note** If you do not have execute permissions to run the command in Linux or macOS, run the `chmod 755 ts` command to obtain permissions and start the Tablestore CLI.

   The following code shows a sample Tablestore startup interface:

```
# Welcome to use Command Line Tool for Aliyun Tablestore. Current Version is '2021-11-1
1'.
#    ________      _      _              _
#   |__    __|    | |    | |            | |
#      | |  __ _  | |__  | |  ___   ___ | |_   ___  _ __    ___
#      | | / _' || '_ \ | | / _ \ / __|| __| / _ \ | '__| / _ \
#      | || (_| || |_) || ||  __/ \__ \| |_ | (_) || |   |  __/
#      |_| \__,_||_.__/ |_| \___| |___/ \__| \___/ |_|    \___|
#
# Please visit our product website: https://www.aliyun.com/product/ots
# You can also join our DingTalk Chat Group (ID: 11789671 or 23307953) to discuss and a
sk Tablestore related questions.
#
tablestore>
```

## Step 2: Activate Tablestore

If Tablestore is activated, skip this operation. You must activate Tablestore only once. You are not charged when you activate Tablestore.

1. Run the config command to configure the AccessKey pair information.

> **Notice** An AccessKey pair, which consists of the AccessKey ID and the AccessKey secret of an Alibaba Cloud account, is required. For information about how to obtain an AccessKey pair, see Obtain an AccessKey pair.

```
config --id NTSVLeBHzgX2iZfcaXXPJ**** --key 7NR2DiotscDbauohSq9kSHX8BDp99bjs7eNpCR7o***
*
```

2. Run the `enable_service` command to activate Tablestore.

## Step 3: Create a public preview instance for the TimeSeries model

For more information, see Create a public preview instance for the TimeSeries model.

## Step 4: Configure access information

Run the config command to configure access information.

The following sample code shows how to configure access information for the instance named myinstance:

```
config --endpoint https://myinstance.cn-hangzhou.ots.aliyuncs.com --instance myinstance --i
d NTSVLeBHzgX2iZfcaXXPJ**** --key 7NR2DiotscDbauohSq9kSHX8BDp99bjs7eNpCR7o****
```

## Step 5: Create and use a time series table

After you create a time series table, you can perform operations on the table or data in the table.

1. Run the following command to create a time series table named mytable:

```
create -m timeseries -t mytable --ttl -1
```

2. Run the `use --ts -t mytable` command to use the time series table named mytable.

## Step 6: Perform data operations

Perform data operations based on your business requirements. You can write time series data, import time series data, query time series data, retrieve time series, scan time series, and update time series.

- Write time series data

  Insert a row of time series data.

```
putts --k '["cpu","localhost",["region=hangzhou","os=ubuntu"]]' --field '[{"c":"fieldname
","v":"fieldvalue"},{"c":"bool_field","v":true},{"c":"double_field","v":1.1},{"c":"int_va
lue","v":10,"isint":true}]' --time 1635162859000000
```

- Import time series data

  Import the time series data from the import_timeseries.txt file to a time series table.

```
import_timeseries --input /temp/import_timeseries.txt
```

  The following example shows the content of a configuration file:

```
cpu,hostname=host_0,region=cn-hangzhou usage_user=58i,usage_system=2i,usage_idle=24i 1609
459200000000000
cpu,hostname=host_1,region=cn-hangzhou usage_user=58i,usage_system=2i,usage_idle=24i 1609
459200000000000
```

- Query time series data

  Query all time series data that is generated before 1667638230000000 in the time series whose metric name is cpu, data source is localhost, and tags are "region=hangzhou" and "os=ubuntu".

```
getts --k '["cpu","localhost",["region=hangzhou","os=ubuntu"]]' --time_start 0 --time_end
1667638230000000 --limit 100
```

- Retrieve time series

  Retrieve the time series whose metric name is cpu and data source is localhost.

```
query_ts_meta --measurement cpu --datasource localhost --limit 10
```

- Scan time series

```
query_ts_meta --limit 10
```

- Update time series

  Modify the properties of the specified time series to "city=nanjing" and "region=jiangning".

```
update_ts_meta --k '["cpu","localhost",["city=hangzhou","region=xihu"]]' --attrs '["city=
nanjing","region=jiangning"]'
```

## Step 7: Use SQL to query data

To use SQL statements to quickly query data in a table, perform the following steps:

1. Run the `sql` command to enter the SQL mode.
2. Run the following command to query all data in the table named mytable:

```
SELECT * FROM mytable;
```

If you want to exit the SQL mode, run the `exit;` command.