

ALIBABA CLOUD

阿里云

短信服务
SDK参考（新版）

文档版本：20210512

 阿里云

法律声明

阿里云提醒您在使用或阅读本文档之前仔细阅读、充分理解本法律声明各条款的内容。如果您阅读或使用本文档，您的阅读或使用行为将被视为对本声明全部内容的认可。

1. 您应当通过阿里云网站或阿里云提供的其他授权通道下载、获取本文档，且仅能用于自身的合法合规的业务活动。本文档的内容视为阿里云的保密信息，您应当严格遵守保密义务；未经阿里云事先书面同意，您不得向任何第三方披露本手册内容或提供给任何第三方使用。
2. 未经阿里云事先书面许可，任何单位、公司或个人不得擅自摘抄、翻译、复制本文档内容的部分或全部，不得以任何方式或途径进行传播和宣传。
3. 由于产品版本升级、调整或其他原因，本文档内容有可能变更。阿里云保留在没有任何通知或者提示下对本文档的内容进行修改的权利，并在阿里云授权通道中不时发布更新后的用户文档。您应当实时关注用户文档的版本变更并通过阿里云授权渠道下载、获取最新版的用户文档。
4. 本文档仅作为用户使用阿里云产品及服务的参考性指引，阿里云以产品及服务的“现状”、“有缺陷”和“当前功能”的状态提供本文档。阿里云在现有技术的基础上尽最大努力提供相应的介绍及操作指引，但阿里云在此明确声明对本文档内容的准确性、完整性、适用性、可靠性等不作任何明示或暗示的保证。任何单位、公司或个人因为下载、使用或信赖本文档而发生任何差错或经济损失的，阿里云不承担任何法律责任。在任何情况下，阿里云均不对任何间接性、后果性、惩戒性、偶然性、特殊性或刑罚性的损害，包括用户使用或信赖本文档而遭受的利润损失，承担责任（即使阿里云已被告知该等损失的可能性）。
5. 阿里云网站上所有内容，包括但不限于著作、产品、图片、档案、资讯、资料、网站架构、网站画面的安排、网页设计，均由阿里云和/或其关联公司依法拥有其知识产权，包括但不限于商标权、专利权、著作权、商业秘密等。非经阿里云和/或其关联公司书面同意，任何人不得擅自使用、修改、复制、公开传播、改变、散布、发行或公开发表阿里云网站、产品程序或内容。此外，未经阿里云事先书面同意，任何人不得为了任何营销、广告、促销或其他目的使用、公布或复制阿里云的名称（包括但不限于单独为或以组合形式包含“阿里云”、“Aliyun”、“万网”等阿里云和/或其关联公司品牌，上述品牌的附属标志及图案或任何类似公司名称、商号、商标、产品或服务名称、域名、图案标示、标志、标识或通过特定描述使第三方能够识别阿里云和/或其关联公司）。
6. 如若发现本文档存在任何错误，请与阿里云取得直接联系。

通用约定

格式	说明	样例
 危险	该类警示信息将导致系统重大变更甚至故障，或者导致人身伤害等结果。	 危险 重置操作将丢失用户配置数据。
 警告	该类警示信息可能会导致系统重大变更甚至故障，或者导致人身伤害等结果。	 警告 重启操作将导致业务中断，恢复业务时间约十分钟。
 注意	用于警示信息、补充说明等，是用户必须了解的内容。	 注意 权重设置为0，该服务器不会再接受新请求。
 说明	用于补充说明、最佳实践、窍门等，不是用户必须了解的内容。	 说明 您也可以通过按Ctrl+A选中全部文件。
>	多级菜单递进。	单击设置> 网络> 设置网络类型。
粗体	表示按键、菜单、页面名称等UI元素。	在结果确认页面，单击确定。
Courier字体	命令或代码。	执行 <code>cd /d C:/window</code> 命令，进入Windows系统文件夹。
斜体	表示参数、变量。	<code>bae log list --instanceid</code> <i>Instance_ID</i>
[] 或者 [a b]	表示可选项，至多选择一个。	<code>ipconfig [-all -t]</code>
{ } 或者 {a b}	表示必选项，至多选择一个。	<code>switch {active stand}</code>

目录

- 1.版本说明 ----- 06
- 2.版本：V3 ----- 08
 - 2.1. Java SDK ----- 08
 - 2.2. Go SDK ----- 09
 - 2.3. .Net SDK ----- 12
 - 2.4. PHP SDK ----- 14
 - 2.5. Node.js SDK ----- 16
 - 2.6. Python SDK ----- 17
- 3.版本：V2 ----- 20
 - 3.1. 短信服务SDK简介 ----- 20
 - 3.2. 使用示例 (Demo) ----- 20
 - 3.3. Java SDK ----- 21
 - 3.4. Go SDK ----- 24
 - 3.5. Python SDK ----- 25
 - 3.6. .Net SDK ----- 26
 - 3.7. Node.js SDK ----- 28
 - 3.8. PHP SDK ----- 29
 - 3.9. 升级SDK ----- 31
 - 3.10. 升级消息回执API ----- 32
 - 3.10.1. 升级说明 ----- 32
 - 3.10.2. .NET Demo ----- 32
 - 3.10.3. PHP Demo ----- 35
 - 3.10.4. Python Demo ----- 36
 - 3.10.5. Go Demo ----- 38
 - 3.11. 调用示例 ----- 40
 - 3.11.1. C# Demo ----- 40

3.11.2. Node.js Demo	41
3.11.3. Python Demo	42
3.11.4. PHP Demo	43
3.11.5. JAVA Demo	44
4.版本: V1	46
4.1. Java SDK	46

1.版本说明

本文为您介绍SDK不同版本之间的区别，帮助您快速找到需要的SDK版本。

版本关系

 说明 推荐用户使用V2版本SDK。

- 升级版SDK：V3版本。
- 原版SDK：
 - CommonRequest版本：V2版本。
 - BusinessRequest版本：V1版本。

升级版SDK和原版SDK区别

对比项	升级版SDK	原版SDK
生成方案	基于DSL语言 Darabonba 。	基于模版生成，各编程语言之间采用不同的模版方案，各个语言之间功能完备程度不一致，使用体验不一。
语言支持	支持主流6门语言的SDK生成。	大部分云产品的老版本SDK无法支持超过3门以上的主流语言。
SDK示例	各语言间的代码在逻辑行为上保持一致，通过中间语言来统一生成，具有校验能力，有效保证示例代码的准确性。	各语言间的代码示例不一致，不能保证代码的正确性。
SDK场景化示例	通过DSL一次编写就能生成多语言的SDK场景化示例，可以协助开发者极大的简化SDK使用的理解成本， SDK 示例中心 。	多语言场景化示例编写难度大，覆盖语言数较少。

CommonRequest版本（V2版本）和BusinessRequest版本（V1版本）的区别

对比项	CommonRequest版本（V2版本）	BusinessRequest版本（V1版本）
定义	为开发者封装了签名逻辑，但是所有API调用都需要用户自己填写参数，请求方法，请求头等全量信息。	为开发者封装了包括签名逻辑，公有参数，公有请求头等所有通用逻辑，用户不用关注API的请求逻辑，只需要关注自身业务参数即可。
签名逻辑	包含，开发者无需关注。	包含，开发者无需关注。
请求方法	需要用户通过文档查询，API是通过POST、GET或其他方法请求后，赋值给request对象的对应属性。	开发者无需关注，SDK中已经生成了相关方法。

对比项	CommonRequest版本（V2版本）	BusinessRequest版本（V1版本）
公有参数	需要开发者查看文档获取请求API对应的产品参数，版本参数，返回体对应格式以及请求的 Action参数或API对应的URI参数。	开发者无需关注，SDK中已经生成了全量公有参数。
业务参数	需要开发者查看文档来查看API需要哪些参数，并且需要按照文档指示的格式来序列化参数，序列化开发和调试成本较高。	SDK中为用户提供了业务参数的结构化 Request对象，按Ide提示即可完成编写，序列化过程由SDK完成，保障正确性。
返回结果	直接返回未经处理的字符串，开发者可通过API文档查看如何解析结果，并通过文档序列化处理后的返回值组成结构化的对象来使用。	SDK中为用户做了返回结果的解析，并返回跟文档一致的结构化 Response对象，按Ide提示即可找到自己需要的参数来使用。
产品支持能力	支持所有产品。	部分云产品不提供原版SDK，所有云产品提供新版SDK。
SDK示例	仅 OpenAPI 开发者门户 提供没有原版SDK 产品的示例。	OpenAPI 开发者门户 不仅提供所有新版 SDK和原版SDK示例，还为超过50款产品提供了 SDK 场景化示例 。

2.版本：V3

2.1. Java SDK

本文为您介绍短信服务Java SDK的使用方法及其示例。

准备工作

- 在安装和使用阿里云SDK前，确保您已经注册阿里云账号并生成访问密钥（AccessKey）。详情请参见[AccessKey](#)。
- 将SDK添加到项目中。

引入短信服务SDK依赖，通过Maven二方库依赖的方式将短信服务的SDK加入到自己的项目中。

```
<dependency>
  <groupId>com.aliyun</groupId>
  <artifactId>dysmsapi20170525</artifactId>
  <version>[2.0.0,3.0.0)</version>
</dependency>
```

发送短信

- 代码示例

```
package com.aliyun.sample;
import com.aliyun.tea.*;
import com.aliyun.dysmsapi20170525.*;
import com.aliyun.dysmsapi20170525.models.*;
import com.aliyun.teaopenapi.*;
import com.aliyun.teaopenapi.models.*;
public class Sample {
    /**
     * 使用AK&SK初始化账号Client
     * @param accessKeyId
     * @param accessKeySecret
     * @return Client
     * @throws Exception
     */
    public static com.aliyun.dysmsapi20170525.Client createClient(String accessKeyId, String accessKeySecret) throws Exception {
        Config config = new Config()
            // 您的AccessKey ID
            .setAccessKeyId(accessKeyId)
            // 您的AccessKey Secret
            .setAccessKeySecret(accessKeySecret);
        // 访问的域名
        config.endpoint = "dysmsapi.aliyuncs.com";
        return new com.aliyun.dysmsapi20170525.Client(config);
    }
    public static void main(String[] args_) throws Exception {
        java.util.List<String> args = java.util.Arrays.asList(args_);
        com.aliyun.dysmsapi20170525.Client client = Sample.createClient("accessKeyId", "accessKeySecret");
        SendSmsRequest sendSmsRequest = new SendSmsRequest()
            .setPhoneNumbers("1503871****")
            .setSignName("阿里大于测试专用")
            .setTemplateCode("SMS_7125****")
            .setTemplateParam("{\"customer\":\"1111\"}");
        // 复制代码运行请自行打印 API 的返回值
        client.sendSms(sendSmsRequest);
    }
}
```

- 结果示例

```
{ "RequestId": "614048FB-0619-4439-A1D5-AA8B218A****", "Message": "OK", "BizId": "386715418801811068^0", "Code": "OK"}
```

相关文档

- 如果您想了解更多示例，请参见[OpenAPI开发者门户](#)。

 说明 登录开发者门户平台，SDK版本选择时，请选择升级版 SDK。

- 短信服务相关API，请参见[API概览](#)。

2.2. Go SDK

本文为您介绍短信服务Go SDK的使用方法及示例。

准备工作

- 在安装和使用阿里云SDK前，确保您已经注册阿里云账号并生成访问密钥（AccessKey）。详情请参见[AccessKey](#)。
- 使用Govendor工具执行以下命令，安装阿里云Go SDK。

```
go get github.com/alibabacloud-go/dysmsapi-20170525/v2
```

发送短信

- 代码示例

```
package main
import (
    "os"
    dysmsapi20170525 "github.com/alibabacloud-go/dysmsapi-20170525/v2/client"
    openapi "github.com/alibabacloud-go/darabonba-openapi/client"
    "github.com/alibabacloud-go/tea/tea"
)
/**
 * 使用AK&SK初始化账号Client
 * @param accessKeyId
 * @param accessKeySecret
 * @return Client
 * @throws Exception
 */
func CreateClient (accessKeyId *string, accessKeySecret *string) (_result *dysmsapi20170525.Client, _err error) {
    config := &openapi.Config{
        // 您的AccessKey ID
        AccessKeyId: accessKeyId,
        // 您的AccessKey Secret
        AccessKeySecret: accessKeySecret,
    }
    // 访问的域名
    config.Endpoint = tea.String("dysmsapi.aliyuncs.com")
    _result = &dysmsapi20170525.Client{}
    _result, _err = dysmsapi20170525.NewClient(config)
    return _result, _err
}
func _main (args []*string) (_err error) {
    client, _err := CreateClient(tea.String("accessKeyId"), tea.String("accessKeySecret"))
    if _err != nil {
        return _err
    }
    sendSmsRequest := &dysmsapi20170525.SendSmsRequest{
        PhoneNumbers: tea.String("1503871****"),
        SignName: tea.String("阿里大于测试专用"),
        TemplateCode: tea.String("SMS_71255707"),
        TemplateParam: tea.String("{\"customer\":\"1111\"}"),
    }
    // 复制代码运行请自行打印 API 的返回值
    _, _err = client.SendSms(sendSmsRequest)
    if _err != nil {
        return _err
    }
    return _err
}
func main() {
    err := _main(os.Args[1:])
    if err != nil {
        panic(err)
    }
}
```

- 结果示例

```
{ "RequestId": "614048FB-0619-4439-A1D5-AA8B218A****", "Message": "OK", "BizId": "386715418801811068^0", "Code": "OK"}
```

相关文档

- 如果您想了解更多示例，请参见[OpenAPI开发者门户](#)。

 说明 登录开发者门户平台，SDK版本选择时，请选择升级版 SDK。

- 短信服务相关API，请参见[API概览](#)。

2.3. .Net SDK

本文为您介绍短信服务.Net SDK的使用方法及其示例。

准备工作

- 在安装和使用阿里云SDK前，确保您已经注册阿里云账号并生成访问密钥（AccessKey）。详情请参见[AccessKey](#)。
- 安装SDK。
 - i. 在解决方案资源管理器面板中，右键单击您的项目，选择管理NuGet程序包菜单。
 - ii. 在打开的NuGet管理面板中单击浏览选项卡输入 `AlibabaCloud.SDK.Dysmsapi20170525` 。
 - iii. 单击安装。

发送短信

- 代码示例

```
using System;
using System.Collections;
using System.Collections.Generic;
using System.IO;
using System.Threading.Tasks;
using Tea;
using Tea.Utils;
namespace AlibabaCloud.SDK.Sample
{
    public class Sample
    {
        /**
         * 使用AK&SK初始化账号Client
         * @param accessKeyId
         * @param accessKeySecret
         * @return Client
         * @throws Exception
         */
        public static AlibabaCloud.SDK.Dysmsapi20170525.Client CreateClient(string accessKeyId, string accessKeySecret)
        {
            AlibabaCloud.OpenApiClient.Models.Config config = new AlibabaCloud.OpenApiClient.Models.Config
            {
                // 您的AccessKey ID
                AccessKeyId = accessKeyId,
                // 您的AccessKey Secret
                AccessKeySecret = accessKeySecret,
            };
            // 访问的域名
            config.Endpoint = "dysmsapi.aliyuncs.com";
            return new AlibabaCloud.SDK.Dysmsapi20170525.Client(config);
        }
        public static void Main(string[] args)
        {
            AlibabaCloud.SDK.Dysmsapi20170525.Client client = CreateClient("accessKeyId", "accessKeySecret");
            AlibabaCloud.SDK.Dysmsapi20170525.Models.SendSmsRequest sendSmsRequest = new AlibabaCloud.SDK.Dysmsapi20170525.Models.SendSmsRequest
            {
                PhoneNumbers = "1503871****",
                SignName = "阿里大于测试专用",
                TemplateCode = "SMS_71255707",
                TemplateParam = "{\"customer\":\"1111\"}",
            };
            // 复制代码运行请自行打印 API 的返回值
            client.SendSms(sendSmsRequest);
        }
    }
}
```

- 结果示例

```
{ "RequestId": "614048FB-0619-4439-A1D5-AA8B218A****", "Message": "OK", "BizId": "386715418801811068^0", "Code": "OK"}
```

相关文档

- 如果您想了解更多示例，请参见[OpenAPI开发者门户](#)。

 说明 登录开发者门户平台，SDK版本选择时，请选择升级版 SDK。

- 短信服务相关API，请参见[API概览](#)。

2.4. PHP SDK

本文为您介绍短信服务PHP SDK的使用方法及其示例。

准备工作

- 在安装和使用阿里云SDK前，确保您已经注册阿里云账号并生成访问密钥（AccessKey）。详情请参见[AccessKey](#)。
- 安装镜像和SDK。

composer镜像。

```
https://developer.aliyun.com/composer
```

安装依赖。

```
composer require alibabacloud/dysmsapi-20170525
```

发送短信

- 代码示例

```

<?php
namespace AlibabaCloud\SDK\Sample;
use AlibabaCloud\SDK\Dysmsapi\V20170525\Dysmsapi;
use Darabonba\OpenApi\Models\Config;
use AlibabaCloud\SDK\Dysmsapi\V20170525\Models\SendSmsRequest;
class Sample {
    /**
     * 使用AK&SK初始化账号Client
     * @param string $accessKeyId
     * @param string $accessKeySecret
     * @return Dysmsapi Client
     */
    public static function createClient($accessKeyId, $accessKeySecret){
        $config = new Config([
            // 您的AccessKey ID
            "accessKeyId" => $accessKeyId,
            // 您的AccessKey Secret
            "accessKeySecret" => $accessKeySecret
        ]);
        // 访问的域名
        $config->endpoint = "dysmsapi.aliyuncs.com";
        return new Dysmsapi($config);
    }
    /**
     * @param string[] $args
     * @return void
     */
    public static function main($args){
        $client = self::createClient("accessKeyId", "accessKeySecret");
        $sendSmsRequest = new SendSmsRequest([
            "phoneNumbers" => "1503871****",
            "signName" => "阿里大于测试专用",
            "templateCode" => "SMS_215180185",
            "templateParam" => "{\"code\":\"1111\", \"code1\":\"1111\", \"code3\":\"1111\"}"
        ]);
        // 复制代码运行请自行打印 API 的返回值
        $client->sendSms($sendSmsRequest);
    }
}
$path = __DIR__ . \DIRECTORY_SEPARATOR . '..' . \DIRECTORY_SEPARATOR . 'vendor' . \DIRECTORY_SEPARATOR . 'autoload.php';
if (file_exists($path)) {
    require_once $path;
}
Sample::main(array_slice($argv, 1));

```

- 结果示例

```
{ "RequestId": "614048FB-0619-4439-A1D5-AA8B218A****", "Message": "OK", "BizId": "386715418801811068^0", "Code": "OK"}
```

相关文档

- 如果您想了解更多示例，请参见[OpenAPI开发者门户](#)。

 说明 登录开发者门户平台，SDK版本选择时，请选择升级版 SDK。

- 短信服务相关API，请参见[API概览](#)。

2.5. Node.js SDK

本文为您介绍短信服务Node.js SDK的使用方法及其示例。

准备工作

- 在安装和使用阿里云SDK前，确保您已经注册阿里云账号并生成访问密钥（AccessKey）。详情请参见[AccessKey](#)。
- 安装短信服务Node.js SDK。

```
npm install @alicloud/dysmsapi20170525@2.0.1 --save
```

发送短信

- 代码示例

```
import Dysmsapi20170525, * as $Dysmsapi20170525 from '@alicloud/dysmsapi20170525';
import OpenApi, * as $OpenApi from '@alicloud/openapi-client';
import * as $tea from '@alicloud/tea-typescript';
export default class Client {
  /**
   * 使用AK&SK初始化账号Client
   * @param accessKeyId
   * @param accessKeySecret
   * @return Client
   * @throws Exception
   */
  static createClient(accessKeyId: string, accessKeySecret: string): Dysmsapi20170525 {
    let config = new $OpenApi.Config({
      // 您的AccessKey ID
      accessKeyId: accessKeyId,
      // 您的AccessKey Secret
      accessKeySecret: accessKeySecret,
    });
    // 访问的域名
    config.endpoint = "dysmsapi.aliyuncs.com";
    return new Dysmsapi20170525(config);
  }
  static async main(args: string[]): Promise<void> {
    let client = Client.createClient("accessKeyId", "accessKeySecret");
    let sendSmsRequest = new $Dysmsapi20170525.SendSmsRequest({
      phoneNumbers: "15038712376",
      signName: "阿里大于测试专用",
      templateCode: "SMS_209335004",
      templateParam: "{\"code\":\"1111\"}",
    });
    // 复制代码运行请自行打印 API 的返回值
    await client.sendSms(sendSmsRequest);
  }
}
Client.main(process.argv.slice(2));
```

- 结果示例

```
{ "RequestId": "614048FB-0619-4439-A1D5-AA8B218A****", "Message": "OK", "BizId": "386715418801811068^0", "Code": "OK" }
```

相关文档

- 如果您想了解更多示例，请参见[OpenAPI开发者门户](#)。

 说明 登录开发者门户平台，SDK版本选择时，请选择升级版 SDK。

- 短信服务相关API，请参见[API概览](#)。

2.6. Python SDK

本文为您介绍短信服务Python SDK的使用方法及其示例。

准备工作

- 在安装和使用阿里云SDK前，确保您已经注册阿里云账号并生成访问密钥（AccessKey）。详情请参见[AccessKey](#)。
- 安装SDK。

```
pip install alibabacloud_dysmsapi20170525
```

发送短信

- 代码示例

```
import sys
from typing import List
from alibabacloud_dysmsapi20170525.client import Client as Dysmsapi20170525Client
from alibabacloud_tea_openapi import models as open_api_models
from alibabacloud_dysmsapi20170525 import models as dysmsapi_20170525_models
class Sample:
    def __init__(self):
        pass
    @staticmethod
    def create_client(
        access_key_id: str,
        access_key_secret: str,
    ) -> Dysmsapi20170525Client:
        """
        使用AK&SK初始化账号Client
        @param access_key_id:
        @param access_key_secret:
        @return: Client
        @throws Exception
        """
        config = open_api_models.Config(
            # 您的AccessKey ID,
            access_key_id=access_key_id,
            # 您的AccessKey Secret,
            access_key_secret=access_key_secret
        )
        # 访问的域名
        config.endpoint = 'dysmsapi.aliyuncs.com'
        return Dysmsapi20170525Client(config)
    @staticmethod
    def main(
        args: List[str],
    ) -> None:
        client = Sample.create_client('accessKeyId', 'accessKeySecret')
        send_sms_request = dysmsapi_20170525_models.SendSmsRequest(
            phone_numbers='1503871****',
            sign_name='阿里大于测试专用',
            template_code='SMS_209335004',
            template_param='{"code":"1111"}'
        )
        # 复制代码运行请自行打印 API 的返回值
        client.send_sms(send_sms_request)
```

```
@staticmethod
async def main_async(
    args: List[str],
) -> None:
    client = Sample.create_client('accessKeyId', 'accessKeySecret')
    send_sms_request = dysmsapi_20170525_models.SendSmsRequest(
        phone_numbers='15038712376',
        sign_name='阿里大于测试专用',
        template_code='SMS_209335004',
        template_param='{"code":"1111"}'
    )
    # 复制代码运行请自行打印 API 的返回值
    await client.send_sms_async(send_sms_request)
if __name__ == '__main__':
    Sample.main(sys.argv[1:])
```

- 结果示例

```
{ "RequestId": "614048FB-0619-4439-A1D5-AA8B218A****", "Message": "OK", "BizId": "386715418801811068^0", "Code": "OK"}
```

相关文档

- 如果您想了解更多示例，请参见[OpenAPI开发者门户](#)。

 说明 登录开发者门户平台，SDK版本选择时，请选择升级版 SDK。

- 短信服务相关API，请参见[API概览](#)。

3.版本：V2

3.1. 短信服务SDK简介

短信服务新版SDK源码已经托管至开源平台Git hub及主流依赖仓库，推荐使用各语言主流的依赖管理工具安装，或通过Git Hub clone的方式使用SDK。所有SDK均只依赖SDK核心库，您可以直接使用SDK核心库，使用CommonRequest方式进行调用，Demo代码可通过OpenAPI开发者门户生成。

目前，短信服务提供以下编程语言的SDK。

说明

- 通过Java SDK拉取消息回执，请下载[Java MNS SDK安装包](#)。
- 通过Node.js SDK拉取消息回执，请下载[Node.js MNS SDK安装包](#)。
- 为优化短信服务底层性能，减少队列调用对您业务的压力，超过3个月无发送量时，MNS消息队列将进入休眠状态，重新发送短信后，MNS队列将自动取消休眠。

SDK	安装说明	DEMO
Java SDK	Java SDK	Java SDK DEMO
.NET SDK	.Net SDK	.NET SDK DEMO
PHP SDK	PHP SDK	PHP SDK DEMO
Python SDK	Python SDK	Python SDK DEMO
Node.js SDK	Node.js SDK	Node.js SDK DEMO
Go SDK	Go SDK	Go SDK DEMO

短信服务SDK升级说明

2019年01月22日，短信服务发布了新版SDK，此日期后接入SDK的用户使用的均为新版SDK；2019年01月22日前接入的SDK的用户使用的是旧版SDK。相较于旧版SDK，新版SDK有以下优势：

- 融入阿里云OpenAPI技术体系，改善用户体验。通过新版短信服务SDK，用户可以使用阿里云资源如API Explorer、CloudShell等工具体验API。
- 阿里云统一维护，降低故障风险。新版短信服务SDK由阿里云SDK统一维护，同时支持自动更新及故障预警，包括Git hub通知和Deprecated提示等。

新版短信服务SDK提供了更稳定、完善的开发环境，建议您尽快[升级SDK](#)。

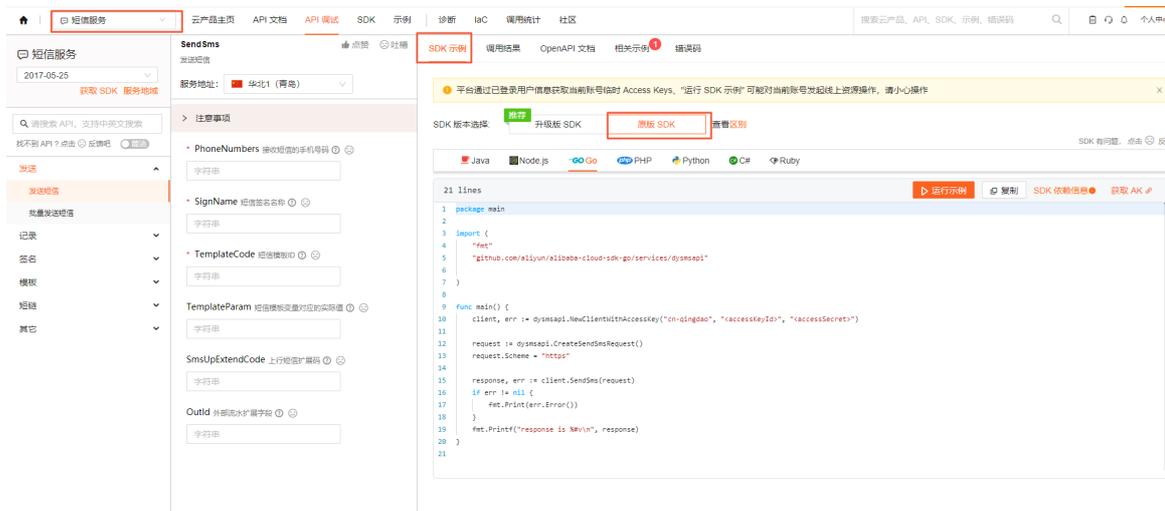
3.2. 使用示例（Demo）

本文为您介绍如何使用OpenAPI 开发者门户调用短信服务API。

查看Demo

- 使用阿里云账号登录[OpenAPI 开发者门户平台](#)。
- 在搜索栏中输入[短信服务](#)。

- 3. 单击选择需要查看Demo的API接口。
- 4. 在左侧参数列中填写参数值，右侧SDK示例页签中会自动同步生成对应SDK的Demo代码。



3.3. Java SDK

您可以通过直接添加Maven依赖或下载阿里云Java SDK的开发工具包的方式安装阿里云Java SDK。

前提条件

在安装和使用阿里云Java SDK前，确保您已经：

- 安装Java环境。阿里云Java SDK要求使用JDK1.6或更高版本。

在Java运行环境配置好的情况下，打开windows的命令行，执行java -version命令，可以检查版本信息。

```

C:\> 命令提示符
Microsoft Windows [版本 10.0.16299.431]
(c) 2017 Microsoft Corporation。保留所有权利。

C:\Users\> java -version
java version "1.8.0_152"
Java(TM) SE Runtime Environment (build 1.8.0_152-b16)
  
```

- 已经注册阿里云账号并生成访问密钥（AccessKey）。详细步骤，请参见AccessKey。

Java SDK安装方式

您可以通过以下两种方式安装Java SDK。

- 导入Maven依赖，适用于使用Maven管理的Java项目。
- 在集成开发环境中导入JAR文件，适用于使用Eclipse或IntelliJ作为集成开发环境的项目。

导入Maven依赖

通过在pom.xml文件中添加Maven依赖安装Java阿里云SDK。您可以在Maven库中查看各云产品的Maven依赖信息。

添加以下依赖安装阿里云Java SDK核心库。

```
<dependency>
  <groupId>com.aliyun</groupId>
  <artifactId>aliyun-java-sdk-core</artifactId>
  <version>4.5.16</version>
</dependency>
```

 **注意** 上述版本号为示例版本。最新版本号，请参见**最新版本号**，jar包可以在版本号中进行下载。

Home » com.aliyun » aliyun-java-sdk-core » 4.5.20

 **Aliyun Java SDK Core » 4.5.20**
 Aliyun Open API SDK for Java Copyright (C) Alibaba Cloud Computing All rights reserved. 版权所有 (C) 阿里云计算有限公司 <http://www.aliyun.com>

HomePage	http://www.aliyun.com
Date	(Mar 29, 2021)
Files	jar (190 KB) View All
Repositories	Central
Used By	531 artifacts

Maven [Gradle](#) [SBT](#) [Ivy](#) [Grape](#) [Leiningen](#) [Buildr](#)

```
<!-- https://mavenrepository.com/artifact/com.aliyun/aliyun-java-sdk-core -->
<dependency>
  <groupId>com.aliyun</groupId>
  <artifactId>aliyun-java-sdk-core</artifactId>
  <version>4.5.20</version>
</dependency>
```

Include comment with link to declaration

在集成开发环境中导入JAR文件

通过导入 [aliyun-java-sdk-core](#) JAR文件的方式安装阿里云Java SDK。

• Eclipse

使用Eclipse完成以下操作，在Eclipse的项目中安装阿里云Java SDK：

- i. 将下载的 *aliyun-java-sdk-core.jar* 文件复制到您的项目文件夹中。
- ii. 在Eclipse中打开您的项目，右键单击该项目，单击Properties。
- iii. 在弹出的对话框中，单击 **Java Build Path > Libraries > Add JARs** 添加下载的JAR文件。
- iv. 单击 **Apply and Close**。

• IntelliJ

使用IntelliJ完成以下操作，在IntelliJ的项目中安装阿里云Java SDK。

- i. 将下载的 *aliyun-java-sdk-core.jar* 文件复制到您的项目文件夹中。
- ii. 在IntelliJ中打开您的项目，在菜单栏中单击 **File > Project Structure**。
- iii. 单击 **Apply**，然后单击 **OK**。

发送短信

- 代码示例

```

import com.aliyuncs.CommonRequest;
import com.aliyuncs.CommonResponse;
import com.aliyuncs.DefaultAcsClient;
import com.aliyuncs.IAcsClient;
import com.aliyuncs.exceptions.ClientException;
import com.aliyuncs.exceptions.ServerException;
import com.aliyuncs.http.MethodType;
import com.aliyuncs.profile.DefaultProfile;
/*
pom.xml
<dependency>
  <groupId>com.aliyun</groupId>
  <artifactId>aliyun-java-sdk-core</artifactId>
  <version>请参见最新版本号</version>
</dependency>
*/
public class SendSms {
  public static void main(String[] args) {
    DefaultProfile profile = DefaultProfile.getProfile("ap-northeast-1", "<accessKeyId>", "<accessSecret>");
  };
  IAcsClient client = new DefaultAcsClient(profile);
  CommonRequest request = new CommonRequest();
  request.setSysMethod(MethodType.POST);
  request.setSysDomain("dysmsapi.ap-southeast-1.aliyuncs.com");
  request.setSysVersion("2017-05-25");
  request.setSysAction("SendSms");
  request.putQueryParameter("PhoneNumbers", "1503871****");
  request.putQueryParameter("SignName", "阿里大于测试专用");
  request.putQueryParameter("TemplateCode", "SMS_209335004");
  request.putQueryParameter("TemplateParam", "{\"code\":\"1111\"}");
  try {
    CommonResponse response = client.getCommonResponse(request);
    System.out.println(response.getData());
  } catch (ServerException e) {
    e.printStackTrace();
  } catch (ClientException e) {
    e.printStackTrace();
  }
}
}
}

```

- 结果示例

```
{ "RequestId": "614048FB-0619-4439-A1D5-AA8B218A****", "Message": "OK", "BizId": "386715418801811068^0", "Code": "OK"}
```

相关文档

- 如果您想了解更多示例，请参见[OpenAPI开发者门户](#)。

 说明 登录开发者门户平台，SDK版本选择时，请选择原版 SDK。

- 短信服务相关API，请参见[API概览](#)。

3.4. Go SDK

可以通过直接安装依赖包的方式安装阿里云Go SDK。

前提条件

在安装和使用阿里云SDK前，确保您已经注册阿里云账号并生成访问密钥（AccessKey）。详情请参见[AccessKey](#)。

安装Go SDK

阿里云Go SDK支持Go 1.7及以上版本，您可以通过以下方式安装Go SDK。

- 使用Glide安装GO SDK（推荐）

执行以下命令，安装阿里云Go SDK：

```
glide get github.com/aliyun/alibaba-cloud-sdk-go
```

- 使用Govendor安装

执行以下命令，安装阿里云Go SDK：

```
go get -u github.com/aliyun/alibaba-cloud-sdk-go/sdk
```

发送短信

- 代码示例

```
package main
import (
    "fmt"
    "github.com/aliyun/alibaba-cloud-sdk-go/services/dysmsapi"
)
func main() {
    client, err := dysmsapi.NewClientWithAccessKey("ap-northeast-1", "<accessKeyId>", "<accessSecret>")
    request := dysmsapi.CreateSendSmsRequest()
    request.Scheme = "https"
    request.PhoneNumbers = "1503871****"
    request.SignName = "阿里大于测试专用"
    request.TemplateCode = "SMS_209335004"
    request.TemplateParam = "{\"code\": \"1111\"}"
    response, err := client.SendSms(request)
    if err != nil {
        fmt.Print(err.Error())
    }
    fmt.Printf("response is %#v\n", response)
}
```

- 结果示例

```
{ "RequestId": "614048FB-0619-4439-A1D5-AA8B218A****", "Message": "OK", "BizId": "386715418801811068^0", "Code": "OK"}
```

相关文档

- 如果您想了解更多示例，请参见[OpenAPI开发者门户](#)。

 说明 登录开发者门户平台，SDK版本选择时，请选择原版 SDK。

- 短信服务相关API，请参见[API概览](#)。

3.5. Python SDK

您可以通过直接添加依赖包或下载阿里云SDK的方式安装阿里云Python SDK。

前提条件

在安装和使用阿里云SDK前，确保您已经注册阿里云账号并生成访问密钥（AccessKey）。请参见[创建AccessKey](#)。

安装Python SDK核心库

- 使用依赖包工具安装（推荐）
 - 如果您使用Python 2.x，执行以下命令，安装阿里云SDK核心库：

```
pip install aliyun-python-sdk-core
```

- 如果您使用Python 3.x，执行以下命令，安装阿里云SDK核心库：

```
pip install aliyun-python-sdk-core-v3
```

- 自行下载安装

您可以使用 `git clone` 或其它手段下载aliyun-python-sdk-core并自行添加解决方案。下载安装包，请参见[SDK 核心库](#)。

发送短信

- 代码示例

```
#!/usr/bin/env python
#coding=utf-8
from aliyunsdkcore.client import AcsClient
from aliyunsdkcore.request import CommonRequest
client = AcsClient('<accessKeyId>', '<accessSecret>', 'ap-northeast-1')
request = CommonRequest()
request.set_accept_format('json')
request.set_domain('dysmsapi.aliyuncs.com')
request.set_method('POST')
request.set_protocol_type('https') # https | http
request.set_version('2017-05-25')
request.set_action_name('SendSms')
request.add_query_param('PhoneNumbers', "1503871****")
request.add_query_param('SignName', "阿里大于测试专用")
request.add_query_param('TemplateCode', "SMS_209335004")
request.add_query_param('TemplateParam', "{\"code\":\"1111\"}")
response = client.do_action(request)
# python2: print(response)
print(str(response, encoding = 'utf-8'))
```

- 结果示例

```
{ "RequestId": "614048FB-0619-4439-A1D5-AA8B218A****", "Message": "OK", "BizId": "386715418801811068^0", "Code": "OK"}
```

相关文档

- 如果您想了解更多示例，请参见[OpenAPI开发者门户](#)。

 说明 登录开发者门户平台，SDK版本选择时，请选择原版 SDK。

- 短信服务相关API，请参见[API概览](#)。

3.6. .Net SDK

您可以通过直接添加NuGet程序包依赖或下载阿里云.NET SDK开发工具包的方式安装阿里云.NET SDK。

前提条件

已经注册阿里云账号并生成访问密钥（AccessKey）。详细步骤，请参见[AccessKey](#)。

安装.NET SDK核心库

- 使用依赖包工具安装（推荐）

您可以通过NuGet程序包管理器来安装，在解决方案资源管理器面板中，右键单击您的项目选择管理NuGet程序包菜单，在打开的NuGet管理面板中单击浏览选项卡输入aliyun-net-sdk-core，选择并单击安装即可。

- 通过.NET CLI工具来安装核心库

```
dotnet add package aliyun-net-sdk-core
```

- 自行下载安装

您可以使用 `git clone` 或其它手段下载aliyun-net-sdk-core并自行添加解决方案。下载安装包，请参见[SDK 核心库](#)。

发送短信

- 代码示例

```
using System;
using System.Collections.Generic;
using Aliyun.Acs.Core;
using Aliyun.Acs.Core.Profile;
using Aliyun.Acs.Core.Exceptions;
using Aliyun.Acs.Core.Http;
namespace CommonRequestDemo
{
    class Program
    {
        static void Main(string[] args)
        {
            IClientProfile profile = DefaultProfile.GetProfile("ap-northeast-1", "<accessKeyId>", "<accessSecret>");
            DefaultAcsClient client = new DefaultAcsClient(profile);
            CommonRequest request = new CommonRequest();
            request.Method = MethodType.POST;
            request.Domain = "dysmsapi.aliyuncs.com";
            request.Version = "2017-05-25";
            request.Action = "SendSms";
            // request.Protocol = ProtocolType.HTTP;
            request.AddQueryParameters("PhoneNumbers", "1503871****");
            request.AddQueryParameters("SignName", "阿里大于测试专用");
            request.AddQueryParameters("TemplateCode", "SMS_209335004");
            request.AddQueryParameters("TemplateParam", "{\"code\":\"1111\"}");
            try {
                CommonResponse response = client.GetCommonResponse(request);
                Console.WriteLine(System.Text.Encoding.Default.GetString(response.HttpResponse.Content));
            }
            catch (ServerException e)
            {
                Console.WriteLine(e);
            }
            catch (ClientException e)
            {
                Console.WriteLine(e);
            }
        }
    }
}
```

- 结果示例

```
{ "RequestId": "614048FB-0619-4439-A1D5-AA8B218A****", "Message": "OK", "BizId": "386715418801811068^0", "Code": "OK"}
```

相关文档

- 如果您想了解更多示例，请参见[OpenAPI开发者门户](#)。

 说明 登录开发者门户平台，SDK版本选择时，请选择**原版 SDK**。

- 短信服务相关API，请参见[API概览](#)。

3.7. Node.js SDK

您可以通过直接添加阿里云SDK依赖包的方式安装阿里云Node.js SDK。

前提条件

在安装和使用阿里云Node.js SDK前，确保您已经：

- 安装Node.js环境。请确认Node.js版本为8.0及以上。
- 已经注册阿里云账号并生成访问密钥（AccessKey）。详细步骤，请参见[AccessKey](#)。

安装Node.js SDK核心库

执行以下命令，安装Node.js SDK核心库。

```
npm install @alicloud/pop-core -S
```

您也可以使用cnpm、yarn等包管理工具来安装@alicloud/pop-core。

安装步骤

请参考以下步骤，使用依赖包工具安装Node.js SDK。

执行以下命令，安装阿里云SDK核心库。

```
npm install @alicloud/pop-core -S
```

您也可以使用cnpm、yarn等包管理工具来安装@alicloud/pop-core。

Git Hub地址：[Git Hub仓库说明](#)。

npm地址：[npm地址](#)。

发送短信

- 代码示例

```
const Core = require('@alicloud/pop-core');
var client = new Core({
  accessKeyId: '<accessKeyId>',
  accessKeySecret: '<accessSecret>',
  endpoint: 'https://dysmsapi.aliyuncs.com',
  apiVersion: '2017-05-25'
});
var params = {
  "PhoneNumbers": "1503871****",
  "SignName": "阿里大于测试专用",
  "TemplateCode": "SMS_209335004",
  "TemplateParam": "{\"code\":\"1111\"}"
}
var requestOptions = {
  method: 'POST'
};
client.request('SendSms', params, requestOptions).then((result) => {
  console.log(JSON.stringify(result));
}, (ex) => {
  console.log(ex);
})
```

- 结果示例

```
{ "RequestId": "614048FB-0619-4439-A1D5-AA8B218A****", "Message": "OK", "BizId": "386715418801811068^0", "Code": "OK"}
```

相关文档

- 如果您想了解更多示例，请参见[OpenAPI开发者门户](#)。

 说明 登录开发者门户平台，SDK版本选择时，请选择原版 SDK。

- 短信服务相关API，请参见[API概览](#)。

3.8. PHP SDK

您可以通过直接添加阿里云SDK依赖包的方式安装阿里云PHP SDK。

前提条件

在安装和使用阿里云PHP SDK前，确保您已经：

- 注册阿里云账号并生成访问密钥（AccessKey）。详细步骤请参见[AccessKey](#)。
- 安装PHP环境。阿里云PHP SDK适用于PHP 5.5.0或更高版本。您可以通过 `php -v` 命令查看当前使用的PHP版本号。
- 安装cURL，并使用TLS后端编译cURL 7.16.2+。
- 已经安装[PHP SDK核心库](#)。

安装PHP SDK核心库

请参考以下步骤，使用Composer安装依赖。

如果在您的系统上全局安装Composer，您可以在项目目录中运行以下内容，将 Alibaba Cloud Client for PHP 添加为依赖项。

```
composer require alibabacloud/client
```

通过Composer和其他方式安装的操作，请参见[安装说明](#)。

发送短信

- 代码示例

```
<?php
use AlibabaCloud\Client\AlibabaCloud;
use AlibabaCloud\Client\Exception\ClientException;
use AlibabaCloud\Client\Exception\ServerException;
// Download: https://github.com/aliyun/openapi-sdk-php
// Usage: https://github.com/aliyun/openapi-sdk-php/blob/master/README.md
AlibabaCloud::accessKeyClient('<accessKeyId>', '<accessSecret>')
    ->regionId('ap-northeast-1')
    ->asDefaultClient();

try {
    $result = AlibabaCloud::rpc()
        ->product('Dysmsapi')
        // ->scheme('https') // https | http
        ->version('2017-05-25')
        ->action('SendSms')
        ->method('POST')
        ->host('dysmsapi.aliyuncs.com')
        ->options([
            'query' => [
                'PhoneNumbers' => "1503871****",
                'SignName' => "阿里大于测试专用",
                'TemplateCode' => "SMS_209335004",
                'TemplateParam' => "{\"code\":\"1111\"}",
            ],
        ])
        ->request();
    print_r($result->toArray());
} catch (ClientException $e) {
    echo $e->getErrorMessage() . PHP_EOL;
} catch (ServerException $e) {
    echo $e->getErrorMessage() . PHP_EOL;
}
```

- 结果示例

```
{ "RequestId": "614048FB-0619-4439-A1D5-AA8B218A****", "Message": "OK", "BizId": "386715418801811068^0", "Code": "OK"}
```

相关文档

- 如果您想了解更多示例，请参见[OpenAPI开发者门户](#)。

 说明 登录开发者门户平台，SDK版本选择时，请选择原版 SDK。

- 短信服务相关API，请参见[API概览](#)。

3.9. 升级SDK

新SDK源码已经托管至开源平台Git hub，可使用Git Hub clone的方式使用SDK，也可以使用依赖管理工具安装，Demo代码可通过OpenAPI Explorer生成，所有SDK均只依赖SDK核心库，使用通用的Request及Response来处理接口请求及响应。

JAVA SDK

如何安装JAVA SDK: [Java SDK](#)。

新版JAVA SDK仅依赖阿里云JAVA SDK核心库，可使用maven安装依赖，使用CommonRequest、CommonResponse来处理接口请求及响应。请使用CommonRequest实例的put QueryParameter方法设置API参数，CommonResponse实例使用get Data方法获取API结果的JSON字符串。

 说明 通过JAVA SDK拉取消息回执，请使用[JAVA MNS SDK](#)。

Node.js SDK

如何安装Node.js SDK: [Node.js SDK](#)。

详情: [@alicloud/pop-core](#)。新Node.js SDK仅依赖@alicloud/pop-core。

 说明 通过Node.js SDK拉取消息回执，请使用[原Node.js MNS SDK](#)。

.NET SDK

如何安装.NET SDK: [.Net SDK](#)。

新版.NET SDK仅依赖阿里云.NET SDK核心库，可使用NuGet安装依赖，使用CommonRequest、CommonResponse来处理接口请求及响应。请使用CommonRequest实例的put QueryParameter方法设置API参数，CommonResponse实例使用get Data方法获取API结果的JSON字符串。

PHP SDK

如何安装PHP SDK: [PHP SDK](#)。

PHP全新SDK上线，并支持composer安装，请按说明重新设置请求参数。

Python SDK

如何安装Python SDK: [Python SDK](#)。

python-sdk-core已统一支持Python 2.0/3.0，与原有SDK兼容，建议参考OpenAPI Explorer提供的Demo重新设置请求参数。

Go SDK

如何安装Go SDK: [Go SDK](#)。

原SDK不支持Golang，使用新SDK请直接参考新Demo即可。

3.10. 升级消息回执API

3.10.1. 升级说明

升级前，请注意以下事项：

已接入SDK的老用户或Java、Node.js用户无需升级消息回执API，新用户的消息回执接入请参考现有Demo。

3.10.2. .NET Demo

依赖.NET语言的阿里云SDK核心库及dybaseapi，其中dybaseapi包用于拉取MNS消息。

- [阿里云SDK核心库](#)
- [dybaseapi](#)

Demo如下：

```
using System;
using Aliyun.Acs.Core;
using Aliyun.Acs.Core.Profile;
using Aliyun.Acs.Core.Exceptions;
using Aliyun.Acs.Dybaseapi.Model.V20170525;
using Aliyun.Acs.Dybaseapi.MNS;
using Aliyun.Acs.Dybaseapi.MNS.Model;
using System.Threading;
using System.Collections.Generic;
using System.Text;
using QueryTokenForMnsQueue_MessageTokenDTO = Aliyun.Acs.Dybaseapi.Model.V20170525.QueryTokenForMnsQueueResponse.QueryTokenForMnsQueue_MessageTokenDTO;
namespace CommonRpc
{
    class Program
    {
        static void Main(string[] args)
        {
            IClientProfile profile = DefaultProfile.GetProfile("cn-hangzhou", "<AccessKeyId>", "<AccessKeySecret>"); // todo: 补充AK信息
            DefaultProfile.AddEndpoint("cn-hangzhou", "cn-hangzhou", "Dybaseapi", "dybaseapi.aliyuncs.com");
            DefaultAcsClient client = new DefaultAcsClient(profile);
            String queueName = "<QueueName>"; // todo: 补充队列名称
            String messageType = "<MessageType>"; // todo: 补充消息类型
            int maxThread = 2;
            for (int i = 0; i < maxThread; i++)
            {
                TestTask testTask = new TestTask("PullMessageTask-thread-" + i, messageType, queueName, client);
                Thread t = new Thread(new ThreadStart(testTask.Handle));
                //启动线程
                t.Start();
            }
            Console.ReadKey();
            try
            {
                QueryTokenForMnsQueueRequest request = new QueryTokenForMnsQueueRequest
```

```

        {
            MessageType = messageType,
            QueueName = queueName
        };
        QueryTokenForMnsQueueResponse response = client.GetAcResponse(request);
        Console.WriteLine(response.MessageTokenDTO.SecurityToken);
    }
    catch (ServerException ex)
    {
        Console.WriteLine(ex.ToString());
    }
    catch (ClientException ex)
    {
        Console.WriteLine(ex.ToString());
    }
}
}
class TestTask
{
    object o = new object();
    const int sleepTime = 50;
    const long bufferTime = 60 * 2; // 过期时间小于2分钟则重新获取, 防止服务器时间误差
    const String mnsAccountEndpoint = "https://1943695596114318.mns.cn-hangzhou.aliyuncs.com/"; // 阿里通信消息的endpoint, 固定
    public String name { get; private set; }
    public String messageType { get; private set; }
    public String QueueName { get; private set; }
    public int TaskID { get; private set; }
    public IAcsClient AcsClient { get; private set; }
    public TestTask(String name, String messageType, String queueName, IAcsClient acsClient)
    {
        this.name = name;
        this.messageType = messageType;
        this.QueueName = queueName;
        this.AcsClient = acsClient;
    }
    readonly Dictionary<string, QueryTokenForMnsQueue_MessageTokenDTO> tokenMap = new Dictionary<string, QueryTokenForMnsQueue_MessageTokenDTO>();
    readonly Dictionary<string, Queue> queueMap = new Dictionary<string, Queue>();
    public QueryTokenForMnsQueue_MessageTokenDTO GetTokenByMessageType(IAcsClient acsClient, String messageType)
    {
        QueryTokenForMnsQueueRequest request = new QueryTokenForMnsQueueRequest
        {
            MessageType = messageType
        };
        QueryTokenForMnsQueueResponse queryTokenForMnsQueueResponse = acsClient.GetAcResponse(request);
        QueryTokenForMnsQueue_MessageTokenDTO token = queryTokenForMnsQueueResponse.MessageTokenDTO;
        return token;
    }
    /// 处理消息
    public void Handle()
    {

```

```
    {
        while (true)
        {
            try
            {
                QueryTokenForMnsQueue_MessageTokenDTO token = null;
                Queue queue = null;
                lock (o)
                {
                    if (tokenMap.ContainsKey(messageType))
                    {
                        token = tokenMap[messageType];
                    }
                    if (queueMap.ContainsKey(QueueName))
                    {
                        queue = queueMap[QueueName];
                    }
                    TimeSpan ts = new TimeSpan(0);
                    if (token != null)
                    {
                        DateTime b = Convert.ToDateTime(token.ExpireTime);
                        DateTime c = Convert.ToDateTime(DateTime.Now);
                        ts = b - c;
                    }
                    if (token == null || ts.TotalSeconds < bufferTime || queue == null)
                    {
                        token = GetTokenByMessageType(AcsClient, messageType);
                        IMNS client = new MNSClient(token.AccessKeyId, token.AccessKeySecret, mnsAccountEndpoint,
token.SecurityToken);
                        queue = client.GetNativeQueue(QueueName);
                        if (tokenMap.ContainsKey(messageType))
                        {
                            tokenMap.Remove(messageType);
                        }
                        if (queueMap.ContainsKey(QueueName))
                        {
                            queueMap.Remove(QueueName);
                        }
                        tokenMap.Add(messageType, token);
                        queueMap.Add(QueueName, queue);
                    }
                }
                BatchReceiveMessageResponse batchReceiveMessageResponse = queue.BatchReceiveMessage(16);
                List<Message> messages = batchReceiveMessageResponse.Messages;
                for (int i = 0; i <= messages.Count - 1; i++)
                {
                    try
                    {
                        byte[] outputb = Convert.FromBase64String(messages[i].Body);
                        string orgStr = Encoding.UTF8.GetString(outputb);
                        Console.WriteLine(orgStr);
                        // TODO 具体消费逻辑,待客户自己实现.
                        // 消费成功的前提下删除消息
                        // queue.DeleteMessage(messages[i].ReceiptHandle);
                    }
                }
            }
        }
    }
}
```

```
    }
    catch (Exception e)
    {
        Console.WriteLine(e.ToString());
    }
}
}
catch (Exception e)
{
    Console.WriteLine(e.ToString());
}
Thread.Sleep(sleepTime);
}
}
}
}
```

3.10.3. PHP Demo

依赖OpenAPI PHP Client包和OpenAPI PHP SDK包。

- OpenAPI PHP Client
 - 下载地址: [openapi-sdk-php-client](#)
 - 安装说明 [OpenAPI PHP Client 安装说明](#)
- OpenAPI PHP SDK
 - 下载地址: [openapi-sdk-php](#)
 - 安装说明 [OpenAPI PHP SDK 安装说明](#)

Demo如下:

```
<?php
use AlibabaCloud\Client\AlibabaCloud;
use AlibabaCloud\Client\Exception\ClientException;
use AlibabaCloud\Client\Exception\ServerException;
use AlibabaCloud\Dybaseapi\MNS\Requests\BatchReceiveMessage;
use AlibabaCloud\Dybaseapi\MNS\Requests\BatchDeleteMessage;
AlibabaCloud::accessKeyClient('<AccessKeyId>', '<AccessSecret>')
    ->regionId('cn-hangzhou')
    ->asGlobalClient();
$queueName = '<QueueName>'; // 队列名称
$messageType = '<MessageType>'; // 需要接收的消息类型
$response = null;
$token = null;
$i = 0;
do {
    try {
        if (null == $token || strtotime($token['ExpireTime']) - time() > 2 * 60) {
            $response = AlibabaCloud::rpcRequest()
                ->product('Dybaseapi')
                ->version('2017-05-25')
                ->action('QueryTokenForMnsQueue')
```

```
->method('POST')
->host("dybaseapi.aliyuncs.com")
->options([
    'query' => [
        'MessageType' => $messageType,
        'QueueName' => $queueName,
    ],
])
->request()
->toArray();
}
$token = $response['MessageTokenDTO'];
$mnsClient = new \AlibabaCloud\Dybaseapi\MNS\MnsClient(
    "http://1943695596114318.mns.cn-hangzhou.aliyuncs.com",
    $token['AccessKeyId'],
    $token['AccessKeySecret'],
    $token['SecurityToken']
);
$mnsRequest = new BatchReceiveMessage(10, 5);
$mnsRequest->setQueueName($queueName);
$mnsResponse = $mnsClient->sendRequest($mnsRequest);
$receiptHandles = Array();
foreach ($mnsResponse->Message as $message) {
    // 用户逻辑:
    // $receiptHandles[] = $message->ReceiptHandle; // 加入$receiptHandles数组中的记录将会被删除
    $messageBody = base64_decode($message->MessageBody); // base64解码后的JSON字符串
    print_r($messageBody . "\n");
}
if (count($receiptHandles) > 0) {
    $deleteRequest = new BatchDeleteMessage($queueName, $receiptHandles);
    $mnsClient->sendRequest($deleteRequest);
}
} catch (ClientException $e) {
    echo $e->getErrorMessage() . PHP_EOL;
} catch (ServerException $e) {
    if ($e->getCode() == 404) {
        $i++;
    }
    echo $e->getErrorMessage() . PHP_EOL;
}
} while ($i < 3);
```

3.10.4. Python Demo

依赖Python语言的阿里云SDK核心库及dybaseapi，其中dybaseapi包用于拉取MNS消息。

- [阿里云SDK核心库（Python）](#)
- [dybaseapi（Python）](#)

Demo如下：

```
#!/usr/bin/env python
# coding=utf8
import time
```

```
from aliyunskcore.acs_exception.exceptions import ServerException
from aliyunskcore.client import AcsClient
from aliyunskdybaseapi.request.v20170525.QueryTokenForMnsQueueRequest import QueryTokenForMnsQueueRequest
from aliyunskcore.profile import region_provider
from datetime import datetime
from aliyunskdybaseapi.mns.account import Account
from aliyunskdybaseapi.mns.queue import *
from aliyunskdybaseapi.mns.mns_exception import *
try:
    import json
except ImportError:
    import simplejson as json
# TODO 需要替换成您需要接收的消息类型
message_type = "<MessageType>"
# TODO 需要替换成您的队列名称。在云通信页面开通相应业务消息后, 就能在页面上获得对应的queueName
queue_name = "<QueueName>"
# 云通信固定的endpoint地址
endpoint = "http://1943695596114318.mns.cn-hangzhou.aliyuncs.com"
acs_client = AcsClient("<AccessKeyId>", "<AccessKeySecret>", "cn-hangzhou")
region_provider.add_endpoint("Dybaseapi", "dybaseapi.aliyuncs.com", "cn-hangzhou")
# 云通信业务token存在失效时间, 需动态更新。
class Token():
    def __init__(self):
        self.token = None
        self.tmp_access_id = None
        self.tmp_access_key = None
        self.expire_time = None
    def is_refresh(self):
        if self.expire_time is None:
            return 1
        # 失效时间与当前系统时间比较, 提前2分钟刷新token
        now = datetime.now()
        expire = datetime.strptime(self.expire_time, "%Y-%m-%d %H:%M:%S")
        if expire <= now or (expire - now).seconds < 120:
            return 1
        return 0
    def refresh(self):
        print("start refresh token...")
        request = QueryTokenForMnsQueueRequest()
        request.set_MessageType(message_type)
        request.set_QueueName(queue_name)
        response = acs_client.do_action_with_exception(request)
        # print response
        if response is None:
            raise ServerException("GET_TOKEN_FAIL", "获取token时无响应")
        response_body = json.loads(response)
        if response_body.get("Code") != "OK":
            raise ServerException("GET_TOKEN_FAIL", "获取token失败")
        sts_token = response_body.get("MessageTokenDTO")
        self.tmp_access_key = sts_token.get("AccessKeySecret")
        self.tmp_access_id = sts_token.get("AccessKeyId")
        self.expire_time = sts_token.get("ExpireTime")
        self.token = sts_token.get("SecurityToken")
```

```
print("finish refresh token...")
# 初始化 token, my_account, my_queue
token, my_account, my_queue = Token(), None, None
# 循环读取删除消息直到队列空
# receive message请求使用long polling方式，通过wait_seconds指定长轮询时间为3秒
## long polling 解析:
### 当队列中有消息时，请求立即返回；
### 当队列中没有消息时，请求在MNS服务器端挂3秒钟，在这期间，有消息写入队列，请求会立即返回消息，3秒后，
请求返回队列没有消息；
wait_seconds = 3
print("%sReceive And Delete Message From Queue%s\nQueueName:%s\nWaitSeconds:%s\n" % (
    10 * "=", 10 * "=", queue_name, wait_seconds))
while True:
    receipt_handles = []
    # 读取消息
    try:
        # token过期是否需要刷新
        if token.is_refresh() == 1:
            # 刷新token
            token.refresh()
            if my_account:
                my_account.mns_client.close_connection()
                my_account = None
            if not my_account:
                my_account = Account(endpoint, token.tmp_access_id, token.tmp_access_key, token.token)
                my_queue = my_account.get_queue(queue_name)
        # 接收消息
        rcv_msgs = my_queue.batch_receive_message(10, wait_seconds)
        for rcv_msg in rcv_msgs:
            # TODO 业务处理
            # receipt_handles.append(rcv_msg.receipt_handle)
            print("Receive Message Succeed! ReceiptHandle:%s MessageBody:%s MessageID:%s" % (
                rcv_msg.receipt_handle, rcv_msg.message_body, rcv_msg.message_id))
    except MNSExceptionBase as e:
        if e.type == "QueueNotExist":
            print("Queue not exist, please create queue before receive message.")
            break
        elif e.type == "MessageNotExist":
            print("Queue is empty! sleep 10s")
            time.sleep(10)
            continue
        print("Receive Message Fail! Exception:%s\n" % e)
        break
    # 删除消息
    try:
        if len(receipt_handles) > 0:
            # my_queue.delete_message(receipt_handles)
            print("Delete Message Succeed! ReceiptHandles:%s" % receipt_handles)
    except MNSExceptionBase as e:
        print("Delete Message Fail! Exception:%s\n" % e)
```

3.10.5. Go Demo

通过本文您可以了解短信服务Go语言的消息回执示例，同时下载相应的SDK安装包进行测试。

单击下载[Go SDK安装包](#)。

Demo如下：

```
package main
import (
    "encoding/base64"
    "fmt"
    "github.com/aliyun/alibaba-cloud-sdk-go/sdk/endpoints"
    "github.com/aliyun/alibaba-cloud-sdk-go/services/dybaseapi"
    "github.com/aliyun/alibaba-cloud-sdk-go/services/dybaseapi/mns"
    "time"
)
const (
    mnsDomain = "1943695596114318.mns.cn-hangzhou.aliyuncs.com"
)
func main(){
    endpoints.AddEndpointMapping("cn-hangzhou", "Dybaseapi", "dybaseapi.aliyuncs.com")
    // 创建client实例
    client, err := dybaseapi.NewClientWithAccessKey(
        "cn-hangzhou", // 您的可用区ID
        "<AccessKeyId>", // 您的Access Key ID
        "<AccessKeySecret>") // 您的Access Key Secret
    if err != nil {
        // 异常处理
        panic(err)
    }
    queueName := "<QueueName>"
    messageType := "<MessageType>"
    var token *dybaseapi.MessageTokenDTO
    for {
        if token == nil || expireTime.Unix()-time.Now().Unix() < 2*60 {
            // 创建 API 请求并设置参数
            request := dybaseapi.CreateQueryTokenForMnsQueueRequest()
            request.MessageType = messageType
            request.QueueName = queueName
            // 发起请求并处理异常
            response, err := client.QueryTokenForMnsQueue(request)
            if err != nil {
                panic(err)
            }
            token = &response.MessageTokenDTO
        }
        expireTime, err = time.ParseInLocation("2006-01-02 15:04:05", token.ExpireTime, time.Local)
        if err != nil {
            panic(err)
        }
    }
    mnsClient, err := mns.NewClientWithStsToken(
        "cn-hangzhou",
        token.AccessKeyId,
        token.AccessKeySecret,
        token.SecurityToken,
    )
}
```

```
if err != nil {
    panic(err)
}
mnsRequest := mns.CreateBatchReceiveMessageRequest()
mnsRequest.Domain = mnsDomain
mnsRequest.QueueName = queueName
mnsRequest.NumOfMessages = "10"
mnsRequest.WaitSeconds = "5"
mnsResponse, err := mnsClient.BatchReceiveMessage(mnsRequest)
if err != nil {
    panic(err)
}
// fmt.Println(mnsResponse)
receiptHandles := make([]string, len(mnsResponse.Message))
for i, message := range mnsResponse.Message {
    messageBody, decodeErr := base64.StdEncoding.DecodeString(message.MessageBody)
    if decodeErr != nil {
        panic(decodeErr)
    }
    fmt.Println(string(messageBody))
    receiptHandles[i] = message.ReceiptHandle
}
if len(receiptHandles) > 0 {
    mnsDeleteRequest := mns.CreateBatchDeleteMessageRequest()
    mnsDeleteRequest.Domain = mnsDomain
    mnsDeleteRequest.QueueName = queueName
    mnsDeleteRequest.SetReceiptHandles(receiptHandles)
    //_, err = mnsClient.BatchDeleteMessage(mnsDeleteRequest) // 取消注释将删除队列中的消息
    if err != nil {
        panic(err)
    }
}
}
```

3.11. 调用示例

3.11.1. C# Demo

本文为您介绍C#调用SendMessageToGlobe发送短信到国际地区的程序样例。

注意事项

参考这个示例您需要注意以下几点：

- 初始化IClientProfile时，第一个参数（RegionId）必须为cn-hangzhou，请勿修改。
- Domain必须为dysmsapi.aliyuncs.com，请勿修改。
- Version必须为2017-05-25，请勿修改。

编写样例程序

```
using System;
using Aliyun.Acs.Core;
using Aliyun.Acs.Core.Profile;
using Aliyun.Acs.Core.Exceptions;
using Aliyun.Acs.Core.Http;
namespace CommonRpc
{
    class Program
    {
        static void Main(string[] args)
        {
            IClientProfile profile = DefaultProfile.GetProfile("cn-hangzhou", "<AccessKeyId>", "<AccessSecret>");
            DefaultAcsClient client = new DefaultAcsClient(profile);
            CommonRequest request = new CommonRequest
            {
                Method = MethodType.POST,
                Domain = "dysmsapi.aliyuncs.com",
                Version = "2017-05-25",
                Action = "SendMessageToGlobe"
            };
            // request.Protocol = ProtocolType.HTTPS;
            request.AddQueryParameters("To", "62123****8901");
            request.AddQueryParameters("PhoneNumber", "156****0987");
            // request.AddQueryParameters("From", "1234567890");
            request.AddQueryParameters("Message", "have a test.");
            // request.AddQueryParameters("Type", "OTP");
            try
            {
                CommonResponse response = client.GetCommonResponse(request);
                Console.WriteLine(response.Data);
            }
            catch (ServerException e)
            {
                Console.WriteLine(e);
            }
            catch (ClientException e)
            {
                Console.WriteLine(e);
            }
        }
    }
}
```

3.11.2. Node.js Demo

本文为您介绍Node.js调用SendMessageToGlobe发送短信到国际地区的程序样例。

注意事项

参考这个示例您需要注意以下几点：

- RegionId必须为cn-hangzhou，请勿修改。

- endpoint必须为“https://dysmsapi.aliyuncs.com”，请勿修改。
- apiVersion必须为2017-05-25，请勿修改。

编写样例程序

```
const Core = require('@alicloud/pop-core');
var client = new Core({
  accessKeyId: '<accessKeyId>',
  accessKeySecret: '<accessSecret>',
  endpoint: 'https://dysmsapi.aliyuncs.com',
  apiVersion: '2017-05-25'
});
var params = {
  "RegionId": "cn-hangzhou",
  "To": "62123****8901",
  "From": "1234",
  "Message": "Have a test.",
  "Type": "OTP"
}
var requestOptions = {
  method: 'POST'
};
client.request('SendMessageToGlobe', params, requestOptions).then((result) => {
  console.log(result);
}, (ex) => {
  console.log(ex);
})
```

3.11.3. Python Demo

本文为您介绍Python调用SendMessageToGlobe发送短信到国际地区的程序样例。

注意事项

参考这个示例您需要注意以下几点：

- 构造AcsClient时，第三个参数（regionId）必须为cn-hangzhou，请勿修改。
- domain必须为dysmsapi.aliyuncs.com，请勿修改。
- version必须为2017-05-25，请勿修改。

编写样例程序

```
#!/usr/bin/env python
#coding=utf-8
import os
from aliyunsdkcore.client import AcsClient
from aliyunsdkcore.request import CommonRequest
client = AcsClient('<accessKeyId>', '<accessSecret>', 'cn-hangzhou')
request = CommonRequest()
request.set_accept_format('json')
request.set_domain('dysmsapi.aliyuncs.com')
request.set_method('POST')
request.set_version('2017-05-25')
request.set_action_name('SendMessageToGlobe')
request.add_query_param('To', '6281****07451')
request.add_query_param('From', '1234')
request.add_query_param('Message', 'Have a test.')
request.add_query_param('Type', 'OTP')
response = client.do_action(request)
# python2: print(response)
print(str(response, encoding = 'utf-8'))
```

3.11.4. PHP Demo

本文为您介绍PHP调用SendMessageToGlobe发送短信到国际地区的程序样例。

注意事项

参考这个示例您需要注意以下几点：

- regionId必须为cn-hangzhou，请勿修改。
- product必须为Dysmsapi，请勿修改。
- host必须为dysmsapi.aliyuncs.com，请勿修改。
- version必须为2017-05-25，请勿修改。

编写样例程序

```
<?php
use AlibabaCloud\Client\AlibabaCloud;
use AlibabaCloud\Client\Exception\ClientException;
use AlibabaCloud\Client\Exception\ServerException;
// Download: https://github.com/aliyun/openapi-sdk-php-client
// Usage: https://github.com/aliyun/openapi-sdk-php-client/blob/master/README-CN.md
AlibabaCloud::accessKeyClient('<accessKeyId>', '<accessSecret>')
->regionId('cn-hangzhou')
->asGlobalClient();
try {
$result = AlibabaCloud::rpcRequest()
->product('Dysmsapi')
->host('dysmsapi.aliyuncs.com')
->version('2017-05-25')
->action('SendMessageToGlobe')
->method('POST')
->options([
'query' => [
'To' => "62123****8901",
// "From" => "1234567890",
'Message' => "have a test.",
// "Type" => "OTP",
],
])
->request();
print_r($result->toArray());
} catch (ClientException $e) {
echo $e->getErrorMessage() . PHP_EOL;
} catch (ServerException $e) {
echo $e->getErrorMessage() . PHP_EOL;
}
```

3.11.5. JAVA Demo

本文为您介绍JAVA调用SendMessageToGlobe发送短信到国际地区的程序样例。

注意事项

参考这个示例您需要注意以下几点：

- 构造DefaultProfile时，第一个参数（regionId）必须为cn-hangzhou，请勿修改。
- domain必须为dysmsapi.aliyuncs.com，请勿修改。
- version必须为2017-05-25，请勿修改。

 说明 Demo中的maven dependency版本号为示例版本。最新版本号，请参见[最新版本号](#)。

编写样例程序

```
package com.alicom.dysms.api;
import com.aliyuncs.CommonRequest;
import com.aliyuncs.CommonResponse;
import com.aliyuncs.DefaultAcsClient;
import com.aliyuncs.IAcsClient;
import com.aliyuncs.exceptions.ClientException;
import com.aliyuncs.exceptions.ServerException;
import com.aliyuncs.http.MethodType;
import com.aliyuncs.profile.DefaultProfile;
/*
pom.xml
<dependency>
  <groupId>com.aliyun</groupId>
  <artifactId>aliyun-java-sdk-core</artifactId>
  <version>4.5.17</version>
</dependency>
*/
public class CommonRpc {
  public static void main(String[] args) {
    //初始化acsClient, <accessKeyId>和<accessSecret>"在短信控制台查询即可
    DefaultProfile profile = DefaultProfile.getProfile("cn-hangzhou", "<accessKeyId>", "<accessSecret>");
    IAcsClient client = new DefaultAcsClient(profile);
    CommonRequest request = new CommonRequest();
    request.setSysMethod(MethodType.POST);
    //域名, 请勿修改
    request.setSysDomain("dysmsapi.aliyuncs.com");
    //API版本号, 请勿修改
    request.setSysVersion("2017-05-25");
    //API名称
    request.setSysAction("SendMessageToGlobe");
    //接收号码, 格式为: 国际码+号码, 必填
    request.putQueryParameter("To", "62123****8901");
    //发送方senderId, 选填
    //request.putQueryParameter("From", "1234****90");
    //短信内容, 必填
    request.putQueryParameter("Message", "have a test.");
    try {
      CommonResponse response = client.getCommonResponse(request);
      System.out.println(response.getData());
    } catch (ServerException e) {
      e.printStackTrace();
    } catch (ClientException e) {
      e.printStackTrace();
    }
  }
}
```

4.版本：V1

4.1. Java SDK

本文为您介绍短信服务Java SDK的使用方法及其示例。

准备工作

注意 本版本仅提供Java相关示例，推荐您使用V2版本SDK完成短信功能对接。更多信息，请参见[短信服务SDK简介](#)。

- 在安装和使用阿里云SDK前，确保您已经注册阿里云账号并生成访问密钥（AccessKey）。详情请参见[创建AccessKey](#)
- 将SDK添加到项目中。

引入短信服务SDK依赖，通过Maven二方库依赖的方式将短信服务的SDK加入到自己的项目中。

```
<dependency>
<groupId>com.aliyun</groupId>
<artifactId>aliyun-java-sdk-core</artifactId>
<version>4.5.16</version>
</dependency>
<dependency>
<groupId>com.aliyun</groupId>
<artifactId>aliyun-java-sdk-dysmsapi</artifactId>
<version>2.1.0</version>
</dependency>
```

说明 上述版本号为示例版本。最新版本号，请参见[最新版本号](#)，jar包可以在版本号中进行下载。

Home » com.aliyun » aliyun-java-sdk-core » 4.5.20

Aliyun Java SDK Core » 4.5.20

Aliyun Open API SDK for Java Copyright (C) Alibaba Cloud Computing All rights reserved. 版权所有 (C) 阿里云计算有限公司 http://www.aliyun.com

HomePage	http://www.aliyun.com
Date	(Mar 29, 2021)
Files	jar (190 KB) View All
Repositories	Central
Used By	531 artifacts

Maven | Gradle | SBT | Ivy | Grape | Leiningen | Buildr

```
<!-- https://mavenrepository.com/artifact/com.aliyun/aliyun-java-sdk-core -->
<dependency>
  <groupId>com.aliyun</groupId>
  <artifactId>aliyun-java-sdk-core</artifactId>
  <version>4.5.20</version>
</dependency>
```

Include comment with link to declaration

发送短信

- 代码示例

```
import com.aliyuncs.dysmsapi.model.v20170525.SendSmsRequest;
import com.aliyuncs.dysmsapi.model.v20170525.SendSmsResponse;
public class SendSms {
    public static void main(String[] args) {
        DefaultProfile profile = DefaultProfile.getProfile("cn-hangzhou", "<accessKeyId>", "<accessSecret>");
        IAcsClient client = new DefaultAcsClient(profile);
        SendSmsRequest request = new SendSmsRequest();
        request.setSysRegionId("cn-hangzhou");
        request.setPhoneNumbers("159512****");
        request.setSignName("云通讯****");
        request.setTemplateCode("SMS_1004****");
        request.setTemplateParam("{\"code\": \"123456\"}");
        try {
            SendSmsResponse response = client.getAcsResponse(request);
        } catch (ClientException e) {
        }
    }
}
```

- 结果示例

```
{ "RequestId": "614048FB-0619-4439-A1D5-AA8B218A****", "Message": "OK", "BizId": "386715418801811068^0", "Code": "OK"}
```

相关文档

短信服务相关API，请参见[API概览](#)。