

消息&短信

SDK 参考

文档版本 01
发布日期 2025-02-13



版权所有 © 华为云计算技术有限公司 2025。保留一切权利。

非经本公司书面许可，任何单位和个人不得擅自摘抄、复制本文档内容的部分或全部，并不得以任何形式传播。

商标声明



HUAWEI和其他华为商标均为华为技术有限公司的商标。

本文档提及的其他所有商标或注册商标，由各自的所有人拥有。

注意

您购买的产品、服务或特性等应受华为云计算技术有限公司商业合同和条款的约束，本文档中描述的全部或部分产品、服务或特性可能不在您的购买或使用范围之内。除非合同另有约定，华为云计算技术有限公司对本文档内容不做任何明示或暗示的声明或保证。

由于产品版本升级或其他原因，本文档内容会不定期进行更新。除非另有约定，本文档仅作为使用指导，本文档中的所有陈述、信息和建议不构成任何明示或暗示的担保。

目录

1 SDK 概述	1
2 应用、签名、模板	3
2.1 Java SDK 使用指导.....	3
2.2 Python SDK 使用指导.....	5
2.3 Go SDK 使用指导.....	7
2.4 Node.js SDK 使用指南.....	8
2.5 .NET SDK 使用指导.....	10
3 发送接收短信	13
3.1 Java SDK 使用指导.....	13

1 SDK 概述

消息&短信（Message&SMS，MSGSMS）服务提供应用侧SDK，您可以直接集成SDK来调用MSGSMS的应用、签名、模板、发送短信相关API，从而实现对MSGSMS服务的快速操作。目前MSGSMS提供如表1-1所示语言SDK供开发者使用。

在线生成 SDK 代码

推荐您使用API在线调试工具[API Explorer](#)，API Explorer支持快速调试和检索，调试API的同时，可以根据您的参数实时生成各种开发语言的SDK示例代码，方便您直接根据示例代码使用SDK。

说明

仅支持在线调测应用、签名、模板API。

SDK 列表

表1-1提供了MSGSMS支持的SDK，您可以在GitHub仓库查看SDK更新历史、获取安装包以及查看指导文档。

表 1-1 SDK 列表

接口名	编程语言	Github地址	参考文档	场景示例代码	工具包
发送接收短信	Java	huaweicloud-sdk-java-v3	Java SDK使用指导	使用SDK发送短信样例	华为云MSGSMS开发工具包（SDK）

接口名	编程语言	Github地址	参考文档	场景示例代码	工具包
应用、签名、模板	Java	huaweicloud-sdk-java-v3	Java SDK使用指导	<ul style="list-style-type: none"> 使用SDK创建、更新及查询短信应用样例 使用SDK创建、查询及删除短信模板样例 使用SDK创建、查询及删除短信签名样例 	华为云MSGSMS开发工具包(SDK)
	Python	huaweicloud-sdk-python-v3	Python SDK使用指导	-	
	Go	huaweicloud-sdk-go-v3	Go SDK使用指导		
	Node.js	huaweicloud-sdk-nodejs-v3	Node.js SDK使用指南		
	.NET	huaweicloud-sdk-net-v3	.Net SDK使用指导		

2 应用、签名、模板

2.1 Java SDK 使用指导

MSGSMS提供Java语言的应用侧SDK供开发者使用。本文介绍Java SDK的安装和配置，及使用Java SDK调用[API参考](#)的示例，您可以参考本章节进行快速集成开发。

开发前准备

- 已[注册](#)华为账号并开通华为云，完成[企业实名认证](#)。
- 已具备Java开发环境，支持Java JDK 1.8及以上版本。
- 已获取华为云账号对应的Access Key（AK）和Secret Access Key（SK）。
请在华为云控制台“我的凭证 > 访问密钥”页面上创建和查看您的AK/SK。具体请参见[访问密钥](#)。

安装 SDK

您可以通过Maven方式获取和安装SDK，首先需要您的操作系统中[下载并安装 Maven](#)，安装完成后您只需要在Java项目的pom.xml文件中加入相应的依赖项即可。

使用SDK前，您需要安装“huaweicloud-sdk-core”和“huaweicloud-sdk-msgsms”，具体的SDK版本号请参见[SDK开发中心](#)。

注意

以下代码示例中的version值，请根据实际的SDK版本号进行替换。

```
<dependency>
  <groupId>com.huaweicloud.sdk</groupId>
  <artifactId>huaweicloud-sdk-core</artifactId>
  <version>3.1.11</version>
</dependency>
<dependency>
  <groupId>com.huaweicloud.sdk</groupId>
  <artifactId>huaweicloud-sdk-msgsms</artifactId>
  <version>3.1.11</version>
</dependency>
```

代码示例

以调用[创建短信应用](#)接口为例，以下代码示例向您展示使用Java SDK的主要步骤：

1. 创建认证。
2. 创建MsgsmsClient实例并初始化。
3. 实例化请求对象。
4. 调用创建短信应用接口。

```
package com.huaweicloud.sdk.test;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.auth.BasicCredentials;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.msgsms.v2.region.MsgsmsRegion;
import com.huaweicloud.sdk.msgsms.v2.*;
import com.huaweicloud.sdk.msgsms.v2.model.*;

public class CreateAppSolution {

    public static void main(String[] args) {
        // 认证用的ak和sk直接写到代码中有很大的安全风险，建议在配置文件或者环境变量中密文存放，使用时解密，确保安全。
        // 本示例以ak和sk保存在环境变量中为例，运行本示例前请先在本地环境中设置环境变量
        // CLOUD_SDK_AK和CLOUD_SDK_SK。
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        // 创建认证
        ICredential auth = new BasicCredentials()
            .withAk(ak)
            .withSk(sk);

        // 创建MsgsmsClient实例并初始化
        MsgsmsClient client = MsgsmsClient.newBuilder()
            .withCredential(auth)
            .withRegion(MsgsmsRegion.valueOf("cn-north-4"))
            .build();

        // 实例化请求对象
        CreateAppRequest request = new CreateAppRequest();
        SmsAppAddReq body = new SmsAppAddReq();
        request.withBody(body);
        try {
            // 调用创建短信应用接口
            CreateAppResponse response = client.createApp(request);
            System.out.println(response.toString());
        } catch (ConnectionException e) {
            e.printStackTrace();
        } catch (RequestTimeoutException e) {
            e.printStackTrace();
        } catch (ServiceResponseException e) {
            e.printStackTrace();
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getRequestId());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

参数	说明
ak	您的华为云账号访问密钥ID（Access Key ID）。

参数	说明
sk	您的华为云账号秘密访问密钥（Secret Access Key）。
MsgsmsRegion .valueOf("cn-north-4")	请替换为您要访问的MSGSMS平台所在区域，当前MSGSMS支持访问的区域，在SDK代码 MsgsmsRegion.java 中已经定义。 您可以在MSGSMS控制台页面左上角查看当前服务所在区域名称。

说明

项目源码及更多详细的使用指导请参考[华为云Java软件开发工具包（Java SDK）](#)。

推荐您使用API在线调试工具[API Explorer](#)，API Explorer支持快速调试和检索，调试API的同时，可以根据您的参数实时生成各种开发语言的SDK示例代码，方便您直接根据示例代码使用SDK。

2.2 Python SDK 使用指导

MSGSMS提供Python语言的应用侧SDK供开发者使用。本文介绍Python SDK的安装和配置，及使用Python SDK调用[API参考](#)的示例，您可以参考本章节进行快速集成开发。

开发前准备

- 已[注册](#)华为账号并开通华为云，完成[企业实名认证](#)。
- 已具备Python开发环境，支持Python 3.3及以上版本。
- 已获取华为云账号对应的Access Key（AK）和Secret Access Key（SK）。

请在华为云控制台“我的凭证 > 访问密钥”页面上创建和查看您的AK/SK。具体请参见[访问密钥](#)。

安装 SDK

执行如下命令安装华为云Python SDK核心库以及相关服务库。

使用SDK前，您需要安装“huaweicloudsdkcore”和“huaweicloudsdkmsgsms”，具体的SDK版本号请参见[SDK开发中心](#)。

- 使用pip安装

执行如下命令安装华为云Python SDK核心库以及相关服务库：

```
# 安装核心库
pip install huaweicloudsdkcore

# 安装MSGSMS服务库
pip install huaweicloudsdkmsgsms
```

- 使用源码安装

执行如下命令安装华为云Python SDK核心库以及相关服务库：

```
# 安装核心库
cd huaweicloudsdkcore-${version}
python setup.py install
```



```
# 安装MSGSMS服务库
cd huaweicloudsdkmsgsms-${version}
python setup.py install
```

代码示例

以调用[创建短信应用](#)接口为例，以下代码示例向您展示使用Python SDK的主要步骤：

1. 创建认证。
2. 创建MsgsmsClient实例并初始化。
3. 实例化请求对象。
4. 调用创建短信应用接口。

```
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkmsgsms.v2.region.msgsms_region import MsgsmsRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkmsgsms.v2 import *

if __name__ == "__main__":
    # 认证用的ak和sk直接写到代码中有很大的安全风险，建议在配置文件或者环境变量中密文存放，使用时解密，确保安全。
    # 本示例以ak和sk保存在环境变量中为例，运行本示例前请先在本地环境中设置环境变量
    # CLOUD_SDK_AK和CLOUD_SDK_SK。
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]

    # 创建认证
    # 创建BasicCredentials实例并初始化
    credentials = BasicCredentials(ak, sk)

    client = MsgsmsClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(MsgsmsRegion.value_of("cn-north-4")) \
        .build()

    try:
        # 实例化请求对象
        request = CreateAppRequest()
        request.body = SmsAppAddReq(
        )
        # 调用创建短信应用接口
        response = client.create_app(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

参数	说明
ak	您的华为云账号访问密钥ID（Access Key ID）。
sk	您的华为云账号秘密访问密钥（Secret Access Key）。
MsgsmsRegion .valueOf("cn-north-4")	请替换为您要访问的MSGSMS平台所在区域，当前MSGSMS支持访问的区域，在SDK代码 msgsms_region.py 中已经定义。 您可以在MSGSMS控制台页面左上角查看当前服务所在区域名称。

📖 说明

项目源码及更多详细的使用指导请参考[华为云Python软件开发工具包（Python SDK）](#)。
推荐您使用API在线调试工具[API Explorer](#)，API Explorer支持快速调试和检索，调试API的同时，可以根据您的参数实时生成各种开发语言的SDK示例代码，方便您直接根据示例代码使用SDK。

2.3 Go SDK 使用指导

MSGSMS提供Go语言的应用侧SDK供开发者使用。本文介绍Go SDK的安装和配置，及使用Go SDK调用[API参考](#)的示例，您可以参考本章节进行快速集成开发。

开发前准备

- 已[注册](#)华为账号并开通华为云，完成[企业实名认证](#)。
- 已具备Go开发环境，支持Go 1.14及以上版本。
- 已获取华为云账号对应的Access Key（AK）和Secret Access Key（SK）。
请在华为云控制台“我的凭证 > 访问密钥”页面上创建和查看您的AK/SK。具体请参见[访问密钥](#)。

安装 SDK

执行如下命令安装华为云Go SDK库以及相关依赖库：

```
# 安装华为云Go库
go get github.com/huaweicloud/huaweicloud-sdk-go-v3

# 安装依赖
go get github.com/json-iterator/go
```

代码示例

以调用[创建短信应用](#)接口为例，以下代码示例向您展示使用Go SDK的主要步骤：

1. 创建认证。
2. 创建MsgsmsClient实例并初始化。
3. 实例化请求对象。
4. 调用创建短信应用接口。

```
package main

import (
    "fmt"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"
    msgsms "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/msgsms/v2"
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/msgsms/v2/model"
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/msgsms/v2/region"
)

func main() {
    // 认证用的ak和sk直接写到代码中有很大的安全风险，建议在配置文件或者环境变量中密文存放，使用时解密，确保安全。
    // 本示例以ak和sk保存在环境变量中为例，运行本示例前请先在本地环境中设置环境变量CLOUD_SDK_AK和CLOUD_SDK_SK。
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    // 创建认证
    auth := basic.NewCredentialsBuilder().
```

```
WithAk(ak).
WithSk(sk).
Build()

// 创建MsgsmsClient实例并初始化
client := msgsms.NewMsgsmsClient(
    msgsms.MsgsmsClientBuilder().
        WithRegion(region.ValueOf("cn-north-4")).
        WithCredential(auth).
        Build())

// 实例化请求对象
request := &model.CreateAppRequest{}
request.Body = &model.SmsAppAddReq{
}
// 调用创建短信应用接口
response, err := client.CreateApp(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
}
```

参数	说明
ak	您的华为云账号访问密钥ID（Access Key ID）。
sk	您的华为云账号秘密访问密钥（Secret Access Key）。
region.ValueOf("cn-north-4")	请替换为您要访问的MSGSMS平台所在区域，当前MSGSMS支持访问的区域，在SDK代码 region.go 中已经定义。您可以在MSGSMS控制台页面左上角查看当前服务所在区域名称。

📖 说明

项目源码及更多详细的使用指导请参考[华为云Go软件开发工具包（Go SDK）](#)。

推荐您使用API在线调试工具[API Explorer](#)，API Explorer支持快速调试和检索，调试API的同时，可以根据您的参数实时生成各种开发语言的SDK示例代码，方便您直接根据示例代码使用SDK。

2.4 Node.js SDK 使用指南

MSGSMS提供Node.js语言的应用侧SDK供开发者使用。本文介绍Node.js SDK的安装和配置，及使用Node.js SDK调用[API参考](#)的示例，您可以参考本章节进行快速集成开发。

开发前准备

- 已[注册](#)华为账号并开通华为云，完成[企业实名认证](#)。
- 已具备Node.js开发环境，支持Node 10.16.1及以上版本。
- 已获取华为云账号对应的Access Key（AK）和Secret Access Key（SK）。

请在华为云控制台“我的凭证 > 访问密钥”页面上创建和查看您的AK/SK。具体请参见[访问密钥](#)。

- 已获取MSGSMS服务对应区域的项目ID。
请在华为云控制台“我的凭证 > API凭证”页面上查看项目ID。具体请参见[API凭证](#)。

安装 SDK

执行如下命令安装华为云Node.js SDK相关依赖库：

```
npm install @huaweicloud/huaweicloud-sdk-core
npm install @huaweicloud/huaweicloud-sdk-msgsms
```

代码示例

以调用[创建短信应用](#)接口为例，以下代码示例向您展示使用Node.js SDK的主要步骤：

1. 创建认证。
2. 创建MsgsmsClient实例并初始化。
3. 实例化请求对象。
4. 调用创建短信应用接口。

```
const core = require('@huaweicloud/huaweicloud-sdk-core');
const msgsms = require("@huaweicloud/huaweicloud-sdk-msgsms/v2/public-api");

// 认证用的ak和sk直接写到代码中有很大的安全风险，建议在配置文件或者环境变量中密文存放，使用时解密，确保安全。
// 本示例以ak和sk保存在环境变量中为例，运行本示例前请先在本地环境中设置环境变量CLOUD_SDK_AK和CLOUD_SDK_SK。
const ak = process.env.CLOUD_SDK_AK;
const sk = process.env.CLOUD_SDK_SK;
const endpoint = "https://msgsms.cn-north-4.myhuaweicloud.com";
const project_id = "";

// 创建认证
const credentials = new core.BasicCredentials()
    .withAk(ak)
    .withSk(sk)
    .withProjectId(project_id)
// 创建MsgsmsClient实例并初始化
const client = msgsms.MsgsmsClient.newBuilder()
    .withCredential(credentials)
    .withEndpoint(endpoint)
    .build();

// 实例化请求对象
const request = new msgsms.CreateAppRequest();
const body = new msgsms.SmsAppAddReq();
request.withBody(body);
// 调用创建短信应用接口
const result = client.createApp(request);
result.then(result => {
    console.log("JSON.stringify(result)::" + JSON.stringify(result));
}).catch(ex => {
    console.log("exception:" + JSON.stringify(ex));
});
```

参数	说明
ak	您的华为云账号访问密钥ID（Access Key ID）。
sk	您的华为云账号秘密访问密钥（Secret Access Key）。
endpoint	请替换为您要访问的MSGSMS服务所在区域的终端节点，在SDK代码 MsgsmsRegion.ts 中已经定义。

参数	说明
project_id	云服务所在区域的项目ID，根据您需要操作的项目所属区域选择对应的项目ID。

📖 说明

项目源码及更多详细的使用指导请参考[华为云Node.js软件开发工具包（Node.js SDK）](#)。推荐您使用API在线调试工具[API Explorer](#)，API Explorer支持快速调试和检索，调试API的同时，可以根据您的参数实时生成各种开发语言的SDK示例代码，方便您直接根据示例代码使用SDK。

2.5 .NET SDK 使用指导

MSGSMS提供.NET语言的应用侧SDK供开发者使用。本文介绍.NET SDK的安装和配置，及使用.NET SDK调用[API参考](#)的示例，您可以参考本章节进行快速集成开发。

开发前准备

- 已[注册](#)华为账号并开通华为云，完成[企业实名认证](#)。
- 已具备.NET开发环境，华为云.NET SDK适用于以下版本：
 - .NET and .NET Core 2.0及以上版本。
 - .NET Framework 4.6.2及以上版本。
- 已获取华为云账号对应的Access Key（AK）和Secret Access Key（SK）。请在华为云控制台“我的凭证 > 访问密钥”页面上创建和查看您的AK/SK。具体请参见[访问密钥](#)。

安装 SDK

使用.NET CLI工具安装SDK。

```
dotnet add package HuaweiCloud.SDK.Core  
dotnet add package HuaweiCloud.SDK.Msgsms
```

代码示例

以调用[创建短信应用](#)接口为例，以下代码示例向您展示使用.NET SDK的主要步骤：

1. 创建认证。
2. 创建MsgsmsClient实例并初始化。
3. 实例化请求对象。
4. 调用创建短信应用接口。

```
using System;  
using System.Collections.Generic;  
using HuaweiCloud.SDK.Core;  
using HuaweiCloud.SDK.Core.Auth;  
using HuaweiCloud.SDK.Msgsms;  
using HuaweiCloud.SDK.Msgsms.V2;  
using HuaweiCloud.SDK.Msgsms.V2.Model;  
  
namespace CreateAppSolution
```

```

{
class Program
{
static void Main(string[] args)
{
// 认证用的ak和sk直接写到代码中有很大的安全风险，建议在配置文件或者环境变量中密文存放，
// 使用时解密，确保安全。
// 本示例以ak和sk保存在环境变量中为例，运行本示例前请先在本地环境中设置环境变量
// CLOUD_SDK_AK和CLOUD_SDK_SK。
string ak = Environment.GetEnvironmentVariable("CLOUD_SDK_AK");
string sk = Environment.GetEnvironmentVariable("CLOUD_SDK_SK");

var config = HttpConfig.GetDefaultConfig();
config.IgnoreSslVerification = true;
// 创建认证
var auth = new BasicCredentials(ak, sk);

// 创建MsgsmsClient实例并初始化
var client = MsgsmsClient.NewBuilder()
    .WithCredential(auth)
    .WithRegion(MsgsmsRegion.ValueOf("cn-north-4"))
    .WithHttpConfig(config)
    .Build();

// 实例化请求对象
var req = new CreateAppRequest
{
};
req.Body = new SmsAppAddReq()
{
};

try
{
// 调用创建短信应用接口
var resp = client.CreateApp(req);
var respStatusCode = resp.HttpStatusCode;
Console.WriteLine(respStatusCode);
}
catch (RequestTimeoutException requestTimeoutException)
{
Console.WriteLine(requestTimeoutException.ErrorMessage);
}
catch (ServiceResponseException clientRequestException)
{
Console.WriteLine(clientRequestException.HttpStatusCode);
Console.WriteLine(clientRequestException.RequestId);
Console.WriteLine(clientRequestException.ErrorCode);
Console.WriteLine(clientRequestException.ErrorMessage);
}
catch (ConnectionException connectionException)
{
Console.WriteLine(connectionException.ErrorMessage);
}
}
}
}
}

```

参数	说明
ak	您的华为云账号访问密钥ID（Access Key ID）。
sk	您的华为云账号秘密访问密钥（Secret Access Key）。

参数	说明
MsgsmsRegion .ValueOf("cn- north-4")	请替换为您要访问的MSGSMS平台所在区域，当前MSGSMS支持访问的区域，在SDK代码 MsgsmsRegion.cs 中已经定义。 您可以在MSGSMS控制台页面左上角查看当前服务所在区域名称。

说明

项目源码及更多详细的使用指导请参考[华为云.Net软件开发工具包（.Net SDK）](#)。

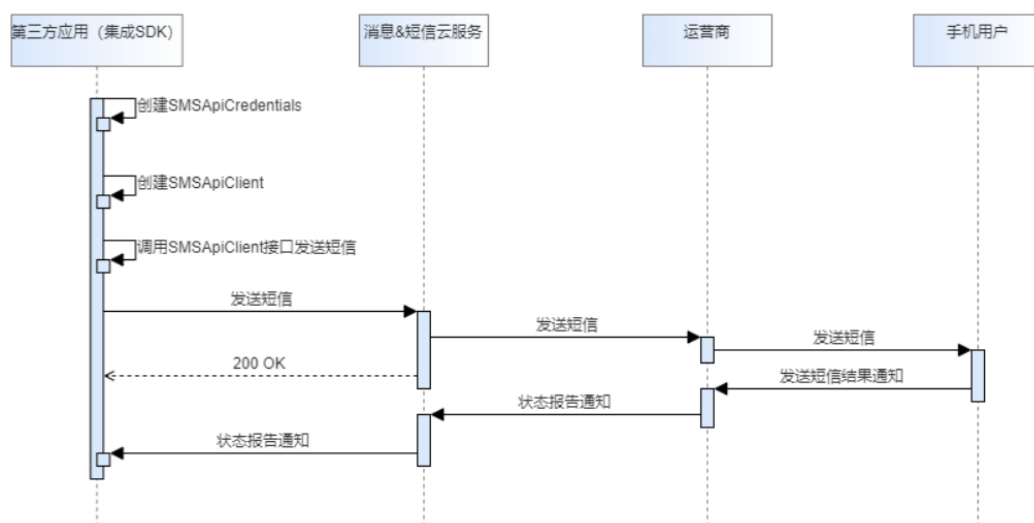
推荐您使用API在线调试工具[API Explorer](#)，API Explorer支持快速调试和检索，调试API的同时，可以根据您的参数实时生成各种开发语言的SDK示例代码，方便您直接根据示例代码使用SDK。

3 发送接收短信

3.1 Java SDK 使用指导

MSGSMS提供Java语言的应用侧SDK供开发者使用，您可以直接集成SDK来调用MSGSMS的短信发送API，从而实现对MSGSMS的快速操作。

开发时序图



开发前准备

- 已[注册](#)华为账号并开通华为云，完成[企业实名认证](#)。
- 已[开通](#)华为云MSGSMS服务。
- 已具备Java开发环境，支持Java JDK 1.8及以上版本。
- 已获取应用的Application Key(app_key)和Application Secret(app_secret)。请在华为云控制台“应用管理”页面上创建和查看您的Application Key和Application Secret。具体请参见[应用管理](#)。
- 已获取您要发送短信的通道号。

由于通道号与签名关联，请在华为云控制台“签名管理”页面上查看您的通道号，如果您还没有通道号，则您需要先提交签名申请，待签名申请通过后，系统会分配通道号。具体请参见[签名管理](#)。

- 已获取您要发送短信的模板ID。
请在华为云控制台“模板管理”页面上创建和查看您的模板ID。具体请参见[模板管理](#)。

获取和安装 SDK

您可以通过Maven方式获取和安装SDK，首先需要在您的操作系统中[下载并安装Maven](#)，安装完成后您只需要在Java项目的pom.xml文件中加入相应的依赖项即可。

使用SDK前，您需要安装“huaweicloud-sdk-smsapi”，具体的SDK版本号请参见[SDK开发中心](#)。

注意

以下代码示例中的version值，请根据实际的SDK版本号进行替换。

```
<dependency>
  <groupId>com.huaweicloud.sdk</groupId>
  <artifactId>huaweicloud-sdk-smsapi</artifactId>
  <version>3.1.125</version>
</dependency>
```

接口参数说明

关于接口参数的详细说明请参见：

- [发送分批短信](#)
- [发送短信](#)

代码示例

须知

- 发送短信为单模板群发短信示例，发送分批短信为多模板群发短信示例。
- 本文档所述Demo在提供服务的过程中，可能会涉及个人数据的使用，建议您遵从国家的相关法律采取足够的措施，以确保用户的个人数据受到充分的保护。
- 本文档所述Demo仅用于功能演示，不允许客户直接进行商业使用。
- 本文档信息仅供参考，不构成任何要约或承诺。

```
package com.huawei.smsapi;

import com.huaweicloud.sdk.core.auth.ICredential;
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
import com.huaweicloud.sdk.core.http.HttpConfig;
import com.huaweicloud.sdk.core.region.Region;
import com.huaweicloud.sdk.smsapi.v1.SMSApiClient;
import com.huaweicloud.sdk.smsapi.v1.SMSApiCredentials;
```

```
import com.huaweicloud.sdk.smsapi.v1.model.BatchSendDiffSmsRequest;
import com.huaweicloud.sdk.smsapi.v1.model.BatchSendDiffSmsRequestBody;
import com.huaweicloud.sdk.smsapi.v1.model.BatchSendDiffSmsResponse;
import com.huaweicloud.sdk.smsapi.v1.model.BatchSendSmsRequest;
import com.huaweicloud.sdk.smsapi.v1.model.BatchSendSmsRequestBody;
import com.huaweicloud.sdk.smsapi.v1.model.BatchSendSmsResponse;
import com.huaweicloud.sdk.smsapi.v1.model.SmsContent;

import java.util.ArrayList;
import java.util.Arrays;

/**
 * Use the SDK interface to send an SMS message.
 * Before using this example, you need to create an SMS application, signature,
 * and template through the console or by invoking the open APIs of the MSGSMS.
 */
public class SendSms {
    public static void main(String[] args) {
        System.out.println("Start HUAWEI CLOUD MSGSMS SEND SMS Java Demo...");

        /*
         * Send an SMS message using a special AK/SK authentication algorithm.
         * When the MSGSMS is used to send SMS messages, the AK is app_key, and the SK is app_secret.
         * There will be security risks if the app_key/app_secret used for authentication is directly written into
         code.
         * We suggest encrypting the app_key/app_secret in the configuration file or environment variables for
         storage.
         * In this sample, the app_key/app_secret is stored in environment variables for identity authentication.
         * Before running this sample, set the environment variables CLOUD_SDK_MSGSMS_APPKEY and
         CLOUD_SDK_MSGSMS_APPSECRET.
         * CLOUD_SDK_MSGSMS_APPKEY indicates the application key (app_key), and
         CLOUD_SDK_MSGSMS_APPSECRET indicates the application secret (app_secret).
         * You can obtain the value from Application Management on the console or by calling the open API
         of Application Management.
         */
        String ak = System.getenv("CLOUD_SDK_MSGSMS_APPKEY");
        String sk = System.getenv("CLOUD_SDK_MSGSMS_APPSECRET");

        /* Creating an SmsApiClient Instance
         * current support region:
         * CN North-Beijing4: RegionId: cn-north-4, endpoint: https://smsapi.cn-
         north-4.myhuaweicloud.com:443
         * CN South-Guangzhou: RegionId: cn-south-1, endpoint: https://smsapi.cn-
         south-1.myhuaweicloud.com:443
         */
        SMSApiClient client = getClient(new Region("cn-north-4", new String[]{"https://smsapi.cn-
        north-4.myhuaweicloud.com: 443"}),
            getCredential(ak, sk));

        try {
            // Example 1: Use POST /sms/batchSendSms/v1 to send an SMS message.
            System.out.println("----Example 1 use POST /sms/batchSendSms/v1 send msg now.");
            int httpcode = batchSendSms(client);
            System.out.println((httpcode == 200) ? "----Example 1 send sms success." : "----Example 1 send
            sms failed.");

            // Example 2: Using the POST /sms/batchSendDiffSms/v1 to Send SMS Messages in Batches
            System.out.println("----Example 2 use POST /sms/batchSendDiffSms/v1 send msg now.");
            httpcode = batchSendDiffSms(client);
            System.out.println((httpcode == 200) ? "----Example 2 send sms success." : "----Example 2 send
            sms failed.");
        } catch (ConnectionException | RequestTimeoutException e) {
            System.out.println(e.getMessage());
        } catch (ServiceResponseException e) {
            System.out.println(e.getHttpStatusCode());
            System.out.println(e.getErrorCode());
            System.out.println(e.getErrorMsg());
        }
    }
}
```

```
/**
 * Sending SMS Messages
 *
 * @param client SMSApiClient Instance
 * @return Indicates whether the SMS message is sent successfully.
 */
private static int batchSendSms(SMSApiClient client) {
    // Construct a request for sending an SMS message.
    BatchSendSmsRequest request = new BatchSendSmsRequest();
    request.withBody(new BatchSendSmsRequestBody()
        // Channel ID for sending SMS messages.
        .withFrom("8824110605****")
        // List of numbers that receive SMS messages.
        // Note: When there are multiple numbers, do not contain spaces between the numbers.
        .withTo("+86137****3774,+86137****3776")
        // Template Id
        .withTemplateId("e1440669a4354ccdb56ebf2283c6****")
        // Template parameter, which must be enclosed in square brackets ([]).
        .withTemplateParas("[\\"12\","\\"23\","\\"e\"])")
        // Status report callback URL. Set this parameter to a valid value. If status reports are not
        // required, you do not need to set this parameter.
        .withStatusCallback("https://test/report"));

    // Invoke the SDK interface to send an SMS message.
    BatchSendSmsResponse response = client.batchSendSms(request);

    // Print the response result.
    System.out.println(response.toString());
    return response.getHttpStatusCode();
}

/**
 * Sending SMS Messages in Batches
 *
 * @param client SMSApiClient Instance
 * @return Whether the batch SMS message is sent successfully.
 */
private static int batchSendDiffSms(SMSApiClient client) {
    //Construct a request for sending SMS messages in batches.
    BatchSendDiffSmsRequest request = new BatchSendDiffSmsRequest();
    request.withBody(new BatchSendDiffSmsRequestBody()
        // Channel ID for sending SMS messages.
        .withFrom("8824110605****")
        // Status report callback URL. Set this parameter to a valid value. If status reports are not
        // required, you do not need to set this parameter.
        .withStatusCallback("https://test/report")
        .withSmsContent(new ArrayList<>(Arrays.asList(
            // Content of the first batch of group SMS messages
            new SmsContent()
                // List of SMS Recipient Numbers
                .withTo(new ArrayList<>((Arrays.asList("+86137****3774", "+86137****3775"))))
                // Template Id
                .withTemplateId("cefada4b8eaa4835864eb5b5eae1****")
                // Template Parameters
                .withTemplateParas(new ArrayList<>(Arrays.asList("1", "23", "45"))),

            // Content of the second batch of group SMS messages
            new SmsContent()
                // List of SMS Recipient Numbers
                .withTo(new ArrayList<>((Arrays.asList("+86137****3777", "+86137****3778"))))
                // Template Id
                .withTemplateId("e1440669a4354ccdb56ebf2283c6****")
                // Template Parameters
                .withTemplateParas(new ArrayList<>(Arrays.asList("3", "4", "5")))))));

    //Invoke the SDK interface to send SMS messages in batche
    BatchSendDiffSmsResponse response = client.batchSendDiffSms(request);
}
```

```
// Print the response result.
System.out.println(response.toString());
return response.getStatusCode();
}

/**
 * Creating an Authentication Credential
 *
 * @param ak Application key (AK) generated after an app is created for the SMS service.
 * @param sk Application Secret (SK) generated after an app is created for the SMS service.
 * @return Authentication credential
 */
private static SMSApiCredentials getCredential(String ak, String sk) {
    return new SMSApiCredentials()
        .withAk(ak)
        .withSk(sk);
}

/**
 * Create SMSApiClient Instance
 *
 * @param region region
 * @param auth Authentication credential
 * @return SMSApiClient Instance
 */
private static SMSApiClient getClient(Region region, ICredential auth) {
    // Use the default configuration.
    HttpConfig config = HttpConfig.getDefaultHttpConfig();
    config.withIgnoreSSLVerification(true);

    // Initializing the Client of the SMSApi Service
    return SMSApiClient.newBuilder()
        .withHttpConfig(config)
        .withCredential(auth)
        .withRegion(region)
        .build();
}
}
```

📖 说明

发送短信SDK默认使用的是okhttp组件发送消息，最大链接数为5；线程池使用ForkJoinPool，默认线程池与服务器的CPU数一样。这两个参数会限制系统的最大并发数，如果出现时延变长、性能无法提升时，可重定义HttpConfig的这两个参数以提升系统性能，例如：

```
HttpConfig config = HttpConfig.getDefaultHttpConfig();
config.withConnectionPool(new ConnectionPool(50, 5L, TimeUnit.MINUTES));
config.withExecutorService(new ForkJoinPool(200));
```

运行结果

- 发送短信 (BatchSendSms)

```
```json
class BatchSendSmsResponse {
 code: 000000
 description: Success
 result: [class SmsID {
 createTime: 2024-11-22T02:16:46Z
 from: 8824110605***
 originTo: +86137****3774
 smsMsgId: eb4e63dd-0945-4231-a48d-6b54c1a8b614_3434030
 status: 000000
 countryId: CN
 total: 1
 }, class SmsID {
 createTime: 2024-11-22T02:16:46Z
 from: 8824110605***
 originTo: +86137****3776
 smsMsgId: eb4e63dd-0945-4231-a48d-6b54c1a8b614_3434031
 }
}
```

```
status: 000000
countryId: CN
total: 1
}]
}
...
```

- 发送分批短信 ( BatchSendDiffSms )

```
``json
class BatchSendDiffSmsResponse {
code: 000000
description: Success
result: [class SmsID {
createTime: 2024-11-22T02:16:46Z
from: 8824110605***
originTo: +86137****3774
smsMsgId: eb4e63dd-0945-4231-a48d-6b54c1a8b614_3435044
status: 000000
countryId: CN
total: 0
}, class SmsID {
createTime: 2024-11-22T02:16:46Z
from: 8824110605***
originTo: +86137****3775
smsMsgId: eb4e63dd-0945-4231-a48d-6b54c1a8b614_3435045
status: 000000
countryId: CN
total: 0
}, class SmsID {
createTime: 2024-11-22T02:16:46Z
from: 8824110605***
originTo: +86137****3777
smsMsgId: eb4e63dd-0945-4231-a48d-6b54c1a8b614_3435046
status: 000000
countryId: CN
total: 1
}, class SmsID {
createTime: 2024-11-22T02:16:46Z
from: 8824110605***
originTo: +86137****3778
smsMsgId: eb4e63dd-0945-4231-a48d-6b54c1a8b614_3435047
status: 000000
countryId: CN
total: 1
}]
}
...
```