

Optical Character Recognition Service

SDK Reference

Issue 01
Date 2024-09-04



Copyright © Huawei Cloud Computing Technologies Co., Ltd. 2024. All rights reserved.

No part of this document may be reproduced or transmitted in any form or by any means without prior written consent of Huawei Cloud Computing Technologies Co., Ltd.

Trademarks and Permissions



HUAWEI and other Huawei trademarks are the property of Huawei Technologies Co., Ltd.

All other trademarks and trade names mentioned in this document are the property of their respective holders.

Notice

The purchased products, services and features are stipulated by the contract made between Huawei Cloud and the customer. All or part of the products, services and features described in this document may not be within the purchase scope or the usage scope. Unless otherwise specified in the contract, all statements, information, and recommendations in this document are provided "AS IS" without warranties, guarantees or representations of any kind, either express or implied.

The information in this document is subject to change without notice. Every effort has been made in the preparation of this document to ensure accuracy of the contents, but all statements, information, and recommendations in this document do not constitute a warranty of any kind, express or implied.

Huawei Cloud Computing Technologies Co., Ltd.

Address: Huawei Cloud Data Center Jiaoxinggong Road
Qianzhong Avenue
Gui'an New District
Gui Zhou 550029
People's Republic of China

Website: <https://www.huaweicloud.com/intl/en-us/>

Contents

1 OCR SDK Overview.....	1
2 Java SDK.....	4
3 Python SDK.....	10
4 Node.js SDK.....	15
5 PHP SDK.....	18
6 C++ SDK.....	23
7 Go SDK.....	28
8 .NET SDK.....	32

1 OCR SDK Overview

SDK Introduction

Optical Character Recognition (OCR) detects and extracts text from images or scanned copies and converts the text into an editable format, freeing you from burdensome manual input while improving the efficiency of your business.

OCR provides services through open APIs. You can obtain inference results by accessing and calling APIs in real time, significantly improving the efficiency of your business.

OCR SDK encapsulates RESTful APIs provided by OCR to simplify development. You can directly call API functions provided by OCR SDKs to use OCR.

The developer guides for SDKs in various programming languages are as follows:

Table 1-1 Developer guides for SDKs in various programming languages

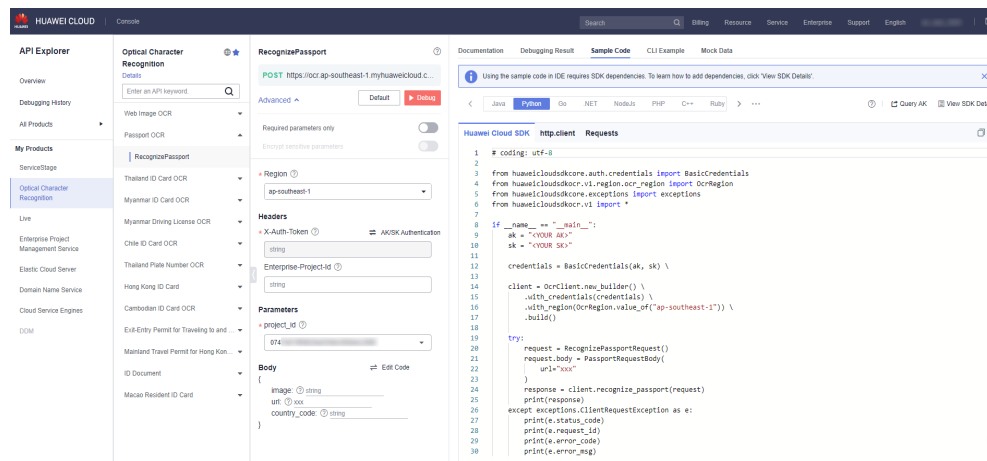
Programming Language	Developer Guide
Java	Java SDK Developer Guide
Python	Python SDK Developer Guide
Node.js	Node.js SDK Developer Guide
PHP	PHP SDK Developer Guide
C++	C++ SDK Developer Guide
Go	Go SDK Developer Guide
NET	NET SDK Developer Guide

Automatic Generation of Sample SDK Code

[API Explorer](#) allows for API search and platform debugging, with features such as quick and comprehensive search, visual debugging, access to help documentation, and online consultation.

You only need to modify API parameters in the API Explorer to automatically generate the corresponding sample code. In addition, you can build, debug, and run code in CloudIDE.

Figure 1-1 API Explorer



Mapping Between Services and APIs

Table 1-2 lists the mappings between OCR services and APIs.

Table 1-2 Mapping between services and APIs

Service	API
ID Document	POST /v2/{project_id}/ocr/id-document
General Table	POST /v2/{project_id}/ocr/general-table
General Text	POST /v2/{project_id}/ocr/general-text
Web Image	POST /v2/{project_id}/ocr/web-image
Passport	POST /v2/{project_id}/ocr/passport
Thailand ID Card	POST /v2/{project_id}/ocr/thailand-id-card
Cambodian ID Card	POST /v2/{project_id}/ocr/cambodian-idcard
Myanmar ID Card	POST /v2/{project_id}/myanmar-id-card
Myanmar Driving License	POST /v2/{project_id}/ocr/myanmar-driver-license
Chile ID Card	POST /v2/{project_id}/ocr/chile-id-card
Vietnam ID Card	POST /v2/{project_id}/ocr/vietnam-id-card

Service	API
Thailand Plate Number	POST /v2/{project_id}/ocr/thailand-license-plate

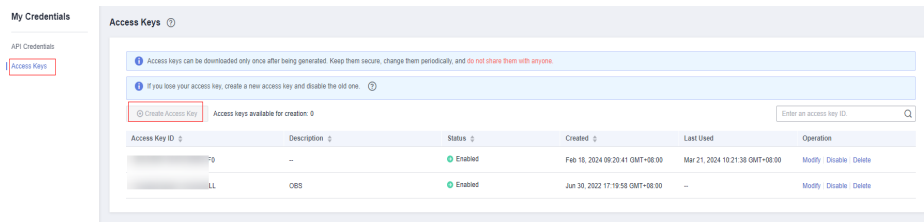
2 Java SDK

This section describes how to use the new Java SDK to quickly develop OCR services.

Prerequisites

- You have registered a Huawei ID and enabled Huawei Cloud services. Your account cannot be in arrears or frozen.
- Java JDK 1.8 or later is available.
- You have obtained an AK and an SK on the [My Credentials](#) > [Access Keys](#) page. The AK and SK are contained in the `credentials.csv` file.

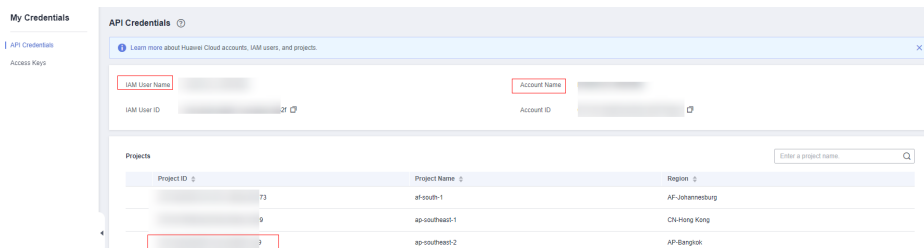
Figure 2-1 Creating an access key



A	B	C	D	E
User Name	Access Key	Secret Access Key		
testuser	LSKM	rIZaQ		
	AK	SK		

- You have obtained the IAM user name, account name, and the project ID of your target region on the [My Credentials](#) > [API Credentials](#) page. The information will be used during service calling. Save it in advance.

Figure 2-2 My Credentials



Installing the SDK

Obtain and install the SDK using Maven. [Download](#) and [install Maven](#) in your operating system. Add the dependency to the **pom.xml** file of the Java project.

Before using the SDK, install **huaweicloud-sdk-core** and **huaweicloud-sdk-ocr**. Obtain the latest version of the SDK package from [SDK Center](#) and replace the version in the code.

```
<dependency>
  <groupId>com.huaweicloud.sdk</groupId>
  <artifactId>huaweicloud-sdk-core</artifactId>
  <version>3.1.5</version>
</dependency>
<dependency>
  <groupId>com.huaweicloud.sdk</groupId>
  <artifactId>huaweicloud-sdk-ocr</artifactId>
  <version>3.1.5</version>
</dependency>
```

NOTE

When a third-party library conflict occurs, for example, a Jackson or OkHttp3 version conflict, you can import the following bundle package (version later than 3.0.40-rc). This package contains all supported services and redirected third-party software on which the SDK depends to avoid conflicts with the libraries on which the service depends.

```
<dependency>
  <groupId>com.huaweicloud.sdk</groupId>
  <artifactId>huaweicloud-sdk-bundle</artifactId>
  <version>[3.0.40-rc, 3.1.0)</version>
</dependency>
```

For details about the Jackson version requirements, see [pom.xml](#).

Getting Started

1. Import dependency modules.

```
package com.huaweicloud.sdk.test;
import com.huaweicloud.sdk.core.auth.ICredential;
// Authenticate the user.
import com.huaweicloud.sdk.core.auth.BasicCredentials;
// Import a request exception class.
import com.huaweicloud.sdk.core.exception.ConnectionException;
import com.huaweicloud.sdk.core.exception.RequestTimeoutException;
import com.huaweicloud.sdk.core.exception.ServiceResponseException;
// Import the OCR client.
import com.huaweicloud.sdk.ocr.v1.region.OcrRegion;
import com.huaweicloud.sdk.ocr.v1.*;
import com.huaweicloud.sdk.ocr.v1.model.*;
```

2. Configure client connection parameters.

- Using the default configuration
// Use the default configuration.
HttpConfig config = HttpConfig.getDefaultHttpConfig();
- (Optional) Configuring a network proxy
// Configure a network proxy as needed. The default protocol of the network proxy is HTTP.
config.withProxyHost("proxy.huaweicloud.com")
 .withProxyPort(8080)
 .withProxyUsername("test")
 .withProxyPassword("test");
- (Optional) Configuring the timeout
// The default connection timeout interval is 60 seconds. You can change it as needed.
config.withTimeout(60);

- (Optional) Configuring an SSL

```
// Configure whether to skip SSL certificate verification as needed.  
// If the "javax.net.ssl.SSLHandshakeException: Received fatal alert: handshake_failure" error is  
// reported during SDK execution, add the following code to skip SSL certificate verification:  
config.withIgnoreSSLVerification(true);
```

3. Configure authentication information.

Configure **ak**, **sk**, and **project_id**. AK is used together with SK to sign requests cryptographically, ensuring that the requests are secret, complete, and correct. There are two authentication methods:

- Initialize authentication information.

```
String ak = System.getenv("HUAWEICLOUD_SDK_AK");  
String sk = System.getenv("HUAWEICLOUD_SDK_SK");
```

 NOTE

- Hard-coded or plaintext AK and SK are risky. For security purposes, encrypt your AK and SK and store them in the configuration file or environment variables.
- In this example, the AK and SK are stored in environment variables for identity authentication. Before running this example, configure environment variables **HUAWEICLOUD_SDK_AK** and **HUAWEICLOUD_SDK_SK**.
- If environment variables are not set, you can skip calling the `System.getenv` method and directly enter the corresponding ak and sk values, for example, `String ak = "ak"; String sk = "sk"`.

- Using a permanent AK and SK

```
BasicCredentials basicCredentials = new BasicCredentials()  
.withAk(ak)  
.withSk(sk)  
.withProjectId(projectId);
```

- Using a temporary AK and SK

```
BasicCredentials basicCredentials = new BasicCredentials()  
.withAk(ak)  
.withSk(sk)  
.withSecurityToken(securityToken)  
.withProjectId(projectId)
```

The authentication parameters are described as follows:

- **ak** and **sk**: access key and secret access key. For details about how to obtain them, see [Prerequisites](#).
- **projectId**: Huawei Cloud project ID. For details about how to obtain the ID, see [Prerequisites](#).
- **securityToken**: security token used for temporary AK/SK authentication. It can be obtained through [a token](#) or [an agency](#).

4. Initialize the client (using either of the following methods).

- (Recommended) Specifying a region

```
// Add the region dependency.  
import com.huaweicloud.sdk.ocr.v1.region.OcrRegion;  
  
// Initialize the client authentication information. If the current client is used, projectId or  
domainId can be left blank.  
ICredential auth = new BasicCredentials()  
.withAk(ak)  
.withSk(sk);  
  
// Initialize the client {Service}Client of a specified cloud service. The following uses OcrClient of  
OCR as an example.  
OcrClient client = OcrClient.newBuilder()  
.withHttpConfig(config)  
.withCredential(auth)  
.withRegion(OcrRegion.valueOf("ap-southeast-2"))  
.build();
```

- Specifying an endpoint for a cloud service

```
// Specify the endpoint for OCR, for example, AP-Bangkok.
String endpoint = "https://ocr.ap-southeast-2.myhuaweicloud.com";
// Initialize the client authentication information. You need to configure projectId or domainId.
// The following uses BasicCredentials as an example.
BasicCredentials basicCredentials = new BasicCredentials()
    .withAk(ak)
    .withSk(sk)
    .withProjectId(projectId);

// Initialize the client {Service}Client of a specified cloud service. The following uses OcrClient of
// a region-level OCR as an example.
OcrClient ocrClient = OcrClient.newBuilder()
    .withHttpConfig(config)
    .withCredential(basicCredentials)
    .withEndpoint(endpoint)
    .build();
```

endpoint indicates the endpoints for Huawei Cloud services. For details, see [Regions and Endpoints](#).

5. Send a request and check the response.

```
// The following uses calling the RecognizePassport API of Passport OCR as an example.
RecognizePassportRequest request = new RecognizePassportRequest();
PassportRequestBody body = new PassportRequestBody();
body.withUrl("Image URL");
request.withBody(body);
try {
    RecognizePassportResponse response = client.recognizePassport(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.getErrorCode());
    System.out.println(e.getErrorMsg());
}
```

6. Handle the exception.

Table 2-1 Exception handling

Level-1 Category	Level-1 Category Description	Level-2 Category	Level-2 Category Description
ConnectionException	Connection exception	HostUnreachableException	The network is unreachable or access is rejected.
		SslHandShakeException	SSL authentication is abnormal.
RequestTimeoutException	Response timeout exception	CallTimeoutException	The server fails to respond to a single request before timeout.

Level-1 Category	Level-1 Category Description	Level-2 Category	Level-2 Category Description
		RetryOutageException	No valid response is returned after the maximum number of retries specified in the retry policy is reached.
ServiceResponseException	Server response exception	ServerResponseException	Internal server error. HTTP response code: [500,].
		ClientRequestException	Invalid request parameter. HTTP response code: [400, 500).

```
// Troubleshooting
try {
    RecognizePassportResponse response = client.recognizePassport(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.getErrorCode());
    System.out.println(e.getErrorMsg());
}
```

 **NOTE**

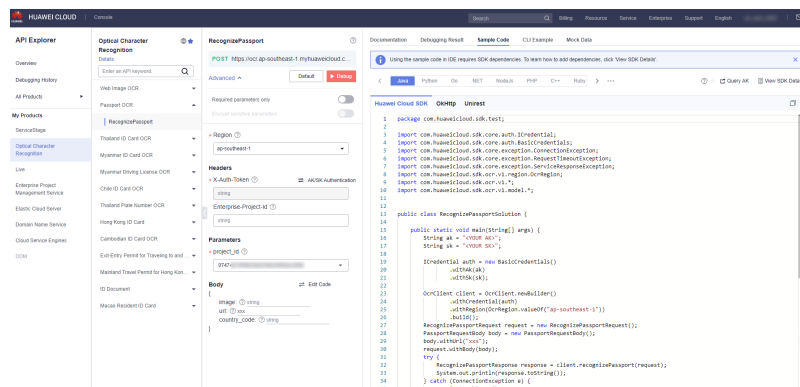
For how to use the asynchronous client and configure logs, see [SDK Center](#) and [Java SDK Usage Guide](#).

Automatic Generation of Sample Code

[API Explorer](#) allows for API search and platform debugging, with features such as quick and comprehensive search, visual debugging, access to help documentation, and online consultation.

You only need to modify API parameters in the API Explorer to automatically generate the corresponding sample code.

Figure 2-3 API Explorer



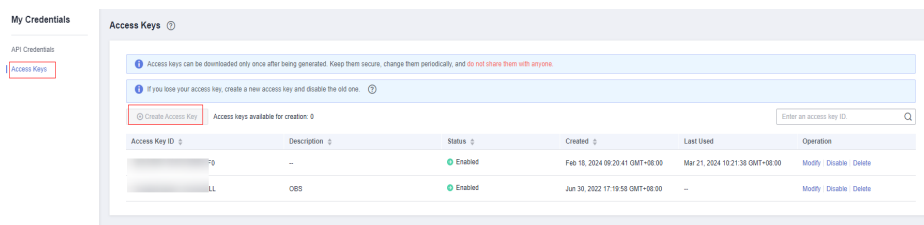
3 Python SDK

This section describes how to use the new Python SDK to quickly develop OCR services.

Prerequisites

- You have registered a Huawei ID and enabled Huawei Cloud services. Your account cannot be in arrears or frozen.
- Python 3 or later is available.
- You have obtained an AK and an SK on the [My Credentials](#) > [Access Keys](#) page. The AK and SK are contained in the `credentials.csv` file.

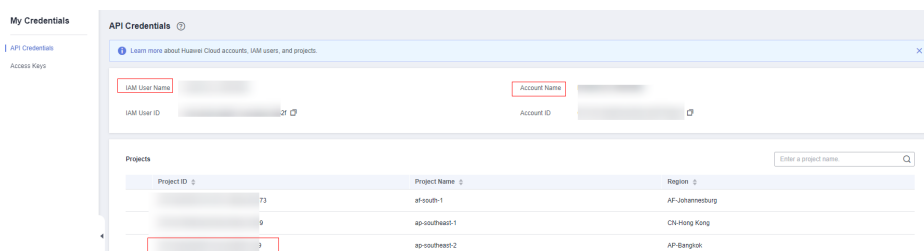
Figure 3-1 Creating an access key



A	B	C	D	E
User Name	Access Key	Secret Access Key		
testuser	LSKM	rIZaQ		
	AK	SK		

- You have obtained the IAM user name, account name, and the project ID of your target region on the [My Credentials](#) > [API Credentials](#) page. The information will be used during service calling. Save it in advance.

Figure 3-2 My Credentials



Installing the SDK

Python 3 or later is supported. Run the `python --version` command to check the current Python version.

Before you use the SDK, install `huaweicloudsdkcore` and `huaweicloudsdkocr` in either of the following ways:

- Using pip

```
# If the message "Successfully installed xxx" is displayed, the installation is successful.
# Install the core library.
pip install huaweicloudsdkcore
# Install the OCR service library.
pip install huaweicloudsdkocr
```

- Using the source code

The SDK version can be queried from [SDK Center](#).

```
# Install the core library.
cd huaweicloudsdkcore-${version}
python setup.py install

# Install the OCR service library.
cd huaweicloudsdkocr-${version}
python setup.py install
```

Getting Started

1. Import dependency modules.

```
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcore.http.http_config import HttpConfig
# Import the huaweicloudsdk{service} library of a specified cloud service.
from huaweicloudsdkocr.v1.region.ocr_region import OcrRegion
from huaweicloudsdkocr.v1 import *
```

2. Configure client connection parameters.

- Using the default configuration

```
# Use the default configuration. If the error message "'HttpConfig' is not defined" is displayed,
check whether the SDK is installed correctly.
config = HttpConfig.get_default_config()
```

- (Optional) Configuring a network proxy

```
# Configure network proxy as needed.
config.proxy_protocol = 'http'
config.proxy_host = 'proxy.huaweicloud.com'
config.proxy_port = 80
config.proxy_user = 'username'
config.proxy_password = 'password'
```

- (Optional) Configuring the timeout

```
# The default connection timeout interval is 60 seconds, and the read timeout interval is 120
seconds. You can set timeout to timeout or (connect timeout, read timeout).
config.timeout = 120
```

- (Optional) Configuring an SSL

```
# Configure whether to skip SSL certificate verification as required.
config.ignore_ssl_verification = True
# Configure the server CA certificate so that the SDK can verify the server certificate.
config.ssl_ca_cert = ssl_ca_cert
```

NOTE

After the client connection parameters are configured, you need to configure the code corresponding to `with_http_config(config)` in the initialized client. For details, see the code following client in [4](#).

3. Configure authentication information.

Configure **ak**, **sk**, and **project_id**. AK is used together with SK to sign requests cryptographically, ensuring that the requests are secret, complete, and correct. There are two authentication methods:

- Initialize authentication information.

```
ak = os.environ.get("HUAWEICLOUD_SDK_AK")
sk = os.environ.get("HUAWEICLOUD_SDK_SK")
```

NOTE

- Hard-coded or plaintext AK and SK are risky. For security purposes, encrypt your AK and SK and store them in the configuration file or environment variables.
- In this example, the AK and SK are stored in environment variables for identity authentication. Before running this example, configure environment variables **HUAWEICLOUD_SDK_AK** and **HUAWEICLOUD_SDK_SK**.

- Using a permanent AK and SK

```
credentials = BasicCredentials(ak, sk, project_id)
```

- Using a temporary AK and SK

```
credentials = BasicCredentials(ak, sk, project_id).with_security_token(security_token)
```

The authentication parameters are described as follows:

- **ak** and **sk**: access key and secret access key. For details about how to obtain them, see [Prerequisites](#).
- **project_id**: Huawei Cloud project ID. For details about how to obtain the ID, see [Prerequisites](#).
- **security_token**: security token used for temporary AK/SK authentication. It can be obtained through [a token](#) or [an agency](#).

4. Initialize the client (using either of the following methods).

- (Recommended) Specifying a region

```
# Add the region dependency.
from huaweicloudsdkocr.v1.region.ocr_region import OcrRegion

# Initialize the client {Service}Client of a specified cloud service. The following uses OcrClient of
OCR as an example.
client = OcrClient.new_builder() \
    .with_http_config(config) \
    .with_credentials(credentials) \
    .with_region(OcrRegion.value_of("ap-southeast-2")) \
    .build()
```

- Specifying an endpoint for a cloud service

```
# Specify the endpoint for OCR, for example, AP-Bangkok.
endpoint = "https://ocr.ap-southeast-2.myhuaweicloud.com"

# Initialize the client {Service}Client of a specified cloud service. The following uses OcrClient of
OCR as an example.
client = OcrClient.new_builder() \
    .with_http_config(config) \
    .with_credentials(credentials) \
    .with_endpoint(endpoint) \
    .build()
```

endpoint indicates the endpoints for Huawei Cloud services. For details, see [Regions and Endpoints](#).

5. Send a request and check the response.

```
# The following uses calling the RecognizePassport API of Passport OCR as an example.
request = RecognizePassportRequest()
```

```
request.body = PassportRequestBody(
    url="Image URL"
)
response = client.recognize_passport(request)
print(response)
```

6. Handle exceptions.

Table 3-1 Exception handling

Level-1 Category	Level-1 Category Description	Level-2 Category	Level-2 Category Description
ConnectionException	Connection exception	HostUnreachableException	The network is unreachable or access is rejected.
		SslHandShakeException	SSL authentication is abnormal.
RequestTimeoutException	Response timeout exception	CallTimeoutException	The server fails to respond to a single request before timeout.
		RetryOutageException	No valid response is returned after the maximum number of retries specified in the retry policy is reached.
ServiceResponseException	Server response exception	ServerResponseException	Internal server error. HTTP response code: [500,].
		ClientRequestException	Invalid request parameter. HTTP response code: [400, 500).

```
# Troubleshooting
try:
    request = RecognizePassportRequest()
    response = client.recognize_passport(request)
    print(response)
except exceptions.ClientResponseException as e:
    print(e.status_code)
    print(e.request_id)
    print(e.error_code)
    print(e.error_msg)
```


 NOTE

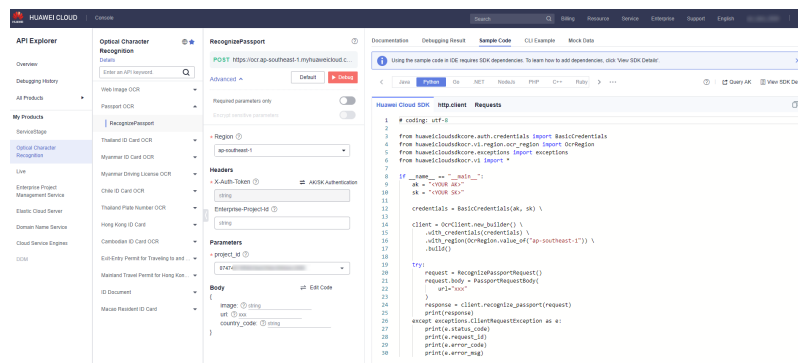
For how to use the asynchronous client and configure logs, see [SDK Center](#) and [Python SDK Usage Guide](#).

Automatic Generation of Sample Code

[API Explorer](#) allows for API search and platform debugging, with features such as quick and comprehensive search, visual debugging, access to help documentation, and online consultation.

You only need to modify API parameters in the API Explorer to automatically generate the corresponding sample code.

Figure 3-3 API Explorer



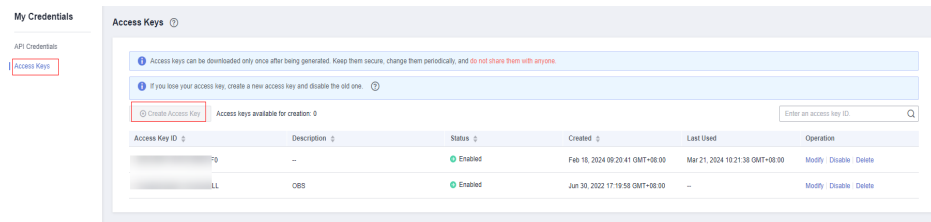
4 Node.js SDK

This section describes how to use the new Node.js SDK to quickly develop OCR services.

Prerequisites

- You have registered a Huawei ID and enabled Huawei Cloud services. Your account cannot be in arrears or frozen.
- Node 10.16.1 or later is available.
- You have obtained an AK and an SK on the [My Credentials](#) > [Access Keys](#) page. The AK and SK are contained in the `credentials.csv` file.

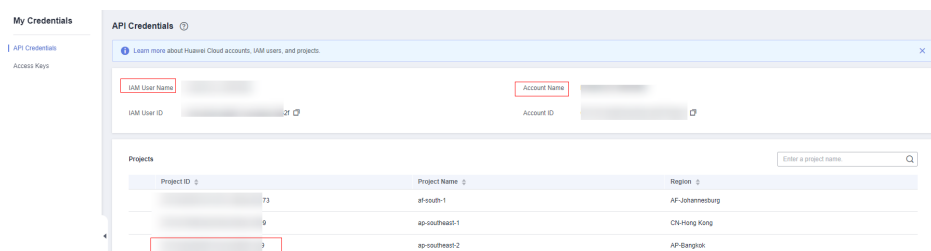
Figure 4-1 Creating an access key



A	B	C	D	E
User Name	Access Key	Secret Access Key		
testuser	LSKM	rIZaQ		
	AK	SK		

- You have obtained the IAM user name, account name, and the project ID of your target region on the [My Credentials](#) > [API Credentials](#) page. The information will be used during service calling. Save it in advance.

Figure 4-2 API Credentials



Installing the SDK

Before you use SDK, install `@huaweicloud/huaweicloud-sdk-core` and `@huaweicloud/huaweicloud-sdk-ocr`.

You are advised to use npm to install the SDK.

```
npm install @huaweicloud/huaweicloud-sdk-core
npm i @huaweicloud/huaweicloud-sdk-ocr
```

Getting Started

1. Import dependency modules.

```
const core = require('@huaweicloud/huaweicloud-sdk-core');
const ocr = require("@huaweicloud/huaweicloud-sdk-ocr");
```

2. Configure client connection parameters.

- Using the default configuration

```
const client = ocr.OcrClient.newBuilder()
```

- (Optional) Configuring a network proxy

```
// (Optional) Use a proxy server.
```

```
client.withProxyAgent("http://username:password@proxy.huaweicloud.com:8080")
```

- (Optional) Configuring an SSL

```
// (Optional) Skip server certificate verification.
```

```
process.env.NODE_TLS_REJECT_UNAUTHORIZED = "0"
```

3. Configure authentication information.

Configure **ak**, **sk**, and **project_id**. AK is used together with SK to sign requests cryptographically, ensuring that the requests are secret, complete, and correct.

Initialize authentication information.

```
const ak = process.env.HUAWEICLOUD_SDK_AK;
const sk = process.env.HUAWEICLOUD_SDK_SK;
const credentials = new core.BasicCredentials()
    .withAk(ak)
    .withSk(sk)
    .withProjectId(project_id)
```

NOTE

- Hard-coded or plaintext AK and SK are risky. For security purposes, encrypt your AK and SK and store them in the configuration file or environment variables.
- In this example, the AK and SK are stored in environment variables for identity authentication. Before running this example, configure environment variables **HUAWEICLOUD_SDK_AK** and **HUAWEICLOUD_SDK_SK**.

The authentication parameters are described as follows:

- **ak** and **sk**: access key and secret access key, respectively. For how to obtain them, see [Prerequisites](#).
- **project_id**: Huawei Cloud project ID. For details about how to obtain the ID, see [Prerequisites](#).

4. Initialize the client.

Specifying an endpoint for a cloud service

```
// Specify the endpoint for OCR, for example, CN North-Beijing4.
```

```
const client = ocr.OcrClient.newBuilder()
    .withCredential(credentials)
    .withEndpoint(endpoint)
    .build();
```

endpoint indicates the endpoints for Huawei Cloud services. For details, see [Regions and Endpoints](#).

- Send a request and check the response.
// The following uses calling the RecognizePassport API of Passport OCR as an example.

```
const request = new ocr.RecognizePassportRequest();
const body = new ocr.PassportRequestBody();
body.withUrl("Image URL");
request.withBody(body);
const result = client.recognizePassport(request);
result.then(result => {
  console.log("JSON.stringify(result)::" + JSON.stringify(result));
}).catch(ex => {
  console.log("exception:" + JSON.stringify(ex));
});
```

 **NOTE**

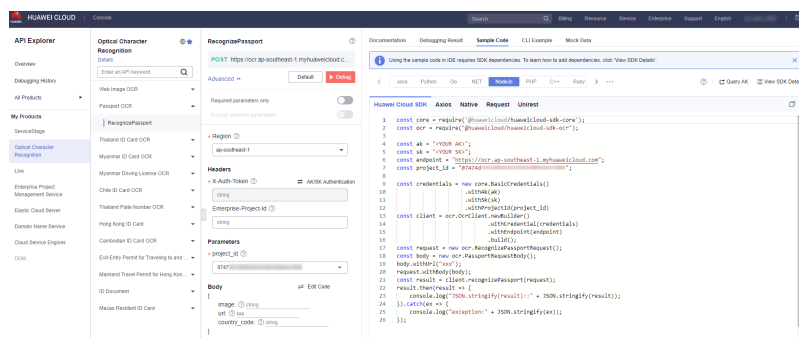
For details about the SDK, see [SDK Center](#) and [Node.js SDK Usage Guide](#).

Automatic Generation of Sample Code

API Explorer allows for API search and platform debugging, with features such as quick and comprehensive search, visual debugging, access to help documentation, and online consultation.

You only need to modify API parameters in the API Explorer to automatically generate the corresponding sample code.

Figure 4-3 API Explorer



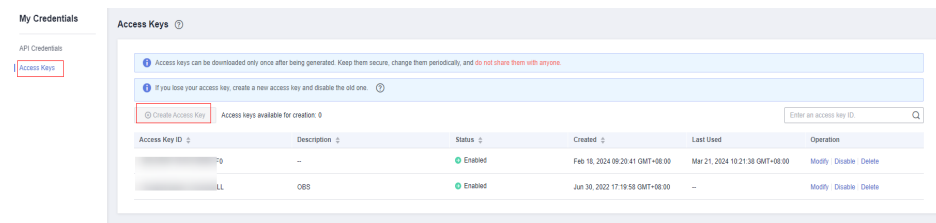
5 PHP SDK

This section describes how to use the new PHP SDK to quickly develop OCR services.

Prerequisites

- You have registered a Huawei ID and enabled Huawei Cloud services. Your account cannot be in arrears or frozen.
- PHP 5.6 or later is available. You can run the `php --version` command to check the version information.
- You have obtained an AK and an SK on the [My Credentials > Access Keys](#) page. The AK and SK are contained in the `credentials.csv` file.

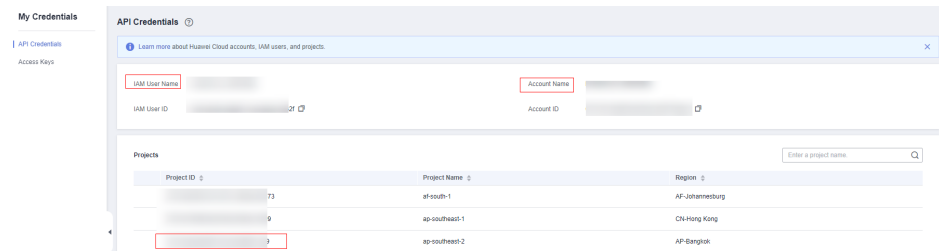
Figure 5-1 Creating an access key



A	B	C	D	E
User Name	Access Key	Secret	Access Key	
testuser	LSKM	rIZaQ		
	AK	SK		

- You have obtained the IAM user name, account name, and the project ID of your target region on the [My Credentials > API Credentials](#) page. The information will be used during service calling. Save it in advance.

Figure 5-2 API Credentials



Installing the SDK

You are advised to use [Composer](#) to install the SDK.

Composer is a dependency manager for PHP. It allows you to declare and install dependencies in a project.

```
// Install Composer.
curl -sS https://getcomposer.org/installer | php
// Install the PHP SDK.
composer require huaweicloud/huaweicloud-sdk-php
```

After the installation is complete, you need to import the autoload file of Composer.

```
require 'path/to/vendor/autoload.php';
```

Getting Started

1. Import dependency modules.

```
<?php
namespace HuaweiCloud\SDK\Ocr\V1\Model;
require_once "vendor/autoload.php";
use HuaweiCloud\SDK\Core\Auth\BasicCredentials;
use HuaweiCloud\SDK\Core\Http\HttpConfig;
use HuaweiCloud\SDK\Core\Exceptions\ConnectionException;
use HuaweiCloud\SDK\Core\Exceptions\RequestTimeoutException;
use HuaweiCloud\SDK\Core\Exceptions\ServiceResponseException;
use HuaweiCloud\SDK\Ocr\V1\OcrClient;
```

2. Configure client connection parameters.

– Using the default configuration

```
// Use the default configuration.
$config = HttpConfig::getDefaultConfig();
```

– (Optional) Configuring a network proxy

```
// Use a proxy server.
$config->setProxyProtocol('http');
$config->setProxyHost('proxy.huawei.com');
$config->setProxyPort(8080);
$config->setProxyUser('username');
$config->setProxyPassword('password');
```

– (Optional) Configuring the timeout

```
// The default connection timeout interval is 60 seconds, and the default read timeout interval
is 120 seconds. You can change the default values as needed.
$config->setTimeout(120);
$config->setConnectionTimeout(60);
```

– (Optional) Configuring an SSL

```
// Skip server certificate verification.
$config->setIgnoreSslVerification(true);
// Configure the server CA certificate so that the SDK can verify the server certificate.
$config->setCertFile("{yourCertFile}");
```

3. Configure authentication information.

Configure **ak**, **sk**, and **projectId**. AK is used together with SK to sign requests cryptographically, ensuring that the requests are secret, complete, and correct.

```
// Specify the endpoint for OCR, for example, AP-Bangkok.
$ak = getenv('HUAWEICLOUD_SDK_AK');
$sk = getenv('HUAWEICLOUD_SDK_SK');
$endpoint = "https://ocr.ap-southeast-2.myhuaweicloud.com";
$projectId = getenv('PROJECT_ID');
$credentials = new BasicCredentials($ak,$sk,$projectId);
```

NOTE

- Hard-coded or plaintext AK and SK are risky. For security purposes, encrypt your AK and SK and store them in the configuration file or environment variables.
- In this example, the AK and SK are stored in environment variables for identity authentication. Before running this example, configure environment variables **HUAWEICLOUD_SDK_AK** and **HUAWEICLOUD_SDK_SK**.

The authentication parameters are described as follows:

- **ak** and **sk**: access key and secret access key, respectively. For how to obtain them, see [Prerequisites](#).
- **projectId**: Huawei Cloud project ID. For details about how to obtain the ID, see [Prerequisites](#).
- **endpoint**: endpoints for Huawei Cloud services. For details, see [Regions and Endpoints](#).

4. Initialize the client.

Specifying an endpoint for a cloud service

```
$client = OcrClient::newBuilder(new OcrClient)
->withHttpConfig($config)
->withEndpoint($endpoint)
->withCredentials($credentials)
->build();
```

5. Send a request and check the response.

// The following uses calling the RecognizePassport API of Passport OCR as an example.

```
$request = new RecognizePassportRequest();
$body = new PassportRequestBody();
$body->setUrl("Image URL");
$request->setBody($body);
try {
    $response = $client->RecognizePassport($request);
} catch (ConnectionException $e) {
    $msg = $e->getMessage();
    echo "\n". $msg . "\n";
} catch (RequestTimeoutException $e) {
    $msg = $e->getMessage();
    echo "\n". $msg . "\n";
} catch (ServiceResponseException $e) {
    echo "\n";
    echo $e->getHttpStatusCode(). "\n";
    echo $e->getErrorCode() . "\n";
    echo $e->getErrMsg() . "\n";
}
echo "\n";
echo $response;
```

6. Handle the exception.

Table 5-1 Exception handling

Level-1 Category	Level-1 Category Description	Level-2 Category	Level-2 Category Description
ConnectionException	Connection exception	HostUnreachableException	The network is unreachable or access is rejected.
		SslHandShakeException	SSL authentication is abnormal.
RequestTimeoutException	Response timeout exception	CallTimeoutException	The server fails to respond to a single request before timeout.
		RetryOutageException	No valid response is returned after the maximum number of retries specified in the retry policy is reached.
ServiceResponseException	Server response exception	ServerResponseException	Internal server error. HTTP response code: [500,].
		ClientRequestException	Invalid request parameter. HTTP response code: [400, 500).

```
try {
    $response = $client->RecognizePassport($request);
} catch (ConnectionException $e) {
    $msg = $e->getMessage();
    echo "\n". $msg ."\n";
} catch (RequestTimeoutException $e) {
    $msg = $e->getMessage();
    echo "\n". $msg ."\n";
} catch (ServiceResponseException $e) {
    echo "\n";
    echo $e->getHttpStatusCode(). "\n";
    echo $e->getErrorCode() . "\n";
    echo $e->getErrMsg() . "\n";
}
echo "\n";
echo $response;
```

 **NOTE**

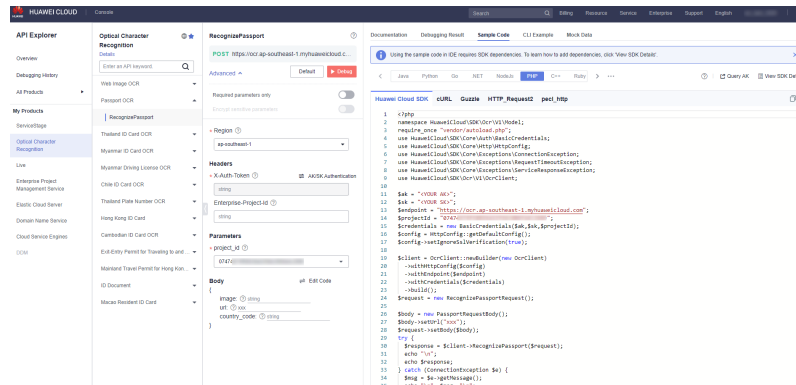
For how to use the asynchronous client and configure logs, see [SDK Center](#) and [PHP SDK Usage Guide](#).

Automatic Generation of Sample Code

API Explorer allows for API search and platform debugging, with features such as quick and comprehensive search, visual debugging, access to help documentation, and online consultation.

You only need to modify API parameters in the API Explorer to automatically generate the corresponding sample code.

Figure 5-3 API Explorer



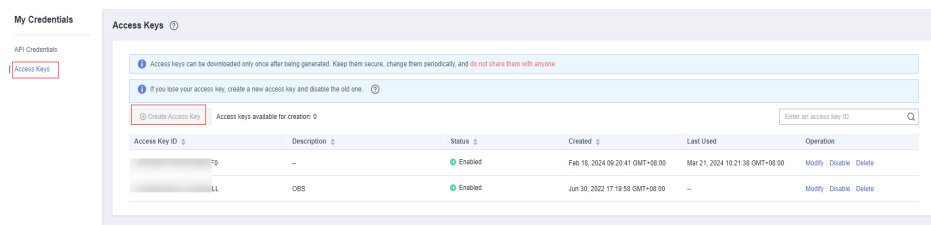
6 C++ SDK

This section describes how to use the new C++ SDK to quickly develop OCR services.

Prerequisites

- You have registered a Huawei ID and enabled Huawei Cloud services. Your account cannot be in arrears or frozen.
- C++ 14 or later and CMake 3.10 or later are available.
- You have obtained an AK and an SK on the [My Credentials](#) > **Access Keys** page. The AK and SK are contained in the **credentials.csv** file.

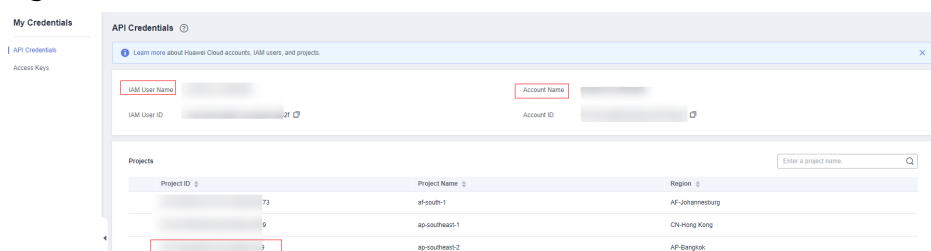
Figure 6-1 Creating an access key



A	B	C	D	E
User Name	Access Key	Secret	Access Key	
testuser	LSKM	rIZaQ		
	AK	SK		

- You have obtained the IAM user name, account name, and the project ID of your target region on the [My Credentials](#) > **API Credentials** page. The information will be used during service calling. Save it in advance.

Figure 6-2 API Credentials



Installing the SDK

- Installing the SDK on a Linux VM
 - a. Obtain dependency packages.

The required third-party software packages are contained in the package management tools of most Linux OSs, for example, Debian and Ubuntu.

```
sudo apt-get install libcurl4-openssl-dev libboost-all-dev libssl-dev libcpr-dev
```

`spdlog` needs to be installed using the source code.

```
git clone https://github.com/gabime/spdlog.git
cd spdlog
mkdir build
cd build
cmake -DCMAKE_POSITION_INDEPENDENT_CODE=ON .. // Used to generate a dynamic library.
make
sudo make install
```
 - b. Perform compilation and installation.

```
git clone https://github.com/huaweicloud/huaweicloud-sdk-cpp-v3.git
cd huaweicloud-sdk-cpp-v3
mkdir build
cd build
cmake ..
make
sudo make install
```

After the operations are complete, the installation directory of the C++ SDK is **/usr/local**.
- Installing the SDK on a Windows VM
 - a. Install `vcpkg` and use it to install required software packages.

```
vcpkg install curl cprestdk boost openssl spdlog
```
 - b. Use CLion for compilation.
 - i. Use CLion to open the **huaweicloud-sdk-cpp-v3** directory.
 - ii. Choose **File > Settings**.
 - iii. Choose **Build > Execution > Deployment > CMake**.
 - iv. Add the following content to CMake options:

```
-DCMAKE_TOOLCHAIN_FILE={your vcpkg install dir}/scripts/buildsystems/vcpkg.cmake
```
 - v. Right-click **CMakeLists.txt** and choose **Load CMake Project** from the shortcut menu.
 - vi. Choose **Build** to start compilation.
 - c. Install the C++ SDK.

After the compilation is complete, choose **Build > Install**.

After the operations are complete, the installation directory of the C++ SDK is **C:\Program File (x86)\huaweicloud-sdk-cpp-v3**.

Getting Started

1. Import dependency modules.

```
//include <cstdlib>
//include <iostream>
//include <string>
//include <memory>
//include <huaweicloud/core/exception/Exceptions.h>
//include <huaweicloud/core/Client.h>
//include <huaweicloud/ocr/v1/OcrClient.h>
using namespace HuaweiCloud::Sdk::Ocr::V1;
```

```
using namespace HuaweiCloud::Sdk::Ocr::V1::Model;
using namespace HuaweiCloud::Sdk::Core;
using namespace HuaweiCloud::Sdk::Core::Exception;
using namespace std;
```

2. Configure client connection parameters.

– Using the default configuration

```
// Use the default configuration.
HttpConfig httpConfig = HttpConfig();
```

– (Optional) Configuring a network proxy

```
// Configure network proxy as needed.
httpConfig.setProxyProtocol("http");
httpConfig.setProxyHost("proxy.huawei.com");
httpConfig.setProxyPort("8080");
httpConfig.setProxyUser("username");
httpConfig.setProxyPassword("password");
```

– (Optional) Configuring the timeout

```
// The default connection timeout interval is 60 seconds, and the default read timeout interval
is 120 seconds. You can change the default values as needed.
httpConfig.setConnectTimeout(60);
httpConfig.setReadTimeout(120);
```

– (Optional) Configuring an SSL

```
// Skip server certificate verification.
httpConfig.setIgnoreSslVerification(true);
```

3. Configure authentication information.

Configure **ak**, **sk**, and **projectId**. AK is used together with SK to sign requests cryptographically, ensuring that the requests are secret, complete, and correct.

```
string ak = getenv("HUAWEICLOUD_SDK_AK");
string sk = getenv("HUAWEICLOUD_SDK_SK");
string projectId = getenv("PROJECT_ID");
auto auth = std::make_unique<BasicCredentials>();
auth->withAk(ak)
    .withSk(sk)
    .withProjectId(projectId);
```

NOTE

- Hard-coded or plaintext AK and SK are risky. For security purposes, encrypt your AK and SK and store them in the configuration file or environment variables.
- In this example, the AK and SK are stored in environment variables for identity authentication. Before running this example, configure environment variables **HUAWEICLOUD_SDK_AK** and **HUAWEICLOUD_SDK_SK**.

The authentication parameters are described as follows:

- **ak** and **sk**: access key and secret access key, respectively. For how to obtain them, see [Prerequisites](#).
- **projectId**: Huawei Cloud project ID. For details about how to obtain the ID, see [Prerequisites](#).

4. Initialize the client.

Specifying an endpoint for a cloud service

```
string endpoint = "https://ocr.cn-north-4.myhuaweicloud.com";
auto client = OcrClient::newBuilder()
    .withCredentials(std::unique_ptr<Credentials>(auth.release()))
    .withHttpConfig(httpConfig)
    .withEndPoint(endpoint)
    .build();
```

endpoint: endpoints for Huawei Cloud services. For details, see [Regions and Endpoints](#).

5. Send a request and check the response.

```
// The following uses calling the RecognizePassport API of Passport OCR as an example.
RecognizePassportRequest request;
PassportRequestBody body;
body.setUrl("Image URL");
request.setBody(body);
```

```
std::cout << "-----begin execute request-----" << std::endl;
```

6. Handle the exception.

Table 6-1 Exception handling

Level-1 Category	Level-1 Category Description	Level-2 Category	Level-2 Category Description
ConnectionException	Connection exception	HostUnreachableException	The network is unreachable or access is rejected.
		SslHandShakeException	SSL authentication is abnormal.
RequestTimeoutException	Response timeout exception	CallTimeoutException	The server fails to respond to a single request before timeout.
		RetryOutageException	No valid response is returned after the maximum number of retries specified in the retry policy is reached.
ServiceResponseException	Server response exception	ServerResponseException	Internal server error. HTTP response code: [500,].
		ClientRequestException	Invalid request parameter. HTTP response code: [400, 500).

```
std::cout << "-----begin execute request-----" << std::endl;
try {
    auto reponse = client->recognizePassport(request);
    std::cout << reponse->getHttpBody() << std::endl;
} catch (HostUnreachableException& e) {
    std::cout << "host unreachable:" << e.what() << std::endl;
} catch (SslHandShakeException& e) {
    std::cout << "ssl handshake error:" << e.what() << std::endl;
} catch (RetryOutageException& e) {
    std::cout << "retryoutage error:" << e.what() << std::endl;
} catch (CallTimeoutException& e) {
    std::cout << "call timeout:" << e.what() << std::endl;
} catch (ServiceResponseException& e) {
```

```

std::cout << "http status code:" << e.getStatusCode() << std::endl;
std::cout << "error code:" << e.getErrorCode() << std::endl;
std::cout << "error msg:" << e.getErrorMsg() << std::endl;
std::cout << "RequestId:" << e.getRequestId() << std::endl;
} catch (exception& e) {
    std::cout << "unknown exception:" << e.what() << std::endl;
}
    
```

NOTE

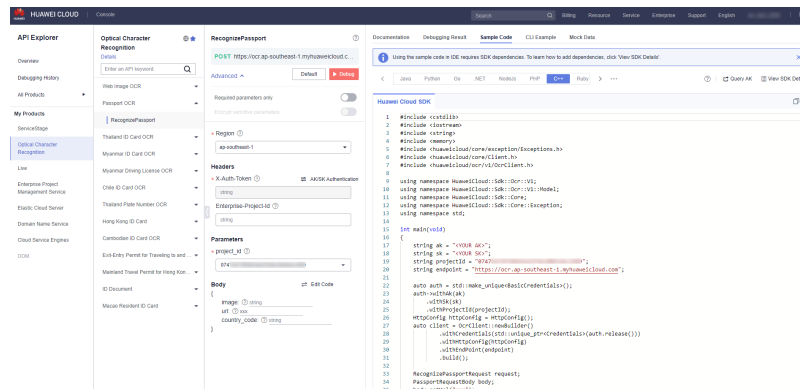
For details about how to use the asynchronous client and configure logs, see [SDK Center](#) and [C++ SDK Usage Guide](#).

Automatic Generation of Sample Code

API Explorer allows for API search and platform debugging, with features such as quick and comprehensive search, visual debugging, access to help documentation, and online consultation.

You only need to modify API parameters in the API Explorer to automatically generate the corresponding sample code.

Figure 6-3 API Explorer



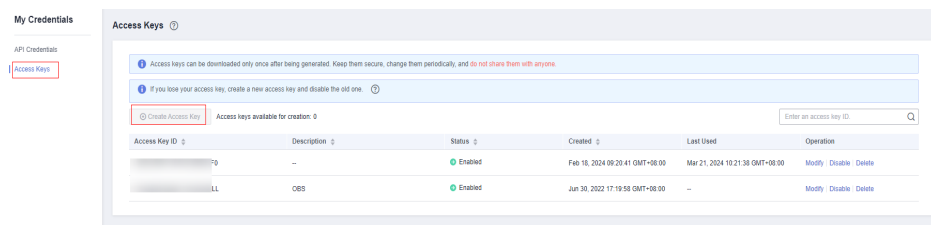
7 Go SDK

This section describes how to use the Go SDK to quickly develop OCR services.

Prerequisites

- You have registered a Huawei ID and enabled Huawei Cloud services. Your account cannot be in arrears or frozen.
- Go 1.14 or later is available. You can run the **go version** command to check the version information.
- You have obtained an AK and an SK on the [My Credentials](#) > **Access Keys** page. The AK and SK are contained in the **credentials.csv** file.

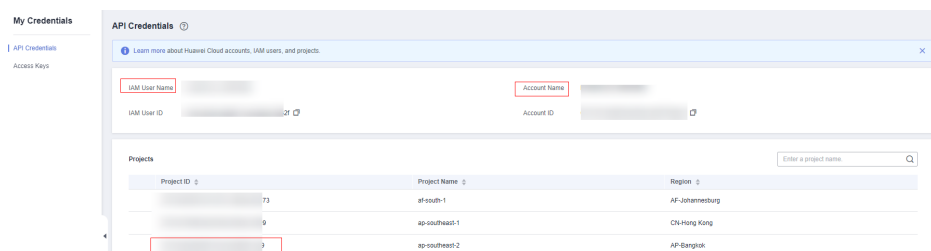
Figure 7-1 Creating an access key



A	B	C	D	E
User Name	Access Key	Secret	Access Key	
testuser	LSKM	rIZaQ		
	AK	SK		

- You have obtained the IAM user name, account name, and the project ID of your target region on the [My Credentials](#) > **API Credentials** page. The information will be used during service calling. Save it in advance.

Figure 7-2 API Credentials



Installing the SDK

Before you use the SDK, you need to install the Huawei Cloud Go SDK library.

```
// Install the Huawei Cloud Go SDK library.  
go get github.com/huaweicloud/huaweicloud-sdk-go-v3
```

Getting Started

1. Import dependency modules.

```
import (  
    "fmt"  
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/core/auth/basic"  
    ocr "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1"  
    "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/model"  
    region "github.com/huaweicloud/huaweicloud-sdk-go-v3/services/ocr/v1/region"  
)
```

2. Configure authentication information.

Configure **ak** and **sk**. AK is used together with SK to sign requests cryptographically, ensuring that the requests are secret, complete, and correct. For details about how to obtain the AK and SK, see [Prerequisites](#).

```
func main() {  
    ak := os.Getenv("HUAWEICLOUD_SDK_AK")  
    sk := os.Getenv("HUAWEICLOUD_SDK_SK")  
    auth := basic.NewCredentialsBuilder().  
        WithAk(ak).  
        WithSk(sk).  
        Build()  
}
```

NOTE

- Hard-coded or plaintext AK and SK are risky. For security purposes, encrypt your AK and SK and store them in the configuration file or environment variables.
 - In this example, the AK and SK are stored in environment variables for identity authentication. Before running this example, configure environment variables **HUAWEICLOUD_SDK_AK** and **HUAWEICLOUD_SDK_SK**.
3. Initialize the client (using either of the following methods).

- (Recommended) Specifying a region for a cloud service

// Initialize the client New{Service}Client of a specified cloud service. The following uses the **AP-Bangkok** region as an example.

```
func main() {  
    client := ocr.NewOcrClient(  
        ocr.OcrClientBuilder().  
            WithRegion(region.ValueOf("ap-southeast-2")).  
            WithCredential(auth).  
            Build()  
    )  
}
```

- Specifying an endpoint for a cloud service

```
func main() {  
    // Specify the endpoint for OCR, for example, AP-Bangkok.  
    endpoint := "https://ocr.ap-southeast-2.myhuaweicloud.com"  
    // If you choose to specify the endpoint, add projectId to the authentication information.  
    auth := basic.NewCredentialsBuilder().  
        WithAk(ak).  
        WithSk(sk).  
        WithProjectId(projectId).  
        Build()  
    // Initialize the client New{Service}Client of a specified cloud service.  
    client := ocr.NewOcrClient(  
        ocr.OcrClientBuilder().  
            WithEndpoint(endpoint).  
    )  
}
```



```
WithCredential(auth).
    Build()
}
```

endpoint indicates the endpoints for Huawei Cloud services. For details, see [Regions and Endpoints](#).

4. Send a request and check the response.

```
// The following uses calling the RecognizePassport API of Passport OCR as an example.
request := &model.RecognizePassportRequest{}
urlPassportRequestBody:= "Image URL"
request.Body = &model.PassportRequestBody{
    Url: &urlPassportRequestBody,
}
response, err := client.RecognizePassport(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
```

5. Handle the exception.

Table 7-1 Exception handling

Level-1 Category	Level-1 Category Description
ServiceResponseError	A service response exception occurs.
url.Error	A URL exception occurs.

```
response, err := client.RecognizePassport(request)
if err == nil {
    fmt.Printf("%+v\n", response)
} else {
    fmt.Println(err)
}
```

 **NOTE**

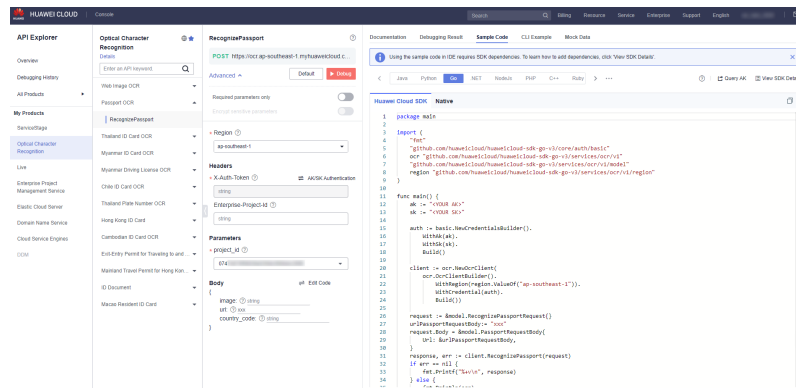
For the SDK, see [SDK Center](#) and [Go SDK Usage Guide](#).

Automatic Generation of Sample Code

[API Explorer](#) allows for API search and platform debugging, with features such as quick and comprehensive search, visual debugging, access to help documentation, and online consultation.

You only need to modify API parameters in the API Explorer to automatically generate the corresponding sample code.

Figure 7-3 API Explorer



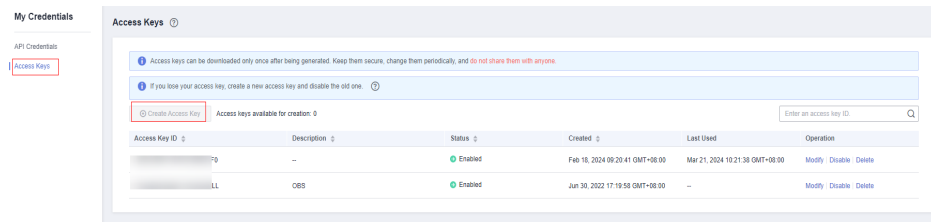
8 .NET SDK

This section describes how to use the .NET SDK to quickly develop OCR services as needed.

Prerequisites

- You have registered a Huawei ID and enabled Huawei Cloud services. Your account cannot be in arrears or frozen.
- .NET Standard 2.0 or later and C# 4.0 or later are available.
- You have obtained an AK and an SK on the [My Credentials > Access Keys](#) page. The AK and SK are contained in the **credentials.csv** file.

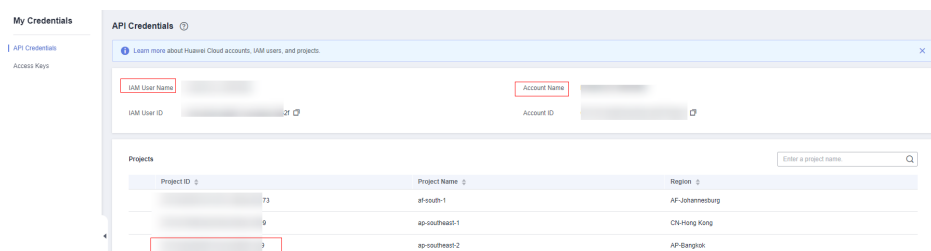
Figure 8-1 Creating an access key



A	B	C	D	E
User Name	Access Key	Secret Access Key		
testuser	LSKM	rIZaQ		
	AK	SK		

- You have obtained the IAM user name, account name, and the project ID of your target region on the [My Credentials > API Credentials](#) page. The information will be used during service calling. Save it in advance.

Figure 8-2 API Credentials



Installing the SDK

Before you use the SDK, install **HuaweiCloud.SDK.Core** and **HuaweiCloud.SDK.Ocr** in either of the following ways:

- Using .NET CLI

```
dotnet add package HuaweiCloud.SDK.Core
dotnet add package HuaweiCloud.SDK.Ocr
```
- Using Package Manager

```
Install-Package HuaweiCloud.SDK.Core
Install-Package HuaweiCloud.SDK.Ocr
```

Getting Started

1. Import dependency modules.

```
using System;
using System.Collections.Generic;
using HuaweiCloud.SDK.Core;
using HuaweiCloud.SDK.Core.Auth;
using HuaweiCloud.SDK.Ocr;
using HuaweiCloud.SDK.Ocr.V1;
using HuaweiCloud.SDK.Ocr.V1.Model;
```

2. Configure client connection parameters.

– Using the default configuration

```
// Use the default configuration.
var config = HttpConfig.GetDefaultConfig();
```

– (Optional) Configuring a network proxy

```
// Configure network proxy as needed.
config.ProxyHost = "proxy.huaweicloud.com";
config.ProxyPort = 8080;
config.ProxyUsername = "test";
config.ProxyPassword = "test";
```

– (Optional) Configuring the timeout

```
// The default connection timeout interval is 120 seconds. You can change it as needed.
config.Timeout = 120;
```

– (Optional) Configuring an SSL

```
// Configure whether to skip SSL certificate verification as needed.
config.IgnoreSslVerification = true;
```

3. Configure authentication information.

Configure **ak** and **sk**. AK is used together with SK to sign requests cryptographically, ensuring that the requests are secret, complete, and correct. For details about how to obtain the AK and SK, see [Prerequisites](#).

```
const string ak = Environment.GetEnvironmentVariable("HUAWEICLOUD_SDK_AK");
const string sk = Environment.GetEnvironmentVariable("HUAWEICLOUD_SDK_SK");
var auth = new BasicCredentials(ak, sk);
```

NOTE

- Hard-coded or plaintext AK and SK are risky. For security purposes, encrypt your AK and SK and store them in the configuration file or environment variables.
 - In this example, the AK and SK are stored in environment variables for identity authentication. Before running this example, configure environment variables **HUAWEICLOUD_SDK_AK** and **HUAWEICLOUD_SDK_SK**.
- ### 4. Initialize the client (using either of the following methods).
- (Recommended) Specifying a region for a cloud service

```
// Initialize the client {Service}Client of a specified cloud service. The following uses OcrClient of OCR as an example.
var client = OcrClient.NewBuilder()
```

```
.WithCredential(auth)
.WithRegion(OcrRegion.ValueOf("ap-southeast-2"))
.WithHttpConfig(config)
.Build();
```

– Specifying an endpoint for a cloud service

```
// Specify the endpoint for OCR, for example, AP-Bangkok.
String endpoint = "https://ocr.ap-southeast-2.myhuaweicloud.com";

// Initialize the client authentication information. You need to enter the corresponding project
ID. The following uses BasicCredentials as an example.
var auth = new BasicCredentials(ak, sk, projectId);

// Initialize the client {Service}Client of a specified cloud service. The following uses OcrClient of
OCR as an example.
var client = OcrClient.NewBuilder()
    .WithCredential(auth)
    .WithEndPoint(endpoint)
    .WithHttpConfig(config)
    .Build();
```

endpoint indicates the endpoints for Huawei Cloud services. For details, see [Regions and Endpoints](#).

5. Send a request and check the response.

```
// The following uses calling the RecognizePassport API of Passport OCR as an example.
var req = new RecognizePassportRequest
{
};
req.Body = new PassportRequestBody()
{
    Url = "Image URL"
};

try
{
    var resp = client.RecognizePassport(req);
    var respStatusCode = resp.HttpStatusCode;
    Console.WriteLine(respStatusCode);
}
```

6. Handle the exception.

Table 8-1 Exception handling

Level-1 Category	Level-1 Category Description	Level-2 Category	Level-2 Category Description
ConnectionException	Connection exception	HostUnreachableException	The network is unreachable or access is rejected.
		SslHandShakeException	SSL authentication is abnormal.
RequestTimeoutException	Response timeout exception	CallTimeoutException	The server fails to respond to a single request before timeout.

Level-1 Category	Level-1 Category Description	Level-2 Category	Level-2 Category Description
		RetryOutageException	No valid response is returned after the maximum number of retries specified in the retry policy is reached.
ServiceResponseException	Server response exception	ServerResponseException	Internal server error. HTTP response code: [500,].
		ClientRequestException	Invalid request parameter. HTTP response code: [400, 500).

```
try
{
    var resp = client.RecognizePassport(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.ErrorCode);
    Console.WriteLine(clientRequestException.ErrorMsg);
}
catch (ConnectionException connectionException)
{
    Console.WriteLine(connectionException.ErrorMessage);
}
```

 **NOTE**

For details about how to use the asynchronous client and configure logs, see [SDK Center](#) and [.NET SDK Usage Guide](#).

Automatic Generation of Sample Code

[API Explorer](#) allows for API search and platform debugging, with features such as quick and comprehensive search, visual debugging, access to help documentation, and online consultation.

You only need to modify API parameters in the API Explorer to automatically generate the corresponding sample code.

Figure 8-3 API Explorer

