CodeArts Check

API Reference

Issue	01
Date	2024-12-02

HUAWEI TECHNOLOGIES CO., LTD.

Trademarks and Permissions

Notice

The purchased products, services and features are stipulated by the contract made between Huawei 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 Technologies Co., Ltd.

Address: Huawei Industrial Base
Bantian, Longgang
Shenzhen 518129
People's Republic of China

Website: https://www.huawei.com

Email: support@huawei.com

Security Declaration

Vulnerability

Huawei's regulations on product vulnerability management are subject to the *Vul. Response Process*. For details about this process, visit the following web page:
https://www.huawei.com/en/psirt/vul-response-process
For vulnerability information, enterprise customers can visit the following web page:
https://securitybulletin.huawei.com/enterprise/en/security-advisory

Contents

1 Before You Start

CodeArts Check manages code quality in the cloud. It allows you to easily perform static checks and security checks on code in multiple programming languages and obtain comprehensive quality reports. CodeArts Check also allows you to view grouped defects with fix suggestions provided, effectively controlling quality.

This document provides application programming interfaces (APIs) for you to perform CodeArts Check operations, such as creating, deleting, and querying tasks. For details about all supported operations, see **API Overview**.

Before calling CodeArts Check APIs, ensure that you are familiar with **basic concepts**.

Endpoints

An endpoint is the request address for calling an API. Endpoints vary with services and regions. For the endpoints of all services, see **Regions and Endpoints**.

Table 1-1 lists the CodeArts Check endpoints. Select a desired one based on the service requirements.

Table 1-1 CodeArts Check endpoints

Region	Region ID	Endpoint
LA-Mexico City2	la-north-2	devcloud.la-north-2.myhuaweicloud.com
LA-Sao Paulo1	sa-brazil-1	devcloud.sa-brazil-1.myhuaweicloud.com
AP-Singapore	ap-southeast-3	devcloud.ap-southeast-3.myhuaweicloud.com

Basic Concepts

- Account

  An account is created upon successful registration with Huawei Cloud. The account has full access permissions for all of its cloud services and resources.

It can be used to reset user passwords and grant user permissions. The account is a payment entity and should not be used directly to perform routine management. For security purposes, create IAM users and grant them permissions for routine management.

- User

  An Identity and Access Management (IAM) user is created using an account to use cloud services. Each IAM user has its own identity credentials (password and access keys).

  An IAM user can view the account ID and user ID on the **My Credentials** page of the console. The domain name, username, and password will be required for API authentication.

- Region

  Regions are divided from the dimensions of geographical location and network latency. Public services, such as Elastic Cloud Server (ECS), Elastic Volume Service (EVS), Object Storage Service (OBS), Virtual Private Cloud (VPC), Elastic IP (EIP), and Image Management Service (IMS), are shared within the same region. Regions are classified into universal regions and dedicated regions.

  - A universal region provides universal cloud services for common tenants.
  - A dedicated region provides particular services for specific tenants.

- Availability Zone (AZ)

  An AZ contains one or more physical data centers. Each AZ has independent cooling, fire extinguishing, moisture-proof, and electricity facilities. Within an AZ, compute, network, storage, and other resources are logically divided into multiple clusters. AZs within a region are interconnected using high-speed optical fibers to support cross-AZ high-availability systems.

- Enterprise Project

  An enterprise project groups and manages enterprise resources in a logically isolated manner. It can contain resources in multiple regions, and allow resources to be added or removed.

  For details about how to obtain enterprise project IDs and features, see the **Enterprise Management User Guide**.

2 API Overview

Table 2-1 CodeArts Check API overview

Type	Subtype	Description
CodeArts Check	**4.1 Task Management**	Task management includes creating, executing, querying, and stopping tasks.
	4.2 Issue Management	Defect management includes defect details, such as issue overview, issue status, cyclomatic complexity, and code repetition rate.
	4.3 Rule Management	Rule management includes obtaining the rule list APIs, creating user-defined rule sets, and querying the rule set list.

3 Calling APIs

3.1 Making an API Request

This section describes the structure of a REST API request, and uses the IAM API for **creating an IAM User** as an example to demonstrate how to call an API.

Request URI

Although a request URI is included in the request header, most programming languages or frameworks require the request URI to be passed separately.

A request URI consists of four parts: *{URI-scheme}* **://** *{Endpoint}* **/** *{resource-path}***?** *{query-string}*

The following table describes the parameters.

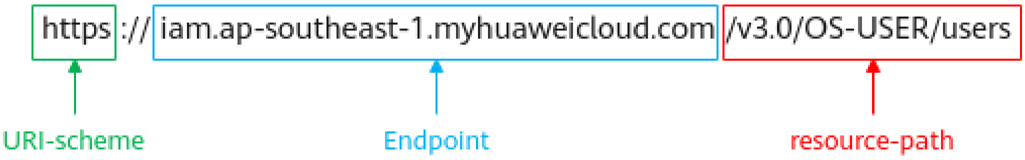
Parameter	Description
URI-scheme	Protocol used to transmit requests. All APIs use **HTTPS**.
Endpoint	Domain name or IP address of the server bearing the REST service. The endpoint varies among services in different regions. It can be obtained from **Regions and Endpoints**. For example, the endpoint of IAM in the **AP-Singapore** region is **iam.ap-southeast-3.myhuaweicloud.com**.
resource-path	Resource path, that is, the API access path, which is obtained from the URI of a specific API. For example, **resource-path** of the API for creating an IAM user is **/v3.0/OS-USER/users**.

Parameter	Description
query-string	(Optional) Query parameter. The query parameter is prefixed with a question mark (**?**), in the format of ***Parameter name*=*Parameter value***. For example, **limit=10** indicates that a maximum of 10 data records will be queried.

For example, if you want to create an IAM user, use the endpoint of any region because IAM is a global service. Obtain the endpoint of the AP-Singapore region (iam.ap-southeast-3.myhuaweicloud.com) and find **resource-path** (**/v3.0/OS-USER/users**) in the URI of the API for **creating an IAM user**. Then, construct them as follows:

```
https://iam.ap-southeast-3.myhuaweicloud.com/v3.0/OS-USER/users
```

Figure 3-1 Example URI



NOTE

To simplify the URI display in this document, each API is provided only with a **resource-path** and a request method. This is because the **URI-scheme** value of all APIs is **HTTPS**, and the endpoints in a region are the same.

Request Methods

HTTP defines the following request methods that can be used to send a request to the server.

- **GET**: requests the server to return specified resources.
- **PUT**: requests the server to update specified resources.
- **POST**: requests the server to add resources or perform special operations.
- **DELETE**: requests the server to delete specified resources, such as an object.
- **HEAD**: same as **GET** except that the server must return only the response header.
- **PATCH**: requests the server to update partial content of a specified resource. If the resource does not exist, a new resource will be created.

For example, in the case of the API for **creating an IAM user**, the request method is **POST**. An example request is as follows:

```
POST https://iam.ap-southeast-3.myhuaweicloud.com/v3.0/OS-USER/users
```

Request Header

You can also add additional header fields to a request, such as fields for specifying a URI and an HTTP method. For example, to request authentication information, add **Content-Type**, which specifies the request body type.

Common request header fields are as follows:

- **Content-Type**: specifies the request body type or format. This field is mandatory and its default value is **application/json**. Other values of this field are provided for specific APIs, if any.
- **Authorization**: specifies signature authentication information. This field is optional. When AK/SK authentication is enabled, this field is automatically specified when SDK is used to sign the request. For more information, see **AK/SK-based Authentication**.
- **X-Sdk-Date**: specifies the time when a request is sent. This field is optional. When AK/SK authentication is enabled, this field is automatically specified when SDK is used to sign the request. For more information, see **AK/SK-based Authentication**.
- **X-Auth-Token**: specifies a user token only for token-based API authentication. The user token is a response to the API used to **obtain a user token**. Only this API does not require authentication.
- **X-Project-ID**: specifies a subproject ID. This parameter is optional. It is used in multi-project scenarios. The **X-Project-ID** field is mandatory in the request header for accessing resources in a sub-project through AK/SK-based authentication.
- **X-Domain-ID**: specifies account ID, which is optional. When you call APIs of global services using AK/SK-based authentication, **X-Domain-ID** needs to be configured in the request header.

The following shows an example request of the API for **creating an IAM user** when AK/SK authentication is used:

```
POST https://iam.ap-southeast-3.myhuaweicloud.com/v3.0/OS-USER/users
Content-Type: application/json
X-Sdk-Date: 20240416T095341Z
Authorization: SDK-HMAC-SHA256 Access=****************, SignedHeaders=content-type;host;x-sdk-date,
Signature=****************
```

Request Body

A request body is often sent in structured format. It corresponds to **Content-Type** in the request header and passes content except the request header.

The request body varies according to APIs. Certain APIs do not require the request body, such as GET and DELETE.

When you call the API for **obtaining a user token**, the request parameters and parameter description can be obtained from the API request. The following provides an example request with a body included. Replace the italic fields with the actual values.

- **accountid** indicates the ID of the account to which the IAM user belongs.
- **username** indicates the IAM username to be created.
- **email** indicates the email address of the IAM user.

- ********** indicates the login password of the IAM user.

```
POST https://iam.ap-southeast-3.myhuaweicloud.com/v3.0/OS-USER/users
Content-Type: application/json
X-Sdk-Date: 20240416T095341Z
Authorization: SDK-HMAC-SHA256 Access=****************, SignedHeaders=content-type;host;x-sdk-date, Signature=****************

{
    "user": {
        "domain_id": "accountid",
        "name": "username",
        "password": "**********",
        "email": "email",
        "description": "IAM user description"
    }
 }
```

If all data required for the API request is available, you can send the request to call the API through curl, Postman, or coding.

3.2 Authentication

Requests for calling an API can be authenticated using either of the following methods:

- Token-based authentication: Requests are authenticated using a token.
- AK/SK-based authentication: Requests are authenticated by encrypting the request body using an AK/SK pair.

Token-based Authentication

NOTE

- The validity period of a token is 24 hours. When using a token for authentication, cache it to prevent frequently calling the IAM API used to obtain a user token.
- Ensure that the token is valid when you use it. Using a token that will soon expire may cause API calling failures.

A token is used to acquire temporary permissions. During API authentication using a token, the token is added to requests to get permissions for calling the API.

When calling the API for **obtaining a user token**, set **auth.scope** in the request body to **project**.

```
{
    "auth": {
        "identity": {
            "methods": [
                "password"
            ],
            "password": {
                "user": {
                    "name": "username",
                    "password": "********",
                    "domain": {
                        "name": "domainname"
                    }
                }
            }
        },
        "scope": {
            "project": {
                "name": "xxxxxxxx"
            }
        }
    }
}
```

After a token is obtained, add field **X-Auth-Token** to the request header to specify the token when other APIs are called. For example, if the token is **ABCDEFJ....**, **X-Auth-Token: ABCDEFJ....** can be added to a request as follows:

```
GET https://iam.ap-southeast-3.myhuaweicloud.com/v3.0/OS-USER/users
Content-Type: application/json
X-Auth-Token: ABCDEFJ....
```

AK/SK-based Authentication

You can use AK/SK to verify the identity of a request sender. In AK/SK authentication, a signature needs to be obtained and then added to requests.

NOTE

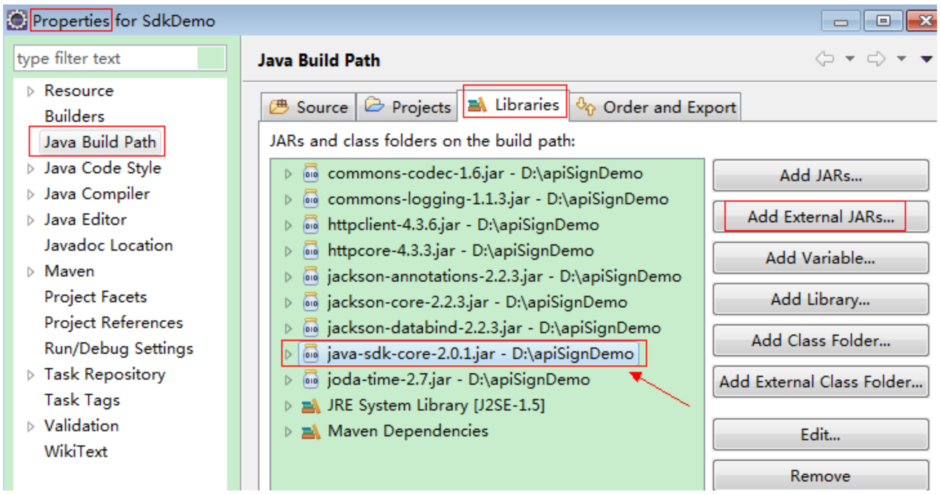
AK: access key ID. It is a unique ID associated with an SK. AK is used together with SK to sign requests.

SK: secret access key. It is used together with an access key ID to identify a sender who initiates a request and to cryptographically sign requests, preventing the request from being modified.

The following uses a demo project to show how to sign a request and use an HTTP client to send an HTTPS request.

Download the demo from **https://github.com/api-gate-way/SdkDemo**.

Decompress the downloaded package to obtain a JAR file. Reference the extracted JAR file to the dependency path, as shown below.



JARs and class folders on the build path:

```
commons-codec-1.6.jar - D:\apiSignDemo
commons-logging-1.1.3.jar - D:\apiSignDemo
httpclient-4.3.6.jar - D:\apiSignDemo
httpcore-4.3.3.jar - D:\apiSignDemo
jackson-annotations-2.2.3.jar - D:\apiSignDemo
jackson-core-2.2.3.jar - D:\apiSignDemo
jackson-databind-2.2.3.jar - D:\apiSignDemo
java-sdk-core-2.0.1.jar - D:\apiSignDemo
joda-time-2.7.jar - D:\apiSignDemo
JRE System Library [J2SE-1.5]
Maven Dependencies
```

Step 1 Create an AK/SK pair. If an AK/SK pair has already been generated, skip this step. Find the downloaded AK/SK file, which is usually named **credentials.csv**.

1. Log in to the management console.
2. Hover over the username and select **My Credentials** from the drop-down list.
3. In the navigation pane, click **Access Keys**.

4. Click **Create Access Key**.
5. Enter your login password.
6. Enter the verification code received by email or SMS message.

   NOTE

   For users created in IAM, if no email address or phone number was specified during the user creation, only a login password is required.

7. Click **OK** to download the access key.

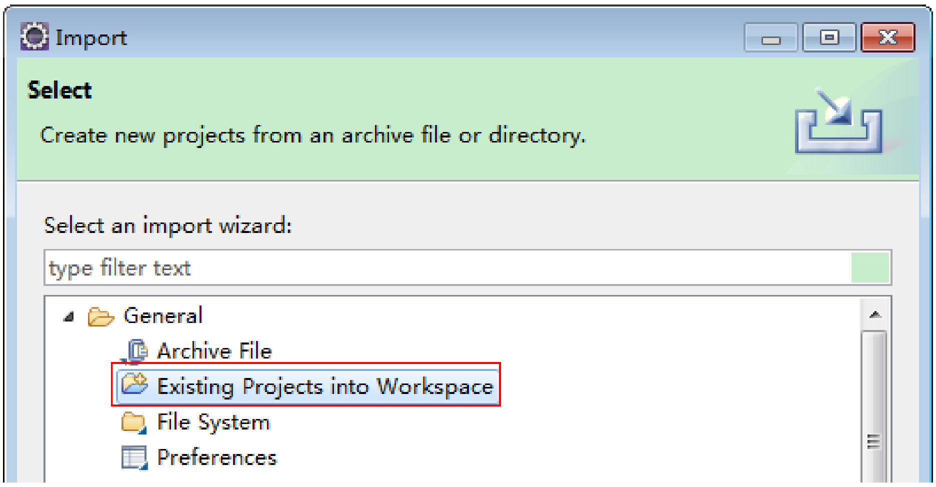
   NOTE

   Keep the key secure.

Step 2 Download and decompress the demo project.

Step 3 Import the demo project to Eclipse.

Figure 3-2 Selecting an existing project



Import

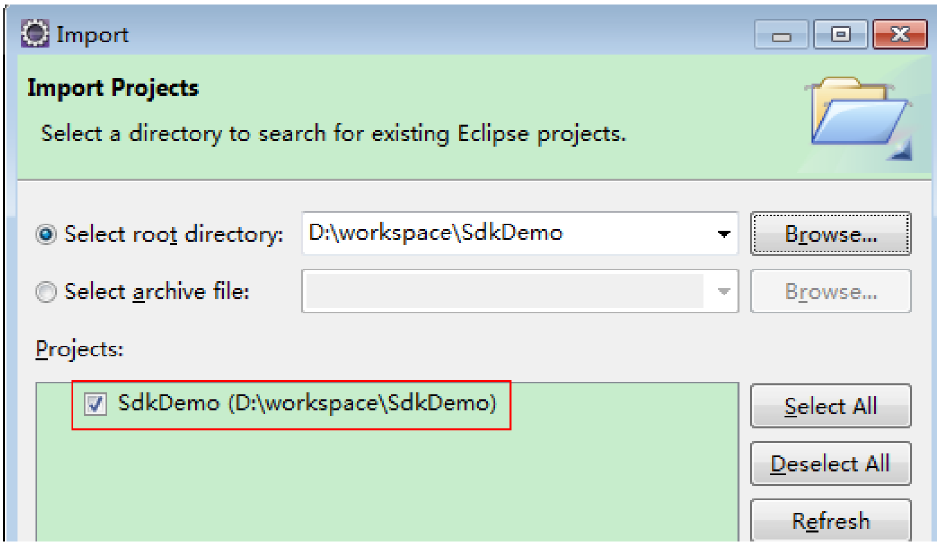
Select

Create new projects from an archive file or directory.

Select an import wizard:

type filter text

Figure 3-3 Selecting the demo project



Import

Import Projects

Select a directory to search for existing Eclipse projects.

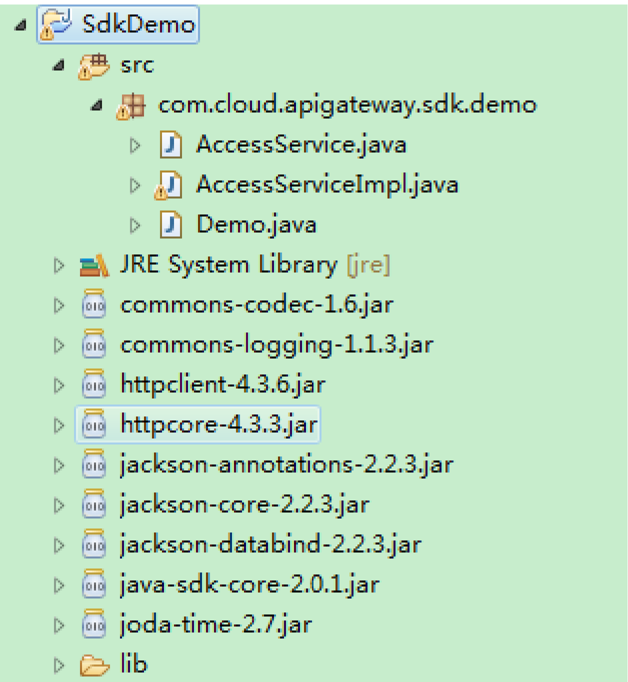
● Select root directory: D:\workspace\SdkDemo

○ Select archive file:

Projects:

☑ SdkDemo (D:\workspace\SdkDemo)

Figure 3-4 Example structure after the demo project is imported



```
SdkDemo
  src
    com.cloud.apigateway.sdk.demo
      AccessService.java
      AccessServiceImpl.java
      Demo.java
  JRE System Library [jre]
  commons-codec-1.6.jar
  commons-logging-1.1.3.jar
  httpclient-4.3.6.jar
  httpcore-4.3.3.jar
  jackson-annotations-2.2.3.jar
  jackson-core-2.2.3.jar
  jackson-databind-2.2.3.jar
  java-sdk-core-2.0.1.jar
  joda-time-2.7.jar
  lib
```

Step 4 Sign the request.

The signature method is integrated into the JAR file imported in **step 3**. Sign a request before sending it. The signature will be added as part of the HTTP header of the request.

The demo code is classified into the following classes to demonstrate:

- **AccessService**: an abstract class that merges the GET, POST, PUT, and DELETE methods into the **access** method
- **Demo**: an execution entry used to simulate GET, POST, PUT, and DELETE request sending
- **AccessServiceImpl**: implements the **access** method, which contains the code required for communication with API Gateway.

1. (Optional) Add a request header.

   Locate and comment out the following lines in the **AccessServiceImpl.java** file, and specify the project ID and account ID.

   ```
   //TODO: Add special headers.
   //request.addHeader("X-Project-Id", "xxxxx");
   //request.addHeader("X-Domain-Id", "xxxxx");
   ```

2. Edit the main method in the **Demo.java** file.

   Replace the bold texts with actual values. If you use other methods, such as POST, PUT, and DELETE, see the corresponding annotations.

   Specify **region**, **serviceName**, **AK/SK**, and **URL** as the actual values. In the demo, the URL for obtaining the VPC is used. Replace it with the required URL. For details on how to obtain the project ID in the URL, see **6.3 Obtaining a Project ID**. For details about the endpoint, see **Regions and Endpoints**.

   ```
   //TODO: Replace region with the name of the region in which the service to be accessed is located.
    private static final String region = "";
   ```

```
 //TODO: Replace serviceName with the name of the service you want to access. For example, ecs,
vpc, iam, and elb.
 private static final String serviceName = "";

 public static void main(String[] args) throws UnsupportedEncodingException
 {
 //TODO: Replace the AK and SK with those obtained on the My Credential page.
 String ak = "ZIRRKMTWP******1WKNKB";
 String sk = "Us0mdMNHk******YrRCnW0ecfzl";

 //TODO: To specify a project ID (multi-project scenarios), add the X-Project-Id header.
 //TODO: To access a global service, such as IAM, DNS, CDN, and TMS, add the X-Domain-Id header
to specify an account ID.
 //TODO: To add a header, find "Add special headers" in the AccessServiceImple.java file.

 //TODO: Test the API
 String url = "https://{Endpoint}/v1/{project_id}/vpcs";
 get(ak, sk, url);

 //TODO: When creating a VPC, replace {project_id} in postUrl with the actual value.
 //String postUrl = "https://serviceEndpoint/v1/{project_id}/cloudservers";
 //String postbody ="{\"vpc\": {\"name\": \"vpc\",\"cidr\": \"192.168.0.0/16\"}}";
 //post(ak, sk, postUrl, postbody);

 //TODO: When querying a VPC, replace {project_id} in url with the actual value.
 //String url = "https://serviceEndpoint/v1/{project_id}/vpcs/{vpc_id}";
 //get(ak, sk, url);

 //TODO: When updating a VPC, replace {project_id} and {vpc_id} in putUrl with the actual values.
 //String putUrl = "https://serviceEndpoint/v1/{project_id}/vpcs/{vpc_id}";
 //String putbody ="{\"vpc\":{\"name\": \"vpc1\",\"cidr\": \"192.168.0.0/16\"}}";
 //put(ak, sk, putUrl, putbody);

 //TODO: When deleting a VPC, replace {project_id} and {vpc_id} in deleteUrl with the actual values.
 //String deleteUrl = "https://serviceEndpoint/v1/{project_id}/vpcs/{vpc_id}";
 //delete(ak, sk, deleteUrl);
 }
```

3. Compile the code and call the API.

   In the **Package Explorer** area on the left, right-click **Demo.java** and choose **Run AS** > **Java Application** to run the demo code.

   You can view the API call logs on the console.

----End

3.3 Response

Status Code

After sending a request, you will receive a response that includes a status code, response header, and response body.

A status code is a group of digits, ranging from 1*xx* to 5*xx*. It indicates the status of a request. For details, see **Status Codes**.

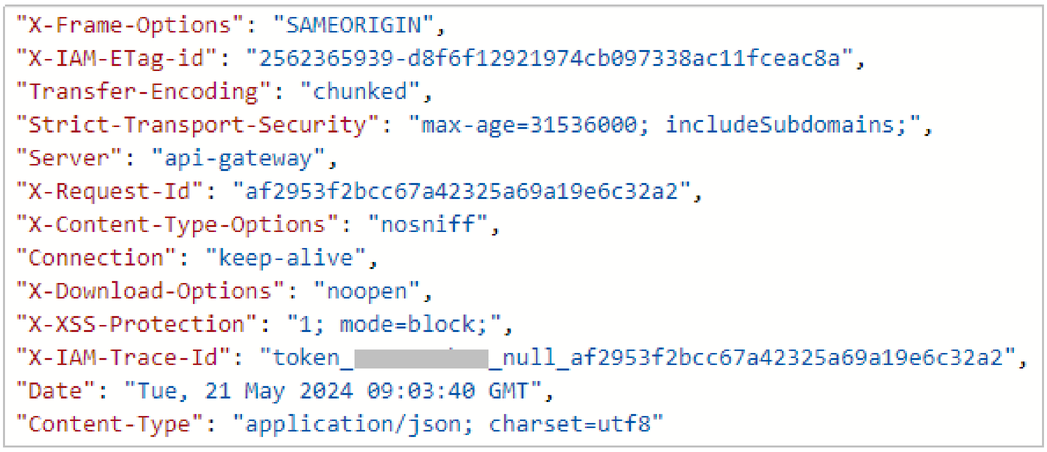
If status code **201** is returned for the API for **creating an IAM user**, the request is successful.

Response Header

Similar to a request, a response also has a header, for example, **content-type**.

For the API for **creating an IAM user**, the message header shown in **Figure 3-5** is returned.

Figure 3-5 Response header



```
"X-Frame-Options": "SAMEORIGIN",
"X-IAM-ETag-id": "2562365939-d8f6f12921974cb097338ac11fceac8a",
"Transfer-Encoding": "chunked",
"Strict-Transport-Security": "max-age=31536000; includeSubdomains;",
"Server": "api-gateway",
"X-Request-Id": "af2953f2bcc67a42325a69a19e6c32a2",
"X-Content-Type-Options": "nosniff",
"Connection": "keep-alive",
"X-Download-Options": "noopen",
"X-XSS-Protection": "1; mode=block;",
"X-IAM-Trace-Id": "token_[redacted]_null_af2953f2bcc67a42325a69a19e6c32a2",
"Date": "Tue, 21 May 2024 09:03:40 GMT",
"Content-Type": "application/json; charset=utf8"
```

Response Body

A response body is often returned in structured format. It corresponds to **content-type** header field and passes content except the response header.

For the API for **creating an IAM user**, the message header shown in the following is returned. The following describes part of the request body.

```
{
    "user": {
        "id": "c131886aec...",
        "name": "IAMUser",
        "description": "IAM User Description",
        "areacode": "",
        "phone": "",
        "email": "***@***.com",
        "status": null,
        "enabled": true,
        "pwd_status": false,
        "access_mode": "default",
        "is_domain_owner": false,
        "xuser_id": "",
        "xuser_type": "",
        "password_expires_at": null,
        "create_time": "2024-05-21T09:03:41.000000",
        "domain_id": "d78cbac1..........",
        "xdomain_id": "30086000........",
        "xdomain_type": "",
        "default_project_id": null
    }
}
```

If an error occurs during API calling, an error code and the corresponding error message will be displayed. The following shows an error response body:

```
{
    "error_msg": "Request body is invalid.",
    "error_code": "IAM.0011"
}
```

In the preceding information, **error_code** is an error code, and **error_msg** describes the error.

4 APIs

4.1 Task Management

4.1.1 Creating a Check Task

Function

This API is used to create a check task.

Calling Method

For details, see **Calling APIs**.

URI

POST /v2/{project_id}/task

Table 4-1 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID.

Request Parameters

Table 4-2 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token. It can be obtained by calling the IAM API used to obtain a user token. The value of **X-Subject-Token** in the response header is the user token.
Content-Type	Yes	String	Media type and encoding format.

Table 4-3 Request body parameters

Parameter	Mandatory	Type	Description
check_type	No	Array of strings	Check type, in array format. The default value is **source**.
git_url	Yes	String	Repository address.
git_branch	Yes	String	Repository branch.
language	Yes	Array of strings	Check language in array format. The value of language can be **cpp**, **java**, **js**, **python**, **php**, **css**, **html**, **go**, **typescript**, or **csharp**.
rule_sets	No	Array of **RuleSetV2** objects	Specified rule set.
task_type	No	String	Check type. The value can be **full** or **inc**. **full** indicates full check, and **inc** indicates MR check.
username	No	String	Name of a user who has the permission to access the repository.
access_token	No	String	Token of a user who has the permission on the repository.
endpoint_id	No	String	Endpoint ID of a user who has the permission on the repository.

Parameter	Mandatory	Type	Description
inc_config	No	**IncConfigV2** object	Parameter related to incremental check.
enable_fossbot	No	Boolean	Whether to enable fossbot check. By default, it is disabled.
resource_pool_id	No	String	Resource pool ID, which can be obtained from the resource pool management page.

Table 4-4 RuleSetV2

Parameter	Mandatory	Type	Description
ruleset_id	No	String	Rule set ID. Obtained from the web page.
language	Yes	String	Check language. The value of language can be **cpp**, **java**, **js**, **python**, **php**, **css**, **html**, **go**, **typescript**, or **csharp**.

Table 4-5 IncConfigV2

Parameter	Mandatory	Type	Description
parent_task_id	No	String	ID of the parent task to be associated. This parameter is required for creating a pipeline or MR task.
git_source_branch	No	String	Code source branch for incremental check.
git_target_branch	No	String	Code target branch for incremental check.
merge_id	No	String	Unique MR ID
event_type	No	String	Webhook triggering event type. merge_request/push_request.
action	No	String	Webhook event status. open/close/update.
title	No	String	MR title.

Response Parameters

Status code: 200

Table 4-6 Response body parameters

Parameter	Type	Description
task_id	String	Task ID.

Status code: 400

Table 4-7 Response body parameters

Parameter	Type	Description
error_code	String	Error code.
error_msg	String	Error message.

Status code: 401

Table 4-8 Response body parameters

Parameter	Type	Description
error_code	String	Error code.
error_msg	String	Error message.

Example Requests

Create a check task: Set the check type to source, the code repository branch to master, and the language to cpp. The code repository URL is displayed based on the site requirements.

```
POST https://{endpoint}/v2/{project_id}/task

{
  "check_type" : [ "source" ],
  "git_url" : "git@code******************958.git",
  "git_branch" : "master",
  "language" : [ "cpp", "js" ]
}
```

Example Responses

Status code: 200

Request succeeded!

```
{
  "task_id" : "d161fd0**********cb3c1d6a783e"
}
```

Status code: 400

Bad Request

```
{
  "error_code" : "CC.00000000",
  "error_msg" : "Network busy. Try again later."
}
```

Status code: 401

Unauthorized

```
{
  "error_code" : "CC.00000003",
  "error_msg" : "Authentication information expired."
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

Create a check task: Set the check type to source, the code repository branch to master, and the language to cpp. The code repository URL is displayed based on the site requirements.

```
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.codecheck.v2.region.CodeCheckRegion;
import com.huaweicloud.sdk.codecheck.v2.*;
import com.huaweicloud.sdk.codecheck.v2.model.*;

import java.util.List;
import java.util.ArrayList;

public class CreateTaskSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
                .withAk(ak)
                .withSk(sk);

        CodeCheckClient client = CodeCheckClient.newBuilder()
                .withCredential(auth)
                .withRegion(CodeCheckRegion.valueOf("<YOUR REGION>"))
                .build();
        CreateTaskRequest request = new CreateTaskRequest();
        request.withProjectId("{project_id}");
        CreateTaskRequestV2 body = new CreateTaskRequestV2();
        List<String> listbodyLanguage = new ArrayList<>();
        listbodyLanguage.add("cpp");
        listbodyLanguage.add("js");
        List<String> listbodyCheckType = new ArrayList<>();
        listbodyCheckType.add("source");
        body.withLanguage(listbodyLanguage);
        body.withGitBranch("master");
        body.withGitUrl("git@code******************958.git");
        body.withCheckType(listbodyCheckType);
        request.withBody(body);
        try {
            CreateTaskResponse response = client.createTask(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());
        }
    }
}
```

Python

Create a check task: Set the check type to source, the code repository branch to master, and the language to cpp. The code repository URL is displayed based on the site requirements.

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcodecheck.v2.region.codecheck_region import CodeCheckRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcodecheck.v2 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]

    credentials = BasicCredentials(ak, sk)

    client = CodeCheckClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CodeCheckRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = CreateTaskRequest()
        request.project_id = "{project_id}"
        listLanguagebody = [
            "cpp",
            "js"
        ]
        listCheckTypebody = [
            "source"
        ]
        request.body = CreateTaskRequestV2(
            language=listLanguagebody,
            git_branch="master",
            git_url="git@code*****************958.git",
            check_type=listCheckTypebody
        )
        response = client.create_task(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

Create a check task: Set the check type to source, the code repository branch to master, and the language to cpp. The code repository URL is displayed based on the site requirements.

```
package main

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

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := codecheck.NewCodeCheckClient(
        codecheck.CodeCheckClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.CreateTaskRequest{}
    request.ProjectId = "{project_id}"
    var listLanguagebody = []string{
        "cpp",
        "js",
    }
    var listCheckTypebody = []string{
        "source",
    }
    request.Body = &model.CreateTaskRequestV2{
        Language: listLanguagebody,
        GitBranch: "master",
        GitUrl: "git@code*****************958.git",
        CheckType: &listCheckTypebody,
    }
    response, err := client.CreateTask(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in **API Explorer**. SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Request succeeded!
400	Bad Request
401	Unauthorized

Error Codes

See **Error Codes**.

4.1.2 Deleting a Check Task

Function

This API is used to delete a check task. A task that is being executed cannot be viewed after being deleted.

Calling Method

For details, see **Calling APIs**.

URI

DELETE /v2/tasks/{task_id}

Table 4-9 Path Parameters

Parameter	Mandatory	Type	Description
task_id	Yes	String	Task ID.

Request Parameters

Table 4-10 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token. It can be obtained by calling the IAM API used to obtain a user token. The value of **X-Subject-Token** in the response header is the user token.

Response Parameters

Status code: 400

Table 4-11 Response body parameters

Parameter	Type	Description
error_code	String	Error code.
error_msg	String	Error message.

Status code: 401

Table 4-12 Response body parameters

Parameter	Type	Description
error_code	String	Error code.
error_msg	String	Error message.

Example Requests

```
DELETE https://{endpoint}/v2/tasks/{task_id}
```

Example Responses

Status code: 200

Request succeeded!

```
null
```

Status code: 400

Bad Request

```
{
  "error_code" : "CC.00000000",
  "error_msg" : "Network busy. Try again later."
}
```

Status code: 401

Unauthorized

```
{
  "error_code" : "CC.00000003",
  "error_msg" : "Authentication information expired."
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
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.codecheck.v2.region.CodeCheckRegion;
import com.huaweicloud.sdk.codecheck.v2.*;
import com.huaweicloud.sdk.codecheck.v2.model.*;


public class DeleteTaskSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great 
security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or 
environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running 
this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
                .withAk(ak)
                .withSk(sk);

        CodeCheckClient client = CodeCheckClient.newBuilder()
                .withCredential(auth)
                .withRegion(CodeCheckRegion.valueOf("<YOUR REGION>"))
                .build();
        DeleteTaskRequest request = new DeleteTaskRequest();
        request.withTaskId("{task_id}");
        try {
            DeleteTaskResponse response = client.deleteTask(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());
        }
    }
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcodecheck.v2.region.codecheck_region import CodeCheckRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcodecheck.v2 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]

    credentials = BasicCredentials(ak, sk)

    client = CodeCheckClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CodeCheckRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = DeleteTaskRequest()
        request.task_id = "{task_id}"
        response = client.delete_task(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

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

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := codecheck.NewCodeCheckClient(
        codecheck.CodeCheckClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
        WithCredential(auth).
        Build())

    request := &model.DeleteTaskRequest{}
    request.TaskId = "{task_id}"
    response, err := client.DeleteTask(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in **API Explorer**. SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Request succeeded!
400	Bad Request
401	Unauthorized

Error Codes

See **Error Codes**.

4.1.3 Querying Tasks

Function

This API is used to query tasks of a project based on project_id.

Calling Method

For details, see **Calling APIs**.

URI

GET /v2/{project_id}/tasks

Table 4-13 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID.

Table 4-14 Query Parameters

Parameter	Mandatory	Type	Description
offset	No	Integer	Offset for pagination.
limit	No	Integer	Number of items displayed on each page. A maximum of 100 items are supported.

Request Parameters

Table 4-15 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token. It can be obtained by calling the IAM API used to obtain a user token. The value of **X-Subject-Token** in the response header is the user token.

Response Parameters

Status code: 200

Table 4-16 Response body parameters

Parameter	Type	Description
tasks	Array of **SimpleTaskInfoV2** objects	Task information.
total	Integer	Total number.

Table 4-17 SimpleTaskInfoV2

Parameter	Type	Description
task_id	String	Task ID.
task_name	String	Task name.
creator_id	String	Creator ID.
git_url	String	Code repository address.

Parameter	Type	Description
git_branch	String	Code repository branch. If the MR mode is used, the value is the source branch.
created_at	String	Creation time.
last_check_time	String	Last check time.

Status code: 400

Table 4-18 Response body parameters

Parameter	Type	Description
error_code	String	Error code.
error_msg	String	Error message.

Status code: 401

Table 4-19 Response body parameters

Parameter	Type	Description
error_code	String	Error code.
error_msg	String	Error message.

Example Requests

```
GET https://{endpoint}/v2/{task_id}/tasks
```

Example Responses

Status code: 200

Request succeeded!

```
{
  "tasks" : [ {
    "task_id" : "435b58e********d01c94a6",
    "task_name" : "CSharp_02201119",
    "creator_id" : "ecadebb5041***9780f3d905e20",
    "git_url" : "git@********_02201119.git",
    "git_branch" : "master",
    "created_at" : "2020-02-20 11:21:11",
    "last_check_time" : "2020-02-20 11:37:46"
  }, {
    "task_id" : "435b58343tg5g36907c1384d01c94a6",
    "task_name" : "CSharp_066",
    "creator_id" : "ecadebb3664*****",
    "git_url" : "git@****_02201119.git",
    "git_branch" : "master",
    "created_at" : "2020-02-20 11:21:11",
    "last_check_time" : "2020-02-20 11:37:46"
  } ],
  "total" : 2
}
```

Status code: 400

Bad Request

```
{
  "error_code" : "CC.00000000",
  "error_msg" : "Network busy. Try again later."
}
```

Status code: 401

Unauthorized

```
{
  "error_code" : "CC.00000003",
  "error_msg" : "Authentication information expired."
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
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.codecheck.v2.region.CodeCheckRegion;
import com.huaweicloud.sdk.codecheck.v2.*;
import com.huaweicloud.sdk.codecheck.v2.model.*;


public class ShowTaskListByProjectIdSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
                .withAk(ak)
                .withSk(sk);

        CodeCheckClient client = CodeCheckClient.newBuilder()
                .withCredential(auth)
                .withRegion(CodeCheckRegion.valueOf("<YOUR REGION>"))
                .build();
        ShowTaskListByProjectIdRequest request = new ShowTaskListByProjectIdRequest();
        request.withProjectId("{project_id}");
        try {
            ShowTaskListByProjectIdResponse response = client.showTaskListByProjectId(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());
        }
    }
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcodecheck.v2.region.codecheck_region import CodeCheckRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcodecheck.v2 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]

    credentials = BasicCredentials(ak, sk)

    client = CodeCheckClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CodeCheckRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ShowTaskListByProjectIdRequest()
        request.project_id = "{project_id}"
        response = client.show_task_list_by_project_id(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

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

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this
example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
```

```
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := codecheck.NewCodeCheckClient(
        codecheck.CodeCheckClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ShowTaskListByProjectIdRequest{}
    request.ProjectId = "{project_id}"
    response, err := client.ShowTaskListByProjectId(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in **API Explorer**. SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Request succeeded!
400	Bad Request
401	Unauthorized

Error Codes

See **Error Codes**.

4.1.4 Executing a Check Task

Function

This API is used to execute a check task.

Calling Method

For details, see **Calling APIs**.

URI

POST /v2/tasks/{task_id}/run

Table 4-20 Path Parameters

Parameter	Mandatory	Type	Description
task_id	Yes	String	Task ID.

Request Parameters

Table 4-21 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	It can be obtained by calling the IAM API used to obtain a user token. The value of **X-Subject-Token** in the response header is the user token.
Content-Type	Yes	String	Media type and encoding format.

Table 4-22 Request body parameters

Parameter	Mandatory	Type	Description
username	No	String	Name of the user who has the permission on the temporary repository corresponding to the task.
access_token	No	String	Token of the user who has the permission on the temporary repository corresponding to the task.
git_url	No	String	Temporary repository address corresponding to the task.
git_branch	No	String	Temporary repository branch corresponding to the task.

Response Parameters

Status code: 200

Table 4-23 Response body parameters

Parameter	Type	Description
exec_id	String	Execution ID.

Status code: 400

Table 4-24 Response body parameters

Parameter	Type	Description
error_code	String	Error code.
error_msg	String	Error message.

Status code: 401

Table 4-25 Response body parameters

Parameter	Type	Description
error_code	String	Error code.
error_msg	String	Error message.

Example Requests

Execute a check task: Set the code repository branch to master. The code repository URL and username are displayed based on the site requirements.

```
POST https://{endpoint}/v2/tasks/{taskid}/run

{
  "git_url" : "http://github.xxxxxxxx.git",
  "git_branch" : "master",
  "username" : "test",
  "access_token" : "xLD56xxxxxxxxxJF3Sdxxxx"
}
```

Example Responses

Status code: 200

Request succeeded!

```
{
  "exec_id" : "d163535d0*******81d6a7654443e"
}
```

Status code: 400

Bad Request

```
{
  "error_code" : "CC.00000000",
  "error_msg" : "Network busy. Try again later."
}
```

Status code: 401

Unauthorized

```
{
  "error_code" : "CC.00000003",
  "error_msg" : "Authentication information expired."
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

Execute a check task: Set the code repository branch to master. The code repository URL and username are displayed based on the site requirements.

```
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.codecheck.v2.region.CodeCheckRegion;
import com.huaweicloud.sdk.codecheck.v2.*;
import com.huaweicloud.sdk.codecheck.v2.model.*;


public class RunTaskSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
                .withAk(ak)
                .withSk(sk);

        CodeCheckClient client = CodeCheckClient.newBuilder()
                .withCredential(auth)
                .withRegion(CodeCheckRegion.valueOf("<YOUR REGION>"))
                .build();
        RunTaskRequest request = new RunTaskRequest();
        request.withTaskId("{task_id}");
        RunRequestV2 body = new RunRequestV2();
        body.withGitBranch("master");
        body.withGitUrl("http://github.xxxxxxxx.git");
        body.withAccessToken("xLD56xxxxxxxxxJF3Sdxxxx");
        body.withUsername("test");
        request.withBody(body);
        try {
            RunTaskResponse response = client.runTask(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());
        }
    }
}
```

Python

Execute a check task: Set the code repository branch to master. The code repository URL and username are displayed based on the site requirements.

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcodecheck.v2.region.codecheck_region import CodeCheckRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcodecheck.v2 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]

    credentials = BasicCredentials(ak, sk)

    client = CodeCheckClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CodeCheckRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = RunTaskRequest()
        request.task_id = "{task_id}"
        request.body = RunRequestV2(
            git_branch="master",
            git_url="http://github.xxxxxxxx.git",
            access_token="xLD56xxxxxxxxxJF3Sdxxxx",
            username="test"
        )
        response = client.run_task(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

Execute a check task: Set the code repository branch to master. The code repository URL and username are displayed based on the site requirements.

```
package main

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

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := codecheck.NewCodeCheckClient(
        codecheck.CodeCheckClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.RunTaskRequest{}
    request.TaskId = "{task_id}"
    gitBranchRunRequestV2:= "master"
    gitUrlRunRequestV2:= "http://github.xxxxxxxx.git"
    accessTokenRunRequestV2:= "xLD56xxxxxxxxxJF3Sdxxxx"
    usernameRunRequestV2:= "test"
    request.Body = &model.RunRequestV2{
        GitBranch: &gitBranchRunRequestV2,
        GitUrl: &gitUrlRunRequestV2,
        AccessToken: &accessTokenRunRequestV2,
        Username: &usernameRunRequestV2,
    }
    response, err := client.RunTask(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in **API Explorer**. SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Request succeeded!
400	Bad Request
401	Unauthorized

Error Codes

See **Error Codes**.

4.1.5 Stopping a Check Task

Function

This API is used to stop a check task based on the task ID.

Calling Method

For details, see **Calling APIs**.

URI

POST /v2/tasks/{task_id}/stop

Table 4-26 Path Parameters

Parameter	Mandatory	Type	Description
task_id	Yes	String	Task ID.

Request Parameters

Table 4-27 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	It can be obtained by calling the IAM API used to obtain a user token. The value of **X-Subject-Token** in the response header is the user token.

Response Parameters

Status code: 400

Table 4-28 Response body parameters

Parameter	Type	Description
error_code	String	Error code.
error_msg	String	Error message.

Status code: 401

Table 4-29 Response body parameters

Parameter	Type	Description
error_code	String	Error code.
error_msg	String	Error message.

Example Requests

```
POST https://{endpoint}/v2/tasks/2b31ed520xxxxxxebedb6e57xxxxxxxx/stop
```

Example Responses

Status code: 200

Request succeeded!

```
null
```

Status code: 400

Bad Request

```
{
  "error_code" : "CC.00000000",
  "error_msg" : "Network busy. Try again later."
}
```

Status code: 401

Unauthorized

```
{
  "error_code" : "CC.00000003",
  "error_msg" : "Authentication information expired."
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
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.codecheck.v2.region.CodeCheckRegion;
import com.huaweicloud.sdk.codecheck.v2.*;
import com.huaweicloud.sdk.codecheck.v2.model.*;


public class StopTaskByIdSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
                .withAk(ak)
                .withSk(sk);

        CodeCheckClient client = CodeCheckClient.newBuilder()
                .withCredential(auth)
                .withRegion(CodeCheckRegion.valueOf("<YOUR REGION>"))
                .build();
        StopTaskByIdRequest request = new StopTaskByIdRequest();
        request.withTaskId("{task_id}");
        try {
            StopTaskByIdResponse response = client.stopTaskById(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());
        }
    }
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcodecheck.v2.region.codecheck_region import CodeCheckRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcodecheck.v2 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]

    credentials = BasicCredentials(ak, sk)

    client = CodeCheckClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CodeCheckRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = StopTaskByIdRequest()
        request.task_id = "{task_id}"
        response = client.stop_task_by_id(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

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

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security 
risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment 
variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this 
example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := codecheck.NewCodeCheckClient(
        codecheck.CodeCheckClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.StopTaskByIdRequest{}
    request.TaskId = "{task_id}"
    response, err := client.StopTaskById(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in **API Explorer**. SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Request succeeded!
400	Bad Request
401	Unauthorized

Error Codes

See **Error Codes**.

4.1.6 Querying the Status of a Task

Function

This API is used to query the task status based on the task ID (**0**: being checked; **1**: failed; **2**: successful; **3**: stopped). Only tasks being executed have progress details.

Calling Method

For details, see **Calling APIs**.

URI

GET /v2/tasks/{task_id}/progress

Table 4-30 Path Parameters

Parameter	Mandatory	Type	Description
task_id	Yes	String	Task ID.

Request Parameters

Table 4-31 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	It can be obtained by calling the IAM API used to obtain a user token. The value of **X-Subject-Token** in the response header is the user token.

Response Parameters

Status code: 200

Table 4-32 Response body parameters

Parameter	Type	Description
task_status	Integer	Task status: **0**: being checked; **1**: failed; **2**: successful; **3**: stopped.

Parameter	Type	Description
progress	**ProgressDetailV2** object	Check progress.

Table 4-33 ProgressDetailV2

Parameter	Type	Description
ratio	String	Progress percentage.
info	String	Progress information.

Status code: 400

Table 4-34 Response body parameters

Parameter	Type	Description
error_code	String	Error code.
error_msg	String	Error message.

Status code: 401

Table 4-35 Response body parameters

Parameter	Type	Description
error_code	String	Error code.
error_msg	String	Error message.

Example Requests

```
GET https://{endpoint}/v2/tasks/2b31ed520xxxxxxebedb6e57xxxxxxxx/progress
```

Example Responses

Status code: 200

Request succeeded!

```
{
  "task_status" : 0,
  "progress" : {
    "ratio" : "50%",
    "info" : "Executing"
  }
}
```

Status code: 400

Bad Request

```
{
  "error_code" : "CC.00000000",
  "error_msg" : "Network busy. Try again later."
}
```

Status code: 401

Unauthorized

```
{
  "error_code" : "CC.00000003",
  "error_msg" : "Authentication information expired."
}
```

Status Codes

Status Code	Description
200	Request succeeded!
400	Bad Request
401	Unauthorized

Error Codes

See **Error Codes**.

4.1.7 Querying Historical Check Results

Function

This API is used to provide statistics on issues scanned each time.

Calling Method

For details, see **Calling APIs**.

URI

GET /v2/{project_id}/tasks/{task_id}/checkrecord

Table 4-36 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID.
task_id	Yes	String	Task ID.

Table 4-37 Query Parameters

Parameter	Mandatory	Type	Description
offset	No	Integer	Offset for pagination.
limit	No	Integer	Number of items displayed on each page. A maximum of 1,000 items are supported.
start_time	No	String	Start time of task. Filter tasks by start time.
end_time	No	String	End time of task. Filter tasks by end time.

Request Parameters

Table 4-38 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token. It can be obtained by calling the IAM API used to obtain a user token. The value of **X-Subject-Token** in the response header is the user token.

Response Parameters

Status code: 200

Table 4-39 Response body parameters

Parameter	Type	Description
data	Array of **CheckRecordDataInfo** objects	Data history.
total	Integer	Total number.

Table 4-40 CheckRecordDataInfo

Parameter	Type	Description
check_time	String	Task execution start time.

Parameter	Type	Description
check_end_time	String	Task execution end time.
issue_counts	CheckRecordIssueCountsInfo object	Severity.

Table 4-41 CheckRecordIssueCountsInfo

Parameter	Type	Description
critical	Integer	Critical issues.
serious	Integer	Major issues.
normal	Integer	Minor issues.
prompt	Integer	Suggestion issues.

Status code: 400

Table 4-42 Response body parameters

Parameter	Type	Description
error_code	String	Error code.
error_msg	String	Error message.

Status code: 401

Table 4-43 Response body parameters

Parameter	Type	Description
error_code	String	Error code.
error_msg	String	Error message.

Example Requests

```
GET https://{endpoint}/v2/{project_id}/tasks/{task_id}/checkrecord
```

Example Responses

Status code: 200

Request succeeded!

```
{
  "total" : 100,
  "data" : [ {
    "check_time" : "2020-02-20 11:37:46",
    "check_end_time" : "2020-02-20 11:47:46",
    "issue_counts" : {
      "critical" : 0,
      "serious" : 1,
      "normal" : 2,
      "prompt" : 3
    }
  }, {
    "check_time" : "2020-02-20 11:37:46",
    "check_end_time" : "2020-02-20 11:47:46",
    "issue_counts" : {
      "critical" : 0,
      "serious" : 1,
      "normal" : 2,
      "prompt" : 3
    }
  } ]
}
```

Status code: 400

Bad Request

```
{
  "error_code" : "CC.00000000",
  "error_msg" : "Network busy. Try again later."
}
```

Status code: 401

Unauthorized

```
{
  "error_code" : "CC.00000003",
  "error_msg" : "Authentication information expired."
}
```

Status Codes

Status Code	Description
200	Request succeeded!
400	Bad Request
401	Unauthorized

Error Codes

See **Error Codes**.

4.1.8 Querying Selected Rule Sets of a Task (Version 2)

Function

This API is used to query selected rule sets of a task (version 2).

Calling Method

For details, see **Calling APIs**.

URI

GET /v2/{project_id}/tasks/{task_id}/rulesets

Table 4-44 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID.
task_id	Yes	String	Task ID.

Request Parameters

Table 4-45 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	No	String	Media type and encoding format.
X-Auth-Token	Yes	String	User token. It can be obtained by calling the IAM API used to obtain a user token. The value of **X-Subject-Token** in the response header is the user token.

Response Parameters

Status code: 200

Table 4-46 Response body parameters

Parameter	Type	Description
[items]	Array<Array<**TaskRulesetInfo**>>	Request succeeded!

Table 4-47 TaskRulesetInfo

Parameter	Type	Description
template_id	String	Rule set ID.
language	String	Rule set language.
template_name	String	Rule set name.
type	String	Rule set status. The value can be **optional** and **selected**.
status	String	Rule set attribute. The value can be **0** (default user rule set) and **1** (default system rule set).

Status code: 400

Table 4-48 Response body parameters

Parameter	Type	Description
error_code	String	Error code.
error_msg	String	Error message.

Status code: 401

Table 4-49 Response body parameters

Parameter	Type	Description
error_code	String	Error code.
error_msg	String	Error message.

Example Requests

```
GET https://{endpoint}/v2/{project_id}/tasks/{task_id}/rulesets
```

Example Responses

Status code: 200

Request succeeded!

```
[ {
  "template_id" : "396e1c2511744f6fa199d33b26038edd",
  "language" : "Java",
  "template_name" : "Java_cmetrics",
  "type" : "selected",
  "status" : "0"
}, {
  "template_id" : "4e988c481b4c46d0a3297fbe343a662d",
  "language" : "html",
  "template_name" : "General Criterion Set",
  "type" : "optional",
  "status" : "1"
} ]
```

Status code: 400

Bad Request

```
{
  "error_code" : "CC.00000000",
  "error_msg" : "Network busy. Try again later."
}
```

Status code: 401

Unauthorized

```
{
  "error_code" : "CC.00000003",
  "error_msg" : "Authentication information expired."
}
```

Status Codes

Status Code	Description
200	Request succeeded!
400	Bad Request
401	Unauthorized

Error Codes

See **Error Codes**.

4.1.9 Querying Selected Rule Sets of a Task (Version 3)

Function

This API is used to query selected rule sets of a task (version 3).

Calling Method

For details, see **Calling APIs**.

URI

GET /v3/{project_id}/tasks/{task_id}/rulesets

Table 4-50 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID.
task_id	Yes	String	Task ID.

Table 4-51 Query Parameters

Parameter	Mandatory	Type	Description
offset	No	Integer	Offset for pagination.
limit	No	Integer	Number of items displayed on each page. A maximum of 1,000 items are supported.

Request Parameters

Table 4-52 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	No	String	Media type and encoding format.
X-Auth-Token	Yes	String	User token. It can be obtained by calling the IAM API used to obtain a user token. The value of **X-Subject-Token** in the response header is the user token.

Response Parameters

Status code: 200

Table 4-53 Response body parameters

Parameter	Type	Description
info	Array of **TaskRulesetInfo** objects	Rule set information.
total	Integer	Total number.

Table 4-54 TaskRulesetInfo

Parameter	Type	Description
template_id	String	Rule set ID.
language	String	Rule set language.
template_name	String	Rule set name.
type	String	Rule set status. The value can be **optional** and **selected**.
status	String	Rule set attribute. The value can be **0** (default user rule set) and **1** (default system rule set).

Status code: 400

Table 4-55 Response body parameters

Parameter	Type	Description
error_code	String	Error code.
error_msg	String	Error message.

Status code: 401

Table 4-56 Response body parameters

Parameter	Type	Description
error_code	String	Error code.
error_msg	String	Error message.

Example Requests

```
GET https://{endpoint}/v3/{project_id}/tasks/{task_id}/rulesets
```

Example Responses

Status code: 200

Request succeeded!

```
{
  "total" : 1,
  "data" : [ {
    "template_id" : "111111xxxx3246",
    "language" : "java",
    "type" : "selected",
    "template_name" : "General Criterion Set",
    "status" : "1"
  } ]
}
```

Status code: 400

Bad Request

```
{
  "error_code" : "CC.00000000",
  "error_msg" : "Network busy. Try again later."
}
```

Status code: 401

Unauthorized

```
{
  "error_code" : "CC.00000003",
  "error_msg" : "Authentication information expired."
}
```

Status Codes

Status Code	Description
200	Request succeeded!
400	Bad Request
401	Unauthorized

Error Codes

See **Error Codes**.

4.1.10 Querying the Check Parameters of a Rule Set (Version 2)

Function

This API is used to query the check parameters of a rule set (version 2).

Calling Method

For details, see **Calling APIs**.

URI

GET /v2/{project_id}/tasks/{task_id}/ruleset/{ruleset_id}/check-parameters

Table 4-57 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID.
task_id	Yes	String	Task ID.
ruleset_id	Yes	String	Rule set ID.

Table 4-58 Query Parameters

Parameter	Mandatory	Type	Description
language	Yes	String	Rule set language.

Request Parameters

Table 4-59 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	No	String	Media type and encoding format.
X-Auth-Token	Yes	String	User token. It can be obtained by calling the IAM API used to obtain a user token. The value of **X-Subject-Token** in the response header is the user token.

Response Parameters

Status code: 200

Table 4-60 Response body parameters

Parameter	Type	Description
[items]	Array<Array< **TaskCheckParamters**>>	Request succeeded!

Table 4-61 TaskCheckParamters

Parameter	Type	Description
check_id	Integer	ID of check tool.
name	String	Compilation parameter name.
checker_configs	Array of **CheckConfigInfo** objects	Configuration of the check parameter.

Table 4-62 CheckConfigInfo

Parameter	Type	Description
name	String	Name of the check parameter.
cfg_key	String	Key value corresponding to the check parameter.
default_value	String	Default value of the check parameter.
option_value	String	Option for the check parameter.
is_required	Integer	**0**: optional; **1**: mandatory.
description	String	Description for the check parameter.
type	Integer	Parameter type. **0**: text; **2**: optional.
status	String	Parameter status. **on**: enabled; **off**: disabled.
value	String	Check parameter value.

Status code: 400

Table 4-63 Response body parameters

Parameter	Type	Description
error_code	String	Error code.
error_msg	String	Error message.

Status code: 401

Table 4-64 Response body parameters

Parameter	Type	Description
error_code	String	Error code.
error_msg	String	Error message.

Example Requests

```
GET https://{endpoint}/v2/{project_id}/tasks/{task_id}/ruleset/{ruleset_id}/check-parameters
```

Example Responses

Status code: 200

Request succeeded!

```
[ {
  "check_id" : 0,
  "name" : "compile_tool",
  "check_configs" : [ {
    "value" : "",
    "name" : "Compile command",
    "cfg_key" : "",
    "default_value" : "",
    "option_value" : "",
    "is_required" : 0,
    "description" : "Compile command.",
    "type" : 0,
    "status" : "on"
  } ]
} ]
```

Status code: 400

Bad Request

```
{
  "error_code" : "CC.00000000",
  "error_msg" : "Network busy. Try again later."
}
```

Status code: 401

Unauthorized

```
{
  "error_code" : "CC.00000003",
  "error_msg" : "Authentication information expired."
}
```

Status Codes

Status Code	Description
200	Request succeeded!
400	Bad Request

Status Code	Description
401	Unauthorized

Error Codes

See **Error Codes**.

4.1.11 Querying the Check Parameters of a Rule Set (Version 3)

Function

This API is used to query the check parameters of a rule set (version 3).

Calling Method

For details, see **Calling APIs**.

URI

GET /v3/{project_id}/tasks/{task_id}/ruleset/{ruleset_id}/check-parameters

Table 4-65 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID.
task_id	Yes	String	Task ID.
ruleset_id	Yes	String	Rule set ID.

Table 4-66 Query Parameters

Parameter	Mandatory	Type	Description
language	Yes	String	Rule set language.
offset	No	Integer	(Optional) Offset for pagination.
limit	No	Integer	(Optional) Number of items displayed on each page.

Request Parameters

Table 4-67 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	No	String	Media type and encoding format.
X-Auth-Token	Yes	String	User token. The token can be obtained by calling the IAM API used to obtain a user token. The value of **X-Subject-Token** in the response header is the user token.

Response Parameters

Status code: 200

Table 4-68 Response body parameters

Parameter	Type	Description
data	Array of **TaskCheckParamters** objects	Data history.
total	Integer	Total number.

Table 4-69 TaskCheckParamters

Parameter	Type	Description
check_id	Integer	ID of check tool.
name	String	Compilation parameter name.
checker_configs	Array of **CheckConfigInfo** objects	Configuration of the check parameter.

Table 4-70 CheckConfigInfo

Parameter	Type	Description
name	String	Name of the check parameter.

Parameter	Type	Description
cfg_key	String	Key value corresponding to the check parameter.
default_value	String	Default value of the check parameter.
option_value	String	Option for the check parameter.
is_required	Integer	**0**: optional; **1**: mandatory.
description	String	Description for the check parameter.
type	Integer	Parameter type. **0**: text; **2**: optional.
status	String	Parameter status. **on**: enabled; **off**: disabled.
value	String	Check parameter value.

Status code: 400

Table 4-71 Response body parameters

Parameter	Type	Description
error_code	String	Error code.
error_msg	String	Error message.

Status code: 401

Table 4-72 Response body parameters

Parameter	Type	Description
error_code	String	Error code.
error_msg	String	Error message.

Example Requests

```
GET https://{endpoint}/v3/{project_id}/tasks/{task_id}/ruleset/{ruleset_id}/check-parameters
```

Example Responses

Status code: 200

Request succeeded!

```
{
  "total" : 1,
  "data" : [ {
    "check_id" : 0,
    "name" : "compile_tool",
    "check_configs" : [ {
      "value" : "",
      "name" : "Compile command",
      "cfg_key" : "",
      "default_value" : "",
      "option_value" : "",
      "is_required" : 0,
      "description" : "Compile command.",
      "type" : 0,
      "status" : "on"
    } ]
  } ]
}
```

Status code: 400

Bad Request

```
{
  "error_code" : "CC.00000000",
  "error_msg" : "Network busy. Try again later."
}
```

Status code: 401

Unauthorized

```
{
  "error_code" : "CC.00000003",
  "error_msg" : "Authentication information expired."
}
```

Status Codes

Status Code	Description
200	Request succeeded!
400	Bad Request
401	Unauthorized

Error Codes

See **Error Codes**.

4.1.12 Configuring the Check Parameters of a Task

Function

This API is used to configure the check parameters of a task.

Calling Method

For details, see **Calling APIs**.

URI

POST /v2/{project_id}/tasks/{task_id}/config-parameters

Table 4-73 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID.
task_id	Yes	String	Task ID.

Request Parameters

Table 4-74 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	It can be obtained by calling the IAM API used to obtain a user token. The value of **X-Subject-Token** in the response header is the user token.
Content-Type	Yes	String	Media type and encoding format.

Table 4-75 Request body parameters

Parameter	Mandatory	Type	Description
check_id	Yes	Integer	ID of check tool.
ruleset_id	Yes	String	Rule set ID.
language	Yes	String	Rule set language.
status	Yes	String	**off**: disabled; **on**: enabled.
task_check_settings	Yes	Array of **TaskCheckSettingsItem** objects	Check parameters.

Table 4-76 TaskCheckSettingsItem

Parameter	Mandatory	Type	Description
cfg_key	Yes	String	Key value corresponding to the check parameter.
status	Yes	String	Parameter status.
cfg_value	No	String	Check parameter value.

Response Parameters

Status code: 400

Table 4-77 Response body parameters

Parameter	Type	Description
error_code	String	Error code.
error_msg	String	Error message.

Status code: 401

Table 4-78 Response body parameters

Parameter	Type	Description
error_code	String	Error code.
error_msg	String	Error message.

Example Requests

Configure the check parameters of a task: Set the language to **csharp** and the status to **on**.

```
POST https://{endpoint}/v2/{project_id}/tasks/{task_id}/config-parameters

{
  "check_id" : 11,
  "ruleset_id" : "b113bxxxxxxxx5738bee86ecxxxxxxxx",
  "language" : "csharp",
  "status" : "on",
  "task_check_settings" : [ {
    "cfg_key" : "csharpCompileTool",
    "status" : "on",
    "cfg_value" : "msbuild"
  } ]
}
```

Example Responses

Status code: 200

Request succeeded!

```
null
```

Status code: 400

Bad Request

```
{
  "error_code" : "CC.00000000",
  "error_msg" : "Network busy. Try again later."
}
```

Status code: 401

Unauthorized

```
{
  "error_code" : "CC.00000003",
  "error_msg" : "Authentication information expired."
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

Configure the check parameters of a task: Set the language to **csharp** and the status to **on**.

```
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.codecheck.v2.region.CodeCheckRegion;
import com.huaweicloud.sdk.codecheck.v2.*;
import com.huaweicloud.sdk.codecheck.v2.model.*;

import java.util.List;
import java.util.ArrayList;

public class ListTaskParameterSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
                .withAk(ak)
                .withSk(sk);

        CodeCheckClient client = CodeCheckClient.newBuilder()
                .withCredential(auth)
                .withRegion(CodeCheckRegion.valueOf("<YOUR REGION>"))
                .build();
        ListTaskParameterRequest request = new ListTaskParameterRequest();
        request.withProjectId("{project_id}");
        request.withTaskId("{task_id}");
        ConfigTaskParameterBody body = new ConfigTaskParameterBody();
        List<TaskCheckSettingsItem> listbodyTaskCheckSettings = new ArrayList<>();
        listbodyTaskCheckSettings.add(
            new TaskCheckSettingsItem()
                .withCfgKey("csharpCompileTool")
                .withStatus("on")
                .withCfgValue("msbuild")
        );
        body.withTaskCheckSettings(listbodyTaskCheckSettings);
        body.withStatus("on");
        body.withLanguage("csharp");
        body.withRulesetId("b113bxxxxxxxx5738bee86ecxxxxxxxx");
        body.withCheckId(11);
        request.withBody(body);
        try {
            ListTaskParameterResponse response = client.listTaskParameter(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());
        }
    }
}
```

Python

Configure the check parameters of a task: Set the language to **csharp** and the status to **on**.

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcodecheck.v2.region.codecheck_region import CodeCheckRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcodecheck.v2 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]

    credentials = BasicCredentials(ak, sk)

    client = CodeCheckClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CodeCheckRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ListTaskParameterRequest()
        request.project_id = "{project_id}"
        request.task_id = "{task_id}"
        listTaskCheckSettingsbody = [
            TaskCheckSettingsItem(
                cfg_key="csharpCompileTool",
                status="on",
                cfg_value="msbuild"
            )
        ]
        request.body = ConfigTaskParameterBody(
            task_check_settings=listTaskCheckSettingsbody,
            status="on",
            language="csharp",
            ruleset_id="b113bxxxxxxxx5738bee86ecxxxxxxxx",
            check_id=11
        )
        response = client.list_task_parameter(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

Configure the check parameters of a task: Set the language to **csharp** and the status to **on**.

```
package main

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

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := codecheck.NewCodeCheckClient(
        codecheck.CodeCheckClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ListTaskParameterRequest{}
    request.ProjectId = "{project_id}"
    request.TaskId = "{task_id}"
    cfgValueTaskCheckSettings:= "msbuild"
    var listTaskCheckSettingsbody = []model.TaskCheckSettingsItem{
        {
            CfgKey: "csharpCompileTool",
            Status: "on",
            CfgValue: &cfgValueTaskCheckSettings,
        },
    }
    request.Body = &model.ConfigTaskParameterBody{
        TaskCheckSettings: listTaskCheckSettingsbody,
        Status: "on",
        Language: "csharp",
        RulesetId: "b113bxxxxxxxx5738bee86ecxxxxxxxx",
        CheckId: int32(11),
    }
    response, err := client.ListTaskParameter(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in **API Explorer**. SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Request succeeded!
400	Bad Request
401	Unauthorized

Error Codes

See **Error Codes**.

4.1.13 Modifying the Rule Set in a Task

Function

This API is used to modify the rule set in a task.

Calling Method

For details, see **Calling APIs**.

URI

PUT /v2/tasks/{task_id}/ruleset

Table 4-79 Path Parameters

Parameter	Mandatory	Type	Description
task_id	Yes	String	Task ID.

Request Parameters

Table 4-80 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Media type and encoding format.
X-Auth-Token	Yes	String	User token. It can be obtained by calling the IAM API used to obtain a user token. The value of **X-Subject-Token** in the response header is the user token.

Table 4-81 Request body parameters

Parameter	Mandatory	Type	Description
[items]	Yes	Array of **UpdateTaskRulesetItem** objects	Rule set modification.

Table 4-82 UpdateTaskRulesetItem

Parameter	Mandatory	Type	Description
language	Yes	String	Rule set language.
rule_set_id	Yes	String	Obtained from the response parameter template_id after invoking API ListTaskRuleset.
if_use	Yes	String	Whether to enable the relationship between the task language and rule set. **1**: yes; **0**: no.
status	Yes	String	New/Old data. Default: 1.

Response Parameters

Status code: 400

Table 4-83 Response body parameters

Parameter	Type	Description
error_code	String	Error code.
error_msg	String	Error message.

Status code: 401

Table 4-84 Response body parameters

Parameter	Type	Description
error_code	String	Error code.
error_msg	String	Error message.

Example Requests

Modify the rule set of a task: Set the language to cpp and the rule status to 1. The rule setting ID is displayed based on the actual situation.

```
PUT https://{endpoint}/v2/tasks/{task_id}/ruleset

[ {
  "language" : "cpp",
  "rule_set_id" : "7be4cfxxxxxxxxfd8d07077cxxxxxxxx",
  "if_use" : "1",
  "status" : "1"
} ]
```

Example Responses

Status code: 200

Request succeeded!

```
null
```

Status code: 400

Bad Request

```
{
  "error_code" : "CC.00000000",
  "error_msg" : "Network busy. Try again later."
}
```

Status code: 401

Unauthorized

```
{
  "error_code" : "CC.00000003",
  "error_msg" : "Authentication information expired."
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

Modify the rule set of a task: Set the language to cpp and the rule status to 1. The rule setting ID is displayed based on the actual situation.

```
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.codecheck.v2.region.CodeCheckRegion;
import com.huaweicloud.sdk.codecheck.v2.*;
import com.huaweicloud.sdk.codecheck.v2.model.*;

import java.util.List;
import java.util.ArrayList;

public class UpdateTaskRulesetSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
                .withAk(ak)
                .withSk(sk);

        CodeCheckClient client = CodeCheckClient.newBuilder()
                .withCredential(auth)
                .withRegion(CodeCheckRegion.valueOf("<YOUR REGION>"))
                .build();
        UpdateTaskRulesetRequest request = new UpdateTaskRulesetRequest();
        request.withTaskId("{task_id}");
        List<UpdateTaskRulesetItem> listbodyUpdateTaskRuleset = new ArrayList<>();
        listbodyUpdateTaskRuleset.add(
            new UpdateTaskRulesetItem()
                .withLanguage("cpp")
                .withRuleSetId("7be4cfxxxxxxxxfd8d07077cxxxxxxxx")
                .withIfUse("1")
                .withStatus("1")
        );
        request.withBody(listbodyUpdateTaskRuleset);
        try {
            UpdateTaskRulesetResponse response = client.updateTaskRuleset(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());
        }
    }
}
```

Python

Modify the rule set of a task: Set the language to cpp and the rule status to 1. The rule setting ID is displayed based on the actual situation.

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcodecheck.v2.region.codecheck_region import CodeCheckRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcodecheck.v2 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]

    credentials = BasicCredentials(ak, sk)

    client = CodeCheckClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CodeCheckRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = UpdateTaskRulesetRequest()
        request.task_id = "{task_id}"
        listUpdateTaskRulesetbody = [
            UpdateTaskRulesetItem(
                language="cpp",
                rule_set_id="7be4cfxxxxxxxxfd8d07077cxxxxxxxxx",
                if_use="1",
                status="1"
            )
        ]
        request.body = listUpdateTaskRulesetbody
        response = client.update_task_ruleset(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

Modify the rule set of a task: Set the language to cpp and the rule status to 1. The rule setting ID is displayed based on the actual situation.

```
package main

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

```
func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := codecheck.NewCodeCheckClient(
        codecheck.CodeCheckClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.UpdateTaskRulesetRequest{}
    request.TaskId = "{task_id}"
    var listUpdateTaskRulesetbody = []model.UpdateTaskRulesetItem{
        {
            Language: "cpp",
            RuleSetId: "7be4cfxxxxxxxxfd8d07077cxxxxxxxx",
            IfUse: "1",
            Status: "1",
        },
    }
    request.Body = &listUpdateTaskRulesetbody
    response, err := client.UpdateTaskRuleset(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in **API Explorer**. SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Request succeeded!
400	Bad Request
401	Unauthorized

Error Codes

See **Error Codes**.

4.1.14 Querying Check Failure Logs

Function

This API is used to query check failure logs. If execute_id is not transferred, the latest check log is queried.

Calling Method

For details, see **Calling APIs**.

URI

GET /v2/{project_id}/tasks/{task_id}/log-detail

Table 4-85 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID.
task_id	Yes	String	Task ID.

Table 4-86 Query Parameters

Parameter	Mandatory	Type	Description
execute_id	No	String	Task execution ID.

Request Parameters

Table 4-87 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	No	String	Media type and encoding format.
X-Auth-Token	Yes	String	User token. It can be obtained by calling the IAM API used to obtain a user token. The value of **X-Subject-Token** in the response header is the user token.

Response Parameters

Status code: 200

Table 4-88 Response body parameters

Parameter	Type	Description
param_info	**ParamInfo** object	Task details.
log_info	Array of **LogInfo** objects	Log information.

Table 4-89 ParamInfo

Parameter	Type	Description
url	String	Repository address.
branch	String	Repository branch.
language	String	Repository language.
exclude_dir	String	Excluded directory.
encode	String	Encoding format.
compile_config	String	Compilation configuration.
rule_template	String	Rule set name.

Table 4-90 LogInfo

Parameter	Type	Description
display_name	String	Log title.
log	String	Log content.
level	String	Log level.
analysis	String	Log analysis.
faq	String	FAQ.

Status code: 400

Table 4-91 Response body parameters

Parameter	Type	Description
error_code	String	Error code.

Parameter	Type	Description
error_msg	String	Error message.

Status code: 401

Table 4-92 Response body parameters

Parameter	Type	Description
error_code	String	Error code.
error_msg	String	Error message.

Example Requests

```
GET https://{endpoint}/v2/{project_id}/tasks/{task_id}/log-detail
```

Example Responses

Status code: 200

Request succeeded!

```
{
  "param_info" : {
    "url" : "git@xxxxxx_only00001/file.git",
    "branch" : "master",
    "language" : "language",
    "exclude_dir" : "aa",
    "encode" : "",
    "compile_config" : "xxx",
    "rule_template" : ""
  },
  "log_info" : [ {
    "log" : "2023-01-04 14:31:33 Running on server:10.75.***.***\n2023-01-04 14:31:34 waiting subJob to execute\n2023-01-04 14:33:38 flush redis cache successfully!\n2023-01-04 14:33:38 all subJob finish!",
    "level" : "",
    "analysis" : "2023-01-04 14:31:33 Running on server:10.75.***.***\n2023-01-04 14:31:34 waiting subJob to execute\n2023-01-04 14:33:38 flush redis cache successfully!\n2023-01-04 14:33:38 all subJob finish!",
    "faq" : "",
    "display_name" : ""
  } ]
}
```

Status code: 400

Bad Request

```
{
  "error_code" : "CC.00000000",
  "error_msg" : "Network busy. Try again later."
}
```

Status code: 401

Unauthorized

```
{
  "error_code" : "CC.00000003",
  "error_msg" : "Authentication information expired."
}
```

Status Codes

Status Code	Description
200	Request succeeded!
400	Bad Request
401	Unauthorized

Error Codes

See **Error Codes**.

4.1.15 Obtaining the Directory Tree of a Task

Function

This API is used to obtain the directory tree of a task.

Calling Method

For details, see **Calling APIs**.

URI

GET /v2/{project_id}/tasks/{task_id}/listpathtree

Table 4-93 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID.
task_id	Yes	String	Task ID.

Table 4-94 Query Parameters

Parameter	Mandatory	Type	Description
current_path	No	String	Directory or file path.
offset	No	Integer	Offset for pagination.
limit	No	Integer	Number of items displayed on each page. A maximum of 1,000 items are supported.

Request Parameters

None

Response Parameters

Status code: 200

Table 4-95 Response body parameters

Parameter	Type	Description
info	Array of **TreeNode** objects	Directory tree information of a task.
total	Integer	Quantity.

Table 4-96 TreeNode

Parameter	Type	Description
file_name	String	Directory or file name.
file_path	String	Directory or file path.
is_leaf	Boolean	Whether this is a leaf node. The options are **true** (yes) and **false** (no).
checkbox_status	String	Ignoring status. The value can be **unchecked** (not ignored), **all** (all ignored), or **half** (half ignored).

Status code: 400

Table 4-97 Response body parameters

Parameter	Type	Description
error_code	String	Error code.
error_msg	String	Error message.

Status code: 401

Table 4-98 Response body parameters

Parameter	Type	Description
error_code	String	Error code.
error_msg	String	Error message.

Example Requests

```
GET https://{endpoint}/v2/{project_id}/tasks/{task_id}/listpathtree
```

Example Responses

Status code: 200

Request succeeded!

```
{
  "info" : [ {
    "file_name" : ".LAST_RELEASE",
    "file_path" : ".LAST_RELEASE",
    "is_leaf" : true,
    "checkbox_status" : "unchecked"
  } ],
  "total" : 1
}
```

Status code: 400

Bad Request

```
{
  "error_code" : "CC.00000000",
  "error_msg" : "Network busy. Try again later."
}
```

Status code: 401

Unauthorized

```
{
  "error_code" : "CC.00000000",
  "error_msg" : "Network busy. Try again later."
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
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.codecheck.v2.region.CodeCheckRegion;
import com.huaweicloud.sdk.codecheck.v2.*;
import com.huaweicloud.sdk.codecheck.v2.model.*;
```

```
public class ShowTaskPathTreeSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
                .withAk(ak)
                .withSk(sk);

        CodeCheckClient client = CodeCheckClient.newBuilder()
                .withCredential(auth)
                .withRegion(CodeCheckRegion.valueOf("<YOUR REGION>"))
                .build();
        ShowTaskPathTreeRequest request = new ShowTaskPathTreeRequest();
        request.withProjectId("{project_id}");
        request.withTaskId("{task_id}");
        try {
            ShowTaskPathTreeResponse response = client.showTaskPathTree(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());
        }
    }
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcodecheck.v2.region.codecheck_region import CodeCheckRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcodecheck.v2 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]

    credentials = BasicCredentials(ak, sk)

    client = CodeCheckClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CodeCheckRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = ShowTaskPathTreeRequest()
        request.project_id = "{project_id}"
        request.task_id = "{task_id}"
        response = client.show_task_path_tree(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

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

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := codecheck.NewCodeCheckClient(
        codecheck.CodeCheckClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.ShowTaskPathTreeRequest{}
    request.ProjectId = "{project_id}"
    request.TaskId = "{task_id}"
    response, err := client.ShowTaskPathTree(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in **API Explorer**. SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Request succeeded!

Status Code	Description
400	Bad Request
401	Unauthorized

Error Codes

See **Error Codes**.

4.1.16 Ignoring Directories of a Task

Function

This API is used to ignore directories of a task.

Calling Method

For details, see **Calling APIs**.

URI

POST /v2/{project_id}/tasks/{task_id}/config-ignorepath

Table 4-99 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID.
task_id	Yes	String	Task ID.

Request Parameters

Table 4-100 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	It can be obtained by calling the IAM API used to obtain a user token. The value of **X-Subject-Token** in the response header is the user token.
Content-Type	Yes	String	Media type and encoding format.

Table 4-101 Request body parameters

Parameter	Mandatory	Type	Description
ignore_path_settings	Yes	Array of **IgnorePathSettingItem** objects	List of nodes that ignore the directory.

Table 4-102 IgnorePathSettingItem

Parameter	Mandatory	Type	Description
file_path	Yes	String	Directory or file path.
checkbox_status	Yes	String	Ignoring status. The value can be **unchecked** (not ignored), **all** (all ignored), or **half** (half ignored).

Response Parameters

Status code: 400

Table 4-103 Response body parameters

Parameter	Type	Description
error_code	String	Error code.
error_msg	String	Error message.

Status code: 401

Table 4-104 Response body parameters

Parameter	Type	Description
error_code	String	Error code.
error_msg	String	Error message.

Example Requests

Ignore directories of a task: The file path is .LAST_RELEASE, and the check status is all.

```
POST https://{endpoint}/v2/{project_id}/tasks/{task_id}/config-ignorepath
```

```
{
  "ignore_path_settings" : [ {
    "file_path" : ".LAST_RELEASE",
    "checkbox_status" : "all"
  } ]
}
```

Example Responses

Status code: 200

Request succeeded!

```
null
```

Status code: 400

Bad Request

```
{
  "error_code" : "CC.00000000",
  "error_msg" : "Network busy. Try again later."
}
```

Status code: 401

Unauthorized

```
{
  "error_code" : "CC.00000003",
  "error_msg" : "Authentication information expired."
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

Ignore directories of a task: The file path is .LAST_RELEASE, and the check status is all.

```
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.codecheck.v2.region.CodeCheckRegion;
import com.huaweicloud.sdk.codecheck.v2.*;
import com.huaweicloud.sdk.codecheck.v2.model.*;

import java.util.List;
import java.util.ArrayList;

public class UpdateIgnorePathSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");
        ICredential auth = new BasicCredentials()
                .withAk(ak)
                .withSk(sk);

        CodeCheckClient client = CodeCheckClient.newBuilder()
                .withCredential(auth)
                .withRegion(CodeCheckRegion.valueOf("<YOUR REGION>"))
                .build();
        UpdateIgnorePathRequest request = new UpdateIgnorePathRequest();
        request.withProjectId("{project_id}");
        request.withTaskId("{task_id}");
        UpdateIgnorePathRequestBody body = new UpdateIgnorePathRequestBody();
        List<IgnorePathSettingItem> listbodyIgnorePathSettings = new ArrayList<>();
        listbodyIgnorePathSettings.add(
            new IgnorePathSettingItem()
                .withFilePath(".LAST_RELEASE")
                .withCheckboxStatus("all")
        );
        body.withIgnorePathSettings(listbodyIgnorePathSettings);
        request.withBody(body);
        try {
            UpdateIgnorePathResponse response = client.updateIgnorePath(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());
        }
    }
}
```

Python

Ignore directories of a task: The file path is .LAST_RELEASE, and the check status is all.

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcodecheck.v2.region.codecheck_region import CodeCheckRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcodecheck.v2 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]

    credentials = BasicCredentials(ak, sk)

    client = CodeCheckClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CodeCheckRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = UpdateIgnorePathRequest()
        request.project_id = "{project_id}"
        request.task_id = "{task_id}"
        listIgnorePathSettingsbody = [
            IgnorePathSettingItem(
                file_path=".LAST_RELEASE",
                checkbox_status="all"
            )
        ]
        request.body = UpdateIgnorePathRequestBody(
            ignore_path_settings=listIgnorePathSettingsbody
        )
        response = client.update_ignore_path(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

Ignore directories of a task: The file path is .LAST_RELEASE, and the check status is all.

```
package main

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

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := codecheck.NewCodeCheckClient(
        codecheck.CodeCheckClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.UpdateIgnorePathRequest{}
    request.ProjectId = "{project_id}"
    request.TaskId = "{task_id}"
    var listIgnorePathSettingsbody = []model.IgnorePathSettingItem{
        {
            FilePath: ".LAST_RELEASE",
            CheckboxStatus: "all",
        },
    }
    request.Body = &model.UpdateIgnorePathRequestBody{
        IgnorePathSettings: listIgnorePathSettingsbody,
    }
    response, err := client.UpdateIgnorePath(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in **API Explorer**. SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Request succeeded!
400	Bad Request
401	Unauthorized

Error Codes

See **Error Codes**.

4.1.17 Querying Advanced Configurations of a Task

Function

This API is used to query advanced configurations of a task.

Calling Method

For details, see **Calling APIs**.

URI

GET /v2/{project_id}/tasks/{task_id}/settings

Table 4-105 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID.
task_id	Yes	String	Task ID.

Table 4-106 Query Parameters

Parameter	Mandatory	Type	Description
config_list	No	String	List of configuration items. The options are **customImage**, **includePaths**, **authId**, **reviewData**, and **taskName**. You are not advised to query multiple configuration items at the same time.
offset	No	Integer	(Optional) Offset for pagination.
limit	No	Integer	(Optional) Number of items displayed on each page.

Request Parameters

Table 4-107 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Media type and encoding format.
X-Auth-Token	Yes	String	User token. The token can be obtained by calling the IAM API used to obtain a user token. The value of **X-Subject-Token** in the response header is the user token.

Response Parameters

Status code: 200

Table 4-108 Response body parameters

Parameter	Type	Description
info	Array of **TaskAdvancedSettings** objects	Information about an advanced configuration.
total	Integer	Total number.

Table 4-109 TaskAdvancedSettings

Parameter	Type	Description
key	String	Name of an advanced configuration.
value	String	Value of an advanced configuration.
option_value	String	Option for an advanced configuration.
description	String	Description of an advanced configuration.

Status code: 400

Table 4-110 Response body parameters

Parameter	Type	Description
error_code	String	Error code.
error_msg	String	Error message.

Status code: 401

Table 4-111 Response body parameters

Parameter	Type	Description
error_code	String	Error code.
error_msg	String	Error message.

Example Requests

```
GET https://{endpoint}/v2/{project_id}/tasks/{task_id}/settings
```

Example Responses

Status code: 200

Request succeeded!

```
{
  "info" : [ {
    "key" : "customImage",
    "value" : "xxxxx",
    "description" : "Custom image.",
    "option_value" : ""
  } ],
  "total" : 1
}
```

Status code: 400

Bad Request

```
{
  "error_code" : "CC.00000000",
  "error_msg" : "Network busy. Try again later."
}
```

Status code: 401

Unauthorized

```
{
  "error_code" : "CC.00000003",
  "error_msg" : "Authentication information expired."
}
```

Status Codes

Status Code	Description
200	Request succeeded!
400	Bad Request
401	Unauthorized

Error Codes

See **Error Codes**.

4.1.18 Configuring Advanced Configurations for a Task

Function

Configure advanced configurations for a task, such as custom image.

Calling Method

For details, see **Calling APIs**.

URI

POST /v2/{project_id}/tasks/{task_id}/settings

Table 4-112 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID.
task_id	Yes	String	Task ID.

Request Parameters

Table 4-113 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	It can be obtained by calling the IAM API used to obtain a user token. The value of **X-Subject-Token** in the response header is the user token.
Content-Type	Yes	String	Media type and encoding format.

Table 4-114 Request body parameters

Parameter	Mandatory	Type	Description
task_advanced_settings	Yes	Array of **TaskAdvancedSettingsItem** objects	Information about an advanced configuration parameter.

Table 4-115 TaskAdvancedSettingsItem

Parameter	Mandatory	Type	Description
key	Yes	String	Name of an advanced configuration.
value	Yes	String	Value of an advanced configuration.

Response Parameters

Status code: 400

Table 4-116 Response body parameters

Parameter	Type	Description
error_code	String	Error code.
error_msg	String	Error message.

Status code: 401

Table 4-117 Response body parameters

Parameter	Type	Description
error_code	String	Error code.
error_msg	String	Error message.

Example Requests

Configure advanced configurations for a task, such as custom image.

```
POST https://{endpoint}/v2/{project_id}/tasks/{task_id}/settings

{
  "task_advanced_settings" : [ {
    "key" : "customImage",
    "value" : "xxxx"
  } ]
}
```

Example Responses

Status code: 200

Request succeeded!

```
null
```

Status code: 400

Bad Request

```
{
  "error_code" : "CC.00000000",
  "error_msg" : "Network busy. Try again later."
}
```

Status code: 401

Unauthorized

```
{
  "error_code" : "CC.00000003",
  "error_msg" : "Authentication information expired."
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

Configure advanced configurations for a task, such as custom image.

```
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.codecheck.v2.region.CodeCheckRegion;
import com.huaweicloud.sdk.codecheck.v2.*;
import com.huaweicloud.sdk.codecheck.v2.model.*;

import java.util.List;
import java.util.ArrayList;

public class UpdateTaskSettingsSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
                .withAk(ak)
                .withSk(sk);

        CodeCheckClient client = CodeCheckClient.newBuilder()
                .withCredential(auth)
                .withRegion(CodeCheckRegion.valueOf("<YOUR REGION>"))
                .build();
        UpdateTaskSettingsRequest request = new UpdateTaskSettingsRequest();
        request.withProjectId("{project_id}");
        request.withTaskId("{task_id}");
        UpdateTaskSettingsRequestBody body = new UpdateTaskSettingsRequestBody();
        List<TaskAdvancedSettingsItem> listbodyTaskAdvancedSettings = new ArrayList<>();
        listbodyTaskAdvancedSettings.add(
            new TaskAdvancedSettingsItem()
                .withKey("customImage")
                .withValue("xxxx")
        );
        body.withTaskAdvancedSettings(listbodyTaskAdvancedSettings);
        request.withBody(body);
        try {
            UpdateTaskSettingsResponse response = client.updateTaskSettings(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());
        }
    }
}
```

Python

Configure advanced configurations for a task, such as custom image.

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcodecheck.v2.region.codecheck_region import CodeCheckRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcodecheck.v2 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]

    credentials = BasicCredentials(ak, sk)

    client = CodeCheckClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CodeCheckRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = UpdateTaskSettingsRequest()
        request.project_id = "{project_id}"
        request.task_id = "{task_id}"
        listTaskAdvancedSettingsbody = [
            TaskAdvancedSettingsItem(
                key="customImage",
                value="xxxx"
            )
        ]
        request.body = UpdateTaskSettingsRequestBody(
            task_advanced_settings=listTaskAdvancedSettingsbody
        )
        response = client.update_task_settings(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

Configure advanced configurations for a task, such as custom image.

```
package main

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

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := codecheck.NewCodeCheckClient(
        codecheck.CodeCheckClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())
    request := &model.UpdateTaskSettingsRequest{}
    request.ProjectId = "{project_id}"
    request.TaskId = "{task_id}"
    var listTaskAdvancedSettingsbody = []model.TaskAdvancedSettingsItem{
        {
            Key: "customImage",
            Value: "xxxx",
        },
    }
    request.Body = &model.UpdateTaskSettingsRequestBody{
        TaskAdvancedSettings: listTaskAdvancedSettingsbody,
    }
    response, err := client.UpdateTaskSettings(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in **API Explorer**. SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Request succeeded!
400	Bad Request
401	Unauthorized

Error Codes

See **Error Codes**.

4.2 Issue Management

4.2.1 Querying Issue Summary

Function

This API is used to query the issue summary based on the check task ID, including the issue overview, issue status, cyclomatic complexity, and duplication rate.

Calling Method

For details, see **Calling APIs**.

URI

GET /v2/tasks/{task_id}/defects-summary

Table 4-118 Path Parameters

Parameter	Mandatory	Type	Description
task_id	Yes	String	Task ID.

Request Parameters

Table 4-119 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	It can be obtained by calling the IAM API used to obtain a user token. The value of **X-Subject-Token** in the response header is the user token.

Response Parameters

Status code: 200

Table 4-120 Response body parameters

Parameter	Type	Description
task_id	String	Task ID.
task_name	String	Task name.
creator_id	String	Creator ID.
git_url	String	Code repository address.
git_branch	String	Code repository branch. If the MR mode is used, the value is the source branch.
last_check_time	String	Last check time.
code_line_total	Integer	Lines of code.
code_line	Integer	Valid lines of code.
code_quality	Number	Code quality.
issue_count	Integer	Number of issues.

Parameter	Type	Description
risk_coefficient	Number	Risk score.
duplication_ratio	String	Repetition ratio.
complexity_count	Integer	Complexity.
duplicated_lines	Integer	Duplicate lines.
comment_lines	Integer	Number of comment lines.
comment_ratio	String	Comment ratio.
duplicated_blocks	Integer	Duplicate blocks.
last_exec_time	String	Last execution time.
check_type	String	Check type.
created_at	String	Creation time.
cyclomatic_complexity_per_method	String	Average cyclomatic complexity.
cyclomatic_complexity_per_file	String	Average code complexity (file).
critical_count	String	Critical issues.
major_count	String	Major issues.
minor_count	String	Minor issues.
suggestion_count	String	Suggestion issues.
is_access	String	Whether the gated check-in is passed.
trigger_type	String	Task triggering mode.
file_duplication_ratio	String	File duplication rate.
new_count	Integer	New issues.
solve_count	Integer	Resolved issues.

Parameter	Type	Description
duplicated_files	Integer	Duplicate files.
new_critical_count	String	New critical issues.
new_major_count	String	New major issues.
new_minor_count	String	New minor issues.
new_suggestion_count	String	New suggestion issues.

Status code: 400

Table 4-121 Response body parameters

Parameter	Type	Description
error_code	String	Error code.
error_msg	String	Error message.

Status code: 401

Table 4-122 Response body parameters

Parameter	Type	Description
error_code	String	Error code.
error_msg	String	Error message.

Example Requests

```
GET https://{endpoint}/v2/tasks/2b31ed520xxxxxxebedb6e57xxxxxxxx/defects-summary
```

Example Responses

Status code: 200

Request succeeded!

```
{
  "task_id" : "435b58ecf7f54c45907c1384d01c94a6",
  "task_name" : "CSharp_02201119",
  "creator_id" : "ecadebb5041146cc96f9780f3d905e20",
  "git_url" : "git@****_02201119.git",
  "git_branch" : "master",
  "last_check_time" : "2020-02-20 11:37:46",
  "code_line_total" : 20,
  "code_line" : 156,
  "code_quality" : 147.4,
  "issue_count" : 3,
  "risk_coefficient" : 23,
  "duplication_ratio" : "0.0%",
  "complexity_count" : 0,
  "duplicated_lines" : 0,
  "comment_lines" : 0,
  "comment_ratio" : "0",
  "duplicated_blocks" : 0,
  "last_exec_time" : "2 minutes and 28 seconds",
  "check_type" : "source",
  "created_at" : "2020-02-20 11:21:11",
  "cyclomatic_complexity_per_method" : "84.0",
  "cyclomatic_complexity_per_file" : "112.0",
  "critical_count" : 2,
  "major_count" : 1,
  "minor_count" : 1,
  "suggestion_count" : 1,
  "is_access" : 0,
  "trigger_type" : 1,
  "file_duplication_ratio" : "33.3%",
  "new_count" : 0,
  "solve_count" : 0,
  "duplicated_files" : 0,
  "new_critical_count" : "0",
  "new_major_count" : "0",
  "new_minor_count" : "0",
  "new_suggestion_count" : "0"
}
```

Status code: 400

Bad Request

```
{
  "error_code" : "CC.00000000",
  "error_msg" : "Network busy. Try again later."
}
```

Status code: 401

Unauthorized

```
{
  "error_code" : "CC.00000003",
  "error_msg" : "Authentication information expired."
}
```

Status Codes

Status Code	Description
200	Request succeeded!
400	Bad Request
401	Unauthorized

Error Codes

See **Error Codes**.

4.2.2 Querying the CMetrics Issue Summary

Function

This API is used to query the CMetrics issue summary based on the check task ID.

Calling Method

For details, see **Calling APIs**.

URI

GET /v2/{project_id}/tasks/{task_id}/metrics-summary

Table 4-123 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID.
task_id	Yes	String	Task ID.

Request Parameters

Table 4-124 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	It can be obtained by calling the IAM API used to obtain a user token. The value of **X-Subject-Token** in the response header is the user token.

Response Parameters

Status code: 200

Table 4-125 Response body parameters

Parameter	Type	Description
task_id	String	Task ID.
task_name	String	Task name.

Parameter	Type	Description
creator_id	String	Creator ID.
git_url	String	Code repository address.
git_branch	String	Code repository branch.
last_check_time	String	Last check time.
last_exec_time	String	Last execution time.
check_type	String	Check type.
created_at	String	Creation time.
metric_info	**MetricInfo** object	Metrics.

Table 4-126 MetricInfo

Parameter	Type	Description
code_size	String	Code size.
raw_lines	String	Total lines.
methods_total	String	Total functions.
cyclomatic_complexity_total	String	Total cyclomatic complexity.
cyclomatic_complexity_per_method	String	Average cyclomatic complexity.
maximum_cyclomatic_complexity	String	Maximum cyclomatic complexity.
huge_cyclomatic_complexity_total	String	Huge cyclomatic complexity functions.
huge_cyclomatic_complexity_ratio	String	Ratio of huge cyclomatic complexity functions.
cca_cyclomatic_complexity_total	String	Total CCA cyclomatic complexity.

Parameter	Type	Description
cca_cyclomatic_complexity_per_method	String	Average CCA cyclomatic complexity.
maximum_cca_cyclomatic_complexity	String	Maximum CCA cyclomatic complexity.
huge_cca_cyclomatic_complexity_total	String	Number of huge CCA cyclomatic complexity functions.
cyclomatic_complexity_adequacy	String	Cyclomatic complexity adequacy.
maximum_depth	String	Maximum depth.
huge_depth_total	String	Number of huge depths.
huge_depth_ratio	String	Ratio of huge depths.
method_lines	String	Number of lines containing functions.
lines_per_method	String	Average code lines containing functions.
huge_method_total	String	Number of huge functions.
huge_method_ratio	String	Ratio of huge functions.
files_total	String	Total files.
folders_total	String	Total directories.
lines_per_file	String	Average code lines in a file.
huge_headerfile_total	String	Huge header files.
huge_headerfile_ratio	String	Ratio of huge header files.
huge_non_headerfile_total	String	Huge source files.
huge_non_headerfile_ratio	String	Ratio of huge source files.
huge_folder_total	String	Huge directories.

Parameter	Type	Description
huge_folder_ratio	String	Ratio of huge source directories.
file_duplication_total	String	Duplicate files.
file_duplication_ratio	String	File duplication rate.
non_hfile_duplication_total	String	Duplicate source files.
non_hfile_duplication_ratio	String	Source file duplication rate.
code_duplication_total	String	Duplicate code.
code_duplication_ratio	String	Code duplication rate.
non_hfile_code_duplication_total	String	Duplicate source files.
non_hfile_code_duplication_ratio	String	Source file code duplication rate.
unsafe_functions_total	String	Unsafe functions.
unsafe_functions_kloc	String	Density of unsafe functions.
redundant_code_total	String	Redundant code.
redundant_code_kloc	String	Density of redundant code blocks.
warning_suppression_total	String	Suppressed alarms.
warning_suppression_kloc	String	Density of suppressed alarms.

Status code: 400

Table 4-127 Response body parameters

Parameter	Type	Description
error_code	String	Error code.
error_msg	String	Error message.

Status code: 401

Table 4-128 Response body parameters

Parameter	Type	Description
error_code	String	Error code.
error_msg	String	Error message.

Example Requests

```
GET https://{endpoint}/v2/{project_id}/tasks/{task_id}/metrics-summary
```

Example Responses

Status code: 200

Request succeeded!

```
{
  "task_id" : "435b58ecf7f54c45907c1384d01c94a6",
  "task_name" : "CSharp_02201119",
  "creator_id" : "ecadebb5041146cc96f9780f3d905e20",
  "git_url" : "git@****_02201119.git",
  "git_branch" : "master",
  "last_check_time" : "2020-02-20 11:37:46",
  "last_exec_time" : "2020-02-20 11:37:46",
  "check_type" : "source",
  "created_at" : "2020-02-20 11:21:11",
  "metric_info" : {
    "code_size" : "133426.00",
    "raw_lines" : "182082.00",
    "methods_total" : "6840.00",
    "cyclomatic_complexity_total" : "27435.00",
    "cyclomatic_complexity_per_method" : "7.43",
    "maximum_cyclomatic_complexity" : "346.00",
    "huge_cyclomatic_complexity_total" : "167.00",
    "huge_cyclomatic_complexity_ratio" : "4.13",
    "cca_cyclomatic_complexity_total" : "26693.00",
    "cca_cyclomatic_complexity_per_method" : "7.25",
    "maximum_cca_cyclomatic_complexity" : "342.00",
    "huge_cca_cyclomatic_complexity_total" : "148.00",
    "cyclomatic_complexity_adequacy" : "21.00",
    "maximum_depth" : "333.00",
    "huge_depth_total" : "8.85",
    "huge_depth_ratio" : "91546.00",
    "method_lines" : "24.91",
    "lines_per_method" : "8.04",
    "huge_method_total" : "12300",
    "huge_method_ratio" : "12300",
    "files_total" : "629.00",
    "folders_total" : "426.00",
    "lines_per_file" : "12300",
    "huge_headerfile_total" : "12300",
    "huge_headerfile_ratio" : "12300",
    "huge_non_headerfile_total" : "12300",
    "huge_non_headerfile_ratio" : "12300",
    "huge_folder_total" : "12300",
    "huge_folder_ratio" : "12300",
    "file_duplication_total" : "12300",
    "file_duplication_ratio" : "12300",
    "non_hfile_duplication_total" : "12300",
    "non_hfile_duplication_ratio" : "12300",
    "code_duplication_total" : "3.28",
    "code_duplication_ratio" : "11782.00",
    "non_hfile_code_duplication_total" : "8.59",
    "non_hfile_code_duplication_ratio" : "11782.00",
    "unsafe_functions_total" : "17.17",
    "unsafe_functions_kloc" : "0.00",
    "redundant_code_total" : "0.00",
    "redundant_code_kloc" : "674.00",
    "warning_suppression_total" : "17.00",
    "warning_suppression_kloc" : "0.45"
  }
}
```

Status code: 400

Bad Request

```
{
  "error_code" : "CC.00000000",
  "error_msg" : "Network busy. Try again later."
}
```

Status code: 401

Unauthorized

```
{
  "error_code" : "CC.00000003",
  "error_msg" : "Authentication information expired."
}
```

Status Codes

Status Code	Description
200	Request succeeded!
400	Bad Request
401	Unauthorized

Error Codes

See **Error Codes**.

4.2.3 Querying Issue Details

Function

This API is used to query issue details by task ID.

Calling Method

For details, see **Calling APIs**.

URI

GET /v2/tasks/{task_id}/defects-detail

Table 4-129 Path Parameters

Parameter	Mandatory	Type	Description
task_id	Yes	String	Task ID.

Table 4-130 Query Parameters

Parameter	Mandatory	Type	Description
offset	No	Integer	Offset for pagination.
limit	No	Integer	Number of items displayed on each page. A maximum of 100 items are supported.
status_ids	No	String	Issue status. Filter issues by status.
severity	No	String	Issue severity. **0**: critical; **1**: major; **2**: minor; **3**: suggestion.

Request Parameters

Table 4-131 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	It can be obtained by calling the IAM API used to obtain a user token. The value of **X-Subject-Token** in the response header is the user token.

Response Parameters

Status code: 200

Table 4-132 Response body parameters

Parameter	Type	Description
defects	Array of **DefectInfoV2** objects	Issue details.
total	Integer	Total number.

Table 4-133 DefectInfoV2

Parameter	Type	Description
defect_id	String	Issue ID.
defect_checker_name	String	Name of the check item corresponding to the issue.
defect_status	String	Issue status. **0**: unresolved; **1**: resolved; **2**: ignored.
rule_system_tags	String	Rule tag. Multiple tags are separated by commas (,).
rule_id	String	Rule ID.
rule_name	String	Rule name.
line_number	String	Line No. of the file where the issue is located.
defect_content	String	Issue description.
defect_level	String	Issue severity. **0**: critical; **1**: major; **2**: minor; **3**: suggestion.
file_path	String	Issue file path.
created_at	String	Creation time.
issue_key	String	Unique ID of an issue.
fragment	Array of **DefectFragmentV2** objects	Issue code snippet details.
events	Array of **DefectEvents** objects	Trace information.

Table 4-134 DefectFragmentV2

Parameter	Type	Description
line_num	String	Line No.
line_content	String	Code line content.
start_offset	Integer	The sequence number of the column where an issue starts.
end_offset	Integer	The sequence number of the column where an issue ends.

Table 4-135 DefectEvents

Parameter	Type	Description
events	Array of **DefectEvents** objects	Trace information.
description	String	Description.
fix_suggestions	Array of strings	Fix suggestion.
line	Integer	Line that has an issue.
end_line	Integer	Line where an issue ends.
code_context_start_line	Integer	Line where an issue starts
main	Boolean	Code snippet.
path	String	Path.
tag	String	Tag.
main_buggy_code	String	Code snippet corresponding to the main event.
code_context	String	Code context.

Status code: 400

Table 4-136 Response body parameters

Parameter	Type	Description
error_code	String	Error code.
error_msg	String	Error message.

Status code: 401

Table 4-137 Response body parameters

Parameter	Type	Description
error_code	String	Error code.
error_msg	String	Error message.

Example Requests

```
GET https://{endpoint}/v2/tasks/2b31ed520xxxxxxebedb6e57xxxxxxxx/defects-summary
```

Example Responses

Status code: 200

Request succeeded!

```
{
  "defects" : [ {
    "line_number" : "81",
    "defect_id" : "5b71a4594bcb4a69a33f576c3564e6cc",
    "rule_system_tags" : "fossscan",
    "rule_id" : "111",
    "rule_name" : "FossScan.1 OpenSource Software",
    "defect_checker_name" : "Disallow fallthrough of `case` statements",
    "defect_status" : "0",
    "defect_content" : "Update or refactor this function so that its implementation doesn't duplicate the one on line 60.",
    "defect_level" : "1",
    "file_path" : "code/types/functions.ts",
    "created_at" : "2020-07-21T02:12:01Z",
    "issue_key" : "AA76E96XXXXXXXXXXXXXXXXXCAAE5",
    "fragment" : [ {
      "line_num" : "78",
      "line_content" : "    export function padding(all: number);",
      "start_offset" : -1,
      "end_offset" : -1
    }, {
      "line_num" : "79",
      "line_content" : "    export function padding(topAndBottom: number, leftAndRight: number);",
      "start_offset" : -1,
      "end_offset" : -1
    }, {
      "line_num" : "80",
      "line_content" : "    export function padding(top: number, right: number, bottom: number, left: number);",
      "start_offset" : -1,
      "end_offset" : -1
    }, {
      "line_num" : "81",
      "line_content" : "    export function padding(a: number, b?: number, c?: number, d?: number) {",
      "start_offset" : 11,
      "end_offset" : 19
    }, {
      "line_num" : "82",
      "line_content" : "        if (b === undefined && c === undefined && d === undefined) {",
      "start_offset" : -1,
      "end_offset" : -1
    } ]
  }, {
    "line_number" : "190",
    "defect_id" : "5b71a4594wtrfsgrew546t466c3564e6cc",
    "rule_system_tags" : "fossscan",
    "rule_id" : "112",
    "rule_name" : "FossScan.1 OpenSource Software",
    "defect_checker_name" : "Disallow fallthrough of `case` statements",
    "defect_status" : "0",
    "defect_content" : "Update or refactor this function so that its implementation doesn't duplicate the one on line 173.",
    "defect_level" : "1",
    "file_path" : "code/types/types.ts",
    "created_at" : "2020-07-21T02:12:01Z",
    "issue_key" : "AA76E96XXXXXXXXXXXXXXXXXFE3B6",
    "fragment" : [ {
      "line_num" : "187",
      "line_content" : "    var _value;",
      "start_offset" : -1,
      "end_offset" : -1
    }, {
      "line_num" : "188",
      "line_content" : "    function getOrSet(): number;",
      "start_offset" : -1,
      "end_offset" : -1
    }, {
      "line_num" : "189",
      "line_content" : "    function getOrSet(value: number);",
      "start_offset" : -1,
      "end_offset" : -1
    }, {
      "line_num" : "190",
      "line_content" : " \"    function getOrSet(value?: number) {\"",
      "start_offset" : 4,
      "end_offset" : 12
    }, {
      "line_num" : "191",
      "line_content" : " \"        if (value === undefined) {\"",
      "start_offset" : -1,
      "end_offset" : -1
    } ]
  } ],
  "total" : 2
}
```

Status code: 400

Bad Request

```
{
  "error_code" : "CC.00000000",
  "error_msg" : "Network busy. Try again later."
}
```

Status code: 401

Unauthorized

```
{
  "error_code" : "CC.00000003",
  "error_msg" : "Authentication information expired."
}
```

Status Codes

Status Code	Description
200	Request succeeded!

Status Code	Description
400	Bad Request
401	Unauthorized

Error Codes

See **Error Codes**.

4.2.4 Querying Issue Details Statistics

Function

This API is used to query issue details statistics based on the task ID.

Calling Method

For details, see **Calling APIs**.

URI

GET /v2/tasks/{task_id}/defects-statistic

Table 4-138 Path Parameters

Parameter	Mandatory	Type	Description
task_id	Yes	String	Task ID.

Request Parameters

Table 4-139 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	It can be obtained by calling the IAM API used to obtain a user token. The value of **X-Subject-Token** in the response header is the user token.

Response Parameters

Status code: 200

Table 4-140 Response body parameters

Parameter	Type	Description
severity	**StatisticSeverityV2** object	Issue level statistics.
status	**StatisticStatusV2** object	Issue status statistics.

Table 4-141 StatisticSeverityV2

Parameter	Type	Description
critical	Integer	Critical issues.
major	Integer	Major issues.
minor	Integer	Minor issues.
suggestion	Integer	Suggestion issues.

Table 4-142 StatisticStatusV2

Parameter	Type	Description
unresolved	Integer	Unresolved.
resolved	Integer	Resolved.
dismissed	Integer	Ignored.

Status code: 400

Table 4-143 Response body parameters

Parameter	Type	Description
error_code	String	Error code.
error_msg	String	Error message.

Status code: 401

Table 4-144 Response body parameters

Parameter	Type	Description
error_code	String	Error code.
error_msg	String	Error message.

Example Requests

```
GET https://{endpoint}/v2/tasks/2b31ed520xxxxxxebedb6e57xxxxxxxx/defects-statistic
```

Example Responses

Status code: 200

Request succeeded!

```
{
  "severity" : {
    "critical" : 120,
    "major" : 877,
    "minor" : 79,
    "suggestion" : 3
  },
  "status" : {
    "unresolved" : 877,
    "resolved" : 79,
    "dismissed" : 5
  }
}
```

Status code: 400

Bad Request

```
{
  "error_code" : "CC.00000000",
  "error_msg" : "Network busy. Try again later."
}
```

Status code: 401

Unauthorized

```
{
  "error_code" : "CC.00000003",
  "error_msg" : "Authentication information expired."
}
```

Status Codes

Status Code	Description
200	Request succeeded!
400	Bad Request
401	Unauthorized

Error Codes

See **Error Codes**.

4.2.5 Changing the Issue Status

Function

This API is used to change the issue status to **Resolved** or **Ignored**.

Calling Method

For details, see **Calling APIs**.

URI

PUT /v2/tasks/{task_id}/defect-status

Table 4-145 Path Parameters

Parameter	Mandatory	Type	Description
task_id	Yes	String	Task ID.

Request Parameters

Table 4-146 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	It can be obtained by calling the IAM API used to obtain a user token. The value of **X-Subject-Token** in the response header is the user token.
Content-Type	Yes	String	Media type and encoding format.

Table 4-147 Request body parameters

Parameter	Mandatory	Type	Description
defect_id	Yes	String	Issue ID. Use commas (,) to separate multiple IDs.
defect_status	Yes	String	Status. **0**: unresolved; **1**: resolved; **2**: ignored.

Response Parameters

Status code: 400

Table 4-148 Response body parameters

Parameter	Type	Description
error_code	String	Error code.
error_msg	String	Error message.

Status code: 401

Table 4-149 Response body parameters

Parameter	Type	Description
error_code	String	Error code.
error_msg	String	Error message.

Example Requests

Change the issue status to **1**. The issue status ID is displayed based on the site requirements.

```
PUT https://{endpoint}/v2/tasks/{taskid}/defect-status

{
  "defect_id" : "363540xxxxxxxx5105099944xxxxxxxx",
  "defect_status" : "1"
}
```

Example Responses

Status code: 200

Request succeeded!

```
null
```

Status code: 400

Bad Request

```
{
  "error_code" : "CC.00000000",
  "error_msg" : "Network busy. Try again later."
}
```

Status code: 401

Unauthorized

```
{
  "error_code" : "CC.00000003",
  "error_msg" : "Authentication information expired."
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

Change the issue status to **1**. The issue status ID is displayed based on the site requirements.

```
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.codecheck.v2.region.CodeCheckRegion;
import com.huaweicloud.sdk.codecheck.v2.*;
import com.huaweicloud.sdk.codecheck.v2.model.*;


public class UpdateDefectStatusSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
                .withAk(ak)
                .withSk(sk);

        CodeCheckClient client = CodeCheckClient.newBuilder()
                .withCredential(auth)
                .withRegion(CodeCheckRegion.valueOf("<YOUR REGION>"))
                .build();
        UpdateDefectStatusRequest request = new UpdateDefectStatusRequest();
        request.withTaskId("{task_id}");
        UpdateDefectRequestBody body = new UpdateDefectRequestBody();
        body.withDefectStatus("1");
        body.withDefectId("363540xxxxxxxx5105099944xxxxxxxx");
        request.withBody(body);
        try {
            UpdateDefectStatusResponse response = client.updateDefectStatus(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());
        }
    }
}
```

Python

Change the issue status to **1**. The issue status ID is displayed based on the site requirements.

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcodecheck.v2.region.codecheck_region import CodeCheckRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcodecheck.v2 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]

    credentials = BasicCredentials(ak, sk)

    client = CodeCheckClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CodeCheckRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = UpdateDefectStatusRequest()
        request.task_id = "{task_id}"
        request.body = UpdateDefectRequestBody(
            defect_status="1",
            defect_id="363540xxxxxxxx5105099944xxxxxxxx"
        )
        response = client.update_defect_status(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

Change the issue status to **1**. The issue status ID is displayed based on the site requirements.

```
package main

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

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")
    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := codecheck.NewCodeCheckClient(
        codecheck.CodeCheckClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.UpdateDefectStatusRequest{}
    request.TaskId = "{task_id}"
    defectStatusUpdateDefectRequestBody:= "1"
    defectIdUpdateDefectRequestBody:= "363540xxxxxxxx5105099944xxxxxxxx"
    request.Body = &model.UpdateDefectRequestBody{
        DefectStatus: &defectStatusUpdateDefectRequestBody,
        DefectId: &defectIdUpdateDefectRequestBody,
    }
    response, err := client.UpdateDefectStatus(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in **API Explorer**. SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Request succeeded!
400	Bad Request
401	Unauthorized

Error Codes

See **Error Codes**.

4.3 Rule Management

4.3.1 Obtaining Rules

Function

This API is used to query the rules based on conditions such as language and issue severity.

Calling Method

For details, see **Calling APIs**.

URI

GET /v2/rules

Table 4-150 Query Parameters

Parameter	Mandatory	Type	Description
rule_languages	No	String	Language corresponding to a rule.
rule_severity	No	String	Issue severity. **0**: critical; **1**: major; **2**: minor; **3**: suggestion.
offset	No	Integer	Offset for pagination.
limit	No	Integer	Number of items on each page.

Request Parameters

Table 4-151 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token. It can be obtained by calling the IAM API used to obtain a user token. The value of **X-Subject-Token** in the response header is the user token.

Response Parameters

Status code: 200

Table 4-152 Response body parameters

Parameter	Type	Description
info	Array of **RuleListItem** objects	Rule information.
total	Integer	Total number.

Table 4-153 RuleListItem

Parameter	Type	Description
rule_set	String	Rule set classification.
rule_id	String	Rule ID.
rule_language	String	Language corresponding to the rule.
rule_name	String	Rule name.
rule_severity	String	Severity of issues corresponding to the rule.
rule_tages	String	Rule tag.
right_example	String	Compliant example.
error_example	String	Noncompliant example.
revise_opinion	String	Fix suggestion.
rule_desc	String	Rule description.

Status code: 400

Table 4-154 Response body parameters

Parameter	Type	Description
error_code	String	Error code.
error_msg	String	Error message.

Status code: 401

Table 4-155 Response body parameters

Parameter	Type	Description
error_code	String	Error code.
error_msg	String	Error message.

Example Requests

```
GET https://{endpoint}/v2/rules
```

Example Responses

Status code: 200

Request succeeded!

```
{
  "info" : [ {
    "rule_id" : "2561",
    "rule_language" : "Java",
    "rule_name" : "\"=+\" cannot be replaced by \"+=\".",
    "rule_severity" : "1",
    "rule_tages" : "bug",
    "rule_desc" : "<p>Use strict equality operator (<code>===</code>) to compare different types. The result is always <code>false</code>. </p>",
    "right_example" : "<pre>\nvar a = 8;\nvar b = \"8\";\n\nif (a == b) {\n  // ...\n}\n</pre>\n<p>or</p>\n<pre>\nvar a = 8;\nvar b = \"8\";\n\nif (a === Number(b)) {\n  // ...\n}\n</pre>",
    "error_example" : "<pre>\nvar a = 8;\nvar b = \"8\";\n\nif (a === b) {  //Noncompliant, always false\n  // ...\n}\n</pre>",
    "revise_opinion" : "None",
    "rule_set" : "HuaWeiJava"
  }, {
    "rule_id" : "8048",
    "rule_language" : "Java",
    "rule_name" : "\"Arrays.stream\" should be used for primitive arrays",
    "rule_severity" : "1",
    "rule_tages" : "performance",
    "rule_desc" : "<p>Use strict equality operator (<code>===</code>) to compare different types. The result is always <code>false</code>. </p>",
    "right_example" : "<pre>\nvar a = 8;\nvar b = \"8\";\n\nif (a == b) {\n  // ...\n}\n</pre>\n<p>or</p>\n<pre>\nvar a = 8;\nvar b = \"8\";\n\nif (a === Number(b)) {\n  // ...\n}\n</pre>",
    "error_example" : "<pre>\nvar a = 8;\nvar b = \"8\";\n\nif (a === b) {  //Noncompliant, always false\n  // ...\n}\n</pre>",
    "revise_opinion" : "None",
    "rule_set" : "HuaWeiJava"
  } ],
  "total" : 2
}
```

Status code: 400

Bad Request

```
{
  "error_code" : "CC.00000000",
  "error_msg" : "Network busy. Try again later."
}
```

Status code: 401

Unauthorized

```
{
  "error_code" : "CC.00000003",
  "error_msg" : "Authentication information expired."
}
```

Status Codes

Status Code	Description
200	Request succeeded!
400	Bad Request
401	Unauthorized

Error Codes

See **Error Codes**.

4.3.2 Customizing a Rule Set

Function

This API is used to combine rules as required.

Calling Method

For details, see **Calling APIs**.

URI

POST /v2/ruleset

Request Parameters

Table 4-156 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Media type and encoding format.
X-Auth-Token	Yes	String	User token. It can be obtained by calling the IAM API used to obtain a user token. The value of **X-Subject-Token** in the response header is the user token.

Table 4-157 Request body parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID.
template_name	Yes	String	New rule set name.
language	Yes	String	Rule set language.
is_default	Yes	String	**0**: with base rule set; **1**: without base rule set.
rule_ids	Yes	String	ID of the newly enabled rule.
uncheck_ids	No	String	ID of the new closed rule.

Parameter	Mandatory	Type	Description
template_id	No	String	Rule set ID.
custom_attributes	No	Array of **CustomAttributes** objects	Customized rule parameter. Modifying rule thresholds is supported.

Table 4-158 CustomAttributes

Parameter	Mandatory	Type	Description
attribute	No	String	Configuration item attribute. Severity indicates the issue severity.
rules	No	Array of **CustomAttributesRule** objects	Rule details.

Table 4-159 CustomAttributesRule

Parameter	Mandatory	Type	Description
rule_id	No	String	Rule ID.
value	No	String	Attribute issue level. **0**: critical; **1**: major; **2**: minor; **3**: suggestion.
rule_config_list	No	Array of **RuleConfig** objects	Rule threshold details.

Table 4-160 RuleConfig

Parameter	Mandatory	Type	Description
id	No	Integer	Rule configuration ID.
rule_id	No	Integer	Rule ID.
default_value	No	String	Default value.
option_value	No	String	Current.
option_key	No	String	Key of the current rule configuration item.

Parameter	Mandatory	Type	Description
option_name	No	String	Name of the current rule configuration item.
template_id	No	String	Rule set ID.
description	No	String	Description.

Response Parameters

Status code: 201

Table 4-161 Response body parameters

Parameter	Type	Description
template_id	String	Rule set ID.

Status code: 400

Table 4-162 Response body parameters

Parameter	Type	Description
error_code	String	Error code.
error_msg	String	Error message.

Status code: 401

Table 4-163 Response body parameters

Parameter	Type	Description
error_code	String	Error code.
error_msg	String	Error message.

Example Requests

Customize a rule set: Set template name to test_cpp and language to cpp. The project ID and template ID corresponding to the rule set are displayed based on the site requirements.

```
POST https://{endpoint}/v2/ruleset

{
  "project_id" : "24b97exxxxxxxxxfb912625b14cxxxxxx",
  "template_id" : "da303dxxxxxxxxaca60dbcc2e2xxxxxx",
  "template_name" : "test_cpp",
  "language" : "cpp",
  "is_default" : "1",
  "rule_ids" : "8139,8138",
  "uncheck_ids" : "1101,1102"
}
```

Example Responses

Status code: 201

Request succeeded!

```
{
  "template_id" : "c53417a4804f45eba9c11991131c9e79"
}
```

Status code: 400

Bad Request

```
{
  "error_code" : "CC.00000000",
  "error_msg" : "Network busy. Try again later."
}
```

Status code: 401

Unauthorized

```
{
  "error_code" : "CC.00000003",
  "error_msg" : "Authentication information expired."
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

Customize a rule set: Set template name to test_cpp and language to cpp. The project ID and template ID corresponding to the rule set are displayed based on the site requirements.

```
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.codecheck.v2.region.CodeCheckRegion;
import com.huaweicloud.sdk.codecheck.v2.*;
import com.huaweicloud.sdk.codecheck.v2.model.*;


public class CreateRulesetSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
environment variables and decrypted during use to ensure security.
```

```
        // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
                .withAk(ak)
                .withSk(sk);

        CodeCheckClient client = CodeCheckClient.newBuilder()
                .withCredential(auth)
                .withRegion(CodeCheckRegion.valueOf("<YOUR REGION>"))
                .build();
        CreateRulesetRequest request = new CreateRulesetRequest();
        Ruleset body = new Ruleset();
        body.withTemplateId("da303dxxxxxxxxaca60dbcc2e2xxxxxx");
        body.withUncheckIds("1101,1102");
        body.withRuleIds("8139,8138");
        body.withIsDefault("1");
        body.withLanguage("cpp");
        body.withTemplateName("test_cpp");
        body.withProjectId("24b97exxxxxxxxfb912625b14cxxxxxx");
        request.withBody(body);
        try {
            CreateRulesetResponse response = client.createRuleset(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());
        }
    }
}
```

Python

Customize a rule set: Set template name to test_cpp and language to cpp. The project ID and template ID corresponding to the rule set are displayed based on the site requirements.

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcodecheck.v2.region.codecheck_region import CodeCheckRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcodecheck.v2 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]

    credentials = BasicCredentials(ak, sk)

    client = CodeCheckClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CodeCheckRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = CreateRulesetRequest()
        request.body = Ruleset(
            template_id="da303dxxxxxxxxaca60dbcc2e2xxxxxx",
            uncheck_ids="1101,1102",
            rule_ids="8139,8138",
            is_default="1",
            language="cpp",
            template_name="test_cpp",
            project_id="24b97exxxxxxxxfb912625b14cxxxxxx"
        )
        response = client.create_ruleset(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

Customize a rule set: Set template name to test_cpp and language to cpp. The project ID and template ID corresponding to the rule set are displayed based on the site requirements.

```
package main

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

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := codecheck.NewCodeCheckClient(
        codecheck.CodeCheckClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.CreateRulesetRequest{}
    templateIdRuleset:= "da303dxxxxxxxxaca60dbcc2e2xxxxxx"
    uncheckIdsRuleset:= "1101,1102"
    request.Body = &model.Ruleset{
        TemplateId: &templateIdRuleset,
        UncheckIds: &uncheckIdsRuleset,
        RuleIds: "8139,8138",
        IsDefault: "1",
        Language: "cpp",
        TemplateName: "test_cpp",
        ProjectId: "24b97exxxxxxxxfb912625b14cxxxxxx",
    }
    response, err := client.CreateRuleset(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in **API Explorer**. SDK sample code can be automatically generated.

Status Codes

Status Code	Description
201	Request succeeded!
400	Bad Request
401	Unauthorized

Error Codes

See **Error Codes**.

4.3.3 Querying Rule Sets

Function

This API is used to query rule sets based on conditions such as the project ID and language.

Calling Method

For details, see **Calling APIs**.

URI

GET /v2/{project_id}/rulesets

Table 4-164 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID.

Table 4-165 Query Parameters

Parameter	Mandatory	Type	Description
category	No	String	Rule set category. **0**: system rule sets; **1**: custom rule sets of current user; **2**: custom rule sets of other users; **0,1,2**: all rule sets.
offset	No	Integer	Offset for pagination.
limit	No	Integer	Number of items on each page.

Request Parameters

Table 4-166 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token. It can be obtained by calling the IAM API used to obtain a user token. The value of **X-Subject-Token** in the response header is the user token.

Response Parameters

Status code: 200

Table 4-167 Response body parameters

Parameter	Type	Description
info	Array of **RulesetItem** objects	Rule set list information.
total	Integer	Total number.

Table 4-168 RulesetItem

Parameter	Type	Description
template_id	String	Rule set ID.

Parameter	Type	Description
language	String	Rule set language.
template_name	String	Rule set name.
creator_id	String	Creator ID.
creator_name	String	Creator name.
template_create_time	String	Creation time.
is_used	String	Usage status. **1**: in use; **0**: idle.
rule_ids	String	Rule IDs in a rule set.
is_default	String	Whether the rule set is the default rule set of the language. **0**: no; **1**: yes.
is_devcloud_project_default	String	Whether the rule set is the default rule set of the language in a project. **0**: no; **1**: yes.
is_default_template	String	Whether the rule set is a system rule set. **0**: no; **1**: yes.

Status code: 400

Table 4-169 Response body parameters

Parameter	Type	Description
error_code	String	Error code.
error_msg	String	Error message.

Status code: 401

Table 4-170 Response body parameters

Parameter	Type	Description
error_code	String	Error code.
error_msg	String	Error message.

Example Requests

```
GET https://{endpoint}/v2/{project_id}/rulesets
```

Example Responses

Status code: 200

Request succeeded!

```
{
  "info" : [ {
    "template_id" : "9698e0cb9d2441c3bd8cec5f8641696e",
    "language" : "Java",
    "template_name" : "test1111",
    "creator_id" : "ae161856f2604229ae12056478a7919f",
    "creator_name" : "Mr. Yang",
    "template_create_time" : "2021-10-21T02:14:55Z",
    "is_used" : "0",
    "rule_ids" : "",
    "is_default" : "0",
    "is_default_template" : "1",
    "is_devcloud_project_default" : "0"
  }, {
    "template_id" : "9698e0cb9d2441c3bd8cec5f8641696e",
    "language" : "cpp",
    "template_name" : "test222",
    "creator_id" : "ae161856f2604229ae12056478a7919f",
    "creator_name" : "Mr. Yang",
    "template_create_time" : "2021-10-21T02:14:55Z",
    "is_used" : "1",
    "rule_ids" : "11152,11153,11154,11155",
    "is_default" : "0",
    "is_default_template" : "1",
    "is_devcloud_project_default" : "1"
  } ],
  "total" : 2
}
```

Status code: 400

Bad Request

```
{
  "error_code" : "CC.00000000",
  "error_msg" : "Network busy. Try again later."
}
```

Status code: 401

Unauthorized

```
{
  "error_code" : "CC.00000003",
  "error_msg" : "Authentication information expired."
}
```

Status Codes

Status Code	Description
200	Request succeeded!
400	Bad Request
401	Unauthorized

Error Codes

See **Error Codes**.

4.3.4 Viewing Rules of a Rule Set

Function

This API is used to query rules based on conditions such as the project ID and rule set ID.

Calling Method

For details, see **Calling APIs**.

URI

GET /v2/{project_id}/ruleset/{ruleset_id}/rules

Table 4-171 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID.
ruleset_id	Yes	String	Rule set ID.

Table 4-172 Query Parameters

Parameter	Mandatory	Type	Description
types	Yes	String	Rule status. **1**: all; **2**: enabled; **3**: disabled.
languages	No	String	Rule language.
tags	No	String	Tag to which a rule belongs.
offset	No	Integer	Offset for pagination.
limit	No	Integer	Number of items on each page.

Request Parameters

Table 4-173 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token. It can be obtained by calling the IAM API used to obtain a user token. The value of **X-Subject-Token** in the response header is the user token.

Response Parameters

Status code: 200

Table 4-174 Response body parameters

Parameter	Type	Description
info	Array of **RuleItem** objects	Rule list of a rule set.
total	Integer	Total number.

Table 4-175 RuleItem

Parameter	Type	Description
rule_id	String	Rule ID.
rule_language	String	Language corresponding to the rule.
rule_name	String	Rule name.
rule_severity	String	Severity of issues corresponding to the rule.
tags	String	Rule tag.
checked	String	Rule status. **0**: disabled; **1**: enabled.
rule_config_list	Array of **RuleConfig** objects	Threshold information about rule parameter configuration.

Table 4-176 RuleConfig

Parameter	Type	Description
id	Integer	Rule configuration ID.
rule_id	Integer	Rule ID.
default_value	String	Default value.
option_value	String	Current.
option_key	String	Key of the current rule configuration item.
option_name	String	Name of the current rule configuration item.
template_id	String	Rule set ID.
description	String	Description.

Status code: 400

Table 4-177 Response body parameters

Parameter	Type	Description
error_code	String	Error code.
error_msg	String	Error message.

Status code: 401

Table 4-178 Response body parameters

Parameter	Type	Description
error_code	String	Error code.
error_msg	String	Error message.

Example Requests

```
GET https://{endpoint}/v2/{project_id}/ruleset/{ruleset_id}/rules
```

Example Responses

Status code: 200

Request succeeded!

```
{
  "info" : [ {
    "rule_language" : "cpp",
    "rule_id" : "1614",
    "checked" : "1",
    "rule_name" : "A buffer must have size of 2 integers if used as parameter of pipe().",
    "rule_severity" : "1",
    "tags" : "cwe",
    "rule_config_list" : [ ]
  }, {
    "rule_language" : "cpp",
    "rule_id" : "1611",
    "checked" : "1",
    "rule_name" : "A buffer must have size of 2 integers if used as parameter of pipe().",
    "rule_severity" : "1",
    "tags" : "cwe",
    "rule_config_list" : [ {
      "id" : 250,
      "rule_id" : 11707,
      "template_id" : "906e7eac47dd4bde9c984f5e6f2a54e8",
      "option_key" : "threshold",
      "option_name" : "Adding a threshold",
      "option_value" : "10",
      "default_value" : "20",
      "description" : "If the code complexity exceeds the threshold, the code is identified as a huge cyclomatic complexity. The difference between the large cyclomatic complexity and cyclomatic complexity statistics is that only the number of switch statements is counted. The parameter is mandatory."
    }, {
      "id" : 251,
      "rule_id" : 11707,
      "template_id" : "906e7eac47dd4bde9c984f5e6f2a54e8",
      "option_key" : "threshold_modify",
      "option_name" : "Modifying a threshold",
      "option_value" : "30",
      "default_value" : "20",
      "description" : "If the code complexity exceeds the threshold, the code is identified as a huge cyclomatic complexity. The difference between the large cyclomatic complexity and cyclomatic complexity statistics is that only the number of switch statements is counted. The modified threshold must be greater than the new threshold. This parameter is optional."
    } ]
  } ],
  "total" : 2
}
```

Status code: 400

Bad Request

```
{
  "error_code" : "CC.00000000",
  "error_msg" : "Network busy. Try again later."
}
```

Status code: 401

Unauthorized

```
{
  "error_code" : "CC.00000003",
  "error_msg" : "Authentication information expired."
}
```

Status Codes

Status Code	Description
200	Request succeeded!
400	Bad Request

Status Code	Description
401	Unauthorized

Error Codes

See **Error Codes**.

4.3.5 Deleting a Custom Rule Set

Function

This API is used to delete a custom rule set. The rule set in use or the default rule set cannot be deleted.

Calling Method

For details, see **Calling APIs**.

URI

DELETE /v2/{project_id}/ruleset/{ruleset_id}

Table 4-179 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID.
ruleset_id	Yes	String	Rule set ID.

Request Parameters

Table 4-180 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token. It can be obtained by calling the IAM API used to obtain a user token. The value of **X-Subject-Token** in the response header is the user token.

Response Parameters

Status code: 400

Table 4-181 Response body parameters

Parameter	Type	Description
error_code	String	Error code.
error_msg	String	Error message.

Status code: 401

Table 4-182 Response body parameters

Parameter	Type	Description
error_code	String	Error code.
error_msg	String	Error message.

Example Requests

```
DELETE https://{endpoint}/v2/{project_id}/ruleset/{ruleset_id}
```

Example Responses

Status code: 200

Request succeeded!

```
null
```

Status code: 400

Bad Request

```
{
  "error_code" : "CC.00000000",
  "error_msg" : "Network busy. Try again later."
}
```

Status code: 401

Unauthorized

```
{
  "error_code" : "CC.00000003",
  "error_msg" : "Authentication information expired."
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
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.codecheck.v2.region.CodeCheckRegion;
import com.huaweicloud.sdk.codecheck.v2.*;
import com.huaweicloud.sdk.codecheck.v2.model.*;


public class DeleteRulesetSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
                .withAk(ak)
                .withSk(sk);

        CodeCheckClient client = CodeCheckClient.newBuilder()
                .withCredential(auth)
                .withRegion(CodeCheckRegion.valueOf("<YOUR REGION>"))
                .build();
        DeleteRulesetRequest request = new DeleteRulesetRequest();
        request.withProjectId("{project_id}");
        request.withRulesetId("{ruleset_id}");
        try {
            DeleteRulesetResponse response = client.deleteRuleset(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());
        }
    }
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcodecheck.v2.region.codecheck_region import CodeCheckRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcodecheck.v2 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security
risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment
variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this
example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
```

```
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]

    credentials = BasicCredentials(ak, sk)

    client = CodeCheckClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CodeCheckRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = DeleteRulesetRequest()
        request.project_id = "{project_id}"
        request.ruleset_id = "{ruleset_id}"
        response = client.delete_ruleset(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

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

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := codecheck.NewCodeCheckClient(
        codecheck.CodeCheckClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.DeleteRulesetRequest{}
    request.ProjectId = "{project_id}"
    request.RulesetId = "{ruleset_id}"
    response, err := client.DeleteRuleset(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in **API Explorer**. SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Request succeeded!
400	Bad Request
401	Unauthorized

Error Codes

See **Error Codes**.

4.3.6 Setting a default rule set for each language in a project.

Function

This API is used to set a default rule set for each language in a project.

Calling Method

For details, see **Calling APIs**.

URI

POST /v2/{project_id}/ruleset/{ruleset_id}/{language}/default

Table 4-183 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID.
ruleset_id	Yes	String	Rule set ID.
language	Yes	String	Rule set language.

Request Parameters

Table 4-184 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	It can be obtained by calling the IAM API used to obtain a user token. The value of **X-Subject-Token** in the response header is the user token.

Response Parameters

Status code: 400

Table 4-185 Response body parameters

Parameter	Type	Description
error_code	String	Error code.
error_msg	String	Error message.

Status code: 401

Table 4-186 Response body parameters

Parameter	Type	Description
error_code	String	Error code.
error_msg	String	Error message.

Example Requests

```
POST https://{endpoint}/v2/{project_id}/{language}/ruleset/{ruleset_id}/default
```

Example Responses

Status code: 200

Request succeeded!

```
null
```

Status code: 400

Bad Request

```
{
  "error_code" : "CC.00000000",
  "error_msg" : "Network busy. Try again later."
}
```

Status code: 401

Unauthorized

```
{
  "error_code" : "CC.00000003",
  "error_msg" : "Authentication information expired."
}
```

SDK Sample Code

The SDK sample code is as follows.

Java

```
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.codecheck.v2.region.CodeCheckRegion;
import com.huaweicloud.sdk.codecheck.v2.*;
import com.huaweicloud.sdk.codecheck.v2.model.*;


public class SetDefaulTemplateSolution {

    public static void main(String[] args) {
        // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great
security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or
environment variables and decrypted during use to ensure security.
        // In this example, AK and SK are stored in environment variables for authentication. Before running
this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
        String ak = System.getenv("CLOUD_SDK_AK");
        String sk = System.getenv("CLOUD_SDK_SK");

        ICredential auth = new BasicCredentials()
                .withAk(ak)
                .withSk(sk);

        CodeCheckClient client = CodeCheckClient.newBuilder()
                .withCredential(auth)
                .withRegion(CodeCheckRegion.valueOf("<YOUR REGION>"))
                .build();
        SetDefaulTemplateRequest request = new SetDefaulTemplateRequest();
        request.withProjectId("{project_id}");
        request.withRulesetId("{ruleset_id}");
        request.withLanguage("{language}");
        try {
            SetDefaulTemplateResponse response = client.setDefaulTemplate(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());
        }
    }
}
```

Python

```
# coding: utf-8

import os
from huaweicloudsdkcore.auth.credentials import BasicCredentials
from huaweicloudsdkcodecheck.v2.region.codecheck_region import CodeCheckRegion
from huaweicloudsdkcore.exceptions import exceptions
from huaweicloudsdkcodecheck.v2 import *

if __name__ == "__main__":
    # The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    # In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak = os.environ["CLOUD_SDK_AK"]
    sk = os.environ["CLOUD_SDK_SK"]

    credentials = BasicCredentials(ak, sk)

    client = CodeCheckClient.new_builder() \
        .with_credentials(credentials) \
        .with_region(CodeCheckRegion.value_of("<YOUR REGION>")) \
        .build()

    try:
        request = SetDefaulTemplateRequest()
        request.project_id = "{project_id}"
        request.ruleset_id = "{ruleset_id}"
        request.language = "{language}"
        response = client.set_defaul_template(request)
        print(response)
    except exceptions.ClientRequestException as e:
        print(e.status_code)
        print(e.request_id)
        print(e.error_code)
        print(e.error_msg)
```

Go

```
package main

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

func main() {
    // The AK and SK used for authentication are hard-coded or stored in plaintext, which has great security risks. It is recommended that the AK and SK be stored in ciphertext in configuration files or environment variables and decrypted during use to ensure security.
    // In this example, AK and SK are stored in environment variables for authentication. Before running this example, set environment variables CLOUD_SDK_AK and CLOUD_SDK_SK in the local environment
    ak := os.Getenv("CLOUD_SDK_AK")
    sk := os.Getenv("CLOUD_SDK_SK")

    auth := basic.NewCredentialsBuilder().
        WithAk(ak).
        WithSk(sk).
        Build()

    client := codecheck.NewCodeCheckClient(
        codecheck.CodeCheckClientBuilder().
            WithRegion(region.ValueOf("<YOUR REGION>")).
            WithCredential(auth).
            Build())

    request := &model.SetDefaulTemplateRequest{}
    request.ProjectId = "{project_id}"
    request.RulesetId = "{ruleset_id}"
    request.Language = "{language}"
    response, err := client.SetDefaulTemplate(request)
    if err == nil {
        fmt.Printf("%+v\n", response)
    } else {
        fmt.Println(err)
    }
}
```

More

For SDK sample code of more programming languages, see the Sample Code tab in **API Explorer**. SDK sample code can be automatically generated.

Status Codes

Status Code	Description
200	Request succeeded!
400	Bad Request
401	Unauthorized

Error Codes

See **Error Codes**.

5 Application Examples

5.1 Example 1: Querying the Task List

Scenario

Query the task list of a project based on **DEVCLOUD_PROJECT_UUID**.

Constraints

None

Involved API

The following API is involved:

Querying the Task List: Query the task list based on **DEVCLOUD_PROJECT_UUID**.

Procedure

Query the task list.

- API information

  URI format: **GET /v2/***{project_id}***/tasks**

- Request example

  ```
  GET https://{endpoint}/v2/2b31ed520xxxxxxebedb6e57xxxxxxxx/tasks
  ```

  For details about the information, see **Endpoints**.

- Example response

  ```
  {
  "result":[
  {
  "projectId":"b9db371361724226937f3280df1d4fc4",
  "devCloudProjectId":"a2b9a42ee6494e0794f5655f8535920e",
  "devCloudProjectName":"TestDemo",
  "codeHubName":"portal-ts",
  ```

```
"codeHubWebUrl":"https://xxxxx/portal-ts",
"codeLineTotal":"63371",
"projectName":"portal-ts-abctest111222",
"creatorId":"ae161856f2604229ae12056478a7919f",
"creatorName":"name",
"domainId":"78d0e09c1ad0425b9b4f3a8fdd066164",
"domainName":"name",
"gitUrl":"git@xxxxx/portal-ts.git",
"branch":"master",
"language":"[
\"html\"
]",
"filePath":"[
\"\"
]",
"lastCheckTime\"":"2019-09-24 15:00:41",
"codeQuality":"141.7",
"codeLine":"55777",
"projectStatus":"2",
"riskCoefficient":"7905",
"starLevel":"0",
"defectInfo":{
"totalCount":"0",
"criticalCount":"0",
"seriousCount":"0",
"normalCount":"0",
"promptCount":"0"
},
"defectStatusInfo":{
"totalCount":"2952",
"newCount":"2952",
"ignoredCount":"0",
"fixedCount":"0"
},
"progress":{

},
"createdAt":"2019-08-08 17:56:40",
"duplications":"24.0%",
"complexityNum":"0",
"complexityAvg":"0",
"duplicatedLines":"15206",
"commentLines":"0",
"commentRatio":"0",
"duplicatedBlocks":"0",
"lastExecTime":"2 minutes 4 seconds",
"checkType":"source",
"dependenceType":"",
"logDatasStr":{

}
},
{
"projectId":"b9db371361724226937f3280df1d4fc4",
"devCloudProjectId":"a2b9a42ee6494e0794f5655f8535920e",
"devCloudProjectName":"TestDemo",
"codeHubName":"portal-ts",
"codeHubWebUrl":"https://xxxxx/portal-ts",
"codeLineTotal":"63371",
"projectName":"portal-ts-abctest111222",
"creatorId":"ae161856f2604229ae12056478a7919f",
"creatorName":"devcloud_devcloud_y00336947_01",
"domainId":"78d0e09c1ad0425b9b4f3a8fdd066164",
"domainName":"name",
"gitUrl":"git@xxxxx/portal-ts.git",
"branch":"master",
"language":"[
\"html\"
]",
"filePath":"[
\"\"
]",
"lastCheckTime\"":"2019-09-24 15:00:41",
"codeQuality":"141.7",
"codeLine":"55777",
"projectStatus":"2",
"riskCoefficient":"7905",
"starLevel":"0",
"defectInfo":{
"totalCount":"0",
"criticalCount":"0",
"seriousCount":"0",
"normalCount":"0",
"promptCount":"0"
},
"defectStatusInfo":{
"totalCount":"2952",
"newCount":"2952",
"ignoredCount":"0",
"fixedCount":"0"
},
"progress":{

},
"createdAt":"2019-08-08 17:56:40",
"duplications":"24.0%",
"complexityNum":"0",
"complexityAvg":"0",
"duplicatedLines":"15206",
"commentLines":"0",
"commentRatio":"0",
"duplicatedBlocks":"0",
"lastExecTime":"2 minutes 4 seconds",
"checkType":"source",
"dependenceType":"",
"logDatasStr":{

}
}
],
"status":"success"
}
```

5.2 Example 2: Querying the Defect Summary

Scenario

Query the defect summary based on the check task ID, including the issue overview, issue status, cyclomatic complexity, and code repetition rate.

Constraints

The task exists.

Involved API

The following API is involved:

4.2.1 Querying Issue Summary: Query the issue overview, including issues, issue status, cyclomatic complexity, and code repetition rate.

Procedure

- API information

  URI format: **GET /v2/tasks/{task_id}/defects-summary**

- Request example

  ```
  GET https://{endpoint}/v2/tasks/2b31ed520xxxxxxebedb6e57xxxxxxxx/defects-summary
  ```

  For details about the information, see **Endpoints**.

- Example response

  ```
  {
  "result":{
  "info":{
  "projectId":"b9db371361724226937f3280df1d4fc4",
  "devCloudProjectId":"a2b9a42ee6494e0794f5655f8535920e",
  "devCloudProjectName":"TestDemo",
  "codeHubName":"portal-ts",
  "codeHubWebUrl":"https://xxxxx/portal-ts",
  "codeLineTotal":"63371",
  "projectName":"portal-ts-abctest111222",
  "creatorId":"ae161856f2604229ae12056478a7919f",
  "creatorName":"name",
  "domainId":"78d0e09c1ad0425b9b4f3a8fdd066164",
  "domainName":"name",
  "gitUrl":"git@xxxxx/portal-ts.git",
  "branch":"master",
  "language":"[
  \"html\"
  ]",
  "filePath":"[
  \"\"
  ]",
  "lastCheckTime\"":"2019-09-24 15:00:41",
  "codeQuality":"141.7",
  "codeLine":"55777",
  "projectStatus":"2",
  "riskCoefficient":"790",
  "starLevel":"0",
  "defectInfo":{
  "totalCount":"0",
  "criticalCount":"0",
  "seriousCount":"0",
  "normalCount":"0",
  "promptCount":"0"
  },
  "defectStatusInfo":{
  "totalCount":"2952",
  "newCount":"2952",
  "ignoredCount":"0",
  "fixedCount":"0"
  },
  "progress":{

  },
  "createdAt":"2019-08-08 17:56:40",
  "duplications":"24.0%",
  "complexityNum":"0",
  "complexityAvg":"0",
  "duplicatedLines":"15206",
  "commentLines":"0",
  "commentRatio":"0",
  "duplicatedBlocks":"0",
  "lastExecTime":"2 minutes 4 seconds",
  "checkType":"source",
  "dependenceType":"",
  "logDatasStr":{

  }
  ```

```
}
},
"status":"success"
}
```

6 Appendixes

6.1 Status Codes

Table 6-1 describes status codes.

Table 6-1 Status codes

Status Code	Message	Description
100	Continue	Continue sending requests. This temporary response is used to inform the client that some requests have been received and not rejected by the server.
101	Switching Protocols	Switch protocols. A protocol can only be switched to a more advanced protocol. For example, switch to a later version of HTTP.
201	Created	The request for creating a resource has been implemented.
202	Accepted	The request has been accepted, but has not finished being processed yet.
203	Non-Authoritative Information	Unauthorized information. The request is successful.

Status Code	Message	Description
204	NoContent	The request has been successfully implemented, but the HTTP response does not contain a response body. This status code is returned in response to an HTTP **OPTIONS** request.
205	Reset Content	The server has successfully processed the request after resetting the content.
206	Partial Content	The server has successfully processed a part of the **GET** request.
300	Multiple Choices	There are multiple choices for a requested resource. A list of resource characteristics and addresses is returned for the client such as a browser to choose from.
301	Moved Permanently	A requested resource has been permanently moved to the given URI indicated in the response.
302	Found	The requested resource has been temporarily moved.
303	See Other	Retrieve another URL, using a **GET** or **POST** method.
304	Not Modified	The requested resource has not been modified. When the server returns this status code, no resource is returned.
305	Use Proxy	The requested resource is available only through a proxy.
306	Unused	This HTTP status code is no longer used.
400	BadRequest	Invalid request. The client should modify the request instead of re-initiating it.
401	Unauthorized	The authorization information provided by the client is incorrect or invalid.
402	Payment Required	Reserved request.
403	Forbidden	Request rejected. The server has received the request and understood it, but refuses to respond to it. The client should not repeat the request without modifications.
404	NotFound	The requested resource cannot be found. The client should modify the request instead of re-initiating it.

Status Code	Message	Description
405	MethodNotAllowed	A request method is not supported for the requested resource. The client should modify the request instead of re-initiating it.
406	Not Acceptable	The server cannot implement the request based on the content characteristics of the request.
407	Proxy Authentication Required	This code is similar to 401, but indicates that the client must first authenticate itself with the proxy.
408	Request Time-out	The server timed out waiting for the request. The client may re-initiate the request without modifications at any time later.
409	Conflict	The request cannot be processed due to a conflict. This status code indicates that the resource that the client attempts to create already exits, or the request fails to be processed because of the update of the conflict request.
410	Gone	The requested resource cannot be found. This status code indicates that the requested resource has been deleted permanently.
411	Length Required	The server fails to process the request which does not contain the **Content-Length** header field.
412	Precondition Failed	The server does not meet one of the preconditions that the requester puts on the request.
413	Request Entity Too Large	The server refuses to process a request because the request entity is too large. The server may disable the connection to prevent the client from sending requests consecutively. If the server temporarily cannot process the request, the response will contain a **Retry-After** header field.
414	Request-URI Too Large	The request URI is too long for the server to process.
415	Unsupported Media Type	The server does not support the media type in the request.
416	Requested range not satisfiable	The requested range is invalid.
417	Expectation Failed	The server fails to meet the requirements of the **Expect** request header field.

Status Code	Message	Description
422	UnprocessableEntity	The request is well-formed but is unable to respond due to semantic errors.
429	TooManyRequests	The client sends too many requests to the server within a given time, exceeding the client's access frequency limit or beyond the server's processing capability. In this case, the client should retry after the time period specified in the **Retry-After** response header.
500	InternalServerError	The server is able to receive the request but it cannot understand the request.
501	Not Implemented	The server does not support the requested function, and therefore cannot implement the request.
502	Bad Gateway	The server acting as a gateway or proxy receives an invalid response from a remote server.
503	ServiceUnavailable	The requested service is invalid. The client should modify the request instead of re-initiating it.
504	ServerTimeout	The server could not return a timely response. The response will reach the client only if the request carries a timeout parameter.
505	HTTP Version not supported	The server does not support the HTTP protocol version used in the request.

6.2 Error Codes

Since no result data is returned in the case of an API call error, you can identify the error cause based on the error codes of each API.

If HTTP status code 400 or 505 is returned after an API call error, you can find the specific error code and description in the response body.

If you are unable to identify the cause of an error, contact customer service and provide the error code for quick troubleshooting.

Format of an Error Response Body

If an error occurs during API calling, an error code and the corresponding error message will be displayed. The following shows an error response body:

```
{
    "error_msg": "The format of message is error",
    "error_code": "AS.0001"
}
```

In the preceding information, **error_code** is an error code, and **error_msg** describes the error.

Error Code Description

If an error code starting with **APIGW** is returned after you call an API, rectify the fault by referring to the instructions provided in **API Gateway Error Codes**.

Table 6-2 Error codes

Status Code	Error Code	Error Message	Description	Solution
200	CC.00050000	parameters not valid	Invalid parameters.	Check the parameter formats.
200	CC.00050001	query task list error	Querying the task list throws an error.	Contact customer service.
200	CC.00050002	query task detail error	Querying task details throws an error.	Contact customer service.
200	CC.00050003	query templates error	Viewing the task configuration template throws an error.	Contact customer service.
200	CC.00050004	query task history error	Querying the task history throws an error.	Contact customer service.
200	CC.00050005	create task error	Creating a task throws an error.	Contact customer service.
200	CC.00050006	service inner error	Internal service error.	Contact customer service.
200	CC.00050007	IAM authentication failed	IAM authentication failed.	Check whether the authentication token is valid.
200	CC.00050010	current user does not have permission	Insufficient permission.	Check the permission.

Status Code	Error Code	Error Message	Description	Solution
200	CC.00050011	not in any project	The current user is not a member of any project of the tenant.	Join the project.
200	CC.00050012	retry later	The network is busy.	Try again later or contact customer service.
200	CC.00050013	project not exist	The project does not exist or has been deleted.	Modify project-related parameters.
200	CC.00050014	call getTaskBasicInfo failed	Failed to call the getTaskBasicInfo API.	Try again later or contact customer service.
200	CC.00050015	task is running	The task is being executed. Try again later.	Try again later.
200	CC.00050016	create task failed	An error occurred when creating a task. Creation failed.	Check related parameters or contact customer service.
200	CC.00050017	call updateIssueStatus failed	Ignore the warning in the latest scanning result.	Ignore the warning in the latest scanning result.
200	CC.00050019	operations that are not supported by domain of the earlier version	Operation not supported by tenants using earlier versions	Operation not supported by tenants using earlier versions
200	CC.00050020	The current task has not been checked or has been checked successfully. Please check or try again after the check is successful	The current task has not been checked or checked successfully. Check or try again after the check is successful.	Check or try again after the check is successful.

6.3 Obtaining a Project ID

Obtaining a Project ID by Calling an API

You can obtain the project ID by calling the API used to **query project information**.

The API for obtaining a project ID is **GET https://***{Endpoint}***/v3/projects/**. *{Endpoint}* indicates the endpoint of IAM, which can be obtained from **Endpoints**. For details about API authentication, see **Authentication**.

In the following example, **id** indicates the project ID.

```
{
    "projects": [
        {
            "domain_id": "65382450e8f64ac0870cd180d14e684b",
            "is_domain": false,
            "parent_id": "65382450e8f64ac0870cd180d14e684b",

            "name": "ap-southeast-3",

            "description": "",
            "links": {
                "next": null,
                "previous": null,
                "self": "https://www.example.com/v3/projects/a4a5d4098fb4474fa22cd05f897d6b99"
            },
            "id": "a4a5d4098fb4474fa22cd05f897d6b99",
            "enabled": true
        }
    ],
    "links": {
        "next": null,
        "previous": null,
        "self": "https://www.example.com/v3/projects"
    }
}
```

Obtaining a Project ID from the Console

A project ID is required for some URLs when an API is called. To obtain a project ID, perform the following operations:

1. Log in to the management console.
2. Hover the mouse pointer over the username in the upper right corner and choose **My Credentials** from the drop-down list.

   Choose **My Credentials** > **API Credentials** to view the project ID in the project list.

6.4 Obtaining an Account ID

An account ID is required for some URLs when an API is called. To obtain an account ID, perform the following operations:

1. Log in to the management console.

2. Hover the mouse pointer over the username in the upper right corner and choose **My Credentials** from the drop-down list.

   Choose **My Credentials** > **API Credentials** to view the account ID.

6.5 CMetrics Specifications

Overview

CMetrics, a Huawei-developed code measurement tool, measures source code from different indicators, including code volume, cyclomatic complexity, lines of code, duplication rate, dangerous functions, and redundant code. The current version is 2.1.5.

Parameters

Table 6-3 Parameters

Parameter	Description
code_size	The number of lines of code without comments and blank lines.
raw_lines	The number of lines of code with comments and blank lines.
methods_total	Total number of functions. Total number of methods in all code files in a project. If the declaration of a Java abstract method does not contain a function body, the method will not be identified as a function.
cyclomatic_complexity_total	Total cyclomatic complexity. Sum of cyclomatic complexity of all functions. The default cyclomatic complexity of a method is 1. Each time a specific keyword appears, the cyclomatic complexity is incremented by 1. When a ternary operator appears, the cyclomatic complexity is incremented by 1. The distinction between cyclomatic complexity and CCA cyclomatic complexity lies in how the switch statement is handled. The cyclomatic complexity is incremented by the number of cases in the switch statement, wile the CCA cyclomatic complexity is incremented by 1 regardless of the number of cases in the switch statement.
cyclomatic_complexity_per_method	Average cyclomatic complexity. Average cyclomatic complexity = Total cyclomatic complexity/Total number of functions

Parameter	Description
maximum_cyclomatic_complexity	Max. cyclomatic complexity of all counted functions.
huge_cyclomatic_complexity_total	Total number of functions with huge cyclomatic complexity. The number of functions whose cyclomatic complexity is greater than the threshold. The threshold is determined by the CMetrics version. For details, see **Table 6-4**. You can modify the threshold based on the Cmetrics rules for some languages.
huge_cyclomatic_complexity_ratio	Proportion of functions with huge cyclomatic complexity. Proportion of huge cyclomatic complexity = Total huge cyclomatic complexity/Total number of functions × 100%
cca_cyclomatic_complexity_total	Total CCA cyclomatic complexity. Sum of the CCA cyclomatic complexity of all functions. The default cyclomatic complexity of a method is 1. Each time a specific keyword appears, the cyclomatic complexity is incremented by 1. When a ternary operator appears, the cyclomatic complexity is incremented by 1. The distinction between cyclomatic complexity and CCA cyclomatic complexity lies in how the switch statement is handled. The cyclomatic complexity is incremented by the number of cases in the switch statement, wile the CCA cyclomatic complexity is incremented by 1 regardless of the number of cases in the switch statement.
cca_cyclomatic_complexity_per_method	Average CCA cyclomatic complexity. Average CCA cyclomatic complexity = Total CCA cyclomatic complexity/Total number of functions
maximum_cca_cyclomatic_complexity	Max. CCA cyclomatic complexity of all counted functions.
huge_cca_cyclomatic_complexity_total	Total number of functions with huge CCA cyclomatic complexity. The number of functions whose cyclomatic complexity is greater than the threshold. The threshold is determined by the CMetrics version. You can modify the threshold based on the Cmetrics rules for some languages.

Parameter	Description
cyclomatic_complexity_adequacy	Cyclomatic complexity adequacy (CCA). Cyclomatic complexity adequacy = (Total number of functions – Total number of functions with huge CCA cyclomatic complexity)/Total number of functions × 100%
maximum_depth	Max. depth of all counted functions.
huge_depth_total	The number of huge depths. The number of functions whose maximum depth is greater than the threshold. The threshold is determined by the CMetrics version. For details, see **Table 6-4**. You can modify the threshold based on the Cmetrics rules for some languages.
huge_depth_ratio	Proportion of huge depths. Proportion of huge depths = The number of huge depths/Total number of functions × 100%
method_lines	The number of lines containing functions. The number of lines in the internal code blocks of functions, excluding the definition lines at the beginning of functions, blank lines, and comment lines.
lines_per_method	Average lines of code containing functions. Average lines of code containing functions = Total number of lines of code with functions/Total number of functions × 100%
huge_method_total	The number of huge functions. The number of functions whose lines of code exceed the threshold. The threshold is determined by the CMetrics version. For details, see **Table 6-4**. You can modify the threshold based on the Cmetrics rules for some languages.
huge_method_ratio	Proportion of huge functions. Proportion of huge functions = The number of huge functions/Total number of functions × 100%
files_total	Total number of files. Total number of files with the source code file name extension in the scanned project directory. Empty files are not counted. For details about the file name extensions corresponding to different languages, see **Table 6-5**.
folders_total	Total number of directories. Total number of directories in a project. Empty directories are not counted.

Parameter	Description
lines_per_file	Average lines of code per file. Average lines of code per file = Total lines of code/Total number of files
huge_headerfile_total	Huge header files. In the C source code file, files with the suffix **.h**, **.hh**, **.hpp**, **.hxx**, **.h++**, **.inc**, or **.inl** are considered as header files. If the number of valid lines of code in a header file is greater than the threshold, the header file is considered as a huge header file. The threshold is determined by the CMetrics version. For details, see **Table 6-4**.
huge_headerfile_ratio	Proportion of huge header files. Proportion of huge header files = The number of huge header files/ Total number of files × 100%
huge_non_headerfile_total	Huge source files. If the file name extension of a source code file is not a header file and the number of valid lines of code is greater than the threshold, the source code file is huge. The threshold is determined by the CMetrics version. For details, see **Table 6-4**. You can modify the threshold based on the Cmetrics rules for some languages.
huge_non_headerfile_ratio	Proportion of huge source files. Proportion of huge source files = The number of huge source files/ Total number of files × 100%
huge_folder_total	The number of huge directories. The total number of files and subdirectories in a directory is counted (excluding the number of subdirectories and files in the subdirectories of the directory node). If the total number of files and subdirectories is greater than the threshold, the directory is considered as a huge directory. Empty sub-directories (sub-directories without source code files) are not counted. The threshold is determined by the CMetrics version. For details, see **Table 6-4**. You can modify the threshold based on the Cmetrics rules for some languages.
huge_folder_ratio	Proportion of huge directories. Proportion of huge directories = The number of huge directories/ Total number of directories × 100%
file_duplication_total	The number of duplicate files. The number of files with the same content. Only the files whose file name extensions meet the source code file name extension requirements of each language are counted. For details about the file name extensions of different languages, see **Table 6-5**.

Parameter	Description
file_duplication_ratio	File duplication rate. File duplication rate = The number of duplicate files/Total number of files × 100%
non_hfile_duplication_total	The number of duplicate source files. The number of duplicate C files with the file name extension of source files. Other languages do not have header files, and all files are source files.
non_hfile_duplication_ratio	Source file duplication rate. Source file duplication rate = The number of duplicate source files/ Total number of files × 100%
code_duplication_total	Duplicate code. Duplicate code refers to non-blank, non-comment lines that are identical and occur consecutively, including import lines. **NOTE** The count of duplicate code is based on the total number of duplicate lines in each file, rather than individual code snippets. For example, if three files have the same 10 lines of duplicate code, the number of duplicate lines of code is 30. If lines 1 to 10 in file A appear in file B and lines 2 to 11 in file A appear in file C, the union of lines 1 to 10 and lines 2 to 11 are counted as duplicate code of file A, that is, 11 lines of duplicate code.
code_duplication_ratio	Duplication rate. Code duplicate rate = The number of duplicate code/Total number of lines of code × 100%
non_hfile_code_duplication_total	Duplicate source files. Total number of duplicate lines in a source file. Only the duplicate lines in the files whose file name extensions meet the requirements of source code files (excluding header files in C/C++) are counted.
non_hfile_code_duplication_ratio	Source file code duplication rate. Source file code duplication rate = The number of duplicate source file code/Total number of lines of code × 100%
unsafe_functions_total	Unsafe functions. The number of times that unsafe functions, such as memcpy and memmove, are called in the code. **NOTE** Currently, only Clike detects unsafe functions.
unsafe_functions_kloc	Density of unsafe functions (the number of unsafe functions per thousand lines of code) Density of dangerous functions = Total number of unsafe functions/ Total number of lines of code × 1000

Parameter	Description
redundant_code_total	Redundant code. The number of redundant code blocks in comments. Code in comments and consecutive words that contain programming language keywords or comply with syntax rules in comments are regarded as redundant code. If a block comment contains multiple lines of redundant code, it is counted as only one redundant code.
redundant_code_kloc	Density of redundant code blocks (the number of lines of redundant code per thousand lines of code). Density of redundant code blocks = The number of lines of redundant code/Total number of lines of code × 1000 **NOTE** If the total number of lines of code is less than 1,000, the density is not calculated and the value is **0**.
warning_suppression_total	The number of suppressed alarms. The number of alarm suppression statements in scanned files. If a string in a comment complies with a specific regular expression, the comment is considered as an alarm suppression statement. For details about the regular expressions corresponding to different languages, see **Table 6-7**. This parameter takes effect only after the CMetrics rule is enabled.
warning_suppression_kloc	Density of suppressed alarms. Density of suppressed alarms = The number of suppressed alarms/ Total number of lines of code × 1000 (The number of redundant lines of code per thousand lines of code) This parameter takes effect only after the CMetrics rule is enabled.

Table 6-4 Default thresholds of the CMetrics tool

Threshold Name	Default Value
huge_cyclomatic_complexity	20
huge_depth	4
huge_method	50
huge_headerfile	500
huge_non_headerfile	2,000
huge_folder	50

Table 6-5 CMetrics languages and file name extensions

Language	Extension
Java	.java
C/C++	.c/.cc/.cpp/.cxx/.cp/.c++/.inc/.inl/.mm/.h/.hh/.hpp/.hxx/.h++
TypeScript	.ts/.tsx
Rust	.rs
ArkTS	.ets
C#	.cs
Go	.go
JavaScript	.js/.jsx
Lua	.lua
Python	.py

Table 6-6 Unsafe C/C++ functions checked by CMetrics

_fstrncat	_vsntprintf	strcatbuff	vos_nsprintf
_fstrncpy	_vsnwprintf	strcatbuffa	vos_nvsprintf
_ftcscat	_vstprintf	strcatbuffw	vos_snprintf
_ftcscpy	bcopy	strcatchainw	vos_sprintf
_getts	copymemory	strcatn	vos_sscanf
_gettws	fscanf	strcatna	vos_strcat
_getws	fwscanf	strcatnw	vos_strcpy
_mbccat	gets	strcatw	vos_strncat
_mbccpy	lstrcat	strcpy	vos_strncpy
_mbscat	lstrcata	strcpya	vos_vsprintf
_mbscpy	lstrcatn	strcpyn	vos_vsscanf
_mbsnbcat	lstrcatnA	strcpyna	vscanf
_mbsnbcpy	lstrcatnW	strcpynw	vsnprintf
_mbsncat	lstrcatw	strcpyw	vsprintf
_mbsncpy	lstrcpy	strncat	vsscanf
_snprintf	lstrcpya	strncata	vswprintf

_sntprintf	lstrcpyn	strncatw	vswscanf
_sntscanf	lstrcpyna	strncpy	vwscanf
_snwprintf	lstrcpynw	strncpya	wcscat
_stprintf	lstrcpyw	strncpyw	wcscpy
_stscanf	lstrncat	swprintf	wcsncat
_tccat	memcpy	swscanf	wcsncpy
_tccpy	memmove	vfscanf	wmemcpy
_tcscat	memset	vfwscanf	wmemmove
_tcscpy	nsprintf	vos_chkmemcpy	wnsprintf
_tcsncat	rtlcopymemory	vos_chkmemmove	wnsprintfA
_tcsncpy	scanf	vos_chkmemset	wnsprintfW
_tscanf	snprintf	vos_chkstrcat	wscanf
_vos_bzero	snscanf	vos_chkstrcpy	wsprintf
_vos_memcpy	sntprintf	vos_chkstrncat	wsprintfa
_vos_memmove	snwscanf	vos_chkstrncpy	wsprintfw
_vos_memset	sprintf	vos_mem_copy	wvnsprintf
_vos_strcat	sprintfa	vos_mem_set	wvnsprintfa
_vos_strcpy	sprintfw	vos_mem_zero	wvnsprintfw
_vos_strncat	sscanf	vos_memcpy	wvsprintf
_vos_strncpy	strcat	vos_memmove	wvsprintfa
_vsnprintf	strcata	vos_memset	wvsprintfw

Table 6-7 Suppression rules in CMetrics

<table>
<tr><th>Language</th><th>Regular Expression for Comment Matching</th></tr>
<tr><td>C/C++, Java, JavaScript, TypeScript</td><td>• r'(/(*|/)\s*(lint\b\s+\S+|coverity\s*\[.+?\]|eslint-disable)'
• r'|^\s*#\s*pragma\s+\w+\s+diagnostic\s+(ignored|warning)'
• r'|^\s*#\s*pragma\s+warning\s*\(\s*disable'
• r'|^\s*@(\w+\.)?SuppressWarnings)'</td></tr>
<tr><td>Python</td><td>r'#\s*pylint\s*:'</td></tr>
</table>

7 Description

Changes between document issues are cumulative. The latest document issue contains all the changes made in earlier issues.

Description	Release Date
This issue is the second official release. It incorporates the following change: Renamed CodeCheck to CodeArts Check.	April 23, 2022
This issue is the first official release.	September 30, 2022