

Repo

API Reference

Issue 01
Date 2023-05-05



Copyright © Huawei Technologies Co., Ltd. 2023. All rights reserved.

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

Trademarks and Permissions



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

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

Notice

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

Contents

1 Before You Start.....	1
1.1 Overview.....	1
1.2 Calling Method.....	1
1.3 Endpoints.....	2
1.4 Concepts.....	2
2 API Overview.....	5
3 Calling APIs.....	12
3.1 Making an API Request.....	12
3.2 Authentication.....	15
3.3 Response.....	16
4 APIs.....	18
4.1 SSHKey.....	18
4.1.1 Obtaining the SSH Key List.....	18
4.1.2 Adding an SSH Key.....	21
4.1.3 Check whether the private key has the permission to pull code.....	23
4.2 Repository.....	25
4.2.1 Obtain files in the branch directory.....	25
4.2.2 Obtaining Warehouse Statistics.....	28
4.2.3 Obtains image files of a specific branch in a repository.....	30
4.2.4 Modifying the Status of a Repository Referenced by a Pipeline.....	31
4.2.5 Checks whether a user has the warehouse administrator permission.....	33
4.2.6 Sets whether a repository is in public or private state.....	35
4.2.7 Obtains the content of a specified file in a specified branch of a repository.....	37
4.2.8 Obtaining the List of Public Sample Templates.....	39
4.2.9 Creating a repository.....	42
4.2.10 Obtain the short ID of the repository based on the repository name and group name to combine the short ID with the URL of the submission details page corresponding to commitid.....	45
4.2.11 Queries the submission of a branch of a repository based on the group name and repository name.....	47
4.2.12 Querying the Branch of a Warehouse.....	50
4.2.13 This API is used to query the submission of a branch of a warehouse based on the warehouse ID. More optional parameters are provided.....	52
4.2.14 Adding a Deployment Key.....	55

4.2.15 Deleting a Repository Deployment Key.....	58
4.2.16 Obtains the last submission statistics of the warehouse.....	59
4.2.17 Repository Statistics.....	63
4.2.18 Delete repositories.....	65
4.2.19 Check the repository creation status.....	67
4.2.20 Querying the Tag List of a Warehouse.....	68
4.2.21 Adding a Tag.....	70
4.2.22 Querying the Details of a Warehouse.....	73
4.2.23 Obtains the number of submitted code lines.....	77
4.2.24 Download Repository.....	79
4.2.25 Creating a Protection Branch.....	80
4.2.26 Obtaining the Repository Branch List.....	84
4.2.27 Obtaining the List of Warehouse Combination Requests.....	87
4.2.28 Obtaining Warehouse Combination Request Details.....	91
4.3 V2Project.....	95
4.3.1 Querying All Warehouses of a User.....	95
4.3.2 Create projects and repositories.....	99
4.3.3 Creating a Project and Forking a Repository.....	102
4.3.4 Querying All Warehouses in a Project.....	105
4.3.5 Associating Branches with Work Items.....	109
4.4 RepoMember.....	111
4.4.1 Obtains all member records in a warehouse.....	111
4.4.2 Add repository members.....	114
4.4.3 Sets the role of a member in the warehouse.....	117
4.4.4 Delete repository members.....	119
4.5 Commit.....	120
4.5.1 Querying the Submission Information of a Warehouse.....	120
4.5.2 Creating a Commit.....	124
4.5.3 Querying Specific Submission Information of a Warehouse.....	128
4.5.4 Querying the Submission Discrepancy Information of a Warehouse.....	132
4.6 File.....	134
4.6.1 Querying File Information of a Warehouse.....	134
4.7 User.....	136
4.7.1 HTTPS account and password verification.....	137
4.8 WebHook.....	138
4.8.1 Querying the Webhook of a Specified Warehouse.....	138
4.8.2 Adding a Hook to a Specified Repository.....	141
4.8.3 Deletes the hook of a specified repository.....	145
4.9 Project.....	146
4.9.1 Obtains the list of repositories that can be set to the public state in a project.....	147
4.9.2 Verifying the Repository Name of a Specified Project.....	149
4.10 V2 Warehouse Management.....	151

4.10.1 Create branches.....	151
5 Application Examples.....	155
5.1 Example 1: Querying All Repositories of a User.....	155
5.2 Example 2: Querying Details About a Specified Repository by Repository UUID.....	156
5.3 Example 3: Querying Branches of a Repository.....	158
6 Appendix.....	160
6.1 Status Codes.....	160
6.2 Error Codes.....	163
6.3 Obtaining a Project ID.....	164
6.4 Obtaining an Account ID.....	165

1 Before You Start

[API Overview](#)

[Calling Method](#)

[Endpoints](#)

[Concepts](#)

[1.1 Overview](#)

[1.2 Calling Method](#)

[1.3 Endpoints](#)

[1.4 Concepts](#)

1.1 Overview

Welcome to CodeArts Repo. CodeArts Repo incorporates Huawei's experience in managing hundreds of billions of lines of code. CodeArts Repo is a Git-based code hosting service that allows you to clone, pull, commit, push, compare, merge, and review code online, as well as create branches.

This document describes how to use APIs to perform operations on CodeArts Repo, such as creating a repository, adding a deployment key, and adding repository members. For details about all supported operations, see [API Overview](#).

If you plan to access CodeArts Repo using an API, ensure that you are familiar with CodeArts Repo concepts. For details, see [Service Overview](#).

1.2 Calling Method

CodeArts Repo supports Representational State Transfer (REST) APIs, allowing you to call APIs using HTTPS. For details about API calling, see [Making an API Request](#).

1.3 Endpoints

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

Table 1 lists the endpoints of CodeArts Repo. Select a desired one based on service requirements.

Table 1-1 CodeArts Repo endpoints

Region Name	Region	Endpoint	Protocol
LA-Mexico City2	la-north-2	codehub-ext.la-north-2.myhuaweicloud.com	HTTPS
LA-Sao Paulo1	sa-brazil-1	codehub-ext.sa-brazil-1.myhuaweicloud.com	HTTPS
AP-Singapore	ap-southeast-3	codehub-ext.ap-southeast-3.myhuaweicloud.com	HTTPS

1.4 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**

A user is created using a domain to use cloud services. Each 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 account 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 and dedicated regions. A universal region provides universal cloud services for common tenants. A dedicated region provides specific services for specific tenants.

For details, see [Region and AZ](#).

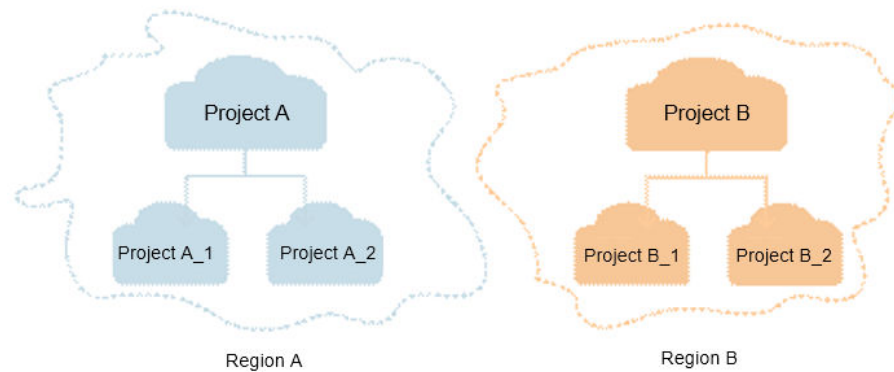
- Availability zone (AZ)

An AZ contains one or more physical data centers. Each AZ has independent cooling, fire extinguishing, moisture-proof, and electricity facilities. Within an AZ, computing, 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.

- Project

Projects group and isolate resources (including compute, storage, network, and other resources) across physical regions. A default project is provided for each region, and subprojects can be created under each default project. Users can be granted permissions to access all resources in a specific project. For more refined access control, create subprojects under a project and create resources in the subprojects. Users can then be assigned permissions to access only specific resources in the subprojects.

Figure 1-1 Project isolation model



You can obtain the project ID on the [My Credentials](#) page.

- git

Most popular distributed version control software originally authored by Linus Torvalds.

- Branching

Branching, in version control and software configuration management, is the duplication of an object (such as a program or a piece of software). Each object can thereafter be modified separately and in parallel so that the objects become different. In this context the objects are called branches. Branches are also known as trees, streams or codelines. The originating branch is sometimes called the parent branch or the upstream branch. Child branches are branches that have a parent. A branch without a parent is referred to as the trunk or the mainline.

- Revision Tag/Tag

A revision tag is a textual label that can be associated with a specific revision of a project maintained by a version control system. This allows the user to define a meaningful name to be given to a particular state of a project that is under version control. This label can then be used in place of the revision identifier for commands supported by the version control system. For

example, in software development, a tag may be used to identify a specific release of the software such as "version 1.2".

- Fork

In software engineering, a project fork happens when developers take a copy of source code from one software package and start independent development on it, creating a distinct and separate piece of software. The term often implies not merely a development branch, but also a split in the developer community; as such, it is a form of schism.

After GitHub appeared, fork became the norm. Forking and merge requests (MRs) together make communities more active instead of divided.

- Pull requests (PRs)/MRs

You can request for merging your own code (written in the fork repository or in a new branch). Code repository administrators review requests and merge the code.

2 API Overview

SSH Key

API	Description
Obtaining the SSH Key List	Obtain the SSH key list.
Adding an SSH Key	Add an SSH key.
Check whether the private key has the permission to pull code	Check whether a private key has the permission to pull code.
Deleting a User Public Key	Call the native GitLab API to delete a public key of a user.

Repository

API	Description
Obtain files in the branch directory	Obtain files in the branch directory.
Obtaining Warehouse Statistics	Obtain repository statistics.
Obtains image files of a specific branch in a repository	Obtain image files of a specific branch in a repository.

API	Description
Modifying the Status of a Repository Referenced by a Pipeline	Modify the status of a repository referenced by a pipeline.
Checks whether a user has the warehouse administrator permission	Check whether a user has the repository administrator permission.
Sets whether a repository is in public or private state	Indicate whether a repository is public or private.
Obtains the content of a specified file in a specified branch of a repository	Obtain the content of a specified file in a specified branch of a repository.
Obtaining the List of Public Sample Templates	Obtain the list of public sample templates.
Creating a repository	Create a repository on a specified project with the specified name. Input parameters include the repository name, template ID, whether to import project members, and home project.
Obtain the short ID of the repository based on the repository name and group name to combine the short ID with the URL of the submission details page corresponding to commitid	Obtain the short ID of a repository, which is used to obtain the URL of the repository details page.
Queries the submission of a branch of a repository based on the group name and repository name	Obtain the commit list by repository group name, repository name, and branch.

API	Description
Querying the Branch of a Warehouse	Obtain the branch list of a specified repository based on the repository ID.
This API is used to query the submission of a branch of a warehouse based on the warehouse ID. More optional parameters are provided	Query the commit of a repository branch based on the repository ID.
Adding a Deployment Key	Add a deployment key.
Deleting a Repository Deployment Key	Delete the repository deployment key.
Obtains the last submission statistics of the warehouse	Obtain the last commit statistics of the repository.
Repository Statistics	Query code commit record statistics of a repository based on the repository short ID.
Delete repositories	Delete a repository based on the 32-bit UUID of the repository.
Check the repository creation status	Obtain the repository status.
Querying the Tag List of a Warehouse	Query the branch corresponding to a specified repository.
Adding a Tag	Create a tag.
Querying the Details of a Warehouse	Obtain repository information based on the repository UUID. The returned information contains the ID, name, group name, and repository access URL.
Obtains the number of submitted code lines	Obtain the number of committed code lines of a specified branch in the code repository on a specified date.
Download Repository	Download a repository in the specified format.

API	Description
Creating a Protection Branch	Create a protection branch.
Obtaining the Repository Branch List	Obtain the repository branch list.
Obtaining the List of Warehouse Combination Requests	Obtain the repository MR list.
Obtaining Warehouse Combination Request Details	Obtain details about a repository MR.

V2Project

API	Description
Querying All Warehouses of a User	Obtain information about all repositories of a user.
Create projects and repositories	After a project is created, the parameters for creating a repository group are generated in the background, including the repository name, template ID, whether to import project members, and home project.
Creating a Project and Forking a Repository	After a repository is created, parameters transferred to the fork repository include the repository name, whether to import project members, and home project.
Querying All Warehouses in a Project	Obtain the repository list. Fuzzy query is supported: If the project UUID is not specified, fuzzy query by repository name or project name is supported; otherwise, fuzzy query by repository name is supported.
Associating Branches with Work Items	Associate a branch with a work item.

RepoMember

API	Description
Obtains all member records in a warehouse	Obtain the repository member list. You can search for a member by keyword.
Add repository members	The calling party codehubportal adds a repository member.
Sets the role of a member in the warehouse	Assign repository operation permissions to repository members.
Delete repository members	Delete repository members

Commit

API	Description
Querying the Submission Information of a Warehouse	Obtain commit information based on the repository short ID. The list of all commits in the path can be queried based on the file path.
Creating a Commit	Multiple files in different directories can be submitted at a time. If the directory does not exist, the system automatically creates a directory. The forcible overwrite option is supported. When the forcible overwrite flag is set to true, the conflict is ignored and the configuration is forcibly submitted.
Querying Specific Submission Information of a Warehouse	Obtain a specific commit by commit ID, branch name, or tag name.
Querying the Submission Discrepancy Information of a Warehouse	Query commit difference information by commit ID.

File

API	Description
Querying File Information of a Warehouse	Obtain information about files in the repository, such as the name, size, and content. Note that the file content is encoded using Base64.

User

API	Description
HTTPS account and password verification	Call the GitLab API to check whether the username and password entered when you upload or download code using HTTPS are valid.

Webhook

API	Description
Querying the Webhook of a Specified Warehouse	Obtain the repository webhook.
Adding a Hook to a Specified Repository	A code commit automatically triggers building and addition of a repository hook.
Deletes the hook of a specified repository	A code commit automatically triggers building and deletion of a repository hook.

Project

API	Description
Obtains the list of repositories that can be set to the public state in a project	Obtain the list of repositories that can be set to the public state under a project.
Obtaining Information About All Warehouses in a Project	Obtain the fuzzy search scope of the repository list. If project_id is not specified, fuzzy search by repository name or project name is supported. Otherwise, fuzzy search by repository name is supported.

API	Description
Verifying the Repository Name of a Specified Project	This API is called when a repository is created. The project UUID and repository name are transferred, and the CoudeHubAdapter API is called to query the database to check whether the repository name is duplicate.

V2 Repository Management

API	Description
Create branches	Create a branch in a specified repository based on the repository ID.

3 Calling APIs

Making an API Request

Authentication

Response

3.1 Making an API Request

3.2 Authentication

3.3 Response

3.1 Making an API Request

This section describes the structure of a REST API request, and calls the IAM API for **obtaining a user token** as an example. The obtained token can then be used to authenticate the calling of other APIs.

Request URI

The format of a request URI is as follows:

{URI-scheme} :// {Endpoint} / {resource-path} ? {query-string}

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

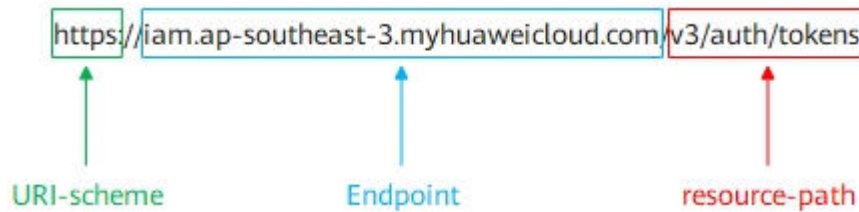
- **URI-scheme:** Protocol used to transmit requests. All APIs use HTTPS.
- **Endpoint:** Domain name or IP address of the server bearing the REST service endpoint. Obtain the value from **Regions and Endpoints**.
- **resource-path:** Access path of an API for performing a specified operation. Obtain the value from the URI of an API. For example, the **resource-path** of the API for obtaining a user token is **/v3/auth/tokens**.
- **query-string:** optional query parameter. Ensure that a question mark (?) is included before a query parameter that is in the format of "Parameter name=Parameter value". For example, **? limit=10** indicates that a maximum of 10 pieces of data is to be viewed.

For example, if you want to obtain a token of IAM in the AP-Singapore region, you need to use the endpoint (iam.ap-southeast-3.huaweicloud.com) of the AP-

Singapore region and find resource-path (/v3/auth/tokens) in the URI after [Obtaining a User Token](#). Here is an example.

```
https://iam.ap-southeast-3.huaweicloud.com/v3/auth/tokens
```

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. The **URI-scheme** of all APIs is **HTTPS**, and the endpoints of all APIs in the same region are identical.

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 used to [obtain a user token](#), the request method is **POST**. The request is as follows:

```
POST https://iam.ap-southeast-3.myhuaweicloud.com/v3/auth/tokens
```

Request Header

You can add additional fields, for example, the fields required by a specified URI or HTTP method, to a request header. For example, to request for the authentication information, add **Content-Type**, which specifies the request body type.

Common request header fields:

- **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 will be provided for specific APIs if any.
- **X-Auth-Token:** A user token only for token-based API authentication. The user token is a response to the API used to [obtain a user token](#). This API is the only one that does not require authentication.

 NOTE

In addition to supporting token-based authentication, public cloud APIs also support authentication using AK/SK. During AK/SK-based authentication, an SDK is used to sign the request, and the **Authorization** (signature information) and **X-Sdk-Date** (time when the request is sent) header fields are automatically added to the request.

For more information, see [AK/SK-based Authentication](#).

The API used to [obtain a user token](#) does not require authentication. Therefore, only the **Content-Type** field needs to be added to requests for calling the API. An example of such requests is as follows:

```
POST https://iam.ap-southeast-3.myhuaweicloud.com/v3/auth/tokens
Content-Type: application/json
```

Request Body

A request body is generally sent in structured format. It corresponds to **Content-Type** in the request header and transfers 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.

In the case of the API used to [obtain 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 **username**, **domainname**, ********* (login password), and **xxxxxxxxxx** (project ID, for example, cn-north-1) with the actual values. To learn how to obtain a project ID, see [Regions and Endpoints](#).

 NOTE

The **scope** parameter specifies where a token takes effect. You can set **scope** to an account or a project under an account. In the following example, the token takes effect only for the resources in a specified project. For more information about this API, see [Obtaining a User Token](#)

```
POST https://iam.ap-southeast-3.myhuaweicloud.com/v3/auth/tokens
Content-Type: application/json
```

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

If all data required for the API request is available, you can send the request to call the API through [curl](#), [Postman](#), or coding. In the response to the API used to [obtain a user token](#), **x-subject-token** is the desired user token. This token can then be used to authenticate the calling of other APIs.

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 Identity and Access Management (IAM) API for obtaining a user token.

A token specifies temporary permissions in a computer system. During API authentication using a token, the token is added to requests to get permissions for calling the API.

When calling the API to [obtain a user token](#), you must set **auth.scope** in the request body to **project**.

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

After a token is obtained, the **X-Auth-Token** header field must be added to requests to specify the token when calling other APIs. For example, if the token is **ABCDEFJ....**, **X-Auth-Token: ABCDEFJ....** should be added to a request as follows:

```
GET https://iam.ap-southeast-3.myhuaweicloud.com/v3/auth/projects
Content-Type: application/json
X-Auth-Token: ABCDEFJ....
```

AK/SK-based Authentication

NOTE

AK/SK authentication supports API requests with a body no larger than 12 MB. For API requests with a larger body, use token authentication.

In AK/SK-based authentication, AK/SK is used to sign requests and the signature is then added to the requests for authentication.

- 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.

In AK/SK-based authentication, you can use an AK/SK to sign requests based on the signature algorithm or use the signing SDK to sign requests. For details about how to sign requests and use the signing SDK, see [AK/SK Signing and Authentication Guide](#).

NOTICE

The signing SDK is only used for signing requests and is different from the SDKs provided by services.

3.3 Response

Status Code

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

A status code is a group of digits, ranging from 1xx to 5xx. It indicates the status of a request. For more information, see [Status Codes](#).

If status code **201** is returned for calling the API used to [obtain a user token](#), the request is successful.

Response Header

Similar to a request, a response also has a header, for example, **Content-Type**.

[Figure 3-2](#) shows the response header fields for the API used to [obtain a user token](#). The **x-subject-token** header field is the desired user token. This token can then be used to authenticate the calling of other APIs.

Figure 3-2 Header fields of the response to the request for obtaining a user token

```

connection → keep-alive

content-type → application/json

date → Tue, 12 Feb 2019 06:52:13 GMT

server → Web Server

strict-transport-security → max-age=31536000; includeSubdomains;

transfer-encoding → chunked

via → proxy A

x-content-type-options → nosniff

x-download-options → noopen

x-frame-options → SAMEORIGIN

x-iam-trace-id → 218d45ab-d674-4995-af3a-2d0255ba41b5

x-subject-token
→ MIIYXQYJKoZIhvcNAQcCoIIYTCCEGoCAQExDTALBglghkgBZQMEAgEwgharBgkqhkiG9w0BBwGgghacBIIWmHsidG9rZW4iOansiZXhwaXJlc19hdCI6IiwMTktMDItMTNUMC
fj3KJs6YgKnpVNRbW2eZ5eb78SZOkajACgkIQ01wi4JIGzrpd18LGXK5btdfq4lqHCYb8P4NaYONYejeAgzJVeFYtLWT1GSO0zxKZmlQHq82HBqHdglZO9fuEbL5dMhdavj+33wEI
xHRCE9I87o+k9-
j+CMZSEB7bUGd5Uj6eRASXl1jipPEGA270g1FruooL6jggIFkNPQuFSOU8+uSsttVwRtnfsc+qTp22Rkd5MCqFGQ8LcuUxC3a+9CMBnOintWW7oeRUUVhVpxk8pxiX1wTEboX-
RzT6MUbvpvGw-oPNFYxJECKnoH3HRozv0vN--n5d6Nbxg==

x-xss-protection → 1; mode=block;

```

Response Body

The body of a response is often returned in structured format as specified in the **Content-Type** header field. The response body transfers content except the response header.

For the API used to **obtain a user token**, the response body is as follows: The following example is part of the response body for the API used to obtain a user token:

```

{
  "token": {
    "expires_at": "2019-02-13T06:52:13.855000Z",
    "methods": [
      "password"
    ],
    "catalog": [
      {
        "endpoints": [
          {
            "region_id": "ap-southeast-3",
            .....

```

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 response body, **error_code** is an error code, and **error_msg** provides information about the error.

4 APIs

- [4.1 SSHKey](#)
- [4.2 Repository](#)
- [4.3 V2Project](#)
- [4.4 RepoMember](#)
- [4.5 Commit](#)
- [4.6 File](#)
- [4.7 User](#)
- [4.8 WebHook](#)
- [4.9 Project](#)
- [4.10 V2 Warehouse Management](#)

4.1 SSHKey

4.1.1 Obtaining the SSH Key List

Function

This interface is used to obtain the SSH key list.

URI

GET /v1/users/sshkey

Request Parameters

Table 4-1 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Specifies the user token. It can be obtained by calling the IAM API (value of X-Subject-Token in the response header).

Response Parameters

Status code: 200

Table 4-2 Response body parameters

Parameter	Type	Description
error	Error object	Response error.
result	PublicKeyList object	Result
status	String	Status code in a response

Table 4-3 Error

Parameter	Type	Description
code	String	Error Codes
message	String	Error message

Table 4-4 PublicKeyList

Parameter	Type	Description
sshkey	Array of PublicKey objects	Keys
total	Integer	Total number of keys

Table 4-5 PublicKey

Parameter	Type	Description
id	String	Key ID Minimum: 1 Maximum: 40
key	String	Key Minimum: 0 Maximum: 2048
title	String	Name

Example Requests

```
GET https://{endpoint}/v1/users/sshkey
```

Example Responses

Status code: 200

OK

```
{
  "result" : {
    "total" : 2,
    "sshkey" : [ {
      "id" : "1015292",
      "title" : "z00464114@SZXG4Z004641141",
      "key" : "ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQDyLDdm3bojppgitxdP854P9i0wDuruTTo
+YSyT6lRb0VEaLCIFvkf3DEtHURAtgJ590hqFHxpSfiLgbJww4Tk4Im3f125meJPvScvV3bp8eHYUYSazP
+MS3NCXFwJzViXRikaA33pZLP1OLPtCTrNdEgX5kj5BbX3V9z8UEFXpq55YZq12BNbs7njgTyTGjAHA
+G2fLS53CO6pUSBLKQj0BCZdDQr5RcrZrX0axFjREATbSDbLHH4/za1VZVSKelphEEJp8OZXkgtlOXVlo+/
scrhIET4qj7oWjJm1BbfHZanfjrJKr6+Z5rnJqa/pE8buOrxri6AWqAky0bmtjSuVf
z00464114@SZXG4Z004641141"
    }, {
      "id" : "1031709",
      "title" : "yumeng17@huawei.com",
      "key" : "ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAQDHcxQrLlYP7NTWjXRjTyc0e5O4EYEJ6HqvK1lLufK676uPu3AOP5csA
2HrQVZdNBCfk50Qon1X0CVWzm+
+PaxrJ9J5rxvEObGZiDasZroRpDLNxVcz01vHNosTZZH3D9dBGfDQVeNYFc1QrsMSXfivCzeq6lcpNTO7Tj6w/
+J1QparPNI3yLB//y+FM71nwbbPFmuJrN7b/o3X1RGrczCY
+idCUcl6CDp9hltwYdoRmntndpVtLwSKeWE1MSLNkvPzENXgtOmxm2LI2eR5h4hNYSxT0NlazzW/UivadFXG/
VNNPl0nU7xcma6HU8nj1E0i46VOB+WyoONL3gh142BV yumeng17@huawei.com"
    }
  ]
},
  "status" : "success"
}
```

Status Codes

Status Code	Description
200	OK

Error Codes

See [Error Codes](#).

4.1.2 Adding an SSH Key

Function

Adding an SSH Key

URI

POST /v1/users/sshkey

Request Parameters

Table 4-6 Request header parameters

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

Table 4-7 Request body parameters

Parameter	Mandatory	Type	Description
key	Yes	String	Key Minimum: 0 Maximum: 2048
title	Yes	String	Name

Response Parameters

Status code: 200

Table 4-8 Response body parameters

Parameter	Type	Description
error	Error object	Response error.
result	PublicKey object	Result

Parameter	Type	Description
status	String	Status code in a response

Table 4-9 Error

Parameter	Type	Description
code	String	Error Codes
message	String	Error message

Table 4-10 PublicKey

Parameter	Type	Description
id	String	Key ID Minimum: 1 Maximum: 40
key	String	Key Minimum: 0 Maximum: 2048
title	String	Name

Example Requests

```
POST https://{endpoint}/v1/users/sshkey
{
  "title" : "z00464114@SZXG4Z004641141",
  "key" : "ssh-rsa AAAAB3NzaXXXXXXXX"
}
```

Example Responses

Status code: 200

OK

```
{
  "result" : {
    "id" : "1034531",
    "title" : "z00464114@SZXG4Z004641141",
    "key" : "ssh-rsa AAAAB3NzaC1yc2XXXXXXXXxxXXX"
  },
  "status" : "success"
}
```

Status Codes

Status Code	Description
200	OK

Error Codes

See [Error Codes](#).

4.1.3 Check whether the private key has the permission to pull code.

Function

Check whether the private key has the permission to pull code.

URI

POST /v1/users/sshkey/privatekey/verify

Request Parameters

Table 4-11 Request header parameters

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

Table 4-12 Request body parameters

Parameter	Mandatory	Type	Description
repository_uid	Yes	String	Repository UUID, which is returned by the CreateRepository interface. Minimum: 32 Maximum: 32
private_key	Yes	String	Private key Minimum: 0 Maximum: 2048

Response Parameters

Status code: 200

Table 4-13 Response body parameters

Parameter	Type	Description
error	Error object	Response error.
result	String	Result
status	String	Status code in a response

Table 4-14 Error

Parameter	Type	Description
code	String	Error Codes
message	String	Error message

Example Requests

```
POST https://{endpoint}/v1/users/sshkey/privatekey/verify
{
  "private_key" : "-----BEGIN RSA PRIVATE KEY-----****-----END RSA PRIVATE KEY-----",
  "repository_uuid" : "9b2fab276e9240e09cd01ed0400fc439"
}
```

Example Responses

Status code: 200

OK

```
{
  "status" : "success"
}
```

Status Codes

Status Code	Description
200	OK

Error Codes

See [Error Codes](#).

4.2 Repository

4.2.1 Obtain files in the branch directory.

Function

Obtain files in the branch directory.

URI

GET /v1/repositories/{repository_uuid}/branch/{branch_name}/sub-files

Table 4-15 Path Parameters

Parameter	Mandatory	Type	Description
repository_uuid	Yes	String	Specifies the repository ID.
branch_name	Yes	String	Branch name.

Table 4-16 Query Parameters

Parameter	Mandatory	Type	Description
path	No	String	Path
offset	No	Integer	Offset
limit	No	Integer	Number of Records

Request Parameters

Table 4-17 Request header parameters

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

Response Parameters

Status code: 200

Table 4-18 Response body parameters

Parameter	Type	Description
error	Error object	Response error.
result	LogsTreeList object	Result
status	String	Response status.

Table 4-19 Error

Parameter	Type	Description
code	String	Error Codes
message	String	Error message

Table 4-20 LogsTreeList

Parameter	Type	Description
trees	Array of LogsTree objects	File Log Tree
total	Integer	By path limit

Table 4-21 LogsTree

Parameter	Type	Description
blob_id	String	Specifies the storage block ID.
commit	Commit object	Details
file_name	String	File Name
file_path	String	File path.
md5	String	MD5
type	String	Storage type

Table 4-22 Commit

Parameter	Type	Description
author_email	String	Author Email
author_name	String	Author
authored_date	String	Author Submission Time
committed_date	String	Submitted
committer_email	String	Submit Author Email
committer_name	String	Requester
format	Object	Details about the file change. The format is determined by the stat_format parameter.
id	String	Submit the corresponding SHA ID.
message	String	Submitted Information
parent_ids	Array of strings	Parent Submission ID

Example Requests

```
GET https://{endpoint}/v1/repositories/{repository_uuid}/branch/{branch_name}/sub-files
```

Example Responses

Status code: 200

OK

```
{
  "result": {
    "trees": [ {
      "blob_id": "ef69ef5dc0c53022fe52a4d02de15f73d96e055c",
      "file_name": "csharpDemo.png",
      "file_path": "images/csharpDemo.png",
      "md5": "8830d25fe6b4e2cd14d75ddb5cae73a7",
      "type": "blob",
      "commit": {
        "id": "1cfd6d81fff948828c7f59bda74c0ef80e53def0",
        "message": "init\n",
        "parent_ids": [ ],
        "author_name": "repo",
        "author_email": "repo@huawei.com",
        "committed_date": "2019-11-13 11:29:26",
        "committer_name": "repo",
        "committer_email": "repo@huawei.com"
      }
    }
  ],
  "total": 5
},
```



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

Status Codes

Status Code	Description
200	OK

Error Codes

See [Error Codes](#).

4.2.2 Obtaining Warehouse Statistics

Function

Obtaining Warehouse Statistics

URI

GET /v1/repositories/{repository_uuid}/statistic-data

Table 4-23 Path Parameters

Parameter	Mandatory	Type	Description
repository_uuid	Yes	String	Specifies the repository ID.

Request Parameters

Table 4-24 Request header parameters

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

Response Parameters

Status code: 200

Table 4-25 Response body parameters

Parameter	Type	Description
error	Error object	Response error.
result	RepositoryStatisticsVO object	Result
status	String	Status code in a response

Table 4-26 Error

Parameter	Type	Description
code	String	Error Codes
message	String	Error message

Table 4-27 RepositoryStatisticsVO

Parameter	Type	Description
repoName	String	Repository name.
commitCount	Integer	Number of the commit times
repoSize	String	Repository capacity
lastCommitTime	String	Last Submission Time
codeLines	Integer	Indicates the number of code lines.
branchCount	Integer	Number of Branches
archiveUrl	String	Code Repository Download Address

Example Requests

```
GET https://{endpoint}/v1/repositories/{repository_uuid}/statistic-data
```

Example Responses

Status code: 200

OK

```
{  
  "result" : {  
    "repoName" : "repoName",  
    "commitCount" : "111",  
    "repoSize" : "0.26 MB",
```

```
"lastCommitTime" : "2020-01-01 11:11:11",
"codeLines" : "368965",
"branchCount" : "222",
"archiveUrl" : "https://devcloud.cn-north-7.ulanhqab.huawei.com/repo/v1/repo/repository/2111663431/
archive"
},
"status" : "success"
}
```

Status Codes

Status Code	Description
200	OK

Error Codes

See [Error Codes](#).

4.2.3 Obtains image files of a specific branch in a repository.

Function

Obtains image files of a specific branch in a repository.

URI

GET /v1/repositories/{repository_uuid}/branch/{branch_name}/image

Table 4-28 Path Parameters

Parameter	Mandatory	Type	Description
repository_uuid	Yes	String	Specifies the repository ID.
branch_name	Yes	String	Branch name.

Table 4-29 Query Parameters

Parameter	Mandatory	Type	Description
path	Yes	String	Image Path

Request Parameters

Table 4-30 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	No	String	Specifies the user token. It can be obtained by calling an IAM API. The value of X-Subject-Token in the response header is the user token.

Response Parameters

None

Example Requests

GET https://{endpoint}/v1/repositories/{repository_uuid}/branch/{branch_name}/image

Example Responses

None

Status Codes

Status Code	Description
200	OK

Error Codes

See [Error Codes](#).

4.2.4 Modifying the Status of a Repository Referenced by a Pipeline

Function

Modifying the Status of a Repository Referenced by a Pipeline

URI

PUT /v1/repositories/{repository_uuid}/pipeline

Table 4-31 Path Parameters

Parameter	Mandatory	Type	Description
repository_uuid	Yes	String	Specifies the repository ID.

Request Parameters

Table 4-32 Request header parameters

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

Response Parameters

Status code: 200

Table 4-33 Response body parameters

Parameter	Type	Description
error	Error object	Response error.
result	Boolean	Result
status	String	Status code in a response

Table 4-34 Error

Parameter	Type	Description
code	String	Error Codes
message	String	Error message

Example Requests

```
PUT https://{endpoint}/v1/repositories/{repository_uuid}/pipeline
```

Example Responses

Status code: 200

OK

```
{
  "result" : true,
  "status" : "success"
}
```

Status Codes

Status Code	Description
200	OK

Error Codes

See [Error Codes](#).

4.2.5 Checks whether a user has the warehouse administrator permission.

Function

Checks whether a user has the warehouse administrator permission.

URI

GET /v1/repositories/{repository_uuid}/master

Table 4-35 Path Parameters

Parameter	Mandatory	Type	Description
repository_uuid	Yes	String	Specifies the repository ID.

Request Parameters

Table 4-36 Request header parameters

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

Response Parameters

Status code: 200

Table 4-37 Response body parameters

Parameter	Type	Description
error	Error object	Response error.
result	Boolean	Result
status	String	Status code in a response

Table 4-38 Error

Parameter	Type	Description
code	String	Error Codes
message	String	Error message

Example Requests

```
GET https://{endpoint}/v1/repositories/{repository_uuid}/master
```

Example Responses

Status code: 200

OK

```
{  
  "result" : true,  
  "status" : "success"  
}
```

Status Codes

Status Code	Description
200	OK

Error Codes

See [Error Codes](#).

4.2.6 Sets whether a repository is in public or private state.

Function

Sets whether a repository is in public or private state.

URI

PUT /v2/repositories/{repository_uuid}/template-status

Table 4-39 Path Parameters

Parameter	Mandatory	Type	Description
repository_uuid	Yes	String	Specifies the repository ID.

Request Parameters

Table 4-40 Request header parameters

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

Table 4-41 Request body parameters

Parameter	Mandatory	Type	Description
template_type	Yes	String	Template Type
code_title	No	String	Code Template Name
creator_name	No	String	Name of the creator
code_description	No	String	Code Template Description
languages	No	Array of strings	Language
platform	No	Array of strings	Template Platform
entertype	No	Array of strings	Template Type

Response Parameters

Status code: 200

Table 4-42 Response body parameters

Parameter	Type	Description
error	Error object	Response error.
result	String	Result
status	String	Status code in a response

Table 4-43 Error

Parameter	Type	Description
code	String	Error Codes
message	String	Error message

Example Requests

```
PUT https://{endpoint}/v2/repositories/{repository_uuid}/template-status
{
  "template_type": "SHARE",
  "code_title": "exampleTitle",
  "creator_name": "exampleName",
  "code_description": "exampleDescription",
  "languages": ['Java'],
  "platform": ['Console'],
  "entertype": ['AI']
}
```

Example Responses

Status code: 200

OK

```
{
  "result" : null,
  "error" : null,
  "status" : "success"
}
```

Status Codes

Status Code	Description
200	OK

Error Codes

See [Error Codes](#).

4.2.7 Obtains the content of a specified file in a specified branch of a repository.

Function

Obtains the content of a specified file in a specified branch of a repository.

URI

GET /v1/repositories/{repository_uuid}/branch/{branch_name}/file

Table 4-44 Path Parameters

Parameter	Mandatory	Type	Description
repository_uuid	Yes	String	Specifies the repository ID.
branch_name	Yes	String	Branch Name

Table 4-45 Query Parameters

Parameter	Mandatory	Type	Description
path	Yes	String	File path.

Request Parameters

Table 4-46 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	No	String	Specifies the user token. It can be obtained by calling an IAM API. The value of X-Subject-Token in the response header is the user token.

Response Parameters

Status code: 200

Table 4-47 Response body parameters

Parameter	Type	Description
error	Error object	Response error.
result	FilesResponseInfo object	Result
status	String	Status code in a response

Table 4-48 Error

Parameter	Type	Description
code	String	Error Codes
message	String	Error message

Table 4-49 FilesResponseInfo

Parameter	Type	Description
file_name	String	File Name
file_path	String	Path
size	String	File size
encoding	String	Encoding Type
ref	String	Branch name.
blob_id	String	File block ID.
file_type	String	File Type
content	String	File content.

Example Requests

GET https://{endpoint}/v1/repositories/{repository_uuid}/branch/{branch_name}/file

Example Responses

Status code: 200

OK

```
{
  "result" : [ {
    "file_name" : "README.md",
    "file_path" : "README.md",
    "size" : 666,
```

```

"encoding" : "base64",
"ref" : "master",
"blob_id" : "75c19b5b1904a72c8a9959c98ae6884d0ace0279",
"file_type" : "text/plain; charset=utf-8",
"content" : "RUVFRUVFRUU="
}],
"status" : "success"
}

```

Status Codes

Status Code	Description
200	OK

Error Codes

See [Error Codes](#).

4.2.8 Obtaining the List of Public Sample Templates

Function

Obtaining the List of Public Sample Templates

URI

GET /v2/repositories/repository-templates

Table 4-50 Query Parameters

Parameter	Mandatory	Type	Description
platform	No	String	Template Platform Type
language	No	String	Language
pipeline	No	String	Whether pipelines are supported
enter_type	No	String	Template category
search	No	String	Template Name
date_order	No	String	Sorting Templates by Date
used_time_order	No	String	Sorting Profile Reference Times
type	No	String	Template Public Type
region	No	String	Region Name
page_no	Yes	Integer	Indicates the page number.

Parameter	Mandatory	Type	Description
page_size	Yes	Integer	Number of data records on each page

Request Parameters

Table 4-51 Request header parameters

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

Response Parameters

Status code: 200

Table 4-52 Response body parameters

Parameter	Type	Description
error	Error object	Response error.
result	TemplateRepositoryList object	Result
status	String	Status code in a response

Table 4-53 Error

Parameter	Type	Description
code	String	Error Codes
message	String	Error message

Table 4-54 TemplateRepositoryList

Parameter	Type	Description
projects	Array of TemplateRepository objects	Template list
total	Integer	Templates

Table 4-55 TemplateRepository

Parameter	Type	Description
id	Integer	Unique ID of a template.
name	String	Template Name
template_name	String	Name of the warehouse associated with the template.
tags	Array of strings	Template Label
description	String	Description
brief_introduction	String	Introduction to Model Templates
auto_pending_pipelines	Integer	Whether to automatically create a pipeline
language	String	Template Language Classification
created_at	String	Time when the template was created.
used_times	Integer	Number of times a profile is referenced
liked_times	Integer	Number of times that a template is liked.
creator_name	String	Template Creator
https_url	String	Template HTTPS Link

Example Requests

```
GET https://{endpoint}/v2/repositories/repository-templates?page_no=1&page_size=10
```

Example Responses

Status code: 200

OK

```
{
  "result": {
```

```
"projects" : [ {
  "id" : 453613,
  "name" : "javaMavenDemo",
  "template_name" : "javaMavenDemo",
  "tags" : [ "UnsupportedPipeline", "OTHERS", "Kunpeng ARM64" ],
  "description" : "javaMavenDemo",
  "brief_introduction" : "javaMavenDemo",
  "auto_pending_pipelines" : 0,
  "language" : "OTHERS",
  "created_at" : "Sep 25, 2019 4:03:31 PM",
  "used_times" : 58,
  "liked_times" : 1,
  "creator_name" : "repo",
  "https_url" : "https://repo.alpha.devcloud.inhuawei.com/chentest00001/javaMavenDemo.git"
}],
"total" : 100
},
"status" : "success"
}
```

Status Codes

Status Code	Description
200	OK

Error Codes

See [Error Codes](#).

4.2.9 Creating a repository

Function

Creates a repository on a specified project with the specified name. Input parameters: repository name, template ID, whether to import project members, and home project

URI

POST /v1/repositories

Request Parameters

Table 4-56 Request header parameters

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

Table 4-57 Request body parameters

Parameter	Mandatory	Type	Description
import_members	No	Integer	Indicates whether to import project members. The options are as follows: 0->Do not import project members; 1->Import project members
name	Yes	String	Repository name. The options are as follows: The value can contain uppercase letters, lowercase letters, digits, hyphens (-), and underscores (_), and must start with a letter.
project_uuid	Yes	String	Specifies the UUID of a project. Minimum: 32 Maximum: 32
template_id	No	String	ID of the replication template.
visibility_level	No	Integer	Warehouse status. The options are as follows: 0: private; 20: public and read-only
import_url	No	String	Encrypting the HTTPS Address of the Template Repository Using Base64
description	No	String	Repository description.
gitignore_id	No	String	Generating the .gitignore File Based on the Programming Language
license_id	No	Integer	License ID
enable_readme	No	Integer	Indicates whether to allow README file generation.
caller	No	String	Invoker

Response Parameters

Status code: 200

Table 4-58 Response body parameters

Parameter	Type	Description
error	Error object	Response error.
result	Repository object	Result
status	String	Response Status

Table 4-59 Error

Parameter	Type	Description
code	String	Error Codes
message	String	Error message

Table 4-60 Repository

Parameter	Type	Description
repository_uuid	String	UUID of the repository to be created. Minimum: 32 Maximum: 32

Example Requests

```
POST https://{endpoint}/v1/repositories
{
  "project_uuid": "4838955a48e2492bbe44b31bc4c272f6",
  "name": "demotest1",
  "import_members": "0",
  "visibility_level": "20",
}
```

Example Responses

Status code: 200

OK

```
{
  "result" : {
    "repository_uuid" : "d161fd00d9194816a455cb3c1d6a783e"
  },
  "status" : "success"
}
```

Status Codes

Status Code	Description
200	OK

Error Codes

See [Error Codes](#).

4.2.10 Obtain the short ID of the repository based on the repository name and group name to combine the short ID with the URL of the submission details page corresponding to commitid.

Function

This interface is used to obtain the short ID of a repository, which is used to obtain the URL of the repository details page.

URI

GET /v1/repositories/repoId

Table 4-61 Query Parameters

Parameter	Mandatory	Type	Description
group_name	Yes	String	Repository group name (Example: git@repo.alpha.devcloud.inh wei.com:Demo00228/ testword.git Group name: Demo00228)
repository_name	Yes	String	Repository name

Request Parameters

Table 4-62 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Specifies the user token. It can be obtained by calling the IAM API (value of X-Subject-Token in the response header).

Response Parameters

Status code: 200

Table 4-63 Response body parameters

Parameter	Type	Description
error	Error object	Response error.
result	Integer	Result
status	String	Status code in a response

Table 4-64 Error

Parameter	Type	Description
code	String	Error Codes
message	String	Error message

Example Requests

```
GET https://{endpoint}/v1/repositories/repoid
```

Example Responses

Status code: 200

OK

```
{  
  "result" : 468485,  
  "status" : "success"  
}
```

Status Codes

Status Code	Description
200	OK

Error Codes

See [Error Codes](#).

4.2.11 Queries the submission of a branch of a repository based on the group name and repository name.

Function

This API is used to obtain the submission list based on the repository group name, repository name, and branch.

URI

GET /v1/repositories/{group_name}/{repository_name}/commits

Table 4-65 Path Parameters

Parameter	Mandatory	Type	Description
group_name	Yes	String	Repository group name (Example: git@repo.alpha.devcloud.inhuatech.com:Demo00228/testword.git Group name: Demo00228)
repository_name	Yes	String	Repository name

Table 4-66 Query Parameters

Parameter	Mandatory	Type	Description
page_index	No	Integer	Pagination Index
page_size	No	Integer	Pagination Index
ref_name	Yes	String	Branch or label name. The SHA format is supported.

Request Parameters

Table 4-67 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Specifies the user token. It can be obtained by calling an IAM API. 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
error	Error object	Response error.
result	CommitList object	Result
status	String	Status code in a response

Table 4-69 Error

Parameter	Type	Description
code	String	Error Codes
message	String	Error message

Table 4-70 CommitList

Parameter	Type	Description
commits	Array of Commit objects	Submission List
total	Integer	Total Submitted

Table 4-71 Commit

Parameter	Type	Description
author_email	String	Author Email
author_name	String	Author
authored_date	String	Author Submission Time
committed_date	String	Submitted
committer_email	String	Submit Author Email
committer_name	String	Requester
format	Object	Details about the file change. The format is determined by the stat_format parameter.
id	String	Submit the corresponding SHA ID.
message	String	Submitted Information
parent_ids	Array of strings	Parent Submission ID

Example Requests

GET https://{endpoint}/v1/repositories/{group_name}/{repository_name}/commits

Example Responses

Status code: 200

OK

```
{
  "result": {
    "total": 1,
    "commits": [ {
      "id": "a27fafa7d8da2dd55f691bef989226d22f50d8ce",
      "message": "Initial commit",
      "parent_ids": [ ],
      "authored_date": null,
      "author_name": "repo",
      "author_email": "repo@huawei.com",
      "committed_date": "2019-10-15 09:40:29",
      "committer_name": "repo",
      "committer_email": "a618e34bd5704be3ae3395dfede06041@huawei.com"
    } ]
  },
  "status": "success"
}
```

Status Codes

Status Code	Description
200	OK

Error Codes

See [Error Codes](#).

4.2.12 Querying the Branch of a Warehouse

Function

This API is used to obtain the branch list of a specified repository based on the repository ID.

URI

GET /v1/repositories/{repository_id}/branches

Table 4-72 Path Parameters

Parameter	Mandatory	Type	Description
repository_id	Yes	String	Primary key ID of a warehouse.

Request Parameters

Table 4-73 Request header parameters

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

Response Parameters

Status code: 200

Table 4-74 Response body parameters

Parameter	Type	Description
error	Error object	Response error.
result	BranchList object	Result
status	String	Status code in a response

Table 4-75 Error

Parameter	Type	Description
code	String	Error Codes
message	String	Error message

Table 4-76 BranchList

Parameter	Type	Description
branches	Array of Branch objects	Branch List of a Specified Repository
total	Integer	Total number of branches in a specified repository.

Table 4-77 Branch

Parameter	Type	Description
is_protected	Boolean	Whether to enable the branch protection function
name	String	Branch Name

Example Requests

```
GET https://{endpoint}/v1/repositories/{repository_id}/branches
```

Example Responses

Status code: 200

OK

```
{  
  "result": {
```



```

"total" : 1,
"branches" : [ {
  "name" : "master",
  "is_protected" : false
} ]
},
"status" : "success"
}

```

Status Codes

Status Code	Description
200	OK

Error Codes

See [Error Codes](#).

4.2.13 This API is used to query the submission of a branch of a warehouse based on the warehouse ID. More optional parameters are provided.

Function

This API queries the submission of a branch of a warehouse based on the warehouse ID.

URI

GET /v1/repositories/{repository_id}/commits

Table 4-78 Path Parameters

Parameter	Mandatory	Type	Description
repository_id	Yes	Integer	Warehouse Primary Key ID

Table 4-79 Query Parameters

Parameter	Mandatory	Type	Description
author	No	String	Requester
begin_date	No	String	Submission start date, in the format of yyyy-MM-dd.
end_date	No	String	Submission end date, in the format of yyyy-MM-dd.

Parameter	Mandatory	Type	Description
message	No	String	Details
page_index	No	Integer	Pagination Index Default: 1
page_size	No	Integer	Number of records on each page. Default: 10
path	No	String	Path
ref_name	Yes	String	Branch or label name. The SHA format is supported.
stat_format	No	String	Submitted file change details (excluding diff)

Request Parameters

Table 4-80 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Specifies the user token. It can be obtained by calling the IAM API (value of X-Subject-Token in the response header).

Response Parameters

Status code: 200

Table 4-81 Response body parameters

Parameter	Type	Description
error	Error object	Response error.
result	CommitList object	Result
status	String	Status code in a response

Table 4-82 Error

Parameter	Type	Description
code	String	Error Codes
message	String	Error message

Table 4-83 CommitList

Parameter	Type	Description
commits	Array of Commit objects	Submission List
total	Integer	Total Submitted

Table 4-84 Commit

Parameter	Type	Description
author_email	String	Author Email
author_name	String	Author
authored_date	String	Author Submission Time
committed_date	String	Submitted
committer_email	String	Submit Author Email
committer_name	String	Requester
format	Object	Details about the file change. The format is determined by the stat_format parameter.
id	String	Submit the corresponding SHA ID.
message	String	Submitted Information
parent_ids	Array of strings	Parent Submission ID

Example Requests

```

null
GET https://{endpoint}/v1/repositories/{repository_id}/commits

```

Example Responses

Status code: 200

OK

```
{
  "result": {
    "total": 1,
    "commits": [ {
      "id": "a27fafa7d8da2dd55f691bef989226d22f50d8ce",
      "message": "Initial commit",
      "parent_ids": [ ],
      "authored_date": null,
      "author_name": "repo",
      "author_email": "repo@huawei.com",
      "committed_date": "2019-10-15 09:40:29",
      "committer_name": "repo",
      "committer_email": "a618e34bd5704be3ae3395dfede06041@huawei.com"
    } ]
  },
  "status": "success"
}
```

Status Codes

Status Code	Description
200	OK

Error Codes

See [Error Codes](#).

4.2.14 Adding a Deployment Key

Function

Adding a Deployment Key

URI

POST /v2/repositories/{repository_id}/deploy-keys

Table 4-85 Path Parameters

Parameter	Mandatory	Type	Description
repository_id	Yes	Integer	Warehouse Primary Key ID

Request Parameters

Table 4-86 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Specifies the user token. It can be obtained by calling the IAM API (value of X-Subject-Token in the response header).

Table 4-87 Request body parameters

Parameter	Mandatory	Type	Description
application	Yes	String	Source of the SSH Key Used for Deployment
can_push	Yes	Boolean	Can the SSH Key Used for Deployment Push Code?
key	Yes	String	SSH key used for deployment
key_title	Yes	String	Name of the SSH key used for deployment

Response Parameters

Status code: 200

Table 4-88 Response body parameters

Parameter	Type	Description
error	Error object	Response error.
result	Key object	Result
status	String	Status code in a response

Table 4-89 Error

Parameter	Type	Description
code	String	Error Codes
message	String	Error message

Table 4-90 Key

Parameter	Type	Description
can_push	Boolean	Whether to allow push
created_at	String	Deployment Key Creation Time
key	String	Deploy Keys
key_id	String	Deployment Key ID Minimum: 1 Maximum: 40
key_title	String	Deployment Key Name

Example Requests

```
POST https://{endpoint}/v2/repositories/{repository_id}/deploy-keys
{
  "key_title": "z00464114@SZXG4Z004641141",
  "key": "ssh-rsa AAAAB3NzaC1yXXXXXXXX",
  "can_push": true,
  "application": ""
}
```

Example Responses

Status code: 200

OK

```
{
  "result": {
    "key_id": "1034546",
    "key_title": "z00464114@SZXG4Z004641141",
    "key": "ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAxxxxxxxxxxxx"
  },
  "status": "success"
}
```

Status Codes

Status Code	Description
200	OK

Error Codes

See [Error Codes](#).

4.2.15 Deleting a Repository Deployment Key

Function

Deleting a Repository Deployment Key

URI

DELETE /v2/repositories/{repository_id}/deploy-keys/{key_id}

Table 4-91 Path Parameters

Parameter	Mandatory	Type	Description
key_id	Yes	Integer	Deployment Key ID
repository_id	Yes	Integer	Warehouse Primary Key ID

Request Parameters

Table 4-92 Request header parameters

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

Response Parameters

Status code: 200

Table 4-93 Response body parameters

Parameter	Type	Description
error	Error object	Response error.
result	Boolean	Result
status	String	Status code in a response

Table 4-94 Error

Parameter	Type	Description
code	String	Error Codes
message	String	Error message

Example Requests

```
DELETE https://{endpoint}/v2/repositories/{repository_id}/deploy-keys/{key_id}
```

Example Responses

Status code: 200

OK

```
{  
  "result" : "false",  
  "status" : "success"  
}
```

Status Codes

Status Code	Description
200	OK

Error Codes

See [Error Codes](#).

4.2.16 Obtains the last submission statistics of the warehouse.

Function

Obtains the last submission statistics of the warehouse.

URI

```
GET /v1/repositories/{repository_id}/statistics
```

Table 4-95 Path Parameters

Parameter	Mandatory	Type	Description
repository_id	Yes	String	Primary key ID of a warehouse.

Table 4-96 Query Parameters

Parameter	Mandatory	Type	Description
branch_name	Yes	String	Branch Name

Request Parameters

Table 4-97 Request header parameters

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

Response Parameters

Status code: 200

Table 4-98 Response body parameters

Parameter	Type	Description
error	Error object	Response error.
result	RepoCommit Statistics object	Result
status	String	Status code in a response

Table 4-99 Error

Parameter	Type	Description
code	String	Error Codes
message	String	Error message

Table 4-100 RepoCommitStatistics

Parameter	Type	Description
all_branch_commits_count	Integer	Total Warehouse Submission Times
codelines	Array of RepoDailyCodeline objects	Daily Submitted Code Lines in Last 15 Days
count	Integer	Total Submission Times of the Corresponding Branch Warehouse
event	RepoStatisticsEvent object	Warehouse Statistics Status
statistics	Array of RepoStatistics objects	Warehouse Statistics List
total	Integer	Warehouse Statistics Times

Table 4-101 RepoDailyCodeline

Parameter	Type	Description
additions	Integer	Add code lines every day.
date	String	Date
deletions	Integer	Deleting Code Lines Every Day

Table 4-102 RepoStatisticsEvent

Parameter	Type	Description
branch	String	Branch Name Minimum: 1 Maximum: 260
created_at	String	Time when warehouse statistics are created.
date	String	Date when warehouse statistics are collected.
id	Integer	ID of a warehouse statistics event.
project_id	Integer	Specifies the repository ID.
status	String	Status of warehouse statistics: waiting for statistics collection, waiting for statistics collection, active statistics collection, and finish statistics collection.

Parameter	Type	Description
updated_at	String	Time when warehouse statistics are updated.
user_id	Integer	Specifies the user ID.

Table 4-103 RepoStatistics

Parameter	Type	Description
add_lines	Integer	Add Code Line
branch	String	Branch Name Minimum: 1 Maximum: 260
commit_count	Integer	Number of the commit times
created_at	String	Time when warehouse statistics are created.
delete_lines	Integer	Deleting Code Lines
id	Integer	Warehouse Statistics Record ID
project_id	Integer	Specifies the repository ID.
updated_at	String	Time when warehouse statistics are updated.
user_name	String	User Name

Example Requests

None

Example Responses

Status code: 200

OK

```
{
  "result": {
    "event": null,
    "total": 0,
    "statistics": [ ],
    "all_branch_commits_count": 1,
    "count": 1,
    "codelines": {
      "additions": 0,
      "deletions": 0,
      "date": "20191015"
    }
  },
  "status": "success"
}
```

Status Codes

Status Code	Description
200	OK

Error Codes

See [Error Codes](#).

4.2.17 Repository Statistics

Function

This API is used to query the code submission record statistics of a repository based on the repository short ID.

URI

POST /v1/repositories/{repository_id}/statistics

Table 4-104 Path Parameters

Parameter	Mandatory	Type	Description
repository_id	Yes	Integer	Primary key ID of a warehouse.

Request Parameters

Table 4-105 Request header parameters

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

Table 4-106 Request body parameters

Parameter	Mandatory	Type	Description
branch_name	Yes	String	Repository Branch Name

Response Parameters

Status code: 200

Table 4-107 Response body parameters

Parameter	Type	Description
error	Error object	Response error.
result	RepoStatistic sLaunch object	Result
status	String	Status code in a response

Table 4-108 Error

Parameter	Type	Description
code	String	Error Codes
message	String	Error message

Table 4-109 RepoStatisticsLaunch

Parameter	Type	Description
can_statistics	Boolean	Indicates whether the warehouse can collect statistics.
join_id	String	Specifies the ID of the sidekiq task.
message	String	Information returned after warehouse statistics collection is started.

Example Requests

```
POST https://{endpoint}/v1/repositories/{repository_id}/statistics
{
  "branch_name": "master"
}
```

Example Responses

Status code: 200

OK

```
{
  "result": {
    "can_statistics": false,
```

```
"message" : "operation success",
"join_id" : "2a48a9fd140dd4efabcc6e07"
},
"status" : "success"
}
```

Status Codes

Status Code	Description
200	OK

Error Codes

See [Error Codes](#).

4.2.18 Delete repositories

Function

Deletes a specified repository based on the 32-bit UUID of the repository.

URI

DELETE /v1/repositories/{repository_uuid}

Table 4-110 Path Parameters

Parameter	Mandatory	Type	Description
repository_uuid	Yes	String	Repository UUID (returned by the CreateRepository interface), which is used to specify the repository to be deleted.

Request Parameters

Table 4-111 Request header parameters

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

Response Parameters

Status code: 200

Table 4-112 Response body parameters

Parameter	Type	Description
error	Error object	Response error.
result	Boolean	Result
status	String	Status code in a response

Table 4-113 Error

Parameter	Type	Description
code	String	Error Codes
message	String	Error message

Example Requests

```
DELETE https://{endpoint}/v1/repositories/{repository_uuid}
```

Example Responses

Status code: 200

OK

```
{  
  "result" : "false",  
  "status" : "success"  
}
```

Status Codes

Status Code	Description
200	OK

Error Codes

See [Error Codes](#).

4.2.19 Check the repository creation status.

Function

Obtains the warehouse status.

URI

GET /v1/repositories/{repository_uuid}/status

Table 4-114 Path Parameters

Parameter	Mandatory	Type	Description
repository_uuid	Yes	String	UUID of the repository, which is used to specify the repository to be viewed.

Request Parameters

Table 4-115 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Specifies a user token. It can be obtained by calling the IAM API (value of X-Subject-Token in the response header).

Response Parameters

Status code: 200

Table 4-116 Response body parameters

Parameter	Type	Description
error	Error object	Response error.
result	Integer	Result
status	String	Status code in a response

Table 4-117 Error

Parameter	Type	Description
code	String	Error Codes
message	String	Error message

Example Requests

```
GET https://{endpoint}/v1/repositories/{repository_uuid}/status
```

Example Responses

Status code: 200

OK

```
{  
  "result" : 0,  
  "status" : "success"  
}
```

Status Codes

Status Code	Description
200	OK

Error Codes

See [Error Codes](#).

4.2.20 Querying the Tag List of a Warehouse

Function

This API is used to query the branch corresponding to a specified repository.

URI

```
GET /v2/repositories/{repository_id}/tags
```

Table 4-118 Path Parameters

Parameter	Mandatory	Type	Description
repository_id	Yes	String	Primary key ID of a warehouse.

Request Parameters

Table 4-119 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Specifies the user token. It can be obtained by calling the IAM API (value of X-Subject-Token in the response header).

Response Parameters

Status code: 200

Table 4-120 Response body parameters

Parameter	Type	Description
error	Error object	Response error.
result	TagList object	Result
status	String	Status code in a response

Table 4-121 Error

Parameter	Type	Description
code	String	Error Codes
message	String	Error message

Table 4-122 TagList

Parameter	Type	Description
tags	Array of Tag objects	Tag list of a specified repository.
total	Integer	Total number of tags in a specified repository.

Table 4-123 Tag

Parameter	Type	Description
is_double_name	Boolean	Indicates whether the name is the same as that of a branch.
name	String	Label Minimum: 1 Maximum: 260

Example Requests

```
GET https://{endpoint}/v2/repositories/{repository_id}/tags
```

Example Responses

Status code: 200

OK

```
{
  "result": {
    "total": 1,
    "tags": [ {
      "name": "abctest",
      "is_double_name": "false"
    } ]
  },
  "status": "success"
}
```

Status Codes

Status Code	Description
200	OK

Error Codes

See [Error Codes](#).

4.2.21 Adding a Tag

Function

Adding a Tag

URI

POST /v2/repositories/{repository_id}/tags

Table 4-124 Path Parameters

Parameter	Mandatory	Type	Description
repository_id	Yes	Integer	Warehouse Primary Key ID

Request Parameters

Table 4-125 Request header parameters

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

Table 4-126 Request body parameters

Parameter	Mandatory	Type	Description
tag_name	Yes	String	Tag Name
ref	Yes	String	Branch Name
message	No	String	Remarks

Response Parameters

Status code: 200

Table 4-127 Response body parameters

Parameter	Type	Description
error	Error object	Response error.
result	AddTagsResponse object	Result
status	String	Status code in a response

Table 4-128 Error

Parameter	Type	Description
code	String	Error Codes
message	String	Error message

Table 4-129 AddTagsResponse

Parameter	Type	Description
name	String	Tag Name
message	String	Remarks
commit	CommitRepo V2 object	Details

Table 4-130 CommitRepoV2

Parameter	Type	Description
id	String	Submit the corresponding SHA ID.
short_id	String	Submit the corresponding short SHA ID.
created_at	String	Created
title	String	Submit Title
parent_ids	Array of strings	Parent Submission ID
message	String	Details
author_name	String	Author
committer_name	String	Requester
committed_date	String	Submitted

Example Requests

```
POST https://{endpoint}/v2/repositories/{repository_id}/tags
{
  "tag_name": "test20220920",
  "ref": "master",
  "message": "test20220920"
}
```

Example Responses

Status code: 200

OK

```
{
  "result": {
    "name": "test_tag_name",
    "message": "test message",
    "commit": {
      "id": "3b68902cf0eb7de3d5c8757e4a990bc3fe0af349",
      "short_id": "3b68902c",
      "created_at": "2022-03-08T03:14:39.000Z",
      "title": "Creating a File",
      "parent_ids": [ "08618c900a4048aae7e4cd88913d3d521eae02eb" ],
      "message": "Creating a File",
      "author_name": "repo",
      "committer_name": "repo",
      "committed_date": "2022-03-08T03:14:39.000Z"
    }
  },
  "status": "success"
}
```

Status Codes

Status Code	Description
200	OK

Error Codes

See [Error Codes](#).

4.2.22 Querying the Details of a Warehouse

Function

This interface is used to obtain warehouse information based on the warehouse UUID (returned by the CreateRepository interface). The returned information contains the ID, name, group name, and repository access URL.

URI

GET /v2/repositories/{repository_uuid}

Table 4-131 Path Parameters

Parameter	Mandatory	Type	Description
repository_uuid	Yes	String	UUID of the repository.

Request Parameters

Table 4-132 Request header parameters

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

Response Parameters

Status code: 200

Table 4-133 Response body parameters

Parameter	Type	Description
error	Error object	Response error.
result	RepoInfoV2 object	Result
status	String	Response status.

Table 4-134 Error

Parameter	Type	Description
code	String	Error Codes
message	String	Error message

Table 4-135 RepoInfoV2

Parameter	Type	Description
created_at	String	Created
creator_name	String	User name of the creator. If the user is a tenant, the user name is the same as the tenant name.
domain_name	String	Tenant name of the creator.

Parameter	Type	Description
group_name	String	Repository group name (Example: git@repo.alpha.devcloud.inhuawei.com:Demo00228/testword.git Group name: Demo00228)
https_url	String	URL used when HTTPS is used to clone the repository.
iam_user_uid	String	IAM user UUID of the user.
is_owner	Integer	Indicates whether the current user is the creator of the warehouse. 1: yes; 0: no
lfs_size	String	LFS capacity of the warehouse, in MB. If the capacity is greater than 1024 MB, the unit is GB.
project_is_deleted	String	Whether the project is deleted
project_uuid	String	Project UUID.
repository_id	Integer	Warehouse Primary Key ID
repository_name	String	Repository name
repository_size	String	Total repository capacity = Repository LFS capacity + Git repository capacity. The unit is MB. If the value is greater than 1024 MB, the unit is GB.
repository_uuid	String	Repository UUID, which is returned by the CreateRepository API.
ssh_url	String	URL used for cloning the repository in SSH mode.
star	Boolean	Indicates whether the current user adds the warehouse to favorites.
status	Integer	Warehouse status. 0: The warehouse is successfully created. 1: The repository is being created. 2: creation failed 3: The warehouse is frozen. 4: The warehouse has been closed.
updated_at	String	Update time.
userRole	Integer	User permission in the repository. The options are as follows: 20: read-only member 30: common member 40: administrator
visibility_level	Integer	Indicates whether the repository is visible. The options are 0 (private repository) and 20 (public repository).

Parameter	Type	Description
web_url	String	Web URL. If you access the URL, the repository details page is displayed.

Example Requests

```
GET https://{endpoint}/v2/repositories/{repository_uuid}
```

Example Responses

Status code: 200

OK

```
{
  "result" : {
    "repository_uuid" : "045ceabb9aea40579d5a759e84fa7f0a",
    "repository_id" : 468485,
    "repository_name" : "CreatRepository_MI9Af",
    "ssh_url" : "git@repo.alpha.devcloud.inhuawei.com:Demo00228/CreatRepository_MI9Af.git",
    "https_url" : "https://repo.alpha.devcloud.inhuawei.com/Demo00228/CreatRepository_MI9Af.git",
    "group_name" : "Demo00228",
    "web_url" : "https://alpha.devcloud.inhuawei.com/repo/468485/home",
    "visibility_level" : 0,
    "created_at" : "2019-10-15 09:40:27",
    "updated_at" : null,
    "repository_size" : null,
    "lfs_size" : null,
    "creator_name" : null,
    "domain_name" : null,
    "star" : null,
    "status" : null,
    "is_owner" : null,
    "iam_user_uuid" : null,
    "project_uuid" : "4838955a48e2492bbe44b31bc4c272f6",
    "project_is_deleted" : "false",
    "userRole" : null
  },
  "status" : "success"
}
```

Status Codes

Status Code	Description
200	OK

Error Codes

See [Error Codes](#).

4.2.23 Obtains the number of submitted code lines.

Function

Obtains the number of submitted code lines of a specified branch in the code repository on a specified date.

URI

GET /v3/repositories/{repository_id}/commit-lines

Table 4-136 Path Parameters

Parameter	Mandatory	Type	Description
repository_id	Yes	Integer	Warehouse Short ID

Table 4-137 Query Parameters

Parameter	Mandatory	Type	Description
ref_name	Yes	String	Branch Name
begin_date	Yes	String	Submission start date, in the format of yyyy-MM-dd.
end_date	Yes	String	Submission end date, in the format yyyy-MM-dd. The interval between begin_date and end_date cannot exceed 60 days.

Request Parameters

Table 4-138 Request header parameters

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

Response Parameters

Status code: 200

Table 4-139 Response body parameters

Parameter	Type	Description
error	Error object	Response error.
result	CommitStatistic object	Number of lines added and deleted in the code
status	String	Status code in a response

Table 4-140 Error

Parameter	Type	Description
code	String	Error Codes
message	String	Error message

Table 4-141 CommitStatistic

Parameter	Type	Description
additions	Integer	Number of added lines.
deletions	Integer	Number of rows deleted

Example Requests

```
GET https://{endpoint}/v2/repositories/907199/commit-lines
```

Example Responses

Status code: 200

OK

```
{
  "result" : {
    "additions" : 9,
    "deletions" : 6
  },
  "status" : "success"
}
```

Status Codes

Status Code	Description
200	OK

Error Codes

See [Error Codes](#).

4.2.24 Download Repository

Function

Download the repository in the specified format.

URI

GET /v2/repositories/{repository_uuid}/archive

Table 4-142 Path Parameters

Parameter	Mandatory	Type	Description
repository_uuid	Yes	String	UUID of the repository.

Table 4-143 Query Parameters

Parameter	Mandatory	Type	Description
sha	Yes	String	Branch Name
format	Yes	String	Format of the Downloaded Package

Request Parameters

Table 4-144 Request header parameters

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

Response Parameters

None

Example Requests

GET https://{endpoint}/v2/repositories/{repository_uuid}/archive

Example Responses

None

Status Codes

Status Code	Description
200	OK

Error Codes

See [Error Codes](#).

4.2.25 Creating a Protection Branch

Function

Creating a Protection Branch

URI

PUT /v2/repositories/{repository_id}/branch/{branch_name}/protect

Table 4-145 Path Parameters

Parameter	Mandatory	Type	Description
repository_id	Yes	Integer	Warehouse Primary Key ID
branch_name	Yes	String	Branch name.

Request Parameters

Table 4-146 Request header parameters

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

Table 4-147 Request body parameters

Parameter	Mandatory	Type	Description
access_level	Yes	AddProtectAccessLevel object	Creating a Protection Branch Permission

Table 4-148 AddProtectAccessLevel

Parameter	Mandatory	Type	Description
push_access_level	Yes	Integer	Submission permission. The options are as follows: 0: no one can submit; 30: developers and administrators can submit; 40: administrators can submit. Enumeration values: <ul style="list-style-type: none"> • 0 • 30 • 40
merge_access_level	Yes	Integer	Merge permission. 0: No one can merge. 30: Developers and administrators can merge. 40: Administrators can merge. The merge permission must be greater than or equal to the submit permission. Enumeration values: <ul style="list-style-type: none"> • 0 • 30 • 40

Response Parameters

Status code: 200

Table 4-149 Response body parameters

Parameter	Type	Description
error	Error object	Response error.
result	AddProtectResponse object	Result

Parameter	Type	Description
status	String	Response Status

Table 4-150 Error

Parameter	Type	Description
code	String	Error Codes
message	String	Error message

Table 4-151 AddProtectResponse

Parameter	Type	Description
name	String	Branch name.
commit	CommitRepo V2 object	Details
protected	Boolean	Whether to protect DWDM.
developers_can_push	Boolean	Allow Developer to Submit
developers_can_merge	Boolean	Allow Developers to Merge
master_can_push	Boolean	Allow Administrator to Submit
master_can_merge	Boolean	Indicates whether to allow administrators to merge.
no_one_can_push	Boolean	No one is allowed to submit.
no_one_can_merge	Boolean	No one is allowed to merge.
in_an_opened_merge_request	Boolean	Indicates whether a combination request is opened.

Table 4-152 CommitRepoV2

Parameter	Type	Description
id	String	Submit the corresponding SHA ID.

Parameter	Type	Description
short_id	String	Submit the corresponding short SHA ID.
created_at	String	Created
title	String	Submit Title
parent_ids	Array of strings	Parent Submission ID
message	String	Details
author_name	String	Author
committer_name	String	Requester
committed_date	String	Submitted

Example Requests

```
PUT https://{endpoint}/v2/repositories/{repository_id}/branch/{branch_name}/protect
{
  "access_level" : {
    "push_access_level" : 40,
    "merge_access_level" : 40
  }
}
```

Example Responses

Status code: 200

OK

```
{
  "result" : {
    "name" : "master",
    "commit" : {
      "id" : "3b68902cf0eb7de3d5c8757e4a990bc3fe0af349",
      "short_id" : "3b68902c",
      "title" : "Creating a File",
      "created_at" : "2022-03-08T03:14:39.000Z",
      "parent_ids" : [ "08618c900a4048aae7e4cd88913d3d521eae02eb" ],
      "message" : "Creating a File",
      "author_name" : "repo",
      "committer_name" : "repo",
      "committed_date" : "2022-03-08T03:14:39.000Z"
    },
    "protected" : true,
    "developers_can_push" : false,
    "developers_can_merge" : false,
    "master_can_push" : false,
    "master_can_merge" : false,
    "no_one_can_push" : true,
    "no_one_can_merge" : true,
    "in_an_opened_merge_request" : false
  },
}
```



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

Status Codes

Status Code	Description
200	OK

Error Codes

See [Error Codes](#).

4.2.26 Obtaining the Repository Branch List

Function

Obtaining the Repository Branch List

URI

GET /v2/repositories/{repository_id}/branches

Table 4-153 Path Parameters

Parameter	Mandatory	Type	Description
repository_id	Yes	Integer	Warehouse Short ID

Table 4-154 Query Parameters

Parameter	Mandatory	Type	Description
page	No	String	Indicates the page number.
per_page	No	String	Number of data records on each page
match	No	String	Match Conditions

Request Parameters

Table 4-155 Request header parameters

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

Response Parameters

Status code: 200

Table 4-156 Response body parameters

Parameter	Type	Description
error	Error object	Response error.
result	BranchResponse object	Result
status	String	Status code in a response

Table 4-157 Error

Parameter	Type	Description
code	String	Error Codes
message	String	Error message

Table 4-158 BranchResponse

Parameter	Type	Description
branches	Array of BranchItem objects	Branch Information
total	Double	Total

Table 4-159 BranchesItem

Parameter	Type	Description
commit	CommitV2 object	Details
diverging_commit_counts	DivergingCommitCounts object	Submitted Discrepancy Qty.
name	String	Branch Name

Table 4-160 CommitV2

Parameter	Type	Description
committed_date	String	Submitted
committer_name	String	Committed By
id	String	Submission ID
message	String	Details
short_id	String	Submit Short ID
title	String	Submit Title

Table 4-161 DivergingCommitCounts

Parameter	Type	Description
ahead	Double	Leading Submissions
behind	Double	Number of Delayed Submissions

Example Requests

```
GET https://{endpoint}/v2/repositories/{repository_id}/branches
```

Example Responses

Status code: 200

OK

```
{
  "result": {
    "total": 1,
    "branches": [ {
      "name": "master",
```

```

"commit" : {
  "id" : "2912b8f2328e798f7d544272ffaebfccccb598ab",
  "title" : "Initial commit",
  "message" : "Initial commit",
  "committed_date" : "2021-12-06T09:24:50.000+08:00",
  "short_id" : "2912b8f2",
  "committer_name" : "repo"
},
"diverging_commit_counts" : {
  "behind" : 0,
  "ahead" : 0
}
}]
},
"status" : "success"
}

```

Status Codes

Status Code	Description
200	OK

Error Codes

See [Error Codes](#).

4.2.27 Obtaining the List of Warehouse Combination Requests

Function

Obtaining the List of Warehouse Combination Requests

URI

GET /v2/repositories/{repository_id}/merge_request

Table 4-162 Path Parameters

Parameter	Mandatory	Type	Description
repository_id	Yes	Integer	Warehouse Short ID

Table 4-163 Query Parameters

Parameter	Mandatory	Type	Description
state	No	String	Match Conditions
page	No	String	Indicates the page number.

Parameter	Mandatory	Type	Description
per_page	No	String	Number of data records on each page
search	No	String	Match Condition

Request Parameters

Table 4-164 Request header parameters

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

Response Parameters

Status code: 200

Table 4-165 Response body parameters

Parameter	Type	Description
error	Error object	Response error.
result	MergeResult object	Result
status	String	Response Status

Table 4-166 Error

Parameter	Type	Description
code	String	Error Codes
message	String	Error message

Table 4-167 MergeResult

Parameter	Type	Description
closed	Double	Closed Merge Requests
merge_requests	Array of MergeRequestsItem objects	Merge Request Details
merged	Double	Number of Merged Requests
opened	Double	Number of merging requests when the function is enabled
total	Double	Total number of combination requests

Table 4-168 MergeRequestsItem

Parameter	Type	Description
author	Author object	Author
closed_at	String	Closing time
created_at	String	Created
description	String	Combination Request Description
devcloud_source_branch	String	Source Branch
id	Double	Combination request ID.
iid	Double	Sequence number of a combination request in the current warehouse.
merge_request_assignee_list	Array of Author objects	Reviewer
merge_status	String	Can Be Merged?
source_branch	String	Source Branch
state	String	Combination Request Status
target_branch	String	Target Branch
title	String	Title
updated_at	String	Update time.

Table 4-169 Author

Parameter	Type	Description
id	Double	User ID.
name	String	User Name
state	String	Status
username	String	User iamId

Example Requests

```
GET https://{endpoint}/v2/repositories/907199/merge_request
```

Example Responses

Status code: 200

OK

```
{
  "result": {
    "total": 1,
    "merge_requests": [ {
      "iid": 1,
      "description": "merge \"branch123\" into \"master\"\nCreate the dsadsad11 file.",
      "created_at": "2021-12-08T16:34:49.000+08:00",
      "title": "tesafrds.",
      "source_branch": "branch123",
      "updated_at": "2021-12-08T16:39:24.000+08:00",
      "id": 6670,
      "state": "opened",
      "closed_at": "2021-12-08T16:34:50.000+08:00",
      "author": {
        "name": "repo",
        "id": 1234,
        "state": "active",
        "username": "e0c068bf68c44bfab87b763ff85da277"
      },
      "target_branch": "master",
      "merge_status": "can_be_merged",
      "devcloud_source_branch": "branch123",
      "merge_request_assignee_list": [ {
        "name": "Example Account",
        "id": 3333,
        "state": "active",
        "username": "a618e34bd5704be3ae3395dfede06041"
      }, {
        "name": "Example Account",
        "id": 6747,
        "state": "active",
        "username": "42dd0277cfb048dea856e4ee352e937d"
      }
    ]
  },
  "merged": 0,
  "closed": 0,
  "opened": 1
},
"status": "success"
}
```

Status Codes

Status Code	Description
200	OK

Error Codes

See [Error Codes](#).

4.2.28 Obtaining Warehouse Combination Request Details

Function

Obtaining Warehouse Combination Request Details

URI

GET /v2/repositories/{repository_id}/merge_request/{merge_request_id}

Table 4-170 Path Parameters

Parameter	Mandatory	Type	Description
repository_id	Yes	Integer	Warehouse Short ID
merge_request_id	Yes	Integer	Combination request ID.

Request Parameters

Table 4-171 Request header parameters

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

Response Parameters

Status code: 200

Table 4-172 Response body parameters

Parameter	Type	Description
error	Error object	Response error.
result	MergeInfoResult object	Result
status	String	Status code in a response

Table 4-173 Error

Parameter	Type	Description
code	String	Error Codes
message	String	Error message

Table 4-174 MergeInfoResult

Parameter	Type	Description
approval_merge_request_approvers	Array of ApprovalMergeRequestApproversItem objects	Reviewer
author	Author object	Author
closed_at	String	Closing time
created_at	String	Creation Time
description	String	Combination Request Description
devcloud_source_branch	String	Source Branch
id	Double	Combination request ID.
iid	Double	Sequence number of a combination request in the current warehouse.
is_source_branch_exist	Boolean	Whether the source branch exists
merge_request_assignee_list	Array of MergeRequestAssigneeList objects	Reviewer

Parameter	Type	Description
merge_request_diff	MergeRequestDiff object	Combining Differences
merge_status	String	Can Be Merged?
source_branch	String	Source Branch
state	String	Combination Request Status
target_branch	String	Target Branch
title	String	Title
updated_at	String	Update time.

Table 4-175 ApprovalMergeRequestApproverItem

Parameter	Type	Description
id	Double	User ID
name	String	User Name
nick_name	String	Nickname of the user.
state	String	Status
username	String	User iamId

Table 4-176 Author

Parameter	Type	Description
id	Double	User ID.
name	String	User Name
state	String	Status
username	String	User iamId

Table 4-177 MergeRequestAssigneeListItem

Parameter	Type	Description
id	Double	User ID.
name	String	User Name
state	String	Status

Parameter	Type	Description
username	String	User iamId

Table 4-178 MergeRequestDiff

Parameter	Type	Description
base_commit_sha	String	Submitting a Base
commits_count	Double	Number of Submitted SMs
created_at	String	Created
head_commit_sha	String	Submitting the head
merge_request_id	Double	Combination request ID.
start_commit_sha	String	Start Submission
updated_at	String	Update time.

Example Requests

GET https://{endpoint}/v2/repositories/907199/merge_request/9011

Example Responses

Status code: 200

OK

```
{
  "result": {
    "iid": 1,
    "approval_merge_request_approvers": [ {
      "nick_name": "emily",
      "name": "emily",
      "id": 7513,
      "state": "optional",
      "username": "0c1de3e61c80d56b1f7dc01164788784"
    }, {
      "nick_name": "lmpstest02",
      "name": "lmpstest02",
      "id": 7745,
      "state": "optional",
      "username": "2ef8c2c835424e988ff33a8339ee1cb4"
    } ],
    "description": "merge \"branch123\" into \"master\"\nCreate the dsadsad11 file.",
    "created_at": "2021-12-08T16:34:49.000+08:00",
    "title": "tesafdr.",
    "source_branch": "branch123",
    "updated_at": "2021-12-09T11:21:02.000+08:00",
  }
}
```

```
"id" : 6670,
"state" : "opened",
"closed_at" : "2021-12-08T16:34:50.000+08:00",
"author" : {
  "name" : "repo",
  "id" : 6879,
  "state" : "active",
  "username" : "e0c068bf68c44bfab87b763ff85da277"
},
"is_source_branch_exist" : true,
"target_branch" : "master",
"merge_status" : "unchecked",
"merge_request_diff" : {
  "merge_request_id" : 6670,
  "head_commit_sha" : "715fc146292edc31a588fbc74349a2b8d18dbc27",
  "updated_at" : "2021-12-09T11:20:58.000+08:00",
  "start_commit_sha" : "02f0482056d3b949699eb69683914466c84529af",
  "created_at" : "2021-12-09T11:20:58.000+08:00",
  "base_commit_sha" : "6e90282f6334433a9fe3be809a7c1eed1afbf10b",
  "commits_count" : 2
},
"devcloud_source_branch" : "branch123",
"merge_request_assignee_list" : [ {
  "name" : "Flying snow in June",
  "id" : 6743,
  "state" : "active",
  "username" : "a618e34bd5704be3ae3395dfede06041"
}, {
  "name" : "General administrator account",
  "id" : 6747,
  "state" : "active",
  "username" : "42dd0277cfb048dea856e4ee352e937d"
} ]
},
"status" : "success"
}
```

Status Codes

Status Code	Description
200	OK

Error Codes

See [Error Codes](#).

4.3 V2Project

4.3.1 Querying All Warehouses of a User

Function

Obtains all warehouse information of a user.

URI

GET /v2/projects/repositories

Table 4-179 Query Parameters

Parameter	Mandatory	Type	Description
page_index	No	Integer	Pagination index, starting from 1.
page_size	No	Integer	Specifies the number of records on each page.
search	No	String	Keyword entered for search

Request Parameters

Table 4-180 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Specifies the user token. It can be obtained by calling the IAM API (value of X-Subject-Token in the response header).

Response Parameters

Status code: 200

Table 4-181 Response body parameters

Parameter	Type	Description
error	Error object	Response error.
result	RepoListInfo V2 object	Result
status	String	Status code in a response

Table 4-182 Error

Parameter	Type	Description
code	String	Error Codes

Parameter	Type	Description
message	String	Error message

Table 4-183 RepoListInfoV2

Parameter	Type	Description
repositories	Array of RepoInfoV2 objects	Repository list
total	Integer	Total number of repositories.

Table 4-184 RepoInfoV2

Parameter	Type	Description
created_at	String	Created
creator_name	String	User name of the creator. If the user is a tenant, the user name is the same as the tenant name.
domain_name	String	Tenant name of the creator.
group_name	String	Repository group name (Example: git@repo.alpha.devcloud.inhuawei.com:Demo00228/testword.git Group name: Demo00228)
https_url	String	URL used when HTTPS is used to clone the repository.
iam_user_uid	String	IAM user UUID of the user.
is_owner	Integer	Indicates whether the current user is the creator of the warehouse. 1: yes; 0: no
lfs_size	String	LFS capacity of the warehouse, in MB. If the capacity is greater than 1024 MB, the unit is GB.
project_is_deleted	String	Whether the project is deleted
project_uuid	String	Project UUID.
repository_id	Integer	Warehouse Primary Key ID
repository_name	String	Repository name

Parameter	Type	Description
repository_size	String	Total repository capacity = Repository LFS capacity + Git repository capacity. The unit is MB. If the value is greater than 1024 MB, the unit is GB.
repository_uuid	String	Repository UUID, which is returned by the CreateRepository API.
ssh_url	String	URL used for cloning the repository in SSH mode.
star	Boolean	Indicates whether the current user adds the warehouse to favorites.
status	Integer	Warehouse status. 0: The warehouse is successfully created. 1: The repository is being created. 2: creation failed 3: The warehouse is frozen. 4: The warehouse has been closed.
updated_at	String	Update time.
userRole	Integer	User permission in the repository. The options are as follows: 20: read-only member 30: common member 40: administrator
visibility_level	Integer	Indicates whether the repository is visible. The options are 0 (private repository) and 20 (public repository).
web_url	String	Web URL. If you access the URL, the repository details page is displayed.

Example Requests

```
GET https://{endpoint}/v2/projects/repositories
```

Example Responses

Status code: 200

OK

```
{
  "result" : {
    "total" : 29,
    "repositories" : {
      "repository_uuid" : "0bf1e7ceb5b341318cb72b42901abe54",
      "repository_id" : 468564,
      "repository_name" : "test",
      "ssh_url" : "git@repo.alpha.devcloud.inhuawei.com:test_new00001/test.git",
      "https_url" : "https://repo.alpha.devcloud.inhuawei.com/test_new00001/test.git",
      "group_name" : "test_new00001",
      "web_url" : "https://alpha.devcloud.inhuawei.com/repo/468564/home",
      "visibility_level" : 0,
      "created_at" : "2019-10-15 11:00:11",
      "updated_at" : "2019-10-15 11:00:11",
```

```
"repository_size" : "0.05M",
"lfs_size" : "0.00M",
"creator_name" : "repo",
"domain_name" : "repo",
"star" : true,
"status" : 0,
"is_owner" : 1,
"iam_user_uuid" : "a618e34bd5704be3ae3395dfede06041",
"project_uuid" : "dc19b3ca6f334ce894a1ec467689797d",
"project_is_deleted" : "false",
"userRole" : null
}
},
"status" : "success"
}
```

Status Codes

Status Code	Description
200	OK

Error Codes

See [Error Codes](#).

4.3.2 Create projects and repositories.

Function

After a project is created, the parameters for creating a warehouse group are generated in the background. The parameters include the warehouse name, template ID, whether to import project members, and home project.

URI

POST /v2/projects/repositories

Request Parameters

Table 4-185 Request header parameters

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

Table 4-186 Request body parameters

Parameter	Mandatory	Type	Description
import_members	No	Integer	Indicates whether to import project members. The options are as follows: 0->Do not import project members; 1->Import project members
project_name	Yes	String	Project name. The options are as follows: The value can contain uppercase letters, lowercase letters, digits, hyphens (-), and underscores (_), and must start with a letter.
repo_name	Yes	String	Repository name. The options are as follows: The value can contain uppercase letters, lowercase letters, digits, hyphens (-), and underscores (_), and must start with a letter. Minimum: 2 Maximum: 128
type	No	String	Project type. The options are normal and scrum.
visibility_level	No	Integer	Whether the warehouse is a public warehouse
external_project_info	No	ExternalKey Message object	Third-party service information

Table 4-187 ExternalKeyMessage

Parameter	Mandatory	Type	Description
external_key_message	No	String	Key information stored by third parties in code hosting
external_service	No	String	External Service Name

Response Parameters

Status code: 200

Table 4-188 Response body parameters

Parameter	Type	Description
error	Error object	Response error.
result	ProjectRepository object	Result
status	String	Status code in a response

Table 4-189 Error

Parameter	Type	Description
code	String	Error Codes
message	String	Error message

Table 4-190 ProjectRepository

Parameter	Type	Description
projectUuid	String	Specifies the UUID of the created project. Minimum: 32 Maximum: 32
repositoryUuid	String	UUID of the repository to be created. Minimum: 32 Maximum: 32

Example Requests

POST https://{endpoint}/v2/projects/repositories

```
{
  "project_name" : "testworld",
  "repo_name" : "demotest1",
  "import_members" : "0",
  "visibility_level" : "20",
  "type" : "normal"
}
```

Example Responses

Status code: 200

OK

```
{
  "result" : {
    "repositoryUuid" : "02637c34ba954ab5bd37ff9537fe05b4",
    "projectUuid" : "1aeda7a6070d4119b7d8b45e5641599b"
  }
}
```

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

Status Codes

Status Code	Description
200	OK

Error Codes

See [Error Codes](#).

4.3.3 Creating a Project and Forking a Repository

Function

After a warehouse is created, the following parameters are transferred to the fork warehouse: warehouse name, whether to import project members, and home project.

URI

POST /v2/projects/repositories/fork

Request Parameters

Table 4-191 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Specifies the user token. It can be obtained by calling the IAM API (value of X-Subject-Token in the response header).

Table 4-192 Request body parameters

Parameter	Mandatory	Type	Description
import_members	No	Integer	Indicates whether to import project members. The options are as follows: 0->Do not import project members; 1->Import project members

Parameter	Mandatory	Type	Description
project_name	Yes	String	Project name. The options are as follows: The value can contain uppercase letters, lowercase letters, digits, hyphens (-), and underscores (_), and must start with a letter.
repo_name	Yes	String	Repository name. The options are as follows: The value can contain uppercase letters, lowercase letters, digits, hyphens (-), and underscores (_), and must start with a letter. Minimum: 2 Maximum: 128
template_id	Yes	String	ID of the replication template.
type	No	String	Project type. The options are normal and scrum.
visibility_level	No	Integer	Whether the warehouse is a public warehouse
external_project_info	No	ExternalKey Message object	Third-party service information

Table 4-193 ExternalKeyMessage

Parameter	Mandatory	Type	Description
external_key_message	No	String	Key information stored by third parties in code hosting
external_service	No	String	External Service Name

Response Parameters

Status code: 200

Table 4-194 Response body parameters

Parameter	Type	Description
error	Error object	Response error.
result	ProjectRepository object	Result
status	String	Status code in a response

Table 4-195 Error

Parameter	Type	Description
code	String	Error Codes
message	String	Error message

Table 4-196 ProjectRepository

Parameter	Type	Description
projectUuid	String	Specifies the UUID of the created project. Minimum: 32 Maximum: 32
repositoryUuid	String	UUID of the repository to be created. Minimum: 32 Maximum: 32

Example Requests

```
POST https://{endpoint}/v2/projects/{project_uuid}/repositories/fork
```

```
{
  "project_name": "testworld",
  "repo_name": "demotest1",
  "template_id": 453489,
  "import_members": 0,
  "visibility_level": 20,
  "type": "normal"
}
```

Example Responses

Status code: 200

OK

```
{
  "result": {
    "repositoryUuid": "9a780fd55cd64333b668ba26132629d2",
  }
}
```

```
"projectUuid" : "278a03286efe45c6b788122bfc166976"  
},  
"status" : "success"  
}
```

Status Codes

Status Code	Description
200	OK

Error Codes

See [Error Codes](#).

4.3.4 Querying All Warehouses in a Project

Function

This interface is used to obtain the repository list. Fuzzy query is supported. If the project UUID is not specified, fuzzy query by repository name or project name is supported. Otherwise, fuzzy query by repository name is supported.

URI

GET /v2/projects/{project_uuid}/repositories

Table 4-197 Path Parameters

Parameter	Mandatory	Type	Description
project_uuid	Yes	String	Project UUID.

Table 4-198 Query Parameters

Parameter	Mandatory	Type	Description
page_index	No	Integer	Pagination index, starting from 1.
page_size	No	Integer	Specifies the number of records on each page.
search	No	String	Keyword entered for search

Request Parameters

Table 4-199 Request header parameters

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

Response Parameters

Status code: 200**Table 4-200** Response body parameters

Parameter	Type	Description
error	Error object	Response error.
result	RepoListInfoV2 object	Result
status	String	Status code in a response

Table 4-201 Error

Parameter	Type	Description
code	String	Error Codes
message	String	Error message

Table 4-202 RepoListInfoV2

Parameter	Type	Description
repositories	Array of RepoInfoV2 objects	Repository list
total	Integer	Total number of repositories.

Table 4-203 RepoInfoV2

Parameter	Type	Description
created_at	String	Created
creator_name	String	User name of the creator. If the user is a tenant, the user name is the same as the tenant name.
domain_name	String	Tenant name of the creator.
group_name	String	Repository group name (Example: git@repo.alpha.devcloud.inhuawei.com:Demo00228/testword.git Group name: Demo00228)
https_url	String	URL used when HTTPS is used to clone the repository.
iam_user_uid	String	IAM user UUID of the user.
is_owner	Integer	Indicates whether the current user is the creator of the warehouse. 1: yes; 0: no
lfs_size	String	LFS capacity of the warehouse, in MB. If the capacity is greater than 1024 MB, the unit is GB.
project_is_deleted	String	Whether the project is deleted
project_uuid	String	Project UUID.
repository_id	Integer	Warehouse Primary Key ID
repository_name	String	Repository name
repository_size	String	Total repository capacity = Repository LFS capacity + Git repository capacity. The unit is MB. If the value is greater than 1024 MB, the unit is GB.
repository_uid	String	Repository UUID, which is returned by the CreateRepository API.
ssh_url	String	URL used for cloning the repository in SSH mode.
star	Boolean	Indicates whether the current user adds the warehouse to favorites.
status	Integer	Warehouse status. 0: The warehouse is successfully created. 1: The repository is being created. 2: creation failed 3: The warehouse is frozen. 4: The warehouse has been closed.

Parameter	Type	Description
updated_at	String	Update time.
userRole	Integer	User permission in the repository. The options are as follows: 20: read-only member 30: common member 40: administrator
visibility_level	Integer	Indicates whether the repository is visible. The options are 0 (private repository) and 20 (public repository).
web_url	String	Web URL. If you access the URL, the repository details page is displayed.

Example Requests

```
GET https://{endpoint}/v2/projects/{project_uuid}/repositories
```

Example Responses

Status code: 200

OK

```
{
  "result": {
    "total": 14,
    "repositories": [
      {
        "repository_uuid": "d0457f74cc5b43d989ded7248f71b4e5",
        "repository_id": 468480,
        "repository_name": "CreatRepository_2qD5i",
        "ssh_url": "git@repo.alpha.devcloud.inhuawei.com:Demo00228/CreatRepository_2qD5i.git",
        "https_url": "https://repo.alpha.devcloud.inhuawei.com/Demo00228/CreatRepository_2qD5i.git",
        "group_name": "Demo00228",
        "web_url": "https://alpha.devcloud.inhuawei.com/repo/468480/home",
        "visibility_level": 0,
        "created_at": "2019-10-15 09:40:26",
        "updated_at": "2019-10-15 09:40:26",
        "repository_size": "0.06M",
        "lfs_size": "0.00M",
        "creator_name": "repo",
        "domain_name": "repo",
        "star": false,
        "status": 0,
        "is_owner": 1,
        "iam_user_uuid": "a618e34bd5704be3ae3395dfede06041",
        "project_uuid": "4838955a48e2492bbe44b31bc4c272f6",
        "project_is_deleted": "false",
        "userRole": null
      }
    ]
  },
  "status": "success"
}
```

Status Codes

Status Code	Description
200	OK

Error Codes

See [Error Codes](#).

4.3.5 Associating Branches with Work Items

Function

Associating Branches with Work Items

URI

POST /v2/projects/issues

Request Parameters

Table 4-204 Request header parameters

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

Table 4-205 Request body parameters

Parameter	Mandatory	Type	Description
branch	Yes	String	Branch Name
project_id	Yes	String	Project ID.
related_id	Yes	Array of strings	Work Item Set
repo_id	Yes	String	Specifies the repository ID.

Response Parameters

Status code: 200

Table 4-206 Response body parameters

Parameter	Type	Description
error	Error object	Response error.
status	String	Status code in a response

Table 4-207 Error

Parameter	Type	Description
code	String	Error Codes
message	String	Error message

Example Requests

```
POST https://{endpoint}/v2/projects/issues
{
  "project_id" : "a6142fa102b740988fe45cc40254d61c",
  "branch" : "unionsdk",
  "repo_id" : "2111616668",
  "related_id" : [ "5561081" ]
}
```

Example Responses

Status code: 200

OK

```
{
  "status" : "success"
}
```

Status Codes

Status Code	Description
200	OK

Error Codes

See [Error Codes](#).

4.4 RepoMember

4.4.1 Obtains all member records in a warehouse.

Function

This interface is used to obtain the warehouse member list. You can search for a member by keyword.

URI

GET /v1/repositories/{repository_uuid}/members

Table 4-208 Path Parameters

Parameter	Mandatory	Type	Description
repository_uuid	Yes	String	Repository UUID, which is returned by the CreateRepository API.

Table 4-209 Query Parameters

Parameter	Mandatory	Type	Description
page_index	No	Integer	Indicates the page number.
page_size	No	Integer	Size of Each Page
subject	No	String	Keyword entered for search

Request Parameters

Table 4-210 Request header parameters

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

Response Parameters

Status code: 200

Table 4-211 Response body parameters

Parameter	Type	Description
error	Error object	Response error.
result	RepositoryMemberList object	Result
status	String	Status code in a response

Table 4-212 Error

Parameter	Type	Description
code	String	Error Codes
message	String	Error message

Table 4-213 RepositoryMemberList

Parameter	Type	Description
total	Integer	Total Warehouse Members
users	Array of RepositoryMember objects	Warehouse Member List

Table 4-214 RepositoryMember

Parameter	Type	Description
description	String	Warehouse Member Description
domain_id	String	Tenant ID Minimum: 8 Maximum: 32
domain_name	String	Tenant Name
email	String	Email
enabled	String	Indicates whether a member is available.

Parameter	Type	Description
name	String	User Name
role	Integer	Repository user permission. The options are as follows: 30->common member, 40->administrator
user_id	String	Specifies the user ID. Minimum: 4 Maximum: 32

Example Requests

GET https://{endpoint}/v1/repositories/{repository_uuid}/members

Example Responses

Status code: 200

OK

```
{
  "result": {
    "total": 2,
    "users": [ {
      "user_id": "5836",
      "name": "repo",
      "domain_name": "repo",
      "description": null,
      "enabled": null,
      "role": 40,
      "domain_id": "159b65b41ead484d8ddff250a4731781",
      "email": null
    }, {
      "user_id": "6057",
      "name": "repo",
      "domain_name": "repo",
      "description": null,
      "enabled": null,
      "role": 30,
      "domain_id": "159b65b41ead484d8ddff250a4731781",
      "email": null
    } ]
  },
  "status": "success"
}
```

Status Codes

Status Code	Description
200	OK

Error Codes

See [Error Codes](#).

4.4.2 Add repository members

Function

Add a warehouse member.

URI

POST /v1/repositories/{repository_uuid}/members

Table 4-215 Path Parameters

Parameter	Mandatory	Type	Description
repository_uuid	Yes	String	Repository UUID, which is returned by the CreateRepository API.

Request Parameters

Table 4-216 Request header parameters

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

Table 4-217 Request body parameters

Parameter	Mandatory	Type	Description
users	No	Array of RepoMemberInfo objects	List of user information to be added.

Table 4-218 RepoMemberInfo

Parameter	Mandatory	Type	Description
domain_id	No	String	Tenant ID of a user. Minimum: 8 Maximum: 32
domain_name	No	String	Tenant name of the user
id	Yes	String	ID of the user to be added. Minimum: 4 Maximum: 32
name	Yes	String	Name of the user to be added
role	Yes	Integer	Permission of the user to be added. The options are as follows: 30->common member, 40->administrator

Response Parameters

Status code: 200

Table 4-219 Response body parameters

Parameter	Type	Description
error	Error object	Response error.
result	Array of CreateRepoMemberResult objects	Result
status	String	Response Status

Table 4-220 Error

Parameter	Type	Description
code	String	Error Codes
message	String	Error message

Table 4-221 CreateRepoMemberResult

Parameter	Type	Description
id	String	Specifies the user ID. Minimum: 32 Maximum: 32
message	String	Creating Warehouse Member Information
name	String	User Name
status	String	Creating a Warehouse Member Status

Example Requests

```
POST https://{endpoint}/v1/repositories/{repository_uuid}/members
{
  "users" : [ {
    "id" : "87b3dc2019f04ee788cb635eb26154ca",
    "role" : 30,
    "name" : "repo"
  } ]
}
```

Example Responses

Status code: 200

OK

```
{
  "result" : {
    "id" : "87b3dc2019f04ee788cb635eb26154ca",
    "name" : null,
    "message" : "add user success",
    "status" : "success"
  },
  "status" : "success"
}
```

Status Codes

Status Code	Description
200	OK

Error Codes

See [Error Codes](#).

4.4.3 Sets the role of a member in the warehouse.

Function

Set the warehouse operation permission for warehouse members,

URI

PUT /v1/repositories/{repository_uuid}/members/{member_id}

Table 4-222 Path Parameters

Parameter	Mandatory	Type	Description
member_id	Yes	String	User ID.
repository_uuid	Yes	String	Repository UUID, which is returned by the CreateRepository API.

Request Parameters

Table 4-223 Request header parameters

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

Table 4-224 Request body parameters

Parameter	Mandatory	Type	Description
role	Yes	Integer	Member permission of a warehouse. The options are as follows: 20 -> read-only member 30 -> common member, 40 -> administrator Enumeration values: <ul style="list-style-type: none"> • 20 • 30 • 40

Response Parameters

Status code: 200

Table 4-225 Response body parameters

Parameter	Type	Description
error	Error object	Response error.
result	Object	Result
status	String	Response Status

Table 4-226 Error

Parameter	Type	Description
code	String	Error Codes
message	String	Error message

Example Requests

```
PUT https://{endpoint}/v1/repositories/{repository_uuid}/members/{member_id}
{
  "role" : "40"
}
```

Example Responses

Status code: 200

OK

```
{
  "result" : {},
  "status" : "success"
}
```

Status Codes

Status Code	Description
200	OK

Error Codes

See [Error Codes](#).

4.4.4 Delete repository members

Function

Delete repository members

URI

DELETE /v1/repositories/{repository_uuid}/members/{member_id}

Table 4-227 Path Parameters

Parameter	Mandatory	Type	Description
member_id	Yes	String	Warehouse Member ID
repository_uuid	Yes	String	Repository UUID, which is returned by the CreateRepository API.

Request Parameters

Table 4-228 Request header parameters

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

Response Parameters

Status code: 200

Table 4-229 Response body parameters

Parameter	Type	Description
error	Error object	Response error.
result	Object	Result
status	String	Response Status

Table 4-230 Error

Parameter	Type	Description
code	String	Error Codes
message	String	Error message

Example Requests

```
DELETE https://{endpoint}/v1/repositories/{repository_uuid}/members/{member_id}
```

Example Responses

Status code: 200

OK

```
{  
  "result" : {},  
  "status" : "success"  
}
```

Status Codes

Status Code	Description
200	OK

Error Codes

See [Error Codes](#).

4.5 Commit

4.5.1 Querying the Submission Information of a Warehouse

Function

This interface is used to obtain submission information based on the repository short ID. The list of all commits in the path can be queried based on the file path.

URI

```
GET /v2/projects/{repo_id}/repository/commits
```

Table 4-231 Path Parameters

Parameter	Mandatory	Type	Description
repo_id	Yes	Integer	Warehouse Short ID

Table 4-232 Query Parameters

Parameter	Mandatory	Type	Description
ref_name	No	String	Branch name or tag name of the repository. If this parameter is left blank, the default branch is queried.
since	No	String	Submitted on or after this date in the format of YYYY-MM-DDTHH:MM:SSZ
until	No	String	Submitted before or on this date in the format of YYYY-MM-DDTHH:MM:SSZ
path	No	String	File path.
all	No	Boolean	Indicates whether to search for each submission in the warehouse.
with_stats	No	Boolean	Whether statistics about each submission are added to the response
page	No	Integer	Page. Minimum: 1 Maximum: 2147483647
per_page	No	Integer	Specifies the number of records on each page. Minimum: 1 Maximum: 100

Request Parameters

Table 4-233 Request header parameters

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

Response Parameters

Status code: 200

Table 4-234 Response body parameters

Parameter	Type	Description
error	Error object	Response error.
result	Array of CommitInfo objects	Submission List
status	String	Response Status

Table 4-235 Error

Parameter	Type	Description
code	String	Error Codes
message	String	Error message

Table 4-236 CommitInfo

Parameter	Type	Description
id	String	Submit the corresponding SHA ID.
short_id	String	Submit the corresponding short SHA ID.
title	String	Submit Title
author_name	String	Author
author_email	String	Author Email

Parameter	Type	Description
committer_name	String	Requester
committer_email	String	Submit Author Email
created_at	String	Created
message	String	Details
parent_ids	Array of strings	Parent Submission ID
committed_date	String	Submitted
authored_date	String	Author Submission Time

Example Requests

```
GET https://{endpoint}/v2/projects/859341/repository/commits
```

Example Responses

Status code: 200

OK

```
{
  "result": [ {
    "id": "85a0a9d5c6e43bc9c3e443ac01f789e24eeef02b",
    "title": "some commit message",
    "message": "some commit message",
    "short_id": "85a0a9d5",
    "author_name": "repo",
    "author_email": null,
    "committer_name": "repo",
    "committer_email": null,
    "created_at": "2021-09-26T03:44:51.000Z",
    "parent_ids": [ "5c114bb316dff4d4a046e09a5c44c816f2433140" ],
    "committed_date": "2021-09-26T03:44:51.000Z",
    "authored_date": null
  }, {
    "id": "5c114bb316dff4d4a046e09a5c44c816f2433140",
    "title": "Updating README.md",
    "message": "Updating README.md",
    "short_id": "5c114bb3",
    "author_name": "repo",
    "author_email": null,
    "committer_name": "repo",
    "committer_email": null,
    "created_at": "2021-08-26T12:02:58.000Z",
    "parent_ids": [ "646fc72bd87eb02e1e84bf3b82179c7f6d22c2c2" ],
    "committed_date": "2021-08-26T12:02:58.000Z",
    "authored_date": null
  } ],
  "status": "success"
}
```


Status Codes

Status Code	Description
200	OK

Error Codes

See [Error Codes](#).

4.5.2 Creating a Commit

Function

Multiple files in different directories can be submitted at a time. If the directory does not exist, the system automatically creates a directory. The forcible overwrite option is supported. When the forcible overwrite flag is set to true, the conflict is ignored and the configuration is forcibly submitted.

URI

POST /v2/projects/{repo_id}/repository/commits

Table 4-237 Path Parameters

Parameter	Mandatory	Type	Description
repo_id	Yes	Integer	Warehouse Short ID

Request Parameters

Table 4-238 Request header parameters

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

Table 4-239 Request body parameters

Parameter	Mandatory	Type	Description
branch	Yes	String	Target Branch

Parameter	Mandatory	Type	Description
commit_message	Yes	String	Details
start_branch	No	String	New branch name when a branch is created.
actions	Yes	Array of CommitAction objects	Submission Processing List
author_email	No	String	Submit the author's e-mail address
author_name	No	String	Name of the author who submits the application.
stats	No	Boolean	Indicates whether to include submission statistics. The default value is true.
force	No	Boolean	Indicates whether to overwrite the target branch. When true, overwrite the target branch with a new commit based on start_branch.

Table 4-240 CommitAction

Parameter	Mandatory	Type	Description
action	Yes	String	Operations to be performed: create, delete, move, update, and chmod
file_path	Yes	String	Full path of a file. For example, lib/class.rb.
previous_path	No	String	Original full path of the file to be moved. For example, lib/class1.rb. This parameter is valid only when the move operation is performed.
content	No	String	File content. This parameter is mandatory for the create and update operations. A move operation that does not specify content retains the existing file content, and any other value of the content overwrites the file content.

Parameter	Mandatory	Type	Description
encoding	No	String	File encoding: text and base64. The default value is text.
last_commit_id	No	String	Last known submission ID. This parameter is valid only when the update, move, or delete operation is performed.
execute_filemode	No	Boolean	Enables or disables the execution mode of a file. This parameter is valid only when the chmod command is executed.

Response Parameters

Status code: 200

Table 4-241 Response body parameters

Parameter	Type	Description
error	Error object	Response error.
result	CreateCommitResponseBody object	Difference List
status	String	Status code in a response

Table 4-242 Error

Parameter	Type	Description
code	String	Error Codes
message	String	Error message

Table 4-243 CreateCommitResponseBody

Parameter	Type	Description
id	String	Submit the corresponding SHA ID.
short_id	String	Submit the corresponding short SHA ID.
title	String	Submit Title

Parameter	Type	Description
author_name	String	Author
author_email	String	Author Email
committer_name	String	Requester
committer_email	String	Submit Author Email
created_at	String	Created
message	String	Details
parent_ids	Array of strings	Parent Submission ID
committed_date	String	Submitted
authored_date	String	Author Submission Time
stats	stats object	Statistics

Table 4-244 stats

Parameter	Type	Description
additions	Integer	Number of lines added due to changes
deletions	Integer	Change the number of deleted rows.
total	Integer	Total number of changed lines.

Example Requests

POST https://{endpoint}/v2/projects/859341/repository/commits

```
{
  "branch": "master",
  "commit_message": "some commit message",
  "actions": [ {
    "action": "create",
    "file_path": "test001.txt",
    "content": "some content"
  } ]
}
```

Example Responses

Status code: 200

OK

```
{
  "result": {
    "id": "85a0a9d5c6e43bc9c3e443ac01f789e24eeef02b",
    "title": "some commit message",
    "message": "some commit message",
    "short_id": "85a0a9d5",
    "author_name": "repo",
    "author_email": null,
    "committer_name": "repo",
    "committer_email": null,
    "created_at": "2021-09-26T03:44:51.000Z",
    "parent_ids": [ "5c114bb316dff4d4a046e09a5c44c816f2433140" ],
    "committed_date": "2021-09-26T03:44:51.000Z",
    "authored_date": null,
    "stats": {
      "additions": 1,
      "deletions": 0,
      "total": 1
    }
  },
  "status": "success"
}
```

Status Codes

Status Code	Description
200	OK

Error Codes

See [Error Codes](#).

4.5.3 Querying Specific Submission Information of a Warehouse

Function

Gets a specific commit identified by a commit id or the name of a branch or tag.

URI

GET /v2/projects/{repo_id}/repository/commits/{sha}

Table 4-245 Path Parameters

Parameter	Mandatory	Type	Description
repo_id	Yes	Integer	Warehouse Short ID
sha	Yes	String	Commit ID, which is the branch name or tag name of the repository.

Table 4-246 Query Parameters

Parameter	Mandatory	Type	Description
stats	No	Boolean	Including submission statistics. The default value is true.

Request Parameters

Table 4-247 Request header parameters

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

Response Parameters

Status code: 200

Table 4-248 Response body parameters

Parameter	Type	Description
error	Error object	Response error.
result	SpecificCommitInfo object	Result
status	String	Status code in a response

Table 4-249 Error

Parameter	Type	Description
code	String	Error Codes
message	String	Error message

Table 4-250 SpecificCommitInfo

Parameter	Type	Description
id	String	Submit the corresponding SHA ID.
short_id	String	Submit the corresponding short SHA ID.
title	String	Submit Title
author_name	String	Author
author_email	String	Author Email
committer_name	String	Requester
committer_email	String	Submit Author Email
created_at	String	Created
message	String	Details
parent_ids	Array of strings	Parent Submission ID
committed_date	String	Submitted
authored_date	String	Author Submission Time
stats	stats object	Number of Changed Lines
last_pipeline	last_pipeline object	Pipeline Information

Table 4-251 stats

Parameter	Type	Description
additions	Integer	Number of lines added due to changes
deletions	Integer	Change the number of deleted rows.
total	Integer	Total number of changed lines.

Table 4-252 last_pipeline

Parameter	Type	Description
id	Integer	Pipeline ID
sha	String	Submit the corresponding SHA ID.

Parameter	Type	Description
ref	String	Branch Name
status	String	Pipeline status.
web_url	String	Pipeline URL

Example Requests

```
GET https://{endpoint}/v2/projects/859341/repository/commits/master
```

Example Responses

Status code: 200

OK

```
{
  "result": {
    "id": "85a0a9d5c6e43bc9c3e443ac01f789e24eeef02b",
    "title": "some commit message",
    "message": "some commit message",
    "short_id": "85a0a9d5",
    "author_name": "repo",
    "author_email": null,
    "committer_name": "repo",
    "committer_email": null,
    "created_at": "2021-09-26T03:44:51.000Z",
    "parent_ids": [ "5c114bb316dff4d4a046e09a5c44c816f2433140" ],
    "committed_date": "2021-09-26T03:44:51.000Z",
    "authored_date": null,
    "stats": {
      "additions": 1,
      "deletions": 0,
      "total": 1
    },
    "last_pipeline": {
      "id": 292811,
      "sha": "2202af08836f0afd762d1b151d15ac510b974bdf",
      "ref": "master",
      "status": "running",
      "web_url": "https://repo.devcloud.cn-north-208.ulanhqab.huawei.com/Pipeline_Beta_sdcs00001/202107171926/pipelines/292811"
    }
  },
  "status": "success"
}
```

Status Codes

Status Code	Description
200	OK

Error Codes

See [Error Codes](#).

4.5.4 Querying the Submission Discrepancy Information of a Warehouse

Function

This API is used to query submission difference information based on commit id.

URI

GET /v2/projects/{repo_id}/repository/commits/{sha}/diff

Table 4-253 Path Parameters

Parameter	Mandatory	Type	Description
repo_id	Yes	Integer	Warehouse Short ID
sha	Yes	String	Commit ID, which is the branch name or tag name of the repository.

Request Parameters

Table 4-254 Request header parameters

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

Response Parameters

Status code: 200

Table 4-255 Response body parameters

Parameter	Type	Description
error	Error object	Response error.
result	Array of DiffCommitInfo objects	Difference List
status	String	Status code in a response

Table 4-256 Error

Parameter	Type	Description
code	String	Error Codes
message	String	Error message

Table 4-257 DiffCommitInfo

Parameter	Type	Description
old_path	String	File Path Before Change
new_path	String	File Path After Change
a_mode	String	File Mode Before Change
b_mode	String	File Mode After Change
new_file	Boolean	Whether a file is added for this change
renamed_file	Boolean	Do you want to rename the file?
deleted_file	Boolean	Indicates whether to delete files during the change.
diff	Boolean	Difference Information

Example Requests

```
GET https://{endpoint}/v2/projects/859341/repository/commits/master/diff
```

Example Responses

Status code: 200

OK

```
{
  "result": [ {
    "diff": "@@ -1,4 +1,7 @@\n y_mr_0721\n+\n+! [CloudIDE run successfully] (./image/screenshot.PNG)\n+\n edwwd\n # css\n **hhrh**",
    "old_path": "README.md",
    "new_path": "README.md",
    "a_mode": "100644",
    "b_mode": "100644",
    "new_file": false,
    "renamed_file": false,
    "deleted_file": false
  } ],
  "status": "success"
}
```

Status Codes

Status Code	Description
200	OK

Error Codes

See [Error Codes](#).

4.6 File

4.6.1 Querying File Information of a Warehouse

Function

Obtains information about files in the repository, such as the name, size, and content. Note that the file content is encoded using Base64.

URI

GET /v2/projects/{repo_id}/repository/files

Table 4-258 Path Parameters

Parameter	Mandatory	Type	Description
repo_id	Yes	Integer	Warehouse Short ID

Table 4-259 Query Parameters

Parameter	Mandatory	Type	Description
file_path	Yes	String	Full path of a file.
ref	Yes	String	Commit ID, which is the branch name or tag name of the repository.

Request Parameters

Table 4-260 Request header parameters

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

Response Parameters

Status code: 200

Table 4-261 Response body parameters

Parameter	Type	Description
error	Error object	Response error.
result	FileContentInfo object	File Details
status	String	Status code in a response

Table 4-262 Error

Parameter	Type	Description
code	String	Error Codes
message	String	Error message

Table 4-263 FileContentInfo

Parameter	Type	Description
file_name	String	Indicates the file name.
file_path	String	File path.
size	Integer	File size
encoding	String	Document Code
content_sha256	String	SHA256-encoded file content

Parameter	Type	Description
ref	String	Branch Name
blob_id	String	blob sha
commit_id	String	Submit the corresponding SHA ID.
last_commit_id	String	SHA ID corresponding to the last submission
content	String	Base64-encoded file content

Example Requests

GET https://{endpoint}/v2/projects/859341/repository/files?README.md&ref=master

Example Responses

Status code: 200

OK

```
{
  "result": {
    "size": 9,
    "encoding": "base64",
    "ref": "master",
    "content": "NDU0NTQ1NDQ1",
    "file_name": "README.md",
    "file_path": "README.md",
    "content_sha256": "e16f4f6c41a79d120484774a625bea8a3c3dcb655aa7c7e793ac48639dc63944",
    "blob_id": "3325fa3142fd66fedc8c3a5218b417f561e0ea2f",
    "commit_id": "a772945ec2765e322cc83e0f467f394effa22cf3",
    "last_commit_id": "a772945ec2765e322cc83e0f467f394effa22cf3"
  },
  "status": "success"
}
```

Status Codes

Status Code	Description
200	OK

Error Codes

See [Error Codes](#).

4.7 User

4.7.1 HTTPS account and password verification

Function

Invoke the GitLab interface to check whether the user name and password entered when the user uploads or downloads code using HTTPS are valid.

URI

POST /v2/user/{iam_user_uuid}/validate-https-info

Table 4-264 Path Parameters

Parameter	Mandatory	Type	Description
iam_user_uuid	Yes	String	User iam_user_uuid

Request Parameters

Table 4-265 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Specifies the user token. It can be obtained by calling the IAM API (value of X-Subject-Token in the response header).

Table 4-266 Request body parameters

Parameter	Mandatory	Type	Description
pwd	Yes	String	HTTPS password

Response Parameters

Status code: 200

Table 4-267 Response body parameters

Parameter	Type	Description
error	Error object	Response error.
result	String	Result

Parameter	Type	Description
status	String	Status code in a response

Table 4-268 Error

Parameter	Type	Description
code	String	Error Codes
message	String	Error message

Example Requests

```
POST https://{endpoint}/v2/user/{iam_user_uuid}/validate-https-info
{
  "pwd" : "repo@123"
}
```

Example Responses

Status code: 200

OK

```
{
  "result" : "verificationPassed",
  "status" : "success"
}
```

Status Codes

Status Code	Description
200	OK

Error Codes

See [Error Codes](#).

4.8 WebHook

4.8.1 Querying the Webhook of a Specified Warehouse

Function

Get warehouse webhook

URI

GET /v1/repositories/{group_name}/{repository_name}/hooks

Table 4-269 Path Parameters

Parameter	Mandatory	Type	Description
group_name	Yes	String	Group Name (Example: git@repo.alpha.devcloud.inhuatech.com:Demo00228/testword.git Group Name: Demo00228)
repository_name	Yes	String	Repository name

Table 4-270 Query Parameters

Parameter	Mandatory	Type	Description
hook_id	No	String	hook id

Request Parameters

Table 4-271 Request header parameters

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

Response Parameters

Status code: 200

Table 4-272 Response body parameters

Parameter	Type	Description
error	Error object	Response error.
result	RepoListHook object	Result

Parameter	Type	Description
status	String	Status code in a response

Table 4-273 Error

Parameter	Type	Description
code	String	Error Codes
message	String	Error message

Table 4-274 RepoListHook

Parameter	Type	Description
hooks	Array of RepoHook objects	Hook List

Table 4-275 RepoHook

Parameter	Type	Description
build_events	Boolean	Indicates whether to trigger the build_events event.
created_at	String	Time when warehouse statistics are created.
enable_ssl_verification	Boolean	Indicates whether to use SSL authentication.
id	Integer	hook id
issues_events	Boolean	Trigger issues_events
merge_requests_events	Boolean	Whether to trigger the merge_requests_events event
note_events	Boolean	Indicates whether to trigger the note_events event.
pipeline_events	Boolean	Whether to trigger the pipeline_events event
project_id	Integer	Specifies the repository ID.
push_events	Boolean	Whether to trigger the push_events event
repository_update_events	Boolean	Whether to trigger the repository_update_events event

Parameter	Type	Description
tag_push_events	Boolean	Whether to trigger the tag_push_events event
wiki_page_events	Boolean	Trigger wiki_page_events

Example Requests

GET https://{endpoint}/v1/repositories/{group_name}/{repository_name}/hooks

Example Responses

Status code: 200

OK

```
{
  "result": {
    "hooks": [ {
      "id": 2111600786,
      "project_id": 2111615967,
      "created_at": "2021-11-09 16:35:15",
      "enable_ssl_verification": false,
      "push_events": true,
      "tag_push_events": false,
      "repository_update_events": false,
      "merge_requests_events": false,
      "issues_events": false,
      "note_events": false,
      "pipeline_events": false,
      "wiki_page_events": false
    } ]
  },
  "status": "success"
}
```

Status Codes

Status Code	Description
200	OK

Error Codes

See [Error Codes](#).

4.8.2 Adding a Hook to a Specified Repository

Function

Submit code to automatically trigger compilation and building and add a repository hook.

URI

POST /v1/repositories/{group_name}/{repository_name}/hooks

Table 4-276 Path Parameters

Parameter	Mandatory	Type	Description
group_name	Yes	String	Group Name (Example: git@repo.alpha.devcloud.inhuatech.com:Demo00228/testword.git Group Name: Demo00228)
repository_name	Yes	String	Repository name

Request Parameters

Table 4-277 Request header parameters

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

Table 4-278 Request body parameters

Parameter	Mandatory	Type	Description
hook_url	Yes	String	Trigger URL
service	Yes	String	Event Source
token	No	String	Security Token
hook_events	Yes	Array of strings	Trigger

Response Parameters

Status code: 200

Table 4-279 Response body parameters

Parameter	Type	Description
error	Error object	Response error.
result	RepoHook object	Result
status	String	Status code in a response

Table 4-280 Error

Parameter	Type	Description
code	String	Error Codes
message	String	Error message

Table 4-281 RepoHook

Parameter	Type	Description
build_events	Boolean	Indicates whether to trigger the build_events event.
created_at	String	Time when warehouse statistics are created.
enable_ssl_verification	Boolean	Indicates whether to use SSL authentication.
id	Integer	hook id
issues_events	Boolean	Trigger issues_events
merge_requests_events	Boolean	Whether to trigger the merge_requests_events event
note_events	Boolean	Indicates whether to trigger the note_events event.
pipeline_events	Boolean	Whether to trigger the pipeline_events event
project_id	Integer	Specifies the repository ID.
push_events	Boolean	Whether to trigger the push_events event
repository_update_events	Boolean	Whether to trigger the repository_update_events event
tag_push_events	Boolean	Whether to trigger the tag_push_events event

Parameter	Type	Description
wiki_page_events	Boolean	Trigger wiki_page_events

Example Requests

```
POST https://{endpoint}/v1/repositories/{group_name}/{repository_name}/hooks
{
  "service" : 11,
  "token" : "test",
  "hook_url" : "https://alpha.devcloud.inhuawei.com/",
  "hook_events" : [ "push_events" ]
}
```

Example Responses

Status code: 200

OK

```
{
  "result" : {
    "id" : 5609,
    "project_id" : 453489,
    "created_at" : "2019-10-23 17:29:56",
    "enable_ssl_verification" : false,
    "push_events" : true,
    "tag_push_events" : false,
    "repository_update_events" : false,
    "merge_requests_events" : false,
    "issues_events" : false,
    "note_events" : false,
    "pipeline_events" : false,
    "wiki_page_events" : false,
    "build_events" : null
  },
  "status" : "success"
}
```

Status Codes

Status Code	Description
200	OK

Error Codes

See [Error Codes](#).

4.8.3 Deletes the hook of a specified repository.

Function

Code submission automatically triggers compilation and building, and the repository hook is deleted.

URI

DELETE /v1/repositories/{group_name}/{repository_name}/hooks/{hook_id}

Table 4-282 Path Parameters

Parameter	Mandatory	Type	Description
group_name	Yes	String	Group Name (Example: git@repo.alpha.devcloud.inhuatech.com:Demo00228/testword.git Group Name: Demo00228)
hook_id	Yes	Integer	Deletes the hook of a specified repository by ID.
repository_name	Yes	String	Repository name

Request Parameters

Table 4-283 Request header parameters

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

Response Parameters

Status code: 200

Table 4-284 Response body parameters

Parameter	Type	Description
error	Error object	Response error.

Parameter	Type	Description
result	Object	Result
status	String	Response Status

Table 4-285 Error

Parameter	Type	Description
code	String	Error Codes
message	String	Error message

Example Requests

```
null
DELETE https://{endpoint}/v1/repositories/{group_name}/{repository_name}/hooks/{hook_id}
```

Example Responses

Status code: 200

OK

```
{
  "result" : "",
  "status" : "success"
}
```

Status Codes

Status Code	Description
200	OK

Error Codes

See [Error Codes](#).

4.9 Project

4.9.1 Obtains the list of repositories that can be set to the public state in a project.

Function

Obtains the list of repositories that can be set to the public state in a project.

URI

GET /v2/projects/{project_uuid}/repositories/template-status

Table 4-286 Path Parameters

Parameter	Mandatory	Type	Description
project_uuid	Yes	String	Project ID

Table 4-287 Query Parameters

Parameter	Mandatory	Type	Description
page_no	No	Integer	Indicates the page number.
page_size	No	Integer	Number of data records on each page

Request Parameters

Table 4-288 Request header parameters

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

Response Parameters

Status code: 200

Table 4-289 Response body parameters

Parameter	Type	Description
error	Error object	Response error.
result	TemplateListInfo object	Result
status	String	Status code in a response

Table 4-290 Error

Parameter	Type	Description
code	String	Error Codes
message	String	Error message

Table 4-291 TemplateListInfo

Parameter	Type	Description
repos	Array of DevstarRepoInfo objects	Repository list
total_count	Integer	Total number of repositories.

Table 4-292 DevstarRepoInfo

Parameter	Type	Description
uuid	String	UUID of the repository.
repo_id	String	Warehouse ID
repo_name	String	Repository name.
ssh_url	String	SSH address of the repository
code_url	String	Repository HTTPS Address
detail_url	String	Repository Preview Address

Example Requests

```
GET https://{endpoint}/v2/projects/{project_uuid}/repositories/template-status
```

Example Responses

Status code: 200

OK

```
{
  "result": {
    "repos": [ {
      "uuid": "36f532939c4b4e2191c10f5353ccf34c",
      "repo_id": 487773,
      "repo_name": "DDDDDDDD",
      "ssh_url": "git@repo.alpha.devcloud.inhuawei.com:CHEN_TEST00001/DDDDDDDD.git",
      "code_url": "https://repo.alpha.devcloud.inhuawei.com/CHEN_TEST00001/DDDDDDDD.git",
      "detail_url": "https://alpha.devcloud.inhuawei.com/repo/487773/home"
    } ],
    "total_count": 100
  },
  "status": "success"
}
```

Status Codes

Status Code	Description
200	OK

Error Codes

See [Error Codes](#).

4.9.2 Verifying the Repository Name of a Specified Project

Function

Generally, this interface is called when a repository is created. The project UUID and repository name are transferred to determine whether the repository name is duplicate. The repository exists. result:false: The repository does not exist. result:true: The repository does not exist.

URI

GET /v1/projects/{project_uuid}/repositories/validation/{repository_name}

Table 4-293 Path Parameters

Parameter	Mandatory	Type	Description
project_uuid	Yes	String	Project UUID.
repository_name	Yes	String	Repository name

Request Parameters

Table 4-294 Request header parameters

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

Response Parameters

Status code: 200

Table 4-295 Response body parameters

Parameter	Type	Description
error	Error object	Response error.
result	Boolean	Result
status	String	Status code in a response

Table 4-296 Error

Parameter	Type	Description
code	String	Error Codes
message	String	Error message

Example Requests

```
GET https://{endpoint}/v1/projects/{project_uuid}/repositories/validation/{repository_name}
```

Example Responses

Status code: 200

OK

```
{
  "result" : true,
  "status" : "success"
}
```

Status Codes

Status Code	Description
200	OK

Error Codes

See [Error Codes](#).

4.10 V2 Warehouse Management

4.10.1 Create branches

Function

This API is used to create a branch in a specified repository based on the repository ID.

URI

POST /v2/repositories/{repository_id}/branches

Table 4-297 Path Parameters

Parameter	Mandatory	Type	Description
repository_id	Yes	String	Primary key ID of a warehouse.

Request Parameters

Table 4-298 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token.

Table 4-299 Request body parameters

Parameter	Mandatory	Type	Description
branch_name	Yes	String	Branch Name

Parameter	Mandatory	Type	Description
ref	Yes	String	Source Branch Name

Response Parameters

Status code: 200

Table 4-300 Response body parameters

Parameter	Type	Description
error	Error object	Response error.
result	AddProtectResponse object	Result
status	String	Response Status

Table 4-301 Error

Parameter	Type	Description
code	String	Error Codes
message	String	Error message

Table 4-302 AddProtectResponse

Parameter	Type	Description
name	String	Branch name.
commit	CommitRepoV2 object	Details
protected	Boolean	Whether to protect DWDM.
developers_can_push	Boolean	Allow Developer to Submit
developers_can_merge	Boolean	Allow Developers to Merge
master_can_push	Boolean	Allow Administrator to Submit
master_can_merge	Boolean	Indicates whether to allow administrators to merge.

Parameter	Type	Description
no_one_can_push	Boolean	No one is allowed to submit.
no_one_can_merge	Boolean	No one is allowed to merge.
in_an_opened_merge_request	Boolean	Indicates whether a combination request is opened.

Table 4-303 CommitRepoV2

Parameter	Type	Description
id	String	Submit the corresponding SHA ID.
short_id	String	Submit the corresponding short SHA ID.
created_at	String	Created
title	String	Submit Title
parent_ids	Array of strings	Parent Submission ID
message	String	Details
author_name	String	Author
committer_name	String	Requester
committed_date	String	Submitted

Example Requests

```
{
  "ref": "master",
  "branch_name": "test"
}
```

Example Responses

Status code: 200

OK

```
{
  "result": {
    "name": "master",
    "commit": {
      "id": "3b68902cf0eb7de3d5c8757e4a990bc3fe0af349",
      "short_id": "3b68902c",

```

```
"title" : "Creating a File",  
"created_at" : "2022-03-08T03:14:39.000Z",  
"parent_ids" : [ "08618c900a4048aae7e4cd88913d3d521eae02eb" ],  
"message" : "Creating a File",  
"author_name" : "repo",  
"committer_name" : "repo",  
"committed_date" : "2022-03-08T03:14:39.000Z"  
},  
"protected" : true,  
"developers_can_push" : false,  
"developers_can_merge" : false,  
"master_can_push" : false,  
"master_can_merge" : false,  
"no_one_can_push" : true,  
"no_one_can_merge" : true,  
"in_an_opened_merge_request" : false  
},  
"status" : "success"  
}
```

Status Codes

Status Code	Description
200	OK

Error Codes

See [Error Codes](#).

5 Application Examples

[5.1 Example 1: Querying All Repositories of a User](#)

[5.2 Example 2: Querying Details About a Specified Repository by Repository UUID](#)

[5.3 Example 3: Querying Branches of a Repository](#)

5.1 Example 1: Querying All Repositories of a User

Scenario

This section uses a CodeArts user as an example to describe how to use an API to query the code repository list.

For details on how to call APIs, see [Calling APIs](#).

Prerequisites

You have enabled CodeArts Repo on the CodeArts console.

Approach

This API is used to query the list of all repositories of the current user based on the user token.

Querying All Repositories of a User

- API information:

URI: GET v2/projects/repositories

- Request example:

GET https://codehub-ext.ap-southeast-3.myhuaweicloud.com/v2/projects/repositories

- Response example:

```
{
  "result": {
    "total": 40,
    "repositories": [
      {
        "star": false,
```



```
"status": 5,
"userRole": null,
"repository_uuid": "25c88b23d4dc400fa7b3e2a152e4d53c",
"repository_id": 522140,
"repository_name": "11111111111111",
"ssh_url": "git@codehub.devcloud.cn-north-208.ulanhqab.huawei.com:testzqxq00001/111111111111111111.git",
"https_url": "https://codehub.devcloud.cn-north-208.ulanhqab.huawei.com/
testzqxq00001/111111111111111111.git",
"group_name": "testzqxq00001",
"web_url": "https://devcloud.cn-north-208.ulanhqab.huawei.com/codehub/522140/home",
"visibility_level": 0,
"created_at": "2020-07-22 10:45:56",
"updated_at": "2020-07-22 10:45:56",
"repository_size": "0.16M",
"lfs_size": "0.00M",
"creator_name": "devcloud_codehub_l00314597_01",
"domain_name": "devcloud_codehub_l00314597_01",
"is_owner": 1,
"iam_user_uuid": "a618e34bd5704be3ae3395dfede06041",
"project_uuid": "a689e057c2304abfa35d4ff65630a90a",
"project_is_deleted": "false"
}
],
},
"status": "success"}
```

5.2 Example 2: Querying Details About a Specified Repository by Repository UUID

Scenario

This section uses a CodeArts user as an example to describe how to use an API to query details about a specified repository.

For details on how to call APIs, see [Calling APIs](#).

Prerequisites

You have enabled CodeArts Repo on the CodeArts console.

Approach

Use the API to view the list of all repositories, obtain the UUID of the repository to be viewed, and use the UUID to query details about the repository in the API. The procedure is as follows:

1. [Query all repositories of a user.](#)
2. [Query details about a repository..](#)

Step 1: Querying All Repositories of a User

- API information:
URI: GET v2/projects/repositories
- Request example:
GET https://codehub-ext.ap-southeast-3.myhuaweicloud.com/v2/projects/repositories
- Response example:
{
 "result": {

```
"total": 40,
"repositories": [
  {
    "star": false,
    "status": 5,
    "userRole": null,
    "repository_uuid": "25c88b23d4dc400fa7b3e2a152e4d53c",
    "repository_id": 522140,
    "repository_name": "11111111111111",
    "ssh_url": "git@codehub.devcloud.cn-
north-208.ulanqab.huawei.com:testzqx00001/1111111111111111.git",
    "https_url": "https://codehub.devcloud.cn-north-208.ulanqab.huawei.com/
testzqx00001/1111111111111111.git",
    "group_name": "testzqx00001",
    "web_url": "https://devcloud.cn-north-208.ulanqab.huawei.com/codehub/522140/home",
    "visibility_level": 0,
    "created_at": "2020-07-22 10:45:56",
    "updated_at": "2020-07-22 10:45:56",
    "repository_size": "0.16M",
    "lfs_size": "0.00M",
    "creator_name": "devcloud_codehub_l00314597_01",
    "domain_name": "devcloud_codehub_l00314597_01",
    "is_owner": 1,
    "iam_user_uuid": "a618e34bd5704be3ae3395dfede06041",
    "project_uuid": "a689e057c2304abfa35d4ff65630a90a",
    "project_is_deleted": "false"
  }
],
"status": "success"}
```

Step 2: Querying Details About a Repository

- API information:
URI: GET v2/repositories/{project_uuid}
- Request example:
GET https://codehub-ext.ap-southeast-3.myhuaweicloud.com/v2/repositories/73de3f67b30a46a4a7305f3be980c3b1

- Response example:

```
{
  "result": {
    "star": null,
    "status": null,
    "userRole": null,
    "repository_uuid": "25c88b23d4dc400fa7b3e2a152e4d53c",
    "repository_id": 522140,
    "repository_name": "11111111111111",
    "ssh_url": "git@codehub.devcloud.cn-
north-208.ulanqab.huawei.com:testzqx00001/1111111111111111.git",
    "https_url": "https://codehub.devcloud.cn-north-208.ulanqab.huawei.com/
testzqx00001/1111111111111111.git",
    "group_name": "testzqx00001",
    "web_url": "https://devcloud.cn-north-208.ulanqab.huawei.com/codehub/522140/home",
    "visibility_level": 0,
    "created_at": "2020-07-22 10:45:56",
    "updated_at": null,
    "repository_size": null,
    "lfs_size": null,
    "creator_name": null,
    "domain_name": null,
    "is_owner": null,
    "iam_user_uuid": null,
    "project_uuid": "a689e057c2304abfa35d4ff65630a90a",
    "project_is_deleted": "false"
  },
  "status": "success"
}
```

5.3 Example 3: Querying Branches of a Repository

Scenario

This section uses a CodeArts user as an example to describe how to use an API to query branches of a specified repository.

For details on how to call APIs, see [Calling APIs](#).

Prerequisites

You have enabled CodeArts Repo on the CodeArts console.

Approach

Query the repository list based on the user token, and then query the branch list of a specified repository based on the repository short ID. The procedure is as follows:

1. [Query all repositories of a user.](#)
2. [Query branches of a repository.](#)

Step 1: Querying All Repositories of a User

- API information:
URI: GET v2/projects/repositories
- Request example:
GET <https://codehub-ext.ap-southeast-3.myhuaweicloud.com/v2/projects/repositories>
- Response example:

```
{
  "result": {
    "total": 40,
    "repositories": [
      {
        "star": false,
        "status": 5,
        "userRole": null,
        "repository_uuid": "25c88b23d4dc400fa7b3e2a152e4d53c",
        "repository_id": 522140,
        "repository_name": "1111111111111111",
        "ssh_url": "git@codehub.devcloud.cn-north-208.ulanqab.huawei.com:testzqx00001/1111111111111111.git",
        "https_url": "https://codehub.devcloud.cn-north-208.ulanqab.huawei.com/testzqx00001/1111111111111111.git",
        "group_name": "testzqx00001",
        "web_url": "https://devcloud.cn-north-208.ulanqab.huawei.com/codehub/522140/home",
        "visibility_level": 0,
        "created_at": "2020-07-22 10:45:56",
        "updated_at": "2020-07-22 10:45:56",
        "repository_size": "0.16M",
        "lfs_size": "0.00M",
        "creator_name": "devcloud_codehub_l00314597_01",
        "domain_name": "devcloud_codehub_l00314597_01",
        "is_owner": 1,
        "iam_user_uuid": "a618e34bd5704be3ae3395dfede06041",
        "project_uuid": "a689e057c2304abfa35d4ff65630a90a",
        "project_is_deleted": "false"
      }
    ]
  }
}
```

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

Step 2: Querying Branches of a Repository

- API information:
URI: GET v1/repositories/{repository_id}/branches
- Request example:
GET https://codehub-ext.ap-southeast-3.myhuaweicloud.com/v1/repositories/522140/branches
- Response example:

```
{
  "result": {
    "total": 1,
    "branches": [
      {
        "name": "master",
        "is_protected": false
      }
    ]
  },
  "status": "success"
}
```

6 Appendix

[6.1 Status Codes](#)

[6.2 Error Codes](#)

[6.3 Obtaining a Project ID](#)

[6.4 Obtaining an Account ID](#)

6.1 Status Codes

Table 6-1 Status Codes

Status Codes	Message	Description
100	Continue	The client should continue with its request. This interim response is used to inform the client that the initial part of the request has been received and has not yet been rejected by the server.
101	Switching Protocols	The protocol should be switched only when it is advantageous to do so. For example, the current HTTP protocol is switched to a later version.
201	Created	The request has been fulfilled and resulted in a new resource being created.
202	Accepted	The request has been accepted for processing, but the processing has not been completed.
203	Non-Authoritative Information	The server successfully processed the request, but is returning information that may be from another source.

Status Codes	Message	Description
204	NoContent	The server has successfully processed the request, but does not return any response. The status code is returned in response to an HTTP OPTIONS request.
205	Reset Content	The server has fulfilled the request, but the requester is required to reset the content.
206	Partial Content	The server has fulfilled the partial GET request for the resource.
300	Multiple Choices	The response contains a list of resource characteristics and addresses from which a user terminal (such as a browser) can choose the most appropriate one.
301	Moved Permanently	This and all future requests have been permanently moved to the given URI indicated in this response.
302	Found	The requested resource has been temporarily moved.
303	See Other	Use GET and POST requests to view the information.
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 authentication 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.

Status Codes	Message	Description
404	NotFound	The requested resource cannot be found. The client should modify the request instead of re-initiating it.
405	MethodNotAllowed	The method specified in the request is not allowed 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 exists, 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 refused to process the request because the request does not specify the length of its content.
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 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.

Status Codes	Message	Description
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.
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 repeat requests after the time specified in the Retry-After header of the response expires.
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

If an error occurs in API calling and no result is returned, you can locate the cause based on the error code of each API. The returned message body contains the specific error code and error information.

Error Response Body Format

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": {  
    "code": "DEV.CH.10001",
```



```
"message": "Param invalid"
},
"status": "failed"
}
```

In the response body, **error_code** is an error code, and **error_msg** provides information about the error.

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](#)

Status Code	Error Code	Error Message	Description	Measure
400	CH.010001	Invalid parameter value format.	Param invalid	Check parameters.
401	CH.080401	You do not have the permission to access the resource.	No Permission	Check the repository role and permission of a user.
403	CH.080403	Access denied.	Forbidden	Check the request path.
404	CH.000404	Resource unavailable.	Not Found	Check whether the resource ID or resource name is correct.
500	CH.000001	Unknown error.	Unknown error	System error. Please wait.
504	CH.080504	Request timed out.	Gateway Timeout	Check the network.

6.3 Obtaining a Project ID

Obtaining a Project ID by Calling an API

A project ID can also be obtained by calling a specific API. For details, see [Querying Project Information Based on the Specified Criteria](#).

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

In the following example, **id** indicates a 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 your cursor over the username and choose **Basic Information** from the drop-down list.
3. On the **Basic Information** page, click **Manage** next to **Security Credentials**.
On the displayed **API Credentials** page, view project IDs in the project list.

If there are multiple projects in one region, expand **Region** and obtain subproject IDs from the **Project ID** column.

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. Point to the username and choose **My Credentials** from the drop-down list.
On the displayed **API Credentials** page, view **Account ID**.