

CodeArts Repo

API Reference

Issue 01
Date 2026-01-07



Copyright © Huawei Cloud Computing Technologies Co., Ltd. 2026. All rights reserved.

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

Trademarks and Permissions



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

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

Notice

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

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

Huawei Cloud Computing Technologies Co., Ltd.

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

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

Contents

1 Before You Start.....	1
1.1 Overview.....	1
1.2 Calling Method.....	1
1.3 Endpoints.....	2
1.4 Concepts.....	2
2 Calling APIs.....	5
2.1 Making an API Request.....	5
2.2 Authentication.....	7
2.3 Response.....	9
3 APIs.....	11
3.1 SSHKey.....	11
3.1.1 Obtaining an SSH Key List.....	11
3.1.2 Adding an SSH Key.....	13
3.1.3 Checking Whether a Private Key Can Be Used to Pull Code.....	16
3.2 Repository.....	17
3.2.1 Obtaining Files in a Branch Directory.....	18
3.2.2 Obtaining Repository Statistics.....	21
3.2.3 Obtaining Image Files of a Specific Branch in a Repository.....	23
3.2.4 Modifying the Status of a Repository Referenced by a Pipeline.....	24
3.2.5 Checking Whether a User Has the Repository Administrator Permission.....	26
3.2.6 Setting a Repository to Public or Private.....	28
3.2.7 Obtaining the Content of a Specified File in a Specified Branch of a Repository.....	30
3.2.8 Obtaining a Public Template List.....	32
3.2.9 Creating a Repository.....	36
3.2.10 Obtaining a Repository Short ID Based on the Repository or Repository Group Name to Combine the URL of the Commit Details Page Corresponding to the Commit ID.....	38
3.2.11 Querying a Commit of a Branch by Repository or Repository Group Name.....	40
3.2.12 Querying Branches of a Specified Repository.....	43
3.2.13 Querying a Commit of a Repository Branch by the Repository ID.....	46
3.2.14 Adding a Deploy Key.....	49
3.2.15 Deleting a Deploy Key.....	51
3.2.16 Obtaining the Last Commit Statistics of a Repository.....	53
3.2.17 Repository Statistics.....	57

3.2.18 Deleting a Repository.....	59
3.2.19 Viewing a Repository Creation Status.....	60
3.2.20 Querying a Tag List of a Repository.....	62
3.2.21 Adding a Tag.....	64
3.2.22 Querying Details About a Repository.....	67
3.2.23 Obtaining the number of committed code lines.....	70
3.2.24 Downloading a Repository.....	72
3.2.25 Creating a Protected Branch.....	74
3.2.26 Deleting a Protected Branch.....	78
3.2.27 Deleting Protected Branches in Batches.....	79
3.2.28 Creating a Protected Tag.....	81
3.2.29 Deleting a Protected Tag.....	84
3.2.30 Obtaining a Repository Branch List.....	85
3.2.31 Obtaining an MR List of a Repository.....	88
3.2.32 Obtaining MR Details of a Repository.....	92
3.3 Project (V2).....	99
3.3.1 Querying All Repositories of a User.....	99
3.3.2 Creating a Project and Repository.....	102
3.3.3 Creating a Project and Forking a Repository.....	106
3.3.4 Querying All Repositories in a Project.....	109
3.4 Repository Member.....	113
3.4.1 Obtaining All Member Records in a Repository.....	113
3.4.2 Adding a Repository Member.....	116
3.4.3 Setting a Role for a Member in the Repository.....	118
3.4.4 Deleting a Repository Member.....	120
3.5 Commit.....	122
3.5.1 Querying a Commit Message of a Repository.....	122
3.5.2 Creating a Commit.....	125
3.5.3 Querying a Specific Commit of a Repository.....	130
3.5.4 Querying the Commit Difference Information of a Repository.....	133
3.6 File.....	135
3.6.1 Querying a Files in a Repository.....	136
3.7 User.....	138
3.7.1 Verifying HTTPS Account and Password.....	138
3.8 Webhook.....	140
3.8.1 Querying a Webhook of a Specified Repository.....	140
3.8.2 Adding a Hook to a Specified Repository.....	143
3.8.3 Deleting the Hook of a Specified Repository.....	146
3.9 Project.....	148
3.9.1 Obtaining the List of Repositories that Can be Set to Public under a Project.....	148
3.9.2 Obtaining Information About All Repositories in a Project.....	151
3.9.3 Verifying a Repository Name of a Specified Project.....	154

3.10 Repository Management (V2).....	156
3.10.1 Creating a Branch.....	156
3.11 Review.....	159
3.11.1 Creating an MR Review.....	159
3.11.2 Replying to an MR Review.....	167
3.11.3 Obtaining Review Settings.....	173
4 Application Examples.....	177
4.1 Example 1: Querying All Repositories of a User.....	177
4.2 Example 2: Querying Details About a Specified Repository by Repository UUID.....	178
4.3 Example 3: Querying Branches of a Repository.....	179
5 Appendix.....	182
5.1 Status Codes.....	182
5.2 Error Codes.....	185
5.3 Obtaining a Project ID.....	186
5.4 Obtaining an Account ID.....	187

1 Before You Start

[Calling Method](#)

[Endpoints](#)

[Concepts](#)

1.1 Overview

Welcome to CodeArts Repo. CodeArts Repo incorporates experience in managing hundreds of billions of lines of code. It 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 deploy key, and adding repository members.

If you plan to access CodeArts Repo using an API, ensure that you are familiar with CodeArts Repo concepts.

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](#).

NOTE

CodeArts Repo API calls: Max. 600 calls for a single API per minute by all users in a region
When multiple users in a region call the same API at the same time, these users share the limit of 600 calls per minute. As a result, the actual number of calls cannot reach the upper limit.

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

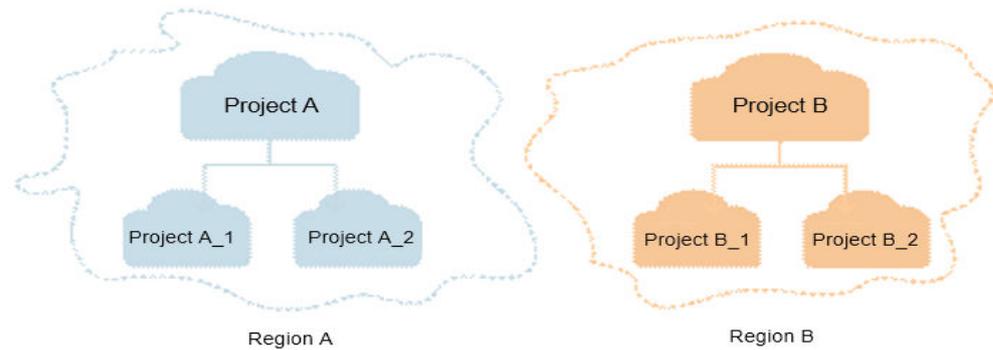
Region Name	Region	Endpoint	Protocol
-------------	--------	----------	----------

1.4 Concepts

- **Account**
An account is created upon successful registration. The account has full access permissions for all of its cloud services and resources. It can be used to reset user passwords and grant user permissions. The account is a payment entity and should not be used directly to perform routine management. For security purposes, create IAM users and grant them permissions for routine management.
- **User**
An Identity and Access Management (IAM) user is created by an account to use cloud services. Each IAM user has its own identity credentials (password and access keys).
An IAM user can view the account ID and user ID on the My Credentials page of the console. The 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
Distributed version management 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/Merge request

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 Calling APIs

Making an API Request

Authentication

Response

2.1 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.

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.region-name-1 hc.cloud/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.

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.region-name-1 hc.cloud/v3/auth/tokens
```

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) 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".

```
Content-Type: application/json
{
  "auth": {
    "identity": {
      "methods": [
        "password"
      ],
      "password": {
        "user": {
          "name": "username",
          "password": "*****",
          "domain": {
            "name": "domainname"
          }
        }
      }
    },
    "scope": {
      "project": {
        "name": "xxxxxxx"
      }
    }
  }
}
```

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.

2.2 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.region-name-1 hc.cloud/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 "API Request Signing Guide".

NOTICE

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

2.3 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 2-1 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 2-1 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 → [REDACTED]
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": "region-name-1",  
.....
```

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.

3 APIs

3.1 SSHKey

3.1.1 Obtaining an SSH Key List

Description

This API is used to obtain an SSH key list.

URI

GET /v1/users/sshkey

Request Parameters

Table 3-1 Request header parameters

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

Response Parameters

Status code: 200

Table 3-2 Response body parameters

Parameter	Type	Description
error	Error object	Response error
result	PublicKeyList object	Response result
status	String	Response status

Table 3-3 Error

Parameter	Type	Description
code	String	Error codes
message	String	Error message

Table 3-4 PublicKeyList

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

Table 3-5 PublicKey

Parameter	Type	Description
id	String	Key ID Minimum length: 1 Maximum length: 40
key	String	Key Minimum length: 0 Maximum length: 2,048
title	String	Key name

Request Examples

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


Request Parameters

Table 3-6 Request header parameters

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

Table 3-7 Request body parameters

Parameter	Mandatory	Type	Description
key	Yes	String	Key Minimum length: 0 Maximum length: 2,048
title	Yes	String	Key name

Response Parameters

Status code: 200

Table 3-8 Response body parameters

Parameter	Type	Description
error	Error object	Response error
result	PublicKey object	Response result
status	String	Response status

Table 3-9 Error

Parameter	Type	Description
code	String	Error codes
message	String	Error message

Table 3-10 PublicKey

Parameter	Type	Description
id	String	Key ID Minimum length: 1 Maximum length: 40
key	String	Key Minimum length: 0 Maximum length: 2,048
title	String	Key name

Request Examples

POST https://{endpoint}/v1/users/sshkey

```
{
  "title" : "repo@test.com",
  "key" : "ssh-rsa
AAAAB3NzaC1yc2EAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAU7xcma6HU8nj1E0i46VObWyOONl3gh142
BV repo@test.com"
}
```

Response Examples

Status code: 200

Request succeeded

```
{
  "result" : {
    "id" : "1034531",
    "title" : "repo@test.com",
    "key" : "ssh-rsa
AAAAB3NzaC1yc2EAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAU7xcma6HU8nj1E0i46VObWyOONl3gh142
BV repo@test.com"
  },
  "status" : "success"
}
```

Status Codes

Status Codes	Description
200	Request succeeded

Error Codes

See Error Codes.

3.1.3 Checking Whether a Private Key Can Be Used to Pull Code

Description

This API is used to check whether a private key can be used to pull code.

URI

POST /v1/users/sshkey/privatekey/verify

Request Parameters

Table 3-11 Request header parameters

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

Table 3-12 Request body parameters

Parameter	Mandatory	Type	Description
repository_uuid	Yes	String	Repository UUID returned by the CreateRepository API Minimum length: 32 Maximum length: 32
private_key	Yes	String	Private key Minimum length: 0 Maximum length: 2,048

Response Parameters

Status code: 200

Table 3-13 Response body parameters

Parameter	Type	Description
error	Error object	Response error

Parameter	Type	Description
result	String	Response result
status	String	Response status

Table 3-14 Error

Parameter	Type	Description
code	String	Error codes
message	String	Error message

Request Examples

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

Response Examples

Status code: 200

Request succeeded

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

Status Codes

Status Codes	Description
200	Request succeeded

Error Codes

See Error Codes.

3.2 Repository

3.2.1 Obtaining Files in a Branch Directory

Description

This API is used to obtain files in the branch directory.

URI

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

Table 3-15 Path parameters

Parameter	Mandatory	Type	Description
repository_uuid	Yes	String	Repository ID
branch_name	Yes	String	Branch name

Table 3-16 Query Parameters

Parameter	Mandatory	Type	Description
path	No	String	File path
offset	No	Integer	Offset
limit	No	Integer	Number of records

Request Parameters

Table 3-17 Request header parameters

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

Response Parameters

Status code: 200

Table 3-18 Response body parameters

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

Table 3-19 Error

Parameter	Type	Description
code	String	Error codes
message	String	Error message

Table 3-20 LogsTreeList

Parameter	Type	Description
trees	Array of LogsTree objects	File log tree
total	Integer	Total number of records

Table 3-21 LogsTree

Parameter	Type	Description
blob_id	String	Storage block ID
commit	Commit object	Commit message
file_name	String	File name
file_path	String	File path
md5	String	MD5
type	String	Storage type

Table 3-22 Commit

Parameter	Type	Description
author_email	String	Author email
author_name	String	Author
authored_date	String	Author commit time
committed_date	String	Commit time
committer_email	String	Committer email
committer_name	String	Committer
format	Object	File change details. The format is determined by the stat_format parameter.
id	String	SHA ID of a commit
message	String	Commit message
parent_ids	Array of strings	Parent commit ID

Request Examples

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

Response Examples

Status code: 200

Request succeeded

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

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

Status Codes

Status Codes	Description
200	Request succeeded

Error codes

See Error Codes.

3.2.2 Obtaining Repository Statistics

Description

This API is used to obtain repository statistics.

URI

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

Table 3-23 Path parameters

Parameter	Mandatory	Type	Description
repository_uuid	Yes	String	Repository ID

Request Parameters

Table 3-24 Request header parameters

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

Response Parameters

Status code: 200

Table 3-25 Response body parameters

Parameter	Type	Description
error	Error object	Response error
result	RepositoryStatisticsVO object	Response result
status	String	Response status

Table 3-26 Error

Parameter	Type	Description
code	String	Error codes
message	String	Error message

Table 3-27 RepositoryStatisticsVO

Parameter	Type	Description
repoName	String	Repository name
commitCount	Integer	Number of commits
repoSize	String	Repository capacity
lastCommitTime	String	Last commit time
codeLines	Integer	Lines of code
branchCount	Integer	Branches
archiveUrl	String	Code repository download address

Request Examples

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

Response Examples

Status code: 200

Request succeeded

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

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

Status Codes

Status Codes	Description
200	Request succeeded

Error codes

See Error Codes.

3.2.3 Obtaining Image Files of a Specific Branch in a Repository

Description

This API is used to obtain image files of a specific branch in a repository.

URI

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

Table 3-28 Path parameters

Parameter	Mandatory	Type	Description
repository_uuid	Yes	String	Repository ID
branch_name	Yes	String	Branch name

Table 3-29 Query Parameters

Parameter	Mandatory	Type	Description
path	Yes	String	Image path

Request Parameters

Table 3-30 Request header parameters

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

Response Parameters

None.

Request Examples

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

Response Examples

None.

Status Codes

Status Codes	Description
200	Request succeeded

Error Codes

See Error Codes.

3.2.4 Modifying the Status of a Repository Referenced by a Pipeline

Description

This API is used to modify the status of a repository referenced by a pipeline.

URI

```
PUT /v1/repositories/{repository_uuid}/pipeline
```

Table 3-31 Path parameters

Parameter	Mandatory	Type	Description
repository_uuid	Yes	String	Repository ID

Request Parameters

Table 3-32 Request header parameters

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

Response Parameters

Status code: 200

Table 3-33 Response body parameters

Parameter	Type	Description
error	Error object	Response error
result	Boolean	Response result
status	String	Response status

Table 3-34 Error

Parameter	Type	Description
code	String	Error codes
message	String	Error message

Request Examples

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

Response Examples

Status code: 200

Request succeeded

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

Status Codes

Status Codes	Description
200	Request succeeded

Error Codes

See Error Codes.

3.2.5 Checking Whether a User Has the Repository Administrator Permission

Description

This API is used to check whether a user has the repository administrator permission.

URI

GET /v1/repositories/{repository_uuid}/master

Table 3-35 Path parameters

Parameter	Mandatory	Type	Description
repository_uuid	Yes	String	Repository ID

Request Parameters

Table 3-36 Request header parameters

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

Response Parameters

Status code: 200**Table 3-37** Response body parameters

Parameter	Type	Description
error	Error object	Response error
result	Boolean	Response result
status	String	Response status

Table 3-38 Error

Parameter	Type	Description
code	String	Error codes
message	String	Error message

Request Examples

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

Response Examples

Status code: 200

Request succeeded

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

Status Codes

Status Codes	Description
200	Request succeeded

Error Codes

See Error Codes.

3.2.6 Setting a Repository to Public or Private

Description

This API is used to set a repository to public or private.

URI

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

Table 3-39 Path parameters

Parameter	Mandatory	Type	Description
repository_uuid	Yes	String	Repository ID

Request Parameters

Table 3-40 Request header parameters

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

Table 3-41 Request body parameters

Parameter	Mandatory	Type	Description
template_type	Yes	String	Template type

Parameter	Mandatory	Type	Description
code_title	No	String	Code template name
creator_name	No	String	Creator name
code_description	No	String	Code template description
languages	No	Array of strings	Template language
platform	No	Array of strings	Template platform
entertype	No	Array of strings	Template type

Response Parameters

Status code: 200

Table 3-42 Response body parameters

Parameter	Type	Description
error	Error object	Response error
result	String	Response result
status	String	Response status

Table 3-43 Error

Parameter	Type	Description
code	String	Error codes
message	String	Error message

Request Examples

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']
}
```

Response Examples

Status code: 200

Request succeeded

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

Status Codes

Status Codes	Description
200	Request succeeded

Error codes

See Error Codes.

3.2.7 Obtaining the Content of a Specified File in a Specified Branch of a Repository

Description

This API is used to obtain the content of a specified file in a specified branch of a repository.

URI

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

Table 3-44 Path parameters

Parameter	Mandatory	Type	Description
repository_uuid	Yes	String	Repository ID
branch_name	Yes	String	Branch name

Table 3-45 Query Parameters

Parameter	Mandatory	Type	Description
path	Yes	String	File path

Request Parameters

Table 3-46 Request header parameters

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

Response Parameters

Status code: 200

Table 3-47 Response body parameters

Parameter	Type	Description
error	Error object	Response error
result	FilesResponseInfo object	Response result
status	String	Response status

Table 3-48 Error

Parameter	Type	Description
code	String	Error codes
message	String	Error message

Table 3-49 FilesResponseInfo

Parameter	Type	Description
file_name	String	File name
file_path	String	File path
size	String	File size
encoding	String	Coding type
ref	String	Branch name

Parameter	Type	Description
blob_id	String	File block ID
file_type	String	File type
content	String	File content

Request Examples

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

Response Examples

Status code: 200

Request succeeded

```
{
  "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 Codes	Description
200	Request succeeded

Error Codes

See Error Codes.

3.2.8 Obtaining a Public Template List

Description

This API is used to obtain a public template list.

URI

```
GET /v2/repositories/repository-templates
```

Table 3-50 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	Templates date order
used_time_order	No	String	Template reference time
type	No	String	Template visibility
region	No	String	Region name
page_no	Yes	Integer	Page number
page_size	Yes	Integer	Number of data records on each page

Request Parameters

Table 3-51 Request header parameters

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

Response Parameters

Status code: 200

Table 3-52 Response body parameters

Parameter	Type	Description
error	Error object	Response error

Parameter	Type	Description
result	TemplateRepositoryList object	Response result
status	String	Response status

Table 3-53 Error

Parameter	Type	Description
code	String	Error codes
message	String	Error message

Table 3-54 TemplateRepositoryList

Parameter	Type	Description
projects	Array of TemplateRepository objects	Template list
total	Integer	Total number of templates

Table 3-55 TemplateRepository

Parameter	Type	Description
id	Integer	Template unique ID
name	String	Template name
template_name	String	Name of the repository associated with the template
tags	Array of strings	Template tag
description	String	Template description
brief_introduction	String	Template overview
auto_pending_pipelines	Integer	Whether to automatically create a pipeline
language	String	Template language category

Parameter	Type	Description
created_at	String	Template creation time
used_times	Integer	Template reference time
liked_times	Integer	Likes
creator_name	String	Template creator
https_url	String	Template HTTPS link

Request Examples

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

Response Examples

Status code: 200

Request succeeded

```
{
  "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.test.com/chentest00001/javaMavenDemo.git"
    } ],
    "total" : 100
  },
  "status" : "success"
}
```

Status Codes

Status Codes	Description
200	Request succeeded

Error Codes

See Error Codes.

3.2.9 Creating a Repository

Description

This API is used to create 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 3-56 Request header parameters

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

Table 3-57 Request body parameters

Parameter	Mandatory	Type	Description
import_members	No	Integer	Whether to import project members. 0 : No. 1 : Yes.
name	Yes	String	Repo name. Start with a letter and use only letters, digits, hyphens (-), and underscores (_).
project_uuid	Yes	String	Project ID. For details, see Obtaining a Project ID . Minimum length: 32 Maximum length: 32
template_id	No	String	Template ID
visibility_level	No	Integer	Repository status. 0 : private; 20 : public and read-only.
import_url	No	String	Base64-encoded HTTP address of the template repository
description	No	String	Repository description

Parameter	Mandatory	Type	Description
gitignore_id	No	String	.gitignore file generated based on the programming language
license_id	No	Integer	License ID
enable_readme	No	Integer	Whether to generate a README file
caller	No	String	Caller

Response Parameters

Status code: 200

Table 3-58 Response body parameters

Parameter	Type	Description
error	Error object	Response error
result	Repository object	Response result
status	String	Response status

Table 3-59 Error

Parameter	Type	Description
code	String	Error codes
message	String	Error message

Table 3-60 Repository

Parameter	Type	Description
repository_uuid	String	Repo UUID Minimum length: 32 Maximum length: 32

Request Examples

```
POST https://{endpoint}/v1/repositories
```

```
{
```

```
"project_uuid": "4838955a48e2492bbe44b31bc4c272f6",  
"name": "demotest1",  
"import_members": "0",  
"visibility_level": "20",  
}
```

Response Examples

Status code: 200

Request succeeded

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

Status Codes

Status Codes	Description
200	Request succeeded

Error Codes

See Error Codes.

3.2.10 Obtaining a Repository Short ID Based on the Repository or Repository Group Name to Combine the URL of the Commit Details Page Corresponding to the Commit ID

Description

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

URI

GET /v1/repositories/repoid

Table 3-61 Query parameters

Parameter	Mandatory	Type	Description
group_name	Yes	String	Repository group name (Segment following the domain name and before the repository name in the clone address. Example: git@repo.alpha.devcloud.test.com:Demo00228/testword.git . Repository group name: Demo00228 .)
repository_name	Yes	String	Repository name

Request Parameters

Table 3-62 Request header parameters

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

Response Parameters

Status code: 200

Table 3-63 Response body parameters

Parameter	Type	Description
error	Error object	Response error
result	Integer	Response result
status	String	Response status

Table 3-64 Error

Parameter	Type	Description
code	String	Error codes
message	String	Error message

Request Examples

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

Response Examples

Status code: 200

Request succeeded

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

Status Codes

Status Codes	Description
200	Request succeeded

Error Codes

See Error Codes.

3.2.11 Querying a Commit of a Branch by Repository or Repository Group Name

Description

This API is used to obtain the commit list by repository or repository group name and branch.

URI

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

Table 3-65 Path parameters

Parameter	Mandatory	Type	Description
group_name	Yes	String	Repository group name (Segment following the domain name and before the repository name in the clone address. Example: git@repo.alpha.devcloud.test.com:Demo00228/testword.git . Repository group name: Demo00228 .)
repository_name	Yes	String	Repository name

Table 3-66 Query parameters

Parameter	Mandatory	Type	Description
page_index	No	Integer	Page index
page_size	No	Integer	Number of records on each page
ref_name	Yes	String	Branch or tag name, in SHA format

Request Parameters

Table 3-67 Request header parameters

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

Response Parameters

Status code: 200

Table 3-68 Response body parameters

Parameter	Type	Description
error	Error object	Response error
result	CommitList object	Response result
status	String	Response status

Table 3-69 Error

Parameter	Type	Description
code	String	Error codes
message	String	Error message

Table 3-70 CommitList

Parameter	Type	Description
commits	Array of Commit objects	Commit list
total	Integer	Total commits

Table 3-71 Commit

Parameter	Type	Description
author_email	String	Author email
author_name	String	Author
authored_date	String	Author commit time
committed_date	String	Commit time
committer_email	String	Committer email
committer_name	String	Committer
format	Object	File change details. The format is determined by the stat_format parameter.

Parameter	Type	Description
id	String	SHA ID of a commit
message	String	Commit message
parent_ids	Array of strings	Parent commit ID

Request Examples

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

Response Examples

Status code: 200

Request succeeded

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

Status Codes

Status Codes	Description
200	Request succeeded

Error Codes

See Error Codes.

3.2.12 Querying Branches of a Specified Repository

Description

This API is used to query the branch list of a specified repository by the repository ID.

URI

GET /v1/repositories/{repository_id}/branches

Table 3-72 Path parameters

Parameter	Mandatory	Type	Description
repository_id	Yes	String	Repository primary key ID

Request Parameters

Table 3-73 Request header parameters

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

Response Parameters

Status code: 200

Table 3-74 Response body parameters

Parameter	Type	Description
error	Error object	Response error
result	BranchList object	Response result
status	String	Response status

Table 3-75 Error

Parameter	Type	Description
code	String	Error codes
message	String	Error message

Table 3-76 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 3-77 Branch

Parameter	Type	Description
is_protected	Boolean	Whether to protect the branch
name	String	Branch name

Request Examples

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

Response Examples

Status code: 200

Request succeeded

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

Status Codes

Status Codes	Description
200	Request succeeded

Error Codes

See Error Codes.

3.2.13 Querying a Commit of a Repository Branch by the Repository ID

Description

This API is used to query a commit of a repository branch by the repository ID.

URI

GET /v1/repositories/{repository_id}/commits

Table 3-78 Path parameters

Parameter	Mandatory	Type	Description
repository_id	Yes	Integer	Repository primary key ID

Table 3-79 Query Parameters

Parameter	Mandatory	Type	Description
author	No	String	Committer
begin_date	No	String	Start commit date, in the format of MMM DD, YYYY.
end_date	No	String	End commit date, in the format of MMM DD, YYYY.
message	No	String	Commit message
page_index	No	Integer	Page index Default value: 1
page_size	No	Integer	Number of records on each page Default value: 10
path	No	String	File path
ref_name	Yes	String	Branch or tag name, in SHA format
stat_format	No	String	Commit file change details (excluding diff)

Request Parameters

Table 3-80 Request header parameters

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

Response Parameters

Status code: 200

Table 3-81 Response body parameters

Parameter	Type	Description
error	Error object	Response error
result	CommitList object	Response result
status	String	Response status

Table 3-82 Error

Parameter	Type	Description
code	String	Error codes
message	String	Error message

Table 3-83 CommitList

Parameter	Type	Description
commits	Array of Commit objects	Commit list
total	Integer	Total commits

Table 3-84 Commit

Parameter	Type	Description
author_email	String	Author email
author_name	String	Author
authored_date	String	Author commit time
committed_date	String	Commit time
committer_email	String	Committer email
committer_name	String	Committer
format	Object	File change details. The format is determined by the stat_format parameter.
id	String	SHA ID of a commit
message	String	Commit message
parent_ids	Array of strings	Parent commit ID

Request Examples

```
null
```

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

Response Examples

Status code: 200

Request succeeded

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

Status Codes

Status Codes	Description
200	Request succeeded

Error Codes

See Error Codes.

3.2.14 Adding a Deploy Key

Description

This API is used to add a deploy key.

URI

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

Table 3-85 Path parameters

Parameter	Mandatory	Type	Description
repository_id	Yes	Integer	Repository primary key ID

Request Parameters

Table 3-86 Request header parameters

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

Table 3-87 Request body parameters

Parameter	Mandatory	Type	Description
application	Yes	String	Source of the SSH key for deployment

Parameter	Mandatory	Type	Description
can_push	Yes	Boolean	Whether the SSH key for deployment can push code
key	Yes	String	SSH key for deployment
key_title	Yes	String	Name of the SSH key for deployment

Response Parameters

Status code: 200

Table 3-88 Response body parameters

Parameter	Type	Description
error	Error object	Response error
result	Key object	Response result
status	String	Response status

Table 3-89 Error

Parameter	Type	Description
code	String	Error codes
message	String	Error message

Table 3-90 Key

Parameter	Type	Description
can_push	Boolean	Whether to allow push
created_at	String	Creation time of the deploy key
key	String	Deploy key
key_id	String	Deployment key ID Minimum length: 1 Maximum length: 40
key_title	String	Deploy key name

Request Examples

```
POST https://{endpoint}/v2/repositories/{repository_id}/deploy-keys
{
  "key_title": "repo@test.com",
  "key": "ssh-rsa
AAAAB3NzaC1yc2EAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAU7xcma6HU8nj1E0i46VObWyOONl3gh142
BV repo@test.com",
  "can_push": true,
  "application": ""
}
```

Response Examples

Status code: 200

Request succeeded

```
{
  "result": {
    "key_id": "1034546",
    "key_title": "repo@test.com",
    "key": "ssh-rsa
AAAAB3NzaC1yc2EAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAU7xcma6HU8nj1E0i46VObWyOONl3gh142
BV repo@test.com"
  },
  "status": "success"
}
```

Status Codes

Status Codes	Description
200	Request succeeded

Error Codes

See Error Codes.

3.2.15 Deleting a Deploy Key

Description

This API is used to delete a deploy key.

URI

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

Table 3-91 Path Parameters

Parameter	Mandatory	Type	Description
key_id	Yes	Integer	Deploy key ID

Parameter	Mandatory	Type	Description
repository_id	Yes	Integer	Repository primary key ID

Request Parameters

Table 3-92 Request header parameters

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

Response Parameters

Status code: 200

Table 3-93 Response body parameters

Parameter	Type	Description
error	Error object	Response error
result	Boolean	Response result
status	String	Response status

Table 3-94 Error

Parameter	Type	Description
code	String	Error codes
message	String	Error message

Request Examples

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

Response Examples

Status code: 200

Request succeeded

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

Status Codes

Status Codes	Description
200	Request succeeded

Error Codes

See Error Codes.

3.2.16 Obtaining the Last Commit Statistics of a Repository

Description

This API is used to obtain the last commit statistics of a repository.

URI

GET /v1/repositories/{repository_id}/statistics

Table 3-95 Path parameters

Parameter	Mandatory	Type	Description
repository_id	Yes	String	Repository primary key ID

Table 3-96 Query Parameters

Parameter	Mandatory	Type	Description
branch_name	Yes	String	Branch name

Request Parameters

Table 3-97 Request header parameters

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

Response Parameters

Status code: 200**Table 3-98** Response body parameters

Parameter	Type	Description
error	Error object	Response error
result	RepoCommitStatistics object	Response result
status	String	Response status

Table 3-99 Error

Parameter	Type	Description
code	String	Error codes
message	String	Error message

Table 3-100 RepoCommitStatistics

Parameter	Type	Description
all_branch_commits_count	Integer	Total commits
codelines	Array of RepoDailyCodeline objects	Daily committed lines in last 15 days
count	Integer	Total commits in the corresponding repository

Parameter	Type	Description
event	RepoStatisticsEvent object	Repository statistics status
statistics	Array of RepoStatistics objects	Repository statistics list
total	Integer	Total collects

Table 3-101 RepoDailyCodeline

Parameter	Type	Description
additions	Integer	Daily increased code lines
date	String	Date
deletions	Integer	Daily deleted code lines

Table 3-102 RepoStatisticsEvent

Parameter	Type	Description
branch	String	Branch name Minimum length: 1 Maximum length: 260
created_at	String	Repository statistics creation time
date	String	Repository statistics date
id	Integer	Repository statistical event ID
project_id	Integer	Repository ID
status	String	Repository statistical status: waiting , active , and finish .
updated_at	String	Repository statistics update time
user_id	Integer	User ID

Table 3-103 RepoStatistics

Parameter	Type	Description
add_lines	Integer	Add code lines

Parameter	Type	Description
branch	String	Branch name Minimum length: 1 Maximum length: 260
commit_count	Integer	Number of commits
created_at	String	Repository statistics creation time
delete_lines	Integer	Deleted code lines
id	Integer	Repository statistics record ID
project_id	Integer	Repository ID
updated_at	String	Repository statistics update time
user_name	String	Username

Request Examples

None.

Response Examples

Status code: 200

Request succeeded

```
{
  "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 Codes	Description
200	Request succeeded

Error Codes

See Error Codes.

3.2.17 Repository Statistics

Description

This API is used to query code commit record statistics of a repository by the repository short ID.

URI

POST /v1/repositories/{repository_id}/statistics

Table 3-104 Path parameters

Parameter	Mandatory	Type	Description
repository_id	Yes	Integer	Repository primary key ID

Request Parameters

Table 3-105 Request header parameters

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

Table 3-106 Request body parameters

Parameter	Mandatory	Type	Description
branch_name	Yes	String	Repository branch name

Response Parameters

Status code: 200

Table 3-107 Response body parameters

Parameter	Type	Description
error	Error object	Response error
result	RepoStatistic sLaunch object	Response result
status	String	Response status

Table 3-108 Error

Parameter	Type	Description
code	String	Error codes
message	String	Error message

Table 3-109 RepoStatisticsLaunch

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

Request Examples

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

Response Examples

Status code: 200

Request succeeded

```
{
  "result": {
    "can_statistics": false,
    "message": "operation success",
    "join_id": "2a48a9fd140dd4efabcc6e07"
  },
  "status": "success"
}
```

Status Codes

Status Codes	Description
200	Request succeeded

Error Codes

See Error Codes.

3.2.18 Deleting a Repository

Description

This API is used to delete a repository by the 32-bit UUID of the repository.

URI

DELETE /v1/repositories/{repository_uuid}

Table 3-110 Path parameters

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

Request Parameters

Table 3-111 Request header parameters

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

Response Parameters

Status code: 200

Table 3-112 Response body parameters

Parameter	Type	Description
error	Error object	Response error
result	Boolean	Response result
status	String	Response status

Table 3-113 Error

Parameter	Type	Description
code	String	Error codes
message	String	Error message

Request Examples

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

Response Examples

Status code: 200

Request succeeded

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

Status Codes

Status Codes	Description
200	Request succeeded

Error Codes

See Error Codes.

3.2.19 Viewing a Repository Creation Status

Description

This API is used to obtain the repository status.

URI

GET /v1/repositories/{repository_uuid}/status

Table 3-114 Path Parameters

Parameter	Mandatory	Type	Description
repository_uuid	Yes	String	UUID of the repository to be viewed

Request Parameters

Table 3-115 Request header parameters

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

Response Parameters

Status code: 200

Table 3-116 Response body parameters

Parameter	Type	Description
error	Error object	Response error
result	Integer	Response result
status	String	Response status

Table 3-117 Error

Parameter	Type	Description
code	String	Error codes
message	String	Error message

Request Examples

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

Response Examples

Status code: 200

Request succeeded

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

Status Codes

Status Codes	Description
200	Request succeeded

Error Codes

See Error Codes.

3.2.20 Querying a Tag List of a Repository

Description

This API is used to query a tag list of a specified repository.

URI

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

Table 3-118 Path parameters

Parameter	Mandatory	Type	Description
repository_id	Yes	String	Repository primary key ID

Request Parameters

Table 3-119 Request header parameters

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

Response Parameters

Status code: 200

Table 3-120 Response body parameters

Parameter	Type	Description
error	Error object	Response error
result	TagList object	Response result
status	String	Response status

Table 3-121 Error

Parameter	Type	Description
code	String	Error codes
message	String	Error message

Table 3-122 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 3-123 Tag

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

Request Examples

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

Response Examples

Status code: 200

Request succeeded

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

Status Codes

Status Codes	Description
200	Request succeeded

Error Codes

See Error Codes.

3.2.21 Adding a Tag

Description

This API is used to add a tag.

URI

POST /v2/repositories/{repository_id}/tags

Table 3-124 Path parameters

Parameter	Mandatory	Type	Description
repository_id	Yes	Integer	Repository primary key ID

Request Parameters

Table 3-125 Request header parameters

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

Table 3-126 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 3-127 Response body parameters

Parameter	Type	Description
error	Error object	Response error
result	AddTagsResponse object	Response result
status	String	Response status

Table 3-128 Error

Parameter	Type	Description
code	String	Error Codes
message	String	Error message

Table 3-129 AddTagsResponse

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

Table 3-130 CommitRepoV2

Parameter	Type	Description
id	String	SHA ID of a commit
short_id	String	Short SHA ID of a commit
created_at	String	Creation time
title	String	Commit title
parent_ids	Array of strings	Parent commit ID
message	String	Commit message
author_name	String	Author
committer_name	String	Committer
committed_date	String	Commit time

Request Examples

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

```
{
  "tag_name": "test20220920",
  "ref": "master",
  "message": "test20220920"
}
```

Response Examples

Status code: 200

Request succeeded

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

Status Codes

Status Codes	Description
200	Request succeeded

Error Codes

See Error Codes.

3.2.22 Querying Details About a Repository

Description

This API is used to obtain details about a repository by UUID (returned by the CreateRepository API). The returned information contains the ID, name, repo group name, and repository access URL.

URI

GET /v2/repositories/{repository_uuid}

Table 3-131 Path parameters

Parameter	Mandatory	Type	Description
repository_uuid	Yes	String	Repository UUID

Request Parameters

Table 3-132 Request header parameters

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

Response Parameters

Status code: 200**Table 3-133** Response body parameters

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

Table 3-134 Error

Parameter	Type	Description
code	String	Error codes
message	String	Error message

Table 3-135 RepoInfoV2

Parameter	Type	Description
created_at	String	Creation time
creator_name	String	Username of the creator. If the user is a tenant, the username is the same as the tenant name.
domain_name	String	Tenant name of the creator

Parameter	Type	Description
group_name	String	Repository group name (Segment following the domain name and before the repository name in the clone address. Example: git@repo.alpha.devcloud.intest.com:Demo00228/testword.git . Repository group name: Demo00228 .)
https_url	String	URL for cloning repositories using HTTPS
iam_user_uid	String	IAM user UUID
is_owner	Integer	Whether the current user is the repository creator. 1 : Yes. 0 : No
lfs_size	String	Repository LFS capacity in MB. If the capacity is greater than 1,024 MB, the unit is GB.
project_is_deleted	String	Whether the project is deleted
project_uuid	String	Project ID. For details, see Obtaining a Project ID .
repository_id	Integer	Repository 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 capacity is greater than 1,024 MB, the unit is GB.
repository_uid	String	Repository UUID returned by the CreateRepository API.
ssh_url	String	URL for cloning repositories using SSH
star	Boolean	Whether you have followed the repository
status	Integer	Repository status. 0 : Created. 1 : Creating. 2 : Failed. 3 : Frozen. 4 : Closed.
updated_at	String	Update time
userRole	Integer	Your permission in the repository. 20 : Read-only member. 30 : Common member. 40 : Administrator.
visibility_level	Integer	Visibility. 0 : Private. 20 : Public
web_url	String	Web URL. If you access it, the repository details page is displayed.

Request Examples

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

Response Examples

Status code: 200

Request succeeded

```
{
  "result": {
    "repository_uuid": "045ceabb9aea40579d5a759e84fa7f0a",
    "repository_id": 468485,
    "repository_name": "CreatRepository_MI9Af",
    "ssh_url": "git@repo.alpha.devcloud.intest.com:Demo00228/CreatRepository_MI9Af.git",
    "https_url": "https://repo.alpha.devcloud.intest.com/Demo00228/CreatRepository_MI9Af.git",
    "group_name": "Demo00228",
    "web_url": "https://alpha.devcloud.intest.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 Codes	Description
200	Request succeeded

Error Codes

See Error Codes.

3.2.23 Obtaining the number of committed code lines

Description

This API is used to obtain the number of committed code lines of a specified branch in the code repository on a specified date.

URI

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

Table 3-136 Path parameters

Parameter	Mandatory	Type	Description
repository_id	Yes	Integer	Repository short ID

Table 3-137 Query Parameters

Parameter	Mandatory	Type	Description
ref_name	Yes	String	Branch name
begin_date	Yes	String	Start commit date, in the format of MMM DD, YYYY.
end_date	Yes	String	End commit date, in the format of MMM DD, YYYY. The interval between begin_date and end_date cannot exceed 60 days.

Request Parameters

Table 3-138 Request header parameters

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

Response Parameters

Status code: 200

Table 3-139 Response body parameters

Parameter	Type	Description
error	Error object	Response error
result	CommitStatistic object	Number of added and deleted code lines
status	String	Response status

Table 3-140 Error

Parameter	Type	Description
code	String	Error codes
message	String	Error message

Table 3-141 CommitStatistic

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

Request Examples

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

Response Examples

Status code: 200

Request succeeded

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

Status Codes

Status Codes	Description
200	Request succeeded

Error Codes

See Error Codes.

3.2.24 Downloading a Repository

Description

This API is used to download a repository in the specified format.

URI

GET /v2/repositories/{repository_uuid}/archive

Table 3-142 Path parameters

Parameter	Mandatory	Type	Description
repository_uuid	Yes	String	Repository UUID

Table 3-143 Query Parameters

Parameter	Mandatory	Type	Description
sha	Yes	String	Branch name
format	Yes	String	Format of the downloaded compressed package

Request Parameters

Table 3-144 Request header parameters

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

Response Parameters

None.

Request Examples

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

Response Examples

None.

Status Codes

Status Codes	Description
200	Request succeeded

Error Codes

See Error Codes.

3.2.25 Creating a Protected Branch

Description

This API is used to create a protected branch.

URI

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

Table 3-145 Path parameters

Parameter	Mandatory	Type	Description
repository_id	Yes	Integer	Repository primary key ID
branch_name	Yes	String	Branch name

Request Parameters

Table 3-146 Request header parameters

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

Table 3-147 Request body parameters

Parameter	Mandatory	Type	Description
access_level	Yes	AddProtectAccessLevel object	Permission to create a protected branch

Table 3-148 AddProtectAccessLevel

Parameter	Mandatory	Type	Description
push_access_level	Yes	Integer	Commit permission. 0 : No one is allowed to commit. 30 : Developers and administrators can commit. 40 : Administrators can commit. Enumerated 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 commit permission. Enumerated values: <ul style="list-style-type: none">• 0• 30• 40

Response Parameters

Status code: 200

Table 3-149 Response body parameters

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

Parameter	Type	Description
status	String	Response status

Table 3-150 Error

Parameter	Type	Description
code	String	Error codes
message	String	Error message

Table 3-151 AddProtectResponse

Parameter	Type	Description
name	String	Branch name
commit	CommitRepo V2 object	Commit message
protected	Boolean	Whether to protect it
developers_can_push	Boolean	Whether a developer can commit
developers_can_merge	Boolean	Whether a developer can merge
master_can_push	Boolean	Whether an administrator can commit
master_can_merge	Boolean	Whether an administrator can merge
no_one_can_push	Boolean	No one can commit
no_one_can_merge	Boolean	No one can merge
in_an_opened_merge_request	Boolean	Whether it is in an open MR

Table 3-152 CommitRepoV2

Parameter	Type	Description
id	String	SHA ID of a commit

Parameter	Type	Description
short_id	String	Short SHA ID of a commit
created_at	String	Creation time
title	String	Commit title
parent_ids	Array of strings	Parent commit ID
message	String	Commit message
author_name	String	Author
committer_name	String	Committer
committed_date	String	Commit time

Request Examples

```
PUT https://{endpoint}/v2/repositories/{repository_id}/branch/{branch_name}/protect
```

```
{
  "access_level" : {
    "push_access_level" : 40,
    "merge_access_level" : 40
  }
}
```

Response Examples

Status code: 200

Request succeeded

```
{
  "result" : {
    "name" : "master",
    "commit" : {
      "id" : "3b68902cf0eb7de3d5c8757e4a990bc3fe0af349",
      "short_id" : "3b68902c",
      "title" : "Create a File",
      "created_at" : "2022-03-08T03:14:39.000Z",
      "parent_ids" : [ "08618c900a4048aae7e4cd88913d3d521eae02eb" ],
      "message" : " Create 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 Codes	Description
200	Request succeeded

Error Codes

See Error Codes.

3.2.26 Deleting a Protected Branch

Description

This API is used to delete a protected branch.

URI

DELETE /v2/repositories/{repository_id}/protected-branches/{branch_name}

Table 3-153 Path parameters

Parameter	Mandatory	Type	Description
repository_id	Yes	Integer	Repository primary key ID
branch_name	Yes	String	Branch name

Request Parameters

Table 3-154 Request header parameters

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

Response Parameters

Status code: 200

Table 3-155 Response body parameters

Parameter	Type	Description
error	Error object	Response error
result	String	Response result
status	String	Response status

Table 3-156 Error

Parameter	Type	Description
code	String	Error codes
message	String	Error message

Request Examples

```
DELETE https://{endpoint}/v2/repositories/{repository_id}/protected_branches/{branch_name}
```

Response Examples

Status code: 200

Request succeeded

```
{  
  "result" : "1034531",  
  "status" : "success"  
}
```

Status Codes

Status Codes	Description
200	Request succeeded

Error Codes

See Error Codes.

3.2.27 Deleting Protected Branches in Batches

Description

This API is used to delete protected branches in batches.

URI

DELETE /v2/repositories/{repository_id}/protected-branches

Table 3-157 Path parameters

Parameter	Mandatory	Type	Description
repository_id	Yes	Integer	Repository primary key ID

Request Parameters

Table 3-158 Request header parameters

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

Table 3-159 Request body parameters

Parameter	Mandatory	Type	Description
names	No	Array of strings	List of protected branch names

Response Parameters

Status code: 200

Table 3-160 Response body parameters

Parameter	Type	Description
error	Error object	Response error
result	String	Response result
status	String	Response status

Table 3-161 Error

Parameter	Type	Description
code	String	Error codes
message	String	Error message

Request Examples

```
DELETE https://{endpoint}/v2/repositories/{repository_id}/protected_branches
{
  "names": [ "test", "test123" ]
}
```

Response Examples

Status code: 200

Request succeeded

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

Status Codes

Status Codes	Description
200	Request succeeded

Error Codes

See Error Codes.

3.2.28 Creating a Protected Tag

Description

This API is used to create a protected tag.

URI

POST /v2/repositories/{repository_id}/protected-tags

Table 3-162 Path parameters

Parameter	Mandatory	Type	Description
repository_id	Yes	Integer	Repository primary key ID

Request Parameters

Table 3-163 Request header parameters

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

Table 3-164 Request body parameters

Parameter	Mandatory	Type	Description
name	No	String	Protected tag name
create_access_level	No	Integer	Permission to create a protected tag

Response Parameters

Status code: 200

Table 3-165 Response body parameters

Parameter	Type	Description
error	Error object	Response error
result	AddProtectTagResponse object	Response result
status	String	Response status

Table 3-166 Error

Parameter	Type	Description
code	String	Error codes
message	String	Error message

Table 3-167 AddProtectTagResponse

Parameter	Type	Description
name	String	Protected tag name
create_access_levels	CreateAccess Levels object	Permission to create a protected tag

Table 3-168 CreateAccessLevels

Parameter	Type	Description
access_level	Integer	Permission
access_level_description	String	Permission description

Request Examples

POST https://{endpoint}/v2/repositories/{repository_id}/protected_tags

```
{
  "name" : "test",
  "create_access_level" : 40
}
```

Response Examples

Status code: 200

Request succeeded

```
{
  "result" : {
    "name" : "test",
    "create_access_levels" : {
      "access_level" : 40,
      "access_level_description" : "Maintainers"
    }
  },
  "status" : "success"
}
```

Status Codes

Status Codes	Description
200	Request succeeded

Error codes

See Error Codes.

3.2.29 Deleting a Protected Tag

Description

This API is used to delete a protected tag.

URI

DELETE /v2/repositories/{repository_id}/protected-tags/{name}

Table 3-169 Path parameters

Parameter	Mandatory	Type	Description
repository_id	Yes	Integer	Repository primary key ID
name	Yes	String	Protected tag name

Request Parameters

Table 3-170 Request header parameters

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

Response Parameters

Status code: 200

Table 3-171 Response body parameters

Parameter	Type	Description
error	Error object	Response error
result	String	Response result
status	String	Response status

Table 3-172 Error

Parameter	Type	Description
code	String	Error codes
message	String	Error message

Request Examples

```
DELETE https://{endpoint}/v2/repositories/{repository_id}/protected_tags/{name}
```

Response Examples

Status code: 200

Request succeeded

```
{  
  "result" : "1034531",  
  "status" : "success"  
}
```

Status Codes

Status Codes	Description
200	Request succeeded

Error Codes

See Error Codes.

3.2.30 Obtaining a Repository Branch List

Description

This API is used to obtain a branch list of a repository.

URI

GET /v2/repositories/{repository_id}/branches

Table 3-173 Path parameters

Parameter	Mandatory	Type	Description
repository_id	Yes	Integer	Repository short ID

Table 3-174 Query Parameters

Parameter	Mandatory	Type	Description
page	No	String	Page number
per_page	No	String	Number of data records on each page
match	No	String	Match condition

Request Parameters

Table 3-175 Request header parameters

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

Response Parameters

Status code: 200

Table 3-176 Response body parameters

Parameter	Type	Description
error	Error object	Response error
result	BranchResponse object	Response result
status	String	Response status

Table 3-177 Error

Parameter	Type	Description
code	String	Error codes
message	String	Error message

Table 3-178 BranchResponse

Parameter	Type	Description
branches	Array of BranchesItem objects	Branch information
total	Double	Total number

Table 3-179 BranchesItem

Parameter	Type	Description
commit	CommitV2 object	Commit message
diverging_commit_counts	DivergingCommitCounts object	Commit discrepancy
name	String	Branch name

Table 3-180 CommitV2

Parameter	Type	Description
committed_date	String	Commit time
committer_name	String	Committer
id	String	Commit ID
message	String	Commit message
short_id	String	Commit short ID
title	String	Commit title

Table 3-181 DivergingCommitCounts

Parameter	Type	Description
ahead	Double	Ahead commit counts
behind	Double	Behind commit counts

Request Examples

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

Response Examples

Status code: 200

Request succeeded

```
{
  "result": {
    "total": 1,
    "branches": [ {
      "name": "master",
      "commit": {
        "id": "2912b8f2328e798f7d544272ffaebfcccb598ab",
        "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 Codes	Description
200	Request succeeded

Error Codes

See Error Codes.

3.2.31 Obtaining an MR List of a Repository

Description

This API is used to obtain an MR list of a repository.

URI

GET /v2/repositories/{repository_id}/merge_request

Table 3-182 Path parameters

Parameter	Mandatory	Type	Description
repository_id	Yes	Integer	Repository short ID

Table 3-183 Query Parameters

Parameter	Mandatory	Type	Description
state	No	String	MR status
page	No	String	Page number
per_page	No	String	Number of data records on each page
search	No	String	Match condition

Request Parameters

Table 3-184 Request header parameters

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

Response Parameters

Status code: 200

Table 3-185 Response body parameters

Parameter	Type	Description
error	Error object	Response error
result	MergeResult object	Response result

Parameter	Type	Description
status	String	Response status

Table 3-186 Error

Parameter	Type	Description
code	String	Error codes
message	String	Error message

Table 3-187 MergeResult

Parameter	Type	Description
closed	Double	Number of closed MRs
merge_requests	Array of MergeRequestsItem objects	MR details
merged	Double	Number of merged requests
opened	Double	Number of open MRs
total	Double	Total number of MRs

Table 3-188 MergeRequestsItem

Parameter	Type	Description
author	Author object	Author
closed_at	String	Close time
created_at	String	Creation time
description	String	MR description
devcloud_source_branch	String	Source branch
id	Double	MR ID
iid	Double	Sequence number of an MR in the current repository

Parameter	Type	Description
merge_request_assignee_list	Array of Author objects	Merger
merge_status	String	Whether it can be merged
source_branch	String	Source branch
state	String	MR status
target_branch	String	Target branch
title	String	Title
updated_at	String	Update time

Table 3-189 Author

Parameter	Type	Description
id	Double	User ID
name	String	Username
state	String	User status
username	String	IAM user ID
avatar_url	String	Avatar URL
web_url	String	web url

Request Examples

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

Response Examples

Status code: 200

Request succeeded

```
{
  "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": "tesaflds.",
      "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 Codes	Description
200	Request succeeded

Error Codes

See Error Codes.

3.2.32 Obtaining MR Details of a Repository

Description

This API is used to obtain MR details of a repository.

URI

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

Table 3-190 Path parameters

Parameter	Mandatory	Type	Description
repository_id	Yes	Integer	Repository short ID
merge_request_id	Yes	Integer	MR ID

Request Parameters

Table 3-191 Request header parameters

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

Response Parameters

Status code: 200**Table 3-192** Response body parameters

Parameter	Type	Description
error	Error object	Response error
result	MergeInfoResult object	Response result
status	String	Response status

Table 3-193 Error

Parameter	Type	Description
code	String	Error codes
message	String	Error message

Table 3-194 MergeInfoResult

Parameter	Type	Description
approval_merge_request_approvers	Array of ApprovalMergeRequestApproverItem objects	Approver
author	Author object	Author
closed_at	String	Close time

Parameter	Type	Description
created_at	String	Creation time
description	String	MR description
devcloud_source_branch	String	Source branch
id	Double	MR ID
iid	Double	Sequence number of an MR in the current repository
is_source_branch_exist	Boolean	Whether the source branch exists
merge_request_assignee_list	Array of MergeRequestAssigneeList objects	Merger
merge_request_diff	MergeRequestDiff object	Merge differences
merge_status	String	Whether it can be merged
source_branch	String	Source branch
state	String	MR status
target_branch	String	Target branch
title	String	Title
updated_at	String	Update time
merged_at	String	Merging time
merged_by	MergedByUser object	Merger
merge_request_related_work_items	Array of MergeRequestRelatedWorkItem objects	Associated work items

Table 3-195 ApprovalMergeRequestApproverItem

Parameter	Type	Description
id	Double	User ID
name	String	Username
nick_name	String	Alias

Parameter	Type	Description
state	String	User status
username	String	IAM user ID

Table 3-196 Author

Parameter	Type	Description
id	Double	User ID
name	String	Username
state	String	User status
username	String	IAM user ID
avatar_url	String	Avatar URL
web_url	String	web url

Table 3-197 MergeRequestAssigneeListItem

Parameter	Type	Description
id	Double	User ID
name	String	Username
state	String	User status
username	String	IAM user ID

Table 3-198 MergeRequestDiff

Parameter	Type	Description
id	Double	MR difference ID
state	String	MR status
merge_request_id	Double	MR ID
created_at	String	Creation time
updated_at	String	Update time
base_commit_sha	String	Base commit

Parameter	Type	Description
real_size	Double	Actual size
head_commit_sha	String	Head commit
start_commit_sha	String	Start commit
commits_count	Double	Number of commits
external_diff	String	External differences
external_diff_store	Double	External storage differences
stored_externally	Boolean	Whether external storage exists
added_lines	Double	Added lines
removed_lines	Double	Deleted lines

Table 3-199 MergedByUser

Parameter	Type	Description
id	Double	User ID
name	String	Username
state	String	User status
username	String	IAM user ID
avatar_url	String	Avatar URL
web_url	String	Homepage URL

Table 3-200 MergeRequestRelatedWorkItem

Parameter	Type	Description
id	Double	id
user_id	String	User ID
repo_id	Double	Repository ID
merge_request_id	Double	MR ID

Parameter	Type	Description
target_branch	String	Target branch
source_branch	String	Source branch
merge_request_title	String	MR title
merge_request_url	String	MR URL
merge_request_state	String	MR status
related_id	String	Associated work item ID
related_title	String	Associated work item title
related_url	String	Associated work item URL
result	Double	Query result
create_at	String	Creation time
update_at	String	Update time

Request Examples

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

Response Examples

Status code: 200

Request succeeded

```
{
  "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": "tesafrds.",
    "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"
} ],
"merged_at" : "2023-08-14T20:51:51.000+08:00",
"merged_by" : {
  "id" : "9124",
  "name" : "CodeHub_beta_dev",
  "username" : "c369c68f1ff84679b5a8ed904d8bff1c",
  "state" : "active",
  "avatar_url" : "",
  "web_url" : "https://devcloud.test.com/c369c68f1ff84679b5a8ed904d8bff1c"
},
"merge_request_related_work_items" : [ {
  "id" : 7343,
  "user_id" : "9124",
  "repo_id" : 2111746921,
  "merge_request_id" : 20637,
  "target_branch" : "master",
  "source_branch" : "test123",
  "merge_request_url" : "https://devcloud.test.com/codehub/2111746921/18/mergedetail?source=test123&target=master",
  "merge_request_state" : "opened",
  "related_id" : "6437435",
  "related_title" : "msm-0115-1000",
  "related_url" : "https://devcloud.test.com/projectman/scrum/e76c13e86364438d9183ca2f395371e1/task/detail/6437435",
  "result" : 1,
  "create_at" : "2023-09-18T20:29:59.000+08:00",
  "update_at" : "2023-09-18T20:29:59.000+08:00"
} ]
},
"status" : "success"
}
```

Status Codes

Status Codes	Description
200	Request succeeded

Error Codes

See Error Codes.

3.3 Project (V2)

3.3.1 Querying All Repositories of a User

Description

This API is used to obtain all repository information of a user.

URI

GET /v2/projects/repositories

Table 3-201 Query parameters

Parameter	Mandatory	Type	Description
page_index	No	Integer	Page index, starting from 1
page_size	No	Integer	Record number on each page
search	No	String	Search keyword

Request Parameters

Table 3-202 Request header parameters

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

Response Parameters

Status code: 200

Table 3-203 Response body parameters

Parameter	Type	Description
error	Error object	Response error
result	RepoListInfo V2 object	Response result
status	String	Response status

Table 3-204 Error

Parameter	Type	Description
code	String	Error codes
message	String	Error message

Table 3-205 RepoListInfoV2

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

Table 3-206 RepoInfoV2

Parameter	Type	Description
created_at	String	Creation time
creator_name	String	Username of the creator. If the user is a tenant, the username is the same as the tenant name.
domain_name	String	Tenant name of the creator
group_name	String	Repository group name (Segment following the domain name and before the repository name in the clone address. Example: git@repo.alpha.devcloud.intest.com:Demo00228/testword.git . Repository group name: Demo00228 .)
https_url	String	URL for cloning repositories using HTTPS

Parameter	Type	Description
iam_user_uuid	String	IAM user UUID
is_owner	Integer	Whether the current user is the repository creator. 1 : Yes. 0 : No
lfs_size	String	Repository LFS capacity in MB. If the capacity is greater than 1,024 MB, the unit is GB.
project_is_deleted	String	Whether the project is deleted
project_uuid	String	Project ID. For details, see Obtaining a Project ID .
repository_id	Integer	Repository 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 capacity is greater than 1,024 MB, the unit is GB.
repository_uuid	String	Repository UUID returned by the CreateRepository API.
ssh_url	String	URL for cloning repositories using SSH
star	Boolean	Whether you have followed the repository
status	Integer	Repository status. 0 : Created. 1 : Creating. 2 : Failed. 3 : Frozen. 4 : Closed.
updated_at	String	Update time
userRole	Integer	Your permission in the repository. 20 : Read-only member. 30 : Common member. 40 : Administrator.
visibility_level	Integer	Visibility. 0 : Private. 20 : Public
web_url	String	Web URL. If you access it, the repository details page is displayed.

Request Examples

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

Response Examples

Status code: 200

Request succeeded

```
{
  "result": {
    "total": 29,
    "repositories": {
      "repository_uuid": "0bf1e7ceb5b341318cb72b42901abe54",
      "repository_id": 468564,
      "repository_name": "test",
      "ssh_url": "git@repo.alpha.devcloud.intest.com:test_new00001/test.git",
      "https_url": "https://repo.alpha.devcloud.intest.com/test_new00001/test.git",
      "group_name": "test_new00001",
      "web_url": "https://alpha.devcloud.intest.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 Codes	Description
200	Request succeeded

Error Codes

See Error Codes.

3.3.2 Creating a Project and Repository

Description

This API is used to create a project first, then create a repository in the project.

URI

POST /v2/projects/repositories

Request Parameters

Table 3-207 Request header parameters

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

Table 3-208 Request body parameters

Parameter	Mandatory	Type	Description
import_members	No	Integer	Whether to import project members. 0 : No. 1 : Yes.
project_name	Yes	String	Project name. Start with a letter and use only letters, digits, hyphens (-), and underscores (_).
repo_name	Yes	String	Repository name. Start with a letter and use only letters, digits, hyphens (-), and underscores (_). Minimum length: 2 Maximum length: 128
type	No	String	Project type, scrum

Parameter	Mandatory	Type	Description
visibility_level	No	Integer	Repository visibility: *Private: Only repository members can read, write and access the repository. The value is 0 . *Public: 1. Read-only for project members: Only project members can read, view and search for repositories of a project or a repository group. The value is 10 . 2. Read-only for tenant members: Only tenant members can read, view and search for repositories of a project or a repository group. The value is 10 . 3. Read-only for all visitors: All visitors can read, view and search for repositories of a project or a repository group. The value is 20 .
external_project_info	No	ExternalKey Message object	Third-party service information

Table 3-209 ExternalKeyMessage

Parameter	Mandatory	Type	Description
external_key_message	No	String	Key information stored by third parties in CodeArts Repo
external_service	No	String	External service name

Response Parameters

Status code: 200

Table 3-210 Response body parameters

Parameter	Type	Description
error	Error object	Response error
result	ProjectRepository object	Response result

Parameter	Type	Description
status	String	Response status

Table 3-211 Error

Parameter	Type	Description
code	String	Error codes
message	String	Error message

Table 3-212 ProjectRepository

Parameter	Type	Description
projectUuid	String	Project ID. For details, see Obtaining a Project ID . Minimum length: 32 Maximum length: 32
repositoryUuid	String	Repository UUID Minimum length: 32 Maximum length: 32

Request Examples

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

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

Response Examples

Status code: 200

Request succeeded

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

Status Codes

Status Codes	Description
200	Request succeeded

Error Codes

See Error Codes.

3.3.3 Creating a Project and Forking a Repository

Description

After a repository is created, parameters transferred to the fork repository include the repository name, whether to import project members, and home project.

URI

POST /v2/projects/repositories/fork

Request Parameters

Table 3-213 Request header parameters

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

Table 3-214 Request body parameters

Parameter	Mandatory	Type	Description
import_members	No	Integer	Whether to import project members. 0 : No. 1 : Yes.
project_name	Yes	String	Project name. Start with a letter and use only letters, digits, hyphens (-), and underscores (_).

Parameter	Mandatory	Type	Description
repo_name	Yes	String	Repository name. Start with a letter and use only letters, digits, hyphens (-), and underscores (_). Minimum length: 2 Maximum length: 128
template_id	Yes	String	Template ID
type	No	String	Project type, scrum
visibility_level	No	Integer	Repository visibility: *Private: Only repository members can read, write and access the repository. The value is 0 . *Public: 1. Read-only for project members: Only project members can read, view and search for repositories of a project or a repository group. The value is 10 . 2. Read-only for tenant members: Only tenant members can read, view and search for repositories of a project or a repository group. The value is 10 . 3. Read-only for all visitors: All visitors can read, view and search for repositories of a project or a repository group. The value is 20 .
external_project_info	No	ExternalKey Message object	Third-party service information

Table 3-215 ExternalKeyMessage

Parameter	Mandatory	Type	Description
external_key_message	No	String	Key information stored by third parties in CodeArts Repo
external_service	No	String	External service name

Response Parameters

Status code: 200

Table 3-216 Response body parameters

Parameter	Type	Description
error	Error object	Response error
result	ProjectRepository object	Response result
status	String	Response status

Table 3-217 Error

Parameter	Type	Description
code	String	Error codes
message	String	Error message

Table 3-218 ProjectRepository

Parameter	Type	Description
projectUuid	String	Project ID. For details, see Obtaining a Project ID . Minimum length: 32 Maximum length: 32
repositoryUuid	String	Repository UUID Minimum length: 32 Maximum length: 32

Request Examples

```
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"
}
```

Response Examples

Status code: 200

Request succeeded

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

```
"repositoryUuid" : "9a780fd55cd64333b668ba26132629d2",
"projectUuid" : "278a03286efe45c6b788122bfc166976"
},
"status" : "success"
}
```

Status Codes

Status Codes	Description
200	Request succeeded

Error Codes

See Error Codes.

3.3.4 Querying All Repositories in a Project

Description

This API is used to obtain the repository list. Fuzzy query is supported: If the **project UUID** is not specified, fuzzy search by repository name or project name is supported. Otherwise, only fuzzy search by repository name is supported.

URI

GET /v2/projects/{project_uuid}/repositories

Table 3-219 Path parameters

Parameter	Mandatory	Type	Description
project_uuid	Yes	String	Project ID. For details, see Obtaining a Project ID .

Table 3-220 Query parameters

Parameter	Mandatory	Type	Description
page_index	No	Integer	Page index, starting from 1
page_size	No	Integer	Record number on each page
search	No	String	Search keyword

Request Parameters

Table 3-221 Request header parameters

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

Response Parameters

Status code: 200

Table 3-222 Response body parameters

Parameter	Type	Description
error	Error object	Response error
result	RepoListInfoV2 object	Response result
status	String	Response status

Table 3-223 Error

Parameter	Type	Description
code	String	Error codes
message	String	Error message

Table 3-224 RepoListInfoV2

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

Table 3-225 RepoInfoV2

Parameter	Type	Description
created_at	String	Creation time
creator_name	String	Username of the creator. If the user is a tenant, the username is the same as the tenant name.
domain_name	String	Tenant name of the creator
group_name	String	Repository group name (Segment following the domain name and before the repository name in the clone address. Example: git@repo.alpha.devcloud.intest.com:Demo00228/testword.git . Repository group name: Demo00228 .)
https_url	String	URL for cloning repositories using HTTPS
iam_user_uid	String	IAM user UUID
is_owner	Integer	Whether the current user is the repository creator. 1 : Yes. 0 : No
lfs_size	String	Repository LFS capacity in MB. If the capacity is greater than 1,024 MB, the unit is GB.
project_is_deleted	String	Whether the project is deleted
project_uuid	String	Project ID. For details, see Obtaining a Project ID .
repository_id	Integer	Repository 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 capacity is greater than 1,024 MB, the unit is GB.
repository_uid	String	Repository UUID returned by the CreateRepository API.
ssh_url	String	URL for cloning repositories using SSH
star	Boolean	Whether you have followed the repository
status	Integer	Repository status. 0 : Created. 1 : Creating. 2 : Failed. 3 : Frozen. 4 : Closed.
updated_at	String	Update time

Parameter	Type	Description
userRole	Integer	Your permission in the repository. 20 : Read-only member. 30 : Common member. 40 : Administrator.
visibility_level	Integer	Visibility. 0 : Private. 20 : Public
web_url	String	Web URL. If you access it, the repository details page is displayed.

Request Examples

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

Response Examples

Status code: 200

Request succeeded

```
{
  "result": {
    "total": 14,
    "repositories": [
      {
        "repository_uuid": "d0457f74cc5b43d989ded7248f71b4e5",
        "repository_id": 468480,
        "repository_name": "CreatRepository_2qD5i",
        "ssh_url": "git@repo.alpha.devcloud.intest.com:Demo00228/CreatRepository_2qD5i.git",
        "https_url": "https://repo.alpha.devcloud.intest.com/Demo00228/CreatRepository_2qD5i.git",
        "group_name": "Demo00228",
        "web_url": "https://alpha.devcloud.intest.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 Codes	Description
200	Request succeeded

Error codes

See Error Codes.

3.4 Repository Member

3.4.1 Obtaining All Member Records in a Repository

Description

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

URI

GET /v1/repositories/{repository_uuid}/members

Table 3-226 Path parameters

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

Table 3-227 Query Parameters

Parameter	Mandatory	Type	Description
page_index	No	Integer	Page number
page_size	No	Integer	Size of each page
subject	No	String	Search keyword

Request Parameters

Table 3-228 Request header parameters

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

Response Parameters

Status code: 200

Table 3-229 Response body parameters

Parameter	Type	Description
error	Error object	Response error
result	RepositoryMemberList object	Response result
status	String	Response status

Table 3-230 Error

Parameter	Type	Description
code	String	Error codes
message	String	Error message

Table 3-231 RepositoryMemberList

Parameter	Type	Description
total	Integer	Total number of repository members
users	Array of RepositoryMember objects	Repository member list

Table 3-232 RepositoryMember

Parameter	Type	Description
description	String	Repository member description
domain_id	String	Tenant ID Minimum length: 8 Maximum length: 32
domain_name	String	Tenant name
email	String	Email address
enabled	String	Whether a member is available

Parameter	Type	Description
name	String	Username
role	Integer	Repository user permission: 30 : Common member. 40 : Administrator.
user_id	String	User ID Minimum length: 4 Maximum length: 32

Request Examples

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

Response Examples

Status code: 200

Request succeeded

```
{
  "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 Codes	Description
200	Request succeeded

Error Codes

See Error Codes.

3.4.2 Adding a Repository Member

Description

This API is used to add a repository member.

URI

POST /v1/repositories/{repository_uuid}/members

Table 3-233 Path parameters

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

Request Parameters

Table 3-234 Request header parameters

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

Table 3-235 Request body parameters

Parameter	Mandatory	Type	Description
users	No	Array of RepoMemberInfo objects	Information list of added users

Table 3-236 RepoMemberInfo

Parameter	Mandatory	Type	Description
domain_id	No	String	Tenant ID Minimum length: 8 Maximum length: 32

Parameter	Mandatory	Type	Description
domain_name	No	String	Tenant name
id	Yes	String	ID of the added user Minimum length: 4 Maximum length: 32
name	Yes	String	Added username
role	Yes	Integer	Permission of the added user. 20 : Viewer. 30 : Common member. 40 : Administrator.

Response Parameters

Status code: 200

Table 3-237 Response body parameters

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

Table 3-238 Error

Parameter	Type	Description
code	String	Error codes
message	String	Error message

Table 3-239 CreateRepoMemberResult

Parameter	Type	Description
id	String	User ID Minimum length: 32 Maximum length: 32
message	String	Information of the added repository member

Parameter	Type	Description
name	String	Username
status	String	Status of the added repository member

Request Examples

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

```
{
  "users": [ {
    "id": "87b3dc2019f04ee788cb635eb26154ca",
    "role": 30,
    "name": "repo"
  } ]
}
```

Response Examples

Status code: 200

Request succeeded

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

Status Codes

Status Codes	Description
200	Request succeeded

Error Codes

See Error Codes.

3.4.3 Setting a Role for a Member in the Repository

Description

This API is used to assign repository operation permissions to repository members.

URI

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

Table 3-240 Path parameters

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

Request Parameters

Table 3-241 Request header parameters

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

Table 3-242 Request body parameters

Parameter	Mandatory	Type	Description
role	Yes	Integer	Repository member permission: 20 : Read-only member. 30 : Common member. 40 : Administrator. Enumerated values: <ul style="list-style-type: none">• 20• 30• 40

Response Parameters

Status code: 200

Table 3-243 Response body parameters

Parameter	Type	Description
error	Error object	Response error
result	Object	Response result

Parameter	Type	Description
status	String	Response status

Table 3-244 Error

Parameter	Type	Description
code	String	Error codes
message	String	Error message

Request Examples

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

Response Examples

Status code: 200

Request succeeded

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

Status Codes

Status Codes	Description
200	Request succeeded

Error Codes

See Error Codes.

3.4.4 Deleting a Repository Member

Description

This API is used to delete a repository member.

URI

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

Table 3-245 Path parameters

Parameter	Mandatory	Type	Description
member_id	Yes	String	Repository member ID
repository_uuid	Yes	String	Repository UUID returned by the CreateRepository API.

Request Parameters

Table 3-246 Request header parameters

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

Response Parameters

Status code: 200

Table 3-247 Response body parameters

Parameter	Type	Description
error	Error object	Response error
result	Object	Response result
status	String	Response status

Table 3-248 Error

Parameter	Type	Description
code	String	Error codes
message	String	Error message

Request Examples

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

Response Examples

Status code: 200

Request succeeded

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

Status Codes

Status Codes	Description
200	Request succeeded

Error Codes

See Error Codes.

3.5 Commit

3.5.1 Querying a Commit Message of a Repository

Description

This API is used to obtain a commit message by the repository short ID. The list of all commits in a path can be queried by the file path.

URI

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

Table 3-249 Path parameters

Parameter	Mandatory	Type	Description
repo_id	Yes	Integer	Repository short ID

Table 3-250 Query Parameters

Parameter	Mandatory	Type	Description
ref_name	No	String	Branch or tag name of a repository. If this parameter is left blank, the default branch is queried.

Parameter	Mandatory	Type	Description
since	No	String	Date after which the commit operation is performed or on the current day. The format is YYYY-MM-DDTHH:MM:SSZ.
until	No	String	Date after which the commit operation is performed or on the current day. The format is YYYY-MM-DDTHH:MM:SSZ.
path	No	String	File path
all	No	Boolean	Whether to search for each commit in a repository
with_stats	No	Boolean	Whether statistics about each commit are added to the response
page	No	Integer	Page number Minimum value: 1 Maximum value: 2,147,483,647
per_page	No	Integer	Record number on each page Minimum value: 1 Maximum value: 100

Request Parameters

Table 3-251 Request header parameters

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

Response Parameters

Status code: 200

Table 3-252 Response body parameters

Parameter	Type	Description
error	Error object	Response error
result	Array of CommitInfo objects	Commit list
status	String	Response status

Table 3-253 Error

Parameter	Type	Description
code	String	Error codes
message	String	Error message

Table 3-254 CommitInfo

Parameter	Type	Description
id	String	SHA ID of a commit
short_id	String	Short SHA ID of a commit
title	String	Commit title
author_name	String	Author
author_email	String	Author email
committer_name	String	Committer
committer_email	String	Committer email
created_at	String	Creation time
message	String	Commit message
parent_ids	Array of strings	Parent commit ID
committed_date	String	Commit time
authored_date	String	Author commit time

Request Examples

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

Response Examples

Status code: 200

Request succeeded

```
{
  "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": " Update README.md",
    "message": "Update 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 Codes	Description
200	Request succeeded

Error Codes

See Error Codes.

3.5.2 Creating a Commit

Description

This API is used to commit multiple files in different directories 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 committed.

URI

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

Table 3-255 Path parameters

Parameter	Mandatory	Type	Description
repo_id	Yes	Integer	Repository short ID

Request Parameters

Table 3-256 Request header parameters

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

Table 3-257 Request body parameters

Parameter	Mandatory	Type	Description
branch	Yes	String	Target branch
commit_message	Yes	String	Commit message
start_branch	No	String	Name of the new branch
actions	Yes	Array of CommitAction objects	Commit processing list
author_email	No	String	Committer email address
author_name	No	String	Committer name
stats	No	Boolean	Whether to include commit statistics. The default value is true .
force	No	Boolean	Whether to overwrite the target branch. When the value is true , overwrite the target branch with a new commit based on start_branch.

Table 3-258 CommitAction

Parameter	Mandatory	Type	Description
action	Yes	String	Operations to be performed: create, delete, move, update, and chmod
file_path	Yes	String	Complete path of the 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.
encoding	No	String	File encoding: text and base64 . The default value is text .
last_commit_id	No	String	Last known commit ID. This parameter is valid only when the update, move, or delete operation is performed.
execute_filemode	No	Boolean	Enable or disable the execution mode of a file. This parameter is valid only when the chmod operation is executed.

Response Parameters

Status code: 200

Table 3-259 Response body parameters

Parameter	Type	Description
error	Error object	Response error

Parameter	Type	Description
result	CreateCommitResponseBody object	Difference list
status	String	Response status

Table 3-260 Error

Parameter	Type	Description
code	String	Error codes
message	String	Error message

Table 3-261 CreateCommitResponseBody

Parameter	Type	Description
id	String	SHA ID of a commit
short_id	String	Short SHA ID of a commit
title	String	Commit title
author_name	String	Author
author_email	String	Author email
committer_name	String	Committer
committer_email	String	Committer email
created_at	String	Creation time
message	String	Commit message
parent_ids	Array of strings	Parent commit ID
committed_date	String	Commit time
authored_date	String	Author commit time
stats	stats object	Statistics

Table 3-262 stats

Parameter	Type	Description
additions	Integer	Number of lines added in changes
deletions	Integer	Number of lines deleted in changes
total	Integer	Total number of changed lines

Request Examples

```
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"
  } ]
}
```

Response Examples

Status code: 200

Request succeeded

```
{
  "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 Codes	Description
200	Request succeeded

Error Codes

See Error Codes.

3.5.3 Querying a Specific Commit of a Repository

Description

This API is used to obtain a specific commit by commit ID, branch name, or tag name.

URI

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

Table 3-263 Path parameters

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

Table 3-264 Query Parameters

Parameter	Mandatory	Type	Description
stats	No	Boolean	It includes commit statistics. The default value is true .

Request Parameters

Table 3-265 Request header parameters

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

Response Parameters

Status code: 200

Table 3-266 Response body parameters

Parameter	Type	Description
error	Error object	Response error
result	SpecificCommitInfo object	Response result
status	String	Response status

Table 3-267 Error

Parameter	Type	Description
code	String	Error codes
message	String	Error message

Table 3-268 SpecificCommitInfo

Parameter	Type	Description
id	String	SHA ID of a commit
short_id	String	Short SHA ID of a commit
title	String	Commit title
author_name	String	Author
author_email	String	Author email
committer_name	String	Committer
committer_email	String	Committer email
created_at	String	Creation time
message	String	Commit message
parent_ids	Array of strings	Parent commit ID
committed_date	String	Commit time
authored_date	String	Author commit time
stats	stats object	Number of changed lines

Parameter	Type	Description
last_pipeline	last_pipeline object	Pipeline information

Table 3-269 stats

Parameter	Type	Description
additions	Integer	Number of lines added in changes
deletions	Integer	Number of lines deleted in changes
total	Integer	Total number of changed lines

Table 3-270 last_pipeline

Parameter	Type	Description
id	Integer	Pipeline ID
sha	String	SHA ID of a commit
ref	String	Branch name
status	String	Pipeline status
web_url	String	Pipeline URL

Request Examples

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

Response Examples

Status code: 200

Request succeeded

```
{
  "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.test.area.test.com/Pipeline_Beta_sdcs00001/202107171926/pipelines/292811"
}
},
"status" : "success"
}
```

Status Codes

Status Codes	Description
200	Request succeeded

Error Codes

See Error Codes.

3.5.4 Querying the Commit Difference Information of a Repository

Description

This API is used to query commit difference information by commit ID.

URI

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

Table 3-271 Path parameters

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

Request Parameters

Table 3-272 Request header parameters

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

Response Parameters

Status code: 200

Table 3-273 Response body parameters

Parameter	Type	Description
error	Error object	Response error
result	Array of DiffCommitInfo objects	Difference list
status	String	Response status

Table 3-274 Error

Parameter	Type	Description
code	String	Error codes
message	String	Error message

Table 3-275 DiffCommitInfo

Parameter	Type	Description
old_path	String	Old file path
new_path	String	New file path
a_mode	String	Old file mode
b_mode	String	New file mode

Parameter	Type	Description
new_file	Boolean	Whether to add files during the modification
renamed_file	Boolean	Whether to rename a file during the modification
deleted_file	Boolean	Whether to delete files during the modification
diff	Boolean	Difference information

Request Examples

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

Response Examples

Status code: 200

Request succeeded

```
{
  "result": [ {
    "diff": "@@ -1,4 +1,7 @@\n y_mr_0721\n+\n+[CloudIDE Executed 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 Codes	Description
200	Request succeeded

Error Codes

See Error Codes.

3.6 File

3.6.1 Querying a Files in a Repository

Description

This API is used to obtain information about a file in a 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 3-276 Path parameters

Parameter	Mandatory	Type	Description
repo_id	Yes	Integer	Repository short ID

Table 3-277 Query Parameters

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

Request Parameters

Table 3-278 Request header parameters

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

Response Parameters

Status code: 200

Table 3-279 Response body parameters

Parameter	Type	Description
error	Error object	Response error
result	FileContentInfo object	File details
status	String	Response status

Table 3-280 Error

Parameter	Type	Description
code	String	Error codes
message	String	Error message

Table 3-281 FileContentInfo

Parameter	Type	Description
file_name	String	File name
file_path	String	File path
size	Integer	File size
encoding	String	File encoding
content_sha256	String	SHA256-encoded file content
ref	String	Branch name
blob_id	String	blob sha
commit_id	String	SHA ID of a commit
last_commit_id	String	SHA ID of the last commit
content	String	Base64-encoded file content

Request Examples

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

Response Examples

Status code: 200

Request succeeded

```
{
  "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 Codes	Description
200	Request succeeded

Error Codes

See Error Codes.

3.7 User

3.7.1 Verifying HTTPS Account and Password

Description

This API is used to check whether the username and password entered when you upload or download code using HTTPS are valid.

URI

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

Table 3-282 Path parameters

Parameter	Mandatory	Type	Description
iam_user_uuid	Yes	String	IAM user UUID

Request Parameters

Table 3-283 Request header parameters

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

Table 3-284 Request body parameters

Parameter	Mandatory	Type	Description
pwd	Yes	String	HTTPS password

Response Parameters

Status code: 200

Table 3-285 Response body parameters

Parameter	Type	Description
error	Error object	Response error
result	String	Response result
status	String	Response status

Table 3-286 Error

Parameter	Type	Description
code	String	Error codes
message	String	Error message

Request Examples

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

Response Examples

Status code: 200

Request succeeded

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

Status Codes

Status Codes	Description
200	Request succeeded

Error Codes

See Error Codes.

3.8 Webhook

3.8.1 Querying a Webhook of a Specified Repository

Description

This API is used to obtain a webhook of a specified repository.

URI

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

Table 3-287 Path parameters

Parameter	Mandatory	Type	Description
group_name	Yes	String	Repository group name (Segment following the domain name and before the repository name in the clone address. Example: git@repo.alpha.devcloud.test.com:Demo00228/testword.git . Repository group name: Demo00228 .)
repository_name	Yes	String	Repository name

Table 3-288 Query parameters

Parameter	Mandatory	Type	Description
hook_id	No	String	hook id

Request Parameters

Table 3-289 Request header parameters

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

Response Parameters

Status code: 200

Table 3-290 Response body parameters

Parameter	Type	Description
error	Error object	Response error
result	RepoListHook object	Response result
status	String	Response status

Table 3-291 Error

Parameter	Type	Description
code	String	Error codes
message	String	Error message

Table 3-292 RepoListHook

Parameter	Type	Description
hooks	Array of RepoHook objects	Hook list

Table 3-293 RepoHook

Parameter	Type	Description
build_events	Boolean	Whether to trigger build_events
created_at	String	Repository statistics creation time
enable_ssl_verification	Boolean	Whether to use SSL authentication
id	Integer	hook id
issues_events	Boolean	Whether to trigger issues_events
merge_requests_events	Boolean	Whether to trigger merge_requests_events
note_events	Boolean	Whether to trigger note_events
pipeline_events	Boolean	Whether to trigger pipeline_events
project_id	Integer	Repository ID
push_events	Boolean	Whether to trigger push_events
repository_update_events	Boolean	Whether to trigger repository_update_events
tag_push_events	Boolean	Whether to trigger tag_push_events
wiki_page_events	Boolean	Whether to trigger wiki_page_events

Request Examples

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

Response Examples

Status code: 200

Request succeeded

```
{  
  "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 Codes	Description
200	Request succeeded

Error Codes

See Error Codes.

3.8.2 Adding a Hook to a Specified Repository

Description

A code commit automatically triggers building and addition of a repository hook.

URI

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

Table 3-294 Path parameters

Parameter	Mandatory	Type	Description
group_name	Yes	String	Repository group name (Segment following the domain name and before the repository name in the clone address. Example: git@repo.alpha.devcloud.test.com:Demo00228/testword.git . Repository group name: Demo00228 .)
repository_name	Yes	String	Repository name

Request Parameters

Table 3-295 Request header parameters

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

Table 3-296 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 events

Response Parameters

Status code: 200

Table 3-297 Response body parameters

Parameter	Type	Description
error	Error object	Response error
result	RepoHook object	Response result
status	String	Response status

Table 3-298 Error

Parameter	Type	Description
code	String	Error codes

Parameter	Type	Description
message	String	Error message

Table 3-299 RepoHook

Parameter	Type	Description
build_events	Boolean	Whether to trigger build_events
created_at	String	Repository statistics creation time
enable_ssl_verification	Boolean	Whether to use SSL authentication
id	Integer	hook id
issues_events	Boolean	Whether to trigger issues_events
merge_requests_events	Boolean	Whether to trigger merge_requests_events
note_events	Boolean	Whether to trigger note_events
pipeline_events	Boolean	Whether to trigger pipeline_events
project_id	Integer	Repository ID
push_events	Boolean	Whether to trigger push_events
repository_update_events	Boolean	Whether to trigger repository_update_events
tag_push_events	Boolean	Whether to trigger tag_push_events
wiki_page_events	Boolean	Whether to trigger wiki_page_events

Request Examples

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

Response Examples

Status code: 200

Request succeeded

```
{
  "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 Codes	Description
200	Request succeeded

Error Codes

See Error Codes.

3.8.3 Deleting the Hook of a Specified Repository

Description

A code commit automatically triggers building and deletion of a repository hook.

URI

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

Table 3-300 Path parameters

Parameter	Mandatory	Type	Description
group_name	Yes	String	Repository group name (Segment following the domain name and before the repository name in the clone address. Example: git@repo.alpha.devcloud.test.com:Demo00228/testword.git . Repository group name: Demo00228 .)

Parameter	Mandatory	Type	Description
hook_id	Yes	Integer	It deletes the hook of a specified repository by ID.
repository_name	Yes	String	Repository name

Request Parameters

Table 3-301 Request header parameters

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

Response Parameters

Status code: 200

Table 3-302 Response body parameters

Parameter	Type	Description
error	Error object	Response error
result	Object	Response result
status	String	Response status

Table 3-303 Error

Parameter	Type	Description
code	String	Error codes
message	String	Error message

Request Examples

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

Response Examples

Status code: 200

Request succeeded

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

Status Codes

Status Codes	Description
200	Request succeeded

Error Codes

See Error Codes.

3.9 Project

3.9.1 Obtaining the List of Repositories that Can be Set to Public under a Project

Description

This API is used to obtain the list of repositories that can be set to public under a project.

URI

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

Table 3-304 Path parameters

Parameter	Mandatory	Type	Description
project_uuid	Yes	String	Project ID. For details, see Obtaining a Project ID .

Table 3-305 Query parameters

Parameter	Mandatory	Type	Description
page_no	No	Integer	Page number

Parameter	Mandatory	Type	Description
page_size	No	Integer	Number of data records on each page

Request Parameters

Table 3-306 Request header parameters

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

Response Parameters

Status code: 200

Table 3-307 Response body parameters

Parameter	Type	Description
error	Error object	Response error
result	TemplateListInfo object	Response result
status	String	Response status

Table 3-308 Error

Parameter	Type	Description
code	String	Error codes
message	String	Error message

Table 3-309 TemplateListInfo

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

Table 3-310 DevstarRepoInfo

Parameter	Type	Description
uuid	String	Repository UUID
repo_id	String	Repository ID
repo_name	String	Repository name
ssh_url	String	Repository SSH address
code_url	String	Repository HTTPS address
detail_url	String	Repository preview address

Request Examples

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

Response Examples

Status code: 200

Request succeeded

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

Status Codes

Status Codes	Description
200	Request succeeded

Error Codes

See Error Codes.

3.9.2 Obtaining Information About All Repositories in a Project

Description

This API is used to obtain the repository list. Fuzzy query is supported: If the *project_id* is not specified, fuzzy search by repository name or project name is supported. Otherwise, only fuzzy search by repository name is supported.

URI

GET /v1/projects/{project_uuid}/repositories

Table 3-311 Path parameters

Parameter	Mandatory	Type	Description
project_uuid	Yes	String	Project ID. For details, see Obtaining a Project ID .

Table 3-312 Query parameters

Parameter	Mandatory	Type	Description
page_index	No	Integer	Page index, starting from 1
page_size	No	Integer	Record number on each page
search	No	String	Search keyword

Request Parameters

Table 3-313 Request header parameters

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

Response Parameters

Status code: 200**Table 3-314** Response body parameters

Parameter	Type	Description
error	Error object	Response error
result	RepoListInfo object	Response result
status	String	Response status

Table 3-315 Error

Parameter	Type	Description
code	String	Error codes
message	String	Error message

Table 3-316 RepoListInfo

Parameter	Type	Description
repositorys	Array of RepoInfo objects	Repository list
total	Integer	Total number of repositories

Table 3-317 RepoInfo

Parameter	Type	Description
createAt	String	Creation time
groupName	String	Repository group name (Segment following the domain name and before the repository name in the clone address. Example: git@repo.alpha.devcloud.intest.com:Demo00228/testword.git . Repository group name: Demo00228 .)
httpUrl	String	https url
id	String	Repository UUID returned by the CreateRepository API. Minimum length: 32 Maximum length: 32
name	String	Repository name Minimum length: 2 Maximum length: 128
projectId	String	Project ID. For details, see Obtaining a Project ID . Minimum length: 32 Maximum length: 32
projectIsDelete	String	Whether the project is deleted
repold	String	Repository primary key ID Minimum length: 1 Maximum length: 40
sshUrl	String	ssh url
visibilityLevel	Integer	Visibility. 0 : Private. 20 : Public
webUrl	String	Web URL access path

Request Examples

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

Response Examples

Status code: 200

Request succeeded

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

```
"total" : 1,
"repositories" : [ {
  "id" : "d0457f74cc5b43d989ded7248f71b4e5",
  "repoid" : "468480",
  "name" : "CreatRepository_2qD5i",
  "sshUrl" : "git@repo.alpha.devcloud.intest.com:Demo00228/CreatRepository_2qD5i.git",
  "httpUrl" : "https://repo.alpha.devcloud.intest.com/Demo00228/CreatRepository_2qD5i.git",
  "groupName" : "Demo00228",
  "webUrl" : "https://alpha.devcloud.intest.com/repo/468480/home",
  "visibilityLevel" : 0,
  "createAt" : "2019-10-15 09:40:26",
  "projectId" : "4838955a48e2492bbe44b31bc4c272f6",
  "projectsDelete" : "false"
} ],
"status" : "success"
}
```

Status Codes

Status Codes	Description
200	Request succeeded

Error Codes

See Error Codes.

3.9.3 Verifying a Repository Name of a Specified Project

Description

This API is called during project creation to check whether a repository name is duplicate in the project. **result:false** indicates that the repository exists, and **result:true** indicates that the repository does not exist.

URI

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

Table 3-318 Path parameters

Parameter	Mandatory	Type	Description
project_uuid	Yes	String	Project ID. For details, see Obtaining a Project ID .
repository_name	Yes	String	Repository name

Request Parameters

Table 3-319 Request header parameters

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

Response Parameters

Status code: 200**Table 3-320** Response body parameters

Parameter	Type	Description
error	Error object	Response error
result	Boolean	Response result
status	String	Response status

Table 3-321 Error

Parameter	Type	Description
code	String	Error codes
message	String	Error message

Request Examples

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

Response Examples

Status code: 200

Request succeeded

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

Status Codes

Status Codes	Description
200	Request succeeded

Error Codes

See Error Codes.

3.10 Repository Management (V2)

3.10.1 Creating a Branch

Description

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 3-322 Path parameters

Parameter	Mandatory	Type	Description
repository_id	Yes	String	Repository primary key ID

Request Parameters

Table 3-323 Request header parameters

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

Table 3-324 Request body parameters

Parameter	Mandatory	Type	Description
branch_name	Yes	String	Branch name
ref	Yes	String	Source branch name

Response Parameters

Status code: 200

Table 3-325 Response body parameters

Parameter	Type	Description
error	Error object	Response error
result	AddProtectResponse object	Response result
status	String	Response status

Table 3-326 Error

Parameter	Type	Description
code	String	Error codes
message	String	Error message

Table 3-327 AddProtectResponse

Parameter	Type	Description
name	String	Branch name
commit	CommitRepoV2 object	Commit message
protected	Boolean	Whether to protect it
developers_can_push	Boolean	Whether a developer can commit
developers_can_merge	Boolean	Whether a developer can merge
master_can_push	Boolean	Whether an administrator can commit
master_can_merge	Boolean	Whether an administrator can merge
no_one_can_push	Boolean	No one can commit
no_one_can_merge	Boolean	No one can merge

Parameter	Type	Description
in_an_opened_merge_request	Boolean	Whether it is in an open MR

Table 3-328 CommitRepoV2

Parameter	Type	Description
id	String	SHA ID of a commit
short_id	String	Short SHA ID of a commit
created_at	String	Creation time
title	String	Commit title
parent_ids	Array of strings	Parent commit ID
message	String	Commit message
author_name	String	Author
committer_name	String	Committer
committed_date	String	Commit time

Request Examples

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

Response Examples

Status code: 200

Request succeeded

```
{
  "result": {
    "name": "master",
    "commit": {
      "id": "3b68902cf0eb7de3d5c8757e4a990bc3fe0af349",
      "short_id": "3b68902c",
      "title": "Create a File",
      "created_at": "2022-03-08T03:14:39.000Z",
      "parent_ids": [ "08618c900a4048aae7e4cd88913d3d521eae02eb" ],
      "message": "Create 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 Codes	Description
200	Request succeeded

Error Codes

See Error Codes.

3.11 Review

3.11.1 Creating an MR Review

Description

This API is used to create an MR review.

URI

POST /v2/repositories/{repository_id}/merge_requests/{merge_request_id}/discussions

Table 3-329 Path parameters

Parameter	Mandatory	Type	Description
repository_id	Yes	Integer	Repository short ID Minimum value: 1 Maximum value: 2,147,483,647
merge_request_id	Yes	Integer	MR No. Minimum value: 1 Maximum value: 2,147,483,647

Request Parameters

Table 3-330 Request header parameters

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

Table 3-331 Request body parameters

Parameter	Mandatory	Type	Description
body	Yes	String	Review content
severity	No	String	Severity
assignee_id	No	String	Assigner ID
review_categories	No	String	Review type
review_modules	No	String	Review module
proposer_id	No	String	Proposer ID
position	No	PositionDto object	Code location information

Table 3-332 PositionDto

Parameter	Mandatory	Type	Description
base_sha	No	String	Base commit node of the source branch
start_sha	No	String	Latest commit node of the target branch
head_sha	No	String	Latest commit node of the source branch
old_path	No	String	Old file path
new_path	No	String	New file path
position_type	No	String	Change type

Parameter	Mandatory	Type	Description
old_line	No	Integer	Old line No.
new_line	No	Integer	New line No.

Response Parameters

Status code: 200

Table 3-333 Response body parameters

Parameter	Type	Description
error	Error object	Response error
result	MergeRequestDiscussionDto object	Response result
status	String	Response status

Table 3-334 Error

Parameter	Type	Description
code	String	Error codes
message	String	Error message

Table 3-335 MergeRequestDiscussionDto

Parameter	Type	Description
id	String	Comment ID
individual_note	Boolean	individual_note
notes	Array of NoteDto objects	Main comment and reply list Array length: 0-1
project_id	Integer	Repository ID Minimum value: 1 Maximum value: 2,147,483,647
noteable_type	String	Object type
commit_id	String	Associated commit ID

Parameter	Type	Description
project_full_path	String	Repository path
a_mode	String	Old file mode
b_mode	String	New file mode
deleted_file	Boolean	Whether to delete files during the modification
new_file	Boolean	Whether to add files during the modification
resolved	Boolean	Whether the review is resolved
archived	Boolean	Whether the review is archived
review_categories	String	Review type
review_categories_cn	String	Review type in Chinese
review_categories_en	String	Review type
review_modules	String	Review module
severity	String	Severity
severity_cn	String	Severity in Chinese
severity_en	String	Severity
assignee	UserBasicDto object	Assignee
proposer	UserBasicDto object	Proposer
merge_request_version_params	MergeRequestVersionParamsDto object	MR version information
diff_file	String	Changed file
added_lines	Integer	Added lines Minimum value: 1 Maximum value: 2,147,483,647
removed_lines	Integer	Deleted lines Minimum value: 1 Maximum value: 2,147,483,647

Table 3-336 NoteDto

Parameter	Type	Description
id	Integer	note id Minimum value: 1 Maximum value: 2,147,483,647
type	String	Note type
body	String	Review content
attachment	String	Attachment
author	UserBasicDto object	Author
created_at	String	Creation time
updated_at	String	Update time
system	Boolean	Whether the log is generated by the system
noteable_id	Integer	Target ID Minimum value: 1 Maximum value: 2,147,483,647
noteable_type	String	Object type
commit_id	String	Associated commit ID
resolvable	Boolean	Whether can be resolved
is_reply	Boolean	Whether the review is replied
resolved_by	UserBasicDto object	Solver
noteable_iid	Integer	Target No. Minimum value: 1 Maximum value: 2,147,483,647
discussion_id	String	Discussion ID
project	String	Project
diff_file	String	Changed file
diff	String	Changed content
archived	Boolean	Whether it is archived
review_categories	String	Review type
review_categories_cn	String	Review type in Chinese

Parameter	Type	Description
review_categories_en	String	Review type
review_modules	String	Review module
severity	String	Severity
severity_cn	String	Severity in Chinese
severity_en	String	Severity
file_path	String	File path
line	String	Line No.
assignee	UserBasicDto object	Assignee
proposer	UserBasicDto object	Proposer
position	PositionDto object	Associated code location
resolved	Boolean	Whether it is resolved
is_outdated	Boolean	Whether it is outdated

Table 3-337 PositionDto

Parameter	Type	Description
base_sha	String	Base commit node of the source branch
start_sha	String	Latest commit node of the target branch
head_sha	String	Latest commit node of the source branch
old_path	String	Old file path
new_path	String	New file path
position_type	String	Change type
old_line	Integer	Old line No.
new_line	Integer	New line No.

Table 3-338 UserBasicDto

Parameter	Type	Description
id	Integer	User ID Minimum value: 1 Maximum value: 2,147,483,647
name	String	Name
username	String	Username
state	String	Status
avatar_url	String	Avatar URL
avatar_path	String	Avatar path
email	String	Email
name_cn	String	Name in Chinese
web_url	String	Homepage
nick_name	String	Alias
tenant_name	String	Tenant name

Table 3-339 MergeRequestVersionParamsDto

Parameter	Type	Description
diff_id	Integer	Latest MR Change ID Minimum value: 1 Maximum value: 2,147,483,647
start_sha	String	Latest commit of the target branch
commit_id	String	Latest commit of the source branch

Request Examples

POST https://{endpoint}/v2/repositories/{repository_id}/merge_requests/{merge_request_iid}/discussions

```
{
  "body": "Review content",
  "position": {
    "new_path": "a.txt",
    "old_path": "a.txt",
    "new_line": 1,
    "old_line": -1
  },
  "severity": "suggestion"
}
```

Response Examples

Status code: 200

Request succeeded

```
{
  "result": {
    "id": "b7d95c35c08e0b328bfafd45408bd2ac28b73f9",
    "individual_note": false,
    "notes": [ {
      "id": 2111886200,
      "type": "DiffNote",
      "body": "Review content",
      "author": {
        "id": 1234,
        "name": "beta",
        "username": "c369c68f1ff84679b5a8ed904d8bff1c",
        "state": "active",
        "name_cn": "beta"
      },
      "created_at": "2023-06-01T17:05:12.000+08:00",
      "updated_at": "2023-06-01T17:05:12.000+08:00",
      "system": false,
      "noteable_id": 19387,
      "noteable_type": "MergeRequest",
      "resolvable": true,
      "is_reply": true,
      "noteable_iid": 2,
      "discussion_id": "8bbe6d4e797169c4c1ce56507fdf1abc11f95601",
      "project": "DevOpsqlcylxm00004/project-test",
      "diff_file": "7",
      "diff": "@@ -0,0 +1,0 @@\n+7",
      "archived": false,
      "review_categories": "regulations",
      "review_categories_cn": "Programming regulations",
      "review_categories_en": "Regulations",
      "review_modules": "aaa,bbb",
      "severity": "suggestion",
      "assignee": {
        "id": 1234,
        "name": "beta",
        "username": "c369c68f1ff84679b5a8ed904d8bff1c",
        "state": "active",
        "name_cn": "beta"
      },
      "proposer": {
        "id": 1234,
        "name": "beta",
        "username": "c369c68f1ff84679b5a8ed904d8bff1c",
        "state": "active",
        "name_cn": "beta"
      },
      "position": {
        "base_sha": "973bcc0211c32dbaa8473561c6767f74e1a81471",
        "start_sha": "973bcc0211c32dbaa8473561c6767f74e1a81471",
        "head_sha": "f5bf0230f9b996ed6c0d64d2b2fef789d91b164c",
        "old_path": "7",
        "new_path": "7",
        "position_type": "text",
        "new_line": 1
      },
      "resolved": false,
      "is_outdated": false
    } ],
    "project_id": 2111699786,
    "noteable_type": "MergeRequest",
    "project_full_path": "DevOpsqlcylxm00004/project-test",
    "a_mode": "0",
  }
}
```

```
"b_mode" : "100644",
"deleted_file" : false,
"new_file" : true,
"resolved" : false,
"archived" : false,
"review_categories" : "regulations",
  "review_categories_cn" : "Programming regulations",
  "review_categories_en" : "Regulations",
  "review_modules" : "aaa,bbb",
  "severity" : "suggestion",
  "severity_cn" : "Suggestion",
  "severity_en" : "Suggestion",
  "assignee" : {
    "id" : 1234,
    "name" : "beta",
    "username" : "c369c68f1ff84679b5a8ed904d8bff1c",
    "state" : "active",
    "name_cn" : "beta"
  },
  "proposer" : {
    "id" : 1234,
    "name" : "beta",
    "username" : "c369c68f1ff84679b5a8ed904d8bff1c",
    "state" : "active",
    "name_cn" : "beta"
  },
  "diff_file" : "@@ -0,0 +1,0 @@\n+7",
  "added_lines" : 1,
  "removed_lines" : 0
},
"status" : "success"
}
```

Status Codes

Status Codes	Description
200	Request succeeded

Error Codes

See Error Codes.

3.11.2 Replying to an MR Review

Description

This API is used to reply to an MR review.

URI

POST /v2/repositories/{repository_id}/merge_requests/{merge_request_iid}/discussions/{discussion_id}/notes

Table 3-340 Path parameters

Parameter	Mandatory	Type	Description
repository_id	Yes	Integer	Repository short ID Minimum value: 1 Maximum value: 2,147,483,647
merge_request_id	Yes	Integer	MR No. Minimum value: 1 Maximum value: 2,147,483,647
discussion_id	Yes	String	Comment ID

Request Parameters

Table 3-341 Request header parameters

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

Table 3-342 Request body parameters

Parameter	Mandatory	Type	Description
body	Yes	String	Review content

Response Parameters

Status code: 200

Table 3-343 Response body parameters

Parameter	Type	Description
error	Error object	Response error
result	NoteDto object	Response result

Parameter	Type	Description
status	String	Response status

Table 3-344 Error

Parameter	Type	Description
code	String	Error codes
message	String	Error message

Table 3-345 NoteDto

Parameter	Type	Description
id	Integer	note id Minimum value: 1 Maximum value: 2,147,483,647
type	String	Note type
body	String	Review content
attachment	String	Attachment
author	UserBasicDto object	Author
created_at	String	Creation time
updated_at	String	Update time
system	Boolean	Whether the log is generated by the system
noteable_id	Integer	Target ID Minimum value: 1 Maximum value: 2,147,483,647
noteable_type	String	Object type
commit_id	String	Associated commit ID
resolvable	Boolean	Whether can be resolved
is_reply	Boolean	Whether the review is replied
resolved_by	UserBasicDto object	Solver

Parameter	Type	Description
noteable_iid	Integer	Target No. Minimum value: 1 Maximum value: 2,147,483,647
discussion_id	String	Discussion ID
project	String	Project
diff_file	String	Changed file
diff	String	Changed content
archived	Boolean	Whether it is archived
review_categories	String	Review type
review_categories_cn	String	Review type in Chinese
review_categories_en	String	Review type
review_modules	String	Review module
severity	String	Severity
severity_cn	String	Severity in Chinese
severity_en	String	Severity
file_path	String	File path
line	String	Line No.
assignee	UserBasicDto object	Assignee
proposer	UserBasicDto object	Proposer
position	PositionDto object	Associated code location
resolved	Boolean	Whether it is resolved
is_outdated	Boolean	Whether it is outdated

Table 3-346 UserBasicDto

Parameter	Type	Description
id	Integer	User ID Minimum value: 1 Maximum value: 2,147,483,647
name	String	Name
username	String	Username
state	String	Status
avatar_url	String	Avatar URL
avatar_path	String	Avatar path
email	String	Email
name_cn	String	Name in Chinese
web_url	String	Homepage
nick_name	String	Alias
tenant_name	String	Tenant name

Table 3-347 PositionDto

Parameter	Type	Description
base_sha	String	Base commit node of the source branch
start_sha	String	Latest commit node of the target branch
head_sha	String	Latest commit node of the source branch
old_path	String	Old file path
new_path	String	New file path
position_type	String	Change type
old_line	Integer	Old line No.
new_line	Integer	New line No.

Request Examples

```
POST https://{endpoint}/v2/repositories/{repository_id}/merge_requests/{merge_request_iid}/discussions/  
{discussion_id}/notes  
  
{  
  "body" : "Reply content"  
}
```

Response Examples

Status code: 200

Request succeeded

```
{
  "result" : {
    "id" : 2111886200,
    "type" : "DiffNote",
    "body" : "Review content",
    "author" : {
      "id" : 1234,
      "name" : "beta",
      "username" : "c369c68f1ff84679b5a8ed904d8bff1c",
      "state" : "active",
      "name_cn" : "beta"
    },
    "created_at" : "2023-06-01T17:05:12.000+08:00",
    "updated_at" : "2023-06-01T17:05:12.000+08:00",
    "system" : false,
    "noteable_id" : 19387,
    "noteable_type" : "MergeRequest",
    "resolvable" : true,
    "is_reply" : true,
    "noteable_iid" : 2,
    "discussion_id" : "8bbe6d4e797169c4c1ce56507fdf1abc11f95601",
    "project" : "DevOpsqlcylxm00004/project-test",
    "diff_file" : "7",
    "diff" : "@@ -0,0 +1,0 @@\n+7",
    "archived" : false,
    "review_categories" : "regulations",
    "review_categories_cn" : "Programming regulations",
    "review_categories_en" : "Regulations",
    "review_modules" : "aaa,bbb",
    "severity" : "suggestion",
    "severity_cn" : "Suggestion",
    "severity_en" : "Suggestion",
    "assignee" : {
      "id" : 1234,
      "name" : "beta",
      "username" : "c369c68f1ff84679b5a8ed904d8bff1c",
      "state" : "active",
      "name_cn" : "beta"
    },
    "proposer" : {
      "id" : 1234,
      "name" : "beta",
      "username" : "c369c68f1ff84679b5a8ed904d8bff1c",
      "state" : "active",
      "name_cn" : "beta"
    },
    "position" : {
      "base_sha" : "973bcc0211c32dbaa8473561c6767f74e1a81471",
      "start_sha" : "973bcc0211c32dbaa8473561c6767f74e1a81471",
      "head_sha" : "f5bf0230f9b996ed6c0d64d2b2fef789d91b164c",
      "old_path" : "7",
      "new_path" : "7",
      "position_type" : "text",
      "new_line" : 1
    },
    "resolved" : false,
    "is_outdated" : false
  },
  "status" : "success"
}
```

Status Codes

Status Codes	Description
200	Request succeeded

Error Codes

See Error Codes.

3.11.3 Obtaining Review Settings

Description

This API is used to obtain review settings.

URI

GET /v2/repositories/{repository_id}/review_setting

Table 3-348 Path parameters

Parameter	Mandatory	Type	Description
repository_id	Yes	Integer	Repository short ID Minimum value: 1 Maximum value: 2,147,483,647

Request Parameters

Table 3-349 Request header parameters

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

Response Parameters

Status code: 200

Table 3-350 Response body parameters

Parameter	Type	Description
error	Error object	Response error
result	ReviewSettingDto object	Response result
status	String	Response status

Table 3-351 Error

Parameter	Type	Description
code	String	Error codes
message	String	Error message

Table 3-352 ReviewSettingDto

Parameter	Type	Description
categories_and_modules_enabled	Boolean	Whether to enable the review category and module
secondary_category_enabled	Boolean	Whether to enable level-2 category
forbidden_add_to_issue	Boolean	Whether to forbid issue association
primary_categories	Array of CategoryDto objects	Level-1 category Array length: 0-100
review_default_categories	Array of strings	Default catalog Array length: 0-100
review_customized_categories	Array of strings	Custom category Array length: 0-100
review_modules	Array of strings	Review module Array length: 0-100
source_id	Integer	Target ID Minimum value: 1 Maximum value: 2,147,483,647

Parameter	Type	Description
source_type	String	Object type
source_path	String	Object path
secondary_category_type	String	Level-2 category type
secondary_categories	Array of CategoryDto objects	Level-2 category

Table 3-353 CategoryDto

Parameter	Type	Description
key	String	Key
name_zh	String	Name in Chinese
name_en	String	Name
sub_categories	Array of CategoryDto objects	Level-2 category Array length: 0-100

Request Examples

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

{
  "body": "Review title",
  "review_categories": "",
  "review_modules": "",
  "severity": "suggestion",
  "assignee_id": 9124
}
```

Response Examples

Status code: 200

Request succeeded

```
{
  "result": {
    "categories_and_modules_enabled": true,
    "secondary_category_enabled": false,
    "primary_categories": [ {
      "key": "design",
      "name_zh": "Code design",
      "name_en": "Code design"
    }, {
      "key": "regulations",
      "name_zh": "Programming regulations",
      "name_en": "Programming guideline"
    } ],
  }
}
```

```
"review_default_categories" : [ "design", "regulations" ],  
"review_customized_categories" : [ "xxx", "yyy" ],  
"review_modules" : [ "aaa", "bbb", "ccc" ],  
"source_id" : 2111699786,  
"source_type" : "Project",  
"source_path" : "DevOpsqlcylxm00004/project-test"  
},  
"status" : "success"  
}
```

Status Codes

Status Codes	Description
200	Request succeeded

Error codes

See Error Codes.

4 Application Examples

4.1 Example 1: Querying All Repositories of a User

Scenario

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

For details on how to call APIs, see [Making an API Request](#).

Prerequisites

You have enabled CodeArts Repo on the 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:

- 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.test.area.test.com:testzqxq00001/11111111111111111111.git",
        "https_url": "https://codehub.devcloud.test.area.test.com/testzqxq00001/11111111111111111111.git",
        "group_name": "testzqxq00001",

```

```
"web_url": "https://devcloud.test.area.test.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"}
```

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

Scenario

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

For details on how to call APIs, see [Making an API Request](#).

Prerequisites

You have enabled CodeArts Repo on the 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:

- 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.test.area.test.com:testzqxq00001/1111111111111111.git",
```

```
"https_url": "https://codehub.devcloud.test.area.test.com/testzqxq00001/11111111111111111111.git",
"group_name": "testzqxq00001",
"web_url": "https://devcloud.test.area.test.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:

- Response example:

```
{
  "result": {
    "star": null,
    "status": null,
    "userRole": null,
    "repository_uuid": "25c88b23d4dc400fa7b3e2a152e4d53c",
    "repository_id": 522140,
    "repository_name": "1111111111111111",
    "ssh_url": "git@codehub.devcloud.test.area.test.com:testzqxq00001/11111111111111111111.git",
    "https_url": "https://codehub.devcloud.test.area.test.com/testzqxq00001/11111111111111111111.git",
    "group_name": "testzqxq00001",
    "web_url": "https://devcloud.test.area.test.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"
}
```

4.3 Example 3: Querying Branches of a Repository

Scenario

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

For details on how to call APIs, see [Making an API Request](#).

Prerequisites

You have enabled CodeArts Repo on the 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:

- 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.test.area.test.com:testzqx00001/1111111111111111.git",
        "https_url": "https://codehub.devcloud.test.area.test.com/testzqx00001/1111111111111111.git",
        "group_name": "testzqx00001",
        "web_url": "https://devcloud.test.area.test.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:

- Response example:

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

```

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

5 Appendix

5.1 Status Codes

Table 5-1 Status Codes

Status Code	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.
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.

Status Code	Message	Description
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.
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.

Status Code	Message	Description
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.
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.

Status Code	Message	Description
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.

5.2 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.

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.

5.3 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": "region-mame-1",
      "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. Click the username and select **My Credentials** from the drop-down list
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.

5.4 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**.