



知识库

API 参考

发布日期 2024-05-16

目录

1 使用前必读.....	1
1.1 概述.....	1
1.2 调用说明.....	1
1.3 终端节点.....	1
1.4 约束与限制.....	1
1.5 基本概念.....	1
2 API 概览.....	3
3 如何调用 API.....	4
3.1 构造请求.....	4
3.2 认证鉴权.....	7
3.3 返回结果.....	11
4 API.....	13
4.1 文档操作.....	13
4.1.1 查询 Wiki 全部文档树结构.....	13
4.1.2 查询文档详情.....	16
4.2 文件操作.....	19
4.2.1 获取文件库下的文件树.....	19
4.2.2 项目上传文件接口.....	22
4.2.3 下载文件.....	24
5 附录.....	27
5.1 获取项目 ID.....	27
5.2 获取租户 ID.....	28
5.3 状态码.....	28
5.4 错误码.....	30

1 使用前必读

1.1 概述

知识库（CodeArts Wiki）是大型企业内部沉淀的团队协作的云端知识库，包括Wiki和文件库，支持知识全生命周期管理，帮助企业完成智能、安全的知识作业，主要包含多人在线协同编辑、文档轻松分享、强大的活文档、历史版本管理、需求管理协同联动、丰富场景化模板等功能。

您可以使用本文档提供的API对CodeArts Wiki进行相关操作，如执行、查询任务等，支持的全部操作请参见[API概览](#)。

1.2 调用说明

CodeArts Wiki提供了REST（Representational State Transfer）风格API，支持您通过HTTPS请求调用，调用方法请参见[如何调用API](#)。

1.3 终端节点

终端节点（Endpoint）即调用API的请求地址，不同服务不同区域的终端节点不同，您可以向管理员获取所有服务的终端节点。

1.4 约束与限制

详细的限制请参见具体API的说明。

1.5 基本概念

- 用户
由账号在IAM中创建的用户，是云服务的使用人员，具有身份凭证（密码和访问密钥）。
通常在调用API的鉴权过程中，您需要用到账号、用户和密码等信息。
- 区域（Region）

从地理位置和网络时延维度划分，同一个Region内共享弹性计算、块存储、对象存储、VPC网络、弹性公网IP、镜像等公共服务。Region分为通用Region和专属Region，通用Region指面向公共租户提供通用云服务的Region；专属Region指只承载同一类业务或只面向特定租户提供业务服务的专用Region。

- 可用区（AZ，Availability Zone）

一个AZ是一个或多个物理数据中心的集合，有独立的风火水电，AZ内逻辑上再将计算、网络、存储等资源划分成多个集群。一个Region中的多个AZ间通过高速光纤相连，以满足用户跨AZ构建高可用性系统的需求。

2 API 概览

表 2-1 API 概览

服务	分类	说明
API	文档操作	文档相关API，如查询Wiki全部文档树结构、查询文档详情操作。
	文件操作	文件相关API，如获取文件库下的文件树、下载文件操作。

3 如何调用 API

3.1 构造请求

本节介绍REST API请求的组成，并以调用IAM的获取用户Token接口说明如何调用API，该API获取用户的Token，Token可以用于调用其他API时鉴权。

请求 URI

请求URI由如下部分组成。

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

尽管请求URI包含在请求消息头中，但大多数语言或框架都要求您从请求消息中单独传递它，所以在此单独强调。

表 3-1 URI 中的参数说明

参数	描述
URI-scheme	表示用于传输请求的协议，当前所有API均采用HTTPS协议。
Endpoint	指定承载REST服务端点的服务器域名或IP，不同服务不同区域的Endpoint不同。
resource-path	资源路径，也即API访问路径。从具体API的URI模块获取，例如“获取用户Token”API的resource-path为“/v3/auth/tokens”。
query-string	查询参数，是可选部分，并不是每个API都有查询参数。查询参数前面需要带一个“？”，形式为“参数名=参数取值”，例如“limit=10”，表示查询不超过10条数据。多个查询参数之间使用“&”隔开。

例如您获取到某区域的Endpoint（iam.xxx.com），并在获取用户Token接口的URI部分找到resource-path（/v3/auth/tokens），拼接起来如下所示。

<https://iam.xxx.com/v3/auth/tokens>

📖 说明

为查看方便，在每个具体API的URI部分，只给出resource-path部分，并将请求方法写在一起。这是因为URI-scheme都是HTTPS，而Endpoint在同一个区域也相同，所以简洁起见将这两部分省略。

请求方法

HTTP请求方法（也称为操作或动词），它告诉服务你正在请求什么类型的操作。

表 3-2 HTTP 方法

方法	说明
GET	请求服务器返回指定资源。
PUT	请求服务器更新指定资源。
POST	请求服务器新增资源或执行特殊操作。
DELETE	请求服务器删除指定资源，如删除对象等。
HEAD	请求服务器资源头部。
PATCH	请求服务器更新资源的部分内容。 当资源不存在的时候，PATCH可能会去创建一个新的资源。

在获取用户Token的URI部分，您可以看到其请求方法为“POST”，则其请求为：

```
POST https://iam.xxx.com/v3/auth/tokens
```

请求消息头

附加请求头字段，如指定的URI和HTTP方法所要求的字段。例如定义消息体类型的请求头“Content-Type”，请求鉴权信息等。

详细的公共请求消息头字段请参见[表3-3](#)。

表 3-3 公共请求消息头

名称	描述	是否必选	示例
Content-Type	发送的实体的MIME类型。推荐用户默认使用application/json，有其他取值时会在具体接口中专门说明。	是	application/json

名称	描述	是否必选	示例
X-Project-Id	project id, 项目编号。 请参考 获取项目ID 章节 获取项目编号。	否 如果是专属云场景 采用AK/SK认证方 式的接口请求或者 多project场景采用 AK/SK认证的接口 请求, 则该字段必 选。	e9993fc787d***** ***aa340f9c0f4
X-Auth-Token	用户Token。 用户Token也就是调用获 取用户Token接口的响应 值, 该接口是唯一不需 要认证的接口。 请求响应成功后在响应 消息头中包含的“X- Subject-Token”的值即 为Token值。	否 使用Token认证时 该字段必选。	注: 以下仅为 Token示例片段 MIIPAgYJKoZlhc NAQcCo...ggg1BBI INPXsidG9rZ

📖 说明

API同时支持使用AK/SK认证, AK/SK认证是使用SDK对请求进行签名, 签名过程会自动往请求中
添加Authorization (签名认证信息)和X-Sdk-Date (请求发送的时间)请求头。

AK/SK认证的详细说明请参见[AK/SK认证](#)。

对于获取用户Token接口, 由于不需要认证, 所以只添加“Content-Type”即可, 添加
消息头后的请求如下所示。

```
POST https://iam.xxx.com/v3/auth/tokens
Content-Type: application/json
```

请求消息体

请求消息体通常以结构化格式 (如JSON或XML) 发出, 与请求消息头中Content-type
对应, 传递除请求消息头之外的内容。若请求消息体中参数支持中文, 则中文字符必
须为UTF-8编码。

每个接口的请求消息体内容不同, 也并不是每个接口都需要有请求消息体 (或者说消
息体为空), GET、DELETE操作类型的接口就不需要消息体, 消息体具体内容需要根
据具体接口而定。

对于获取用户Token接口, 您可以从接口的请求部分看到所需的请求参数及参数说明。
将消息体加入后的请求如下所示, 加粗的斜体字段需要根据实际值填写, 其中
username为用户名, ***tenantname***为用户所属的租户名称, ***********为用户登录密码,
xxxxxxxxxxxxxxxxxxx为project的ID, 获取方法请参见[获取项目ID](#)。

📖 说明

scope参数定义了Token的作用域, 下面示例中获取的Token仅能访问project下的资源。您还可
以设置Token的作用域为某个账号下所有资源或账号的某个project下的资源, 详细定义请参见获
取用户Token接口。

```
POST https://iam.xxx.com/v3/auth/tokens
Content-Type: application/json
```

```
{
  "auth": {
    "identity": {
      "methods": [
        "password"
      ],
      "password": {
        "user": {
          "name": "username",
          "password": "*****",
          "domain": {
            "name": "tenantname"
          }
        }
      }
    },
    "scope": {
      "project": {
        "id": "XXXXXXXXXXXXXXXXXXXX"
      }
    }
  }
}
```

到这里为止这个请求需要的内容就具备齐全了，您可以使用curl、Postman或直接编写代码等方式发送请求调用API。对于获取用户Token接口，返回的响应消息头中“x-subject-token”就是需要获取的用户Token。有了Token之后，您就可以使用Token认证调用其他API。

3.2 认证鉴权

调用接口有如下两种认证方式，您可以选择其中一种进行认证鉴权。推荐使用AK/SK认证，其安全性比Token认证要高。

- Token认证：通过Token认证通用请求。
- AK/SK认证：通过AK（Access Key ID）/SK（Secret Access Key）加密调用请求。

Token 认证

📖 说明

Token的有效期为24小时，需要使用一个Token鉴权时，可以先缓存起来，避免频繁调用。

Token在计算机系统中代表令牌（临时）的意思，拥有Token就代表拥有某种权限。Token认证就是在调用API的时候将Token加到请求消息头，从而通过身份认证，获得操作API的权限。

Token可通过调用获取用户Token接口获取，调用本服务API需要project级别的Token，即调用获取用户Token接口时，请求body中auth.scope的取值需要选择project，如下所示。

```
{
  "auth": {
    "identity": {
      "methods": [
        "password"
      ],
      "password": {
        "user": {
```

```

    "name": "username",
    "password": "*****#",
    "domain": {
      "name": "domainname"
    }
  },
  "scope": {
    "project": {
      "name": "xxxxxxxxx"
    }
  }
}

```

获取Token后，再调用其他接口时，您需要在请求消息头中添加“X-Auth-Token”，其值即为Token。例如Token值为“ABCDEFJ...”，则调用接口时将“X-Auth-Token: ABCDEFJ...” 加到请求消息头即可，如下所示。

```

GET https://iam.xxx.com/v3/auth/projects
Content-Type: application/json
X-Auth-Token: ABCDEFJ....

```

AK/SK 认证

📖 说明

AK/SK签名认证方式仅支持消息体大小12MB以内，12MB以上的请求请使用Token认证。

AK/SK认证就是使用AK/SK对请求进行签名，在请求时将签名信息添加到消息头，从而通过身份认证。

- AK(Access Key ID)：访问密钥ID。与私有访问密钥关联的唯一标识符；访问密钥ID和私有访问密钥一起使用，对请求进行加密签名。
- SK(Secret Access Key)：与访问密钥ID结合使用的密钥，对请求进行加密签名，可标识发送方，并防止请求被修改。

使用AK/SK认证时，您可以基于签名算法使用AK/SK对请求进行签名，也可以使用专门的签名SDK对请求进行签名。

须知

签名SDK只提供签名功能，与服务提供的SDK不同，使用时请注意。

以下结合一个[Demo](#)来介绍如何对一个请求进行签名，并通过HTTP Client发送一个HTTPS请求的过程。如果您不使用Demo工程，也可以向管理员获取API网关签名工具，并在其他工程中引用。

1. 生成AK/SK。如果已生成过AK/SK，则可跳过此步骤，找到原来已下载的AK/SK文件，文件名一般为：credentials.csv。
 - a. 登录管理控制台。
 - b. 单击用户名，在下拉列表中单击“我的凭证”。
 - c. 单击“访问密钥”。
 - d. 单击“新增访问密钥”，根据需要在弹框中输入描述，单击“确定”。
 - e. 下载成功后，在credentials文件中即可获取AK和SK信息。

须知

- 每个用户仅允许新增两个访问密钥。
 - 为保证访问密钥的安全，访问密钥仅在初次生成时自动下载，后续不可再次通过管理控制台界面获取。请在生成后妥善保管。
2. 获取示例代码，并解压缩。
 3. 通过import方式将示例工程导入到Eclipse。

图 3-1 选择已存在的工程

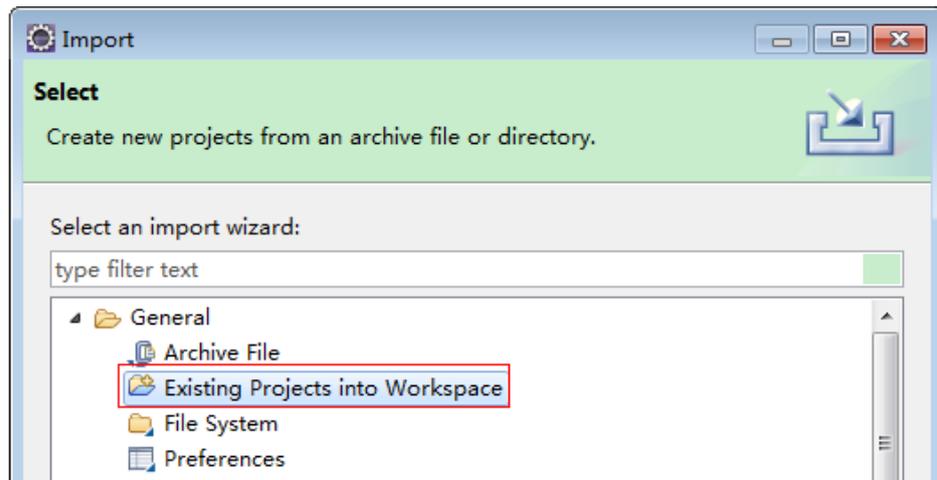


图 3-2 选择解压后的示例代码

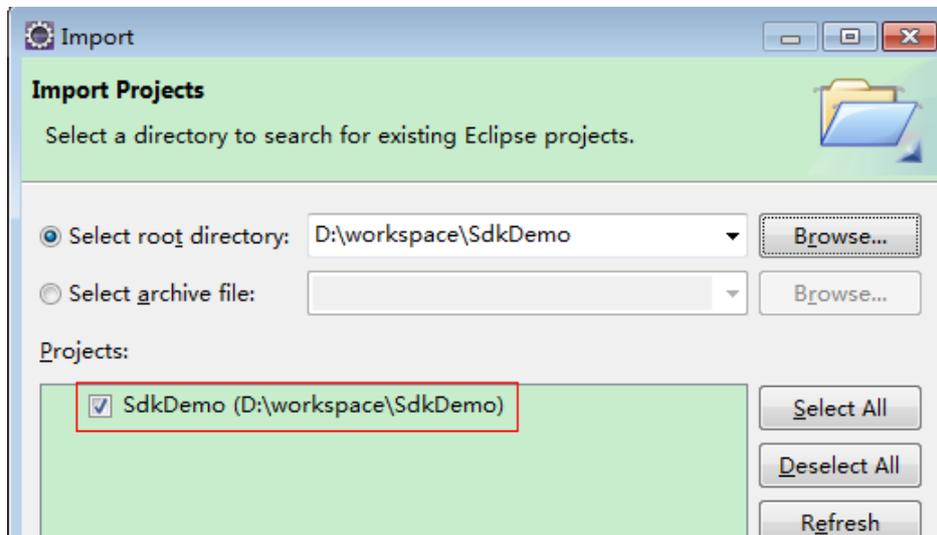
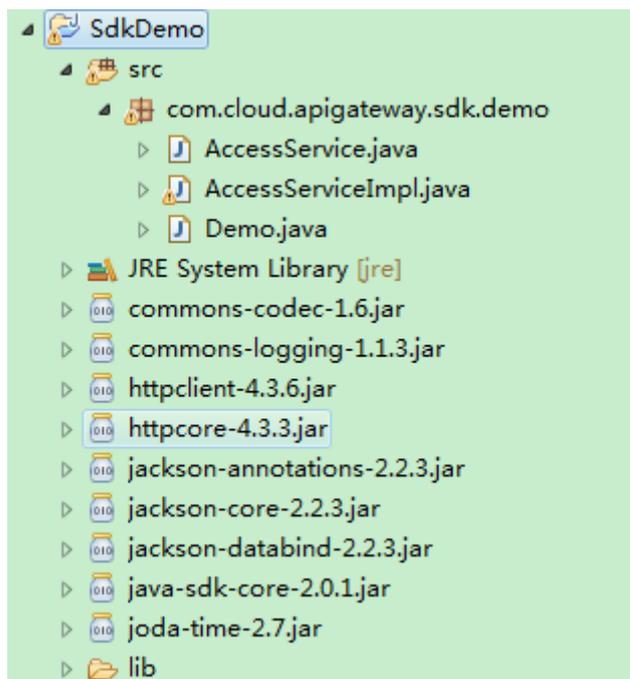


图 3-3 导入成功后工程结构示例



4. 对请求进行签名。

签名方法集成在3引入的java-sdk-core-x.x.x.jar文件中。发送请求前，需要对请求内容进行签名，得到的签名结果将作为http头部信息一起发送。

Demo代码分成三个类进行演示：

- AccessService：抽象类，将GET/POST/PUT/DELETE归一成access方法。
- Demo：运行入口，模拟用户进行GET/POST/PUT/DELETE请求。
- AccessServiceImpl：实现access方法，具体与API网关通信的代码都在access方法中。

a. 编辑“Demo.java”文件中的main方法，将以下内容替换为实际获取到的值。

如果调用其他方法，如POST，PUT，DELETE等，请参考对应注释方法。

注意替换 region、serviceName、AK/SK 和 URL，Demo中使用了获取VPC的 URL，请替换为您需要的URL：

URL中project_id获取请参见[获取项目ID](#)。

```
//TODO: Replace region with the name of the region in which the service to be accessed is
located.
private static final String region = "";
//TODO: Replace vpc with the name of the service you want to access. For example, ecs, vpc,
iam, and elb.
private static final String serviceName = "";
public static void main(String[] args) throws UnsupportedEncodingException
{
//TODO: Replace the AK and SK with those obtained on the My Credential page.
String ak = "ZIRRKMTWP*****1WKNKB";
String sk = "Us0mdMNHk*****YrCnW0ecfzl";
//TODO: To specify a project ID (multi-project scenarios), add the X-Project-Id header.
//TODO: To access a global service, such as IAM, DNS, CDN, and TMS, add the X-Domain-Id
header to specify an account ID.
//TODO: To add a header, find "Add special headers" in the AccessServiceImpl.java file.
//TODO: Test the API
String url = "https://{Endpoint}/v1/{project_id}/vpcs";
get(ak, sk, url);
```

```
//TODO: When creating a VPC, replace {project_id} in postUrl with the actual value.
//String postUrl = "https://serviceEndpoint/v1/{project_id}/cloudservers";
//String postbody = "{\"vpc\": {\"name\": \"vpc\", \"cidr\": \"192.168.0.0/16\"}}";
//post(ak, sk, postUrl, postbody);
//TODO: When querying a VPC, replace {project_id} in url with the actual value.
//String url = "https://serviceEndpoint/v1/{project_id}/vpcs/{vpc_id}";
//get(ak, sk, url);
//TODO: When updating a VPC, replace {project_id} and {vpc_id} in putUrl with the actual
values.
//String putUrl = "https://serviceEndpoint/v1/{project_id}/vpcs/{vpc_id}";
//String putbody = "{\"vpc\": {\"name\": \"vpc1\", \"cidr\": \"192.168.0.0/16\"}}";
//put(ak, sk, putUrl, putbody);
//TODO: When deleting a VPC, replace {project_id} and {vpc_id} in deleteUrl with the actual
values.
//String deleteUrl = "https://serviceEndpoint/v1/{project_id}/vpcs/{vpc_id}";
//delete(ak, sk, deleteUrl);
}
```

b. 编译与运行接口调用。

在左侧“Package Explorer”中找到“Demo.java”，右键选择“Run AS > Java Application”并单击“运行”。

可在控制台查看调用日志。

3.3 返回结果

状态码

请求发送以后，您会收到响应，包含状态码、响应消息头和消息体。

状态码是一组从1xx到5xx的数字代码，状态码表示了请求响应的状态，完整的状态码列表请参见[状态码](#)。

对于获取用户Token接口，如果调用后返回状态码为“201”，则表示请求成功。

响应消息头

对应请求消息头，响应同样也有消息头，如“Content-type”。

对于获取用户Token接口，返回如[图3-4](#)所示的消息头，其中“X-Subject-Token”就是需要获取的用户Token。有了Token之后，您就可以使用Token认证调用其他API。

图 3-4 获取用户 Token 响应消息头

Header	Value
Connection	keep-alive
X-IAM-Trace-Id	token_cn-north-1_null_9d154bc98611b9f6ab8f742d3c975d16
Cache-Control	no-cache, no-store, must-revalidate
Pragma	no-cache
Expires	Thu, 01 Jan 1970 00:00:00 GMT
X-Subject-Token	MllcEAYJKoZlthvcNAQcCollcATCCG-0CAQExDTALBglghkgBZQMEAgEwghoiBglqhik...
X-Request-Id	ed61e620e38ca386fc8d6b116c7fd555
Server	api-gateway
Strict-Transport-Security	max-age=31536000; includeSubdomains;
X-Frame-Options	SAMEORIGIN

响应消息体

响应消息体通常以结构化格式（如JSON或XML）返回，与响应消息头中Content-type对应，传递除响应消息头之外的内容。

对于获取用户Token接口，返回如下消息体。为篇幅起见，这里只展示部分内容。

```
{
  "token": {
    "expires_at": "2019-02-13T06:52:13.855000Z",
    "methods": [
      "password"
    ],
    "catalog": [
      {
        "endpoints": [
          {
            "region_id": "az-01",
            .....

```

当接口调用出错时，会返回错误码及错误信息说明，错误响应的Body体格式如下所示。

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

其中，error_code表示错误码，error_msg表示错误描述信息。

4 API

4.1 文档操作

4.1.1 查询 Wiki 全部文档树结构

功能介绍

查询某Wiki全部文档树结构

URI

GET /v1/openapi/doc/{zhishiku_id}/tree

表 4-1 路径参数

参数	是否必选	参数类型	描述
zhishiku_id	是	String	知识库id 最小长度：1 最大长度：64

请求参数

无

响应参数

状态码： 400

表 4-2 响应 Body 参数

参数	参数类型	描述
error_details	String	错误描述 最小长度：8 最大长度：36
suggestion	String	错误描述详情 最小长度：2 最大长度：512
error_code	String	错误编码 最小长度：1 最大长度：64
error_msg	String	错误信息 最小长度：1 最大长度：64

请求示例

请求pkbwf4nyak文档下的结构数据 其中pkbwf4nyak为文档id

```
/v1/openapi/doc/pkbwf4nyak/tree
```

响应示例

状态码： 200

OK

```
{
  "id": "pkbwf4nyak",
  "children": [ {
    "id": "73e2eccd36174e79bba45c7502aaa09b",
    "name": "未命名页面181619",
    "type": 14,
    "permission": 2,
    "children": [ ],
    "recycle": false,
    "can_create": false,
    "can_share": true,
    "can_delete": true,
    "can_recover": false,
    "can_rename": false,
    "can_association": false,
    "is_curr_user_admin": false,
    "region_id": "region-01",
    "domain_id": "f7285aa58f304ef49ac6667b902e3508",
    "creator_name": "devcloud_docman_wx491639",
    "creator_id": "78de0d103af7417aa9108644f564ede3",
    "create_time": "2024-01-02 18:16:19",
    "update_time": "2024-01-02 18:16:19",
    "updater_id": "78de0d103af7417aa9108644f564ede3",
    "updater_name": "devcloud_docman_wx491639",
    "order_no": 524288,
    "parent_id": "pkbwf4nyak",
```

```

"parent_path" : "/pkbwf4nyak",
"likePath" : "/pkbwf4nyak/73e2eccd36174e79bba45c7502aaa09b",
"old_id" : 0,
"delete_flag" : 0,
"delete_time" : null,
"operate_time" : null,
"is_public" : "0",
"is_share" : 0,
"belong_id" : "pkbwf4nyak",
"zsk_belong_id" : "",
"belong_type" : 1,
"is_favourite" : "no",
"invite_num" : 0,
"is_publish" : false,
"extend_attribute" : {
  "doc_width" : "standard",
  "enable_tag" : "1",
  "disable_copy" : "1",
  "enable_thumbs" : "1",
  "enable_comments" : "1",
  "enable_watermark" : "1",
  "enable_attachments" : "1"
},
"primitive_creator_name" : "",
"primitive_updater_name" : ""
}],
"isZhishiku" : true,
"can_create" : false,
"can_share" : false,
"can_delete" : false,
"can_recover" : false,
"can_rename" : false,
"can_association" : false,
"is_curr_user_admin" : true,
"region_id" : "region-01",
"domain_id" : "f7285aa58f304ef49ac6667b902e3508",
"name" : "Scrum_ILC",
"doc_count" : 7,
"file_count" : 0,
"creator_name" : "devcloud_docman_wx491639",
"primitive_creator_name" : "",
"creator_id" : "78de0d103af7417aa9108644f564ede3",
"update_time" : "2023-11-15 09:28:13",
"role_id" : "",
"role_name" : "",
"belong_id" : "555a2970b5ca4249b78fae7608dd02ef",
"belong_type" : 1,
"belong_name" : "",
"is_public" : "0",
"delete_flag" : 0,
"delete_time" : null,
"is_favourite" : "",
"icon" : 4,
"is_share" : 0,
"team_belong_path" : "",
"team_belong_name" : "",
"privilege_json" : null,
"privilege_belong_id" : "pkbwf4nyak",
"extend_attribute" : {
  "doc_width" : "standard",
  "enable_tag" : "1",
  "disable_copy" : "1",
  "enable_thumbs" : "1",
  "enable_comments" : "1",
  "enable_watermark" : "1",
  "enable_attachments" : "1"
},
"sort_order" : 0
}

```

状态码： 400

Bad request

```
{  
  "error_code" : "ZHISHIKU.001002",  
  "error_msg" : "参数错误",  
  "error_details" : "参数错误: invalid type",  
  "suggestion" : ""  
}
```

状态码

状态码	描述
200	OK
400	Bad request

错误码

请参见[错误码](#)。

4.1.2 查询文档详情

功能介绍

查询文档详情

URI

GET /v1/openapi/content/doc/{id}

表 4-3 路径参数

参数	是否必选	参数类型	描述
id	是	String	文档id 最小长度： 1 最大长度： 64

请求参数

无

响应参数

状态码： 400

表 4-4 响应 Body 参数

参数	参数类型	描述
error_details	String	错误描述 最小长度：8 最大长度：36
suggestion	String	错误描述详情 最小长度：2 最大长度：512
error_code	String	错误编码 最小长度：1 最大长度：64
error_msg	String	错误信息 最小长度：1 最大长度：64

请求示例

查询id为48408ac634154fa4ab3b5d29a6f15ac0的文档详情，其中48408ac634154fa4ab3b5d29a6f15ac0为文档id

```
/v1/openapi/doc/content/48408ac634154fa4ab3b5d29a6f15ac0?isPreview=false
```

响应示例

状态码：200

ok

```
{
  "id": "48408ac634154fa4ab3b5d29a6f15ac0",
  "name": "代岩测试20210104-1",
  "type": 1,
  "permission": 2,
  "children": [ ],
  "edits": 4,
  "views": 17,
  "isCopy": false,
  "id4Count": "",
  "recycle": false,
  "can_create": false,
  "can_share": true,
  "can_delete": true,
  "can_recover": false,
  "can_rename": false,
  "can_association": false,
  "is_curr_user_admin": false,
  "region_id": "region-01",
  "domain_id": "f7285aa58f304ef49ac6667b902e3508",
  "creator_name": "devcloud_docman_wx491639",
  "creator_id": "78de0d103af7417aa9108644f564ede3",
  "create_time": "2024-01-04 15:36:14",
  "update_time": "2024-01-04 19:00:54",
}
```

```

"updater_id" : "78de0d103af7417aa9108644f564ede3",
"updater_name" : "devcloud_docman_wx491639",
"order_no" : 3604480,
"parent_id" : "c3b561d9fd0e49338bf2a974d018b645",
"parent_path" : "/c3b561d9fd0e49338bf2a974d018b645",
"likePath" : "/c3b561d9fd0e49338bf2a974d018b645/48408ac634154fa4ab3b5d29a6f15ac0",
"old_id" : 0,
"delete_flag" : 0,
"delete_time" : null,
"operate_time" : "2024-01-05 10:35:20",
"is_public" : "0",
"is_share" : 0,
"belong_id" : "c3b561d9fd0e49338bf2a974d018b645",
"zsk_belong_id" : "",
"belong_type" : 0,
"is_favourite" : "no",
"invite_num" : 0,
"is_publish" : false,
"extend_attribute" : {
  "copies" : "[\\"1e1feb3b5a5444b3bca718d76fecb186\\",\\"da0f7ab2a12d41c487d02fb84e9c51fe
\\",\\"489f1694b32e4b30846def4b8c6632ea\\"]",
  "doc_width" : "standard",
  "enable_tag" : "1",
  "disable_copy" : "1",
  "enable_thumbs" : "1",
  "enable_comments" : "1",
  "enable_watermark" : "1",
  "enable_attachments" : "1"
},
"primitive_creator_name" : "",
"primitive_updater_name" : "",
"doc_region_id" : "region-01",
"tree_id" : "48408ac634154fa4ab3b5d29a6f15ac0",
"is_markdown" : 0,
"editor_doc_id" : "48408ac634154fa4ab3b5d29a6f15ac0",
"template_id" : "",
"pre_view" : "",
"cur_content" : {
  "id" : "f861d53ab668429ea537daa94e7d81d1",
  "content" : "文档内容",
  "version" : "",
  "name" : "",
  "can_create" : false,
  "can_share" : false,
  "can_delete" : false,
  "can_recover" : false,
  "can_rename" : false,
  "can_association" : false,
  "is_curr_user_admin" : false,
  "region_id" : "region-01",
  "tree_id" : "48408ac634154fa4ab3b5d29a6f15ac0",
  "creator_name" : "devcloud_docman_wx491639",
  "creator_id" : "78de0d103af7417aa9108644f564ede3",
  "create_time" : "2024-01-04 19:00:54",
  "editor_revs" : 200
},
"now_db_time" : null
}

```

状态码: 400

Bad request

```

{
  "error_code" : "ZHISHIKU.001002",
  "error_msg" : "参数错误",
  "error_details" : "参数错误: invalid type",
  "suggestion" : ""
}

```

状态码

状态码	描述
200	ok
400	Bad request

错误码

请参见[错误码](#)。

4.2 文件操作

4.2.1 获取文件库下的文件树

功能介绍

获取文件库下的文件树

URI

GET /v1/openapi/clouddrive/{zhishiku_id}/tree

表 4-5 路径参数

参数	是否必选	参数类型	描述
zhishiku_id	是	String	文件库id 最小长度：1 最大长度：64

请求参数

无

响应参数

状态码： 400

表 4-6 响应 Body 参数

参数	参数类型	描述
error_details	String	错误描述 最小长度：8 最大长度：36
suggestion	String	错误描述详情 最小长度：2 最大长度：512
error_code	String	错误编码 最小长度：1 最大长度：64
error_msg	String	错误信息 最小长度：1 最大长度：64

请求示例

示例中的d4f44ed652ea427da9b5137f8b0ce7a0为文件库id，直接拼接在在请求url中

```
/v1/openapi/clouddrive/d4f44ed652ea427da9b5137f8b0ce7a0/tree
```

响应示例

状态码： 200

ok

```
{
  "id": "p4kl0r32xp",
  "children": [ {
    "id": "1389a0e369b543789ca02c9a86e77ab8",
    "name": "232131",
    "type": 9,
    "permission": 2,
    "children": [ ],
    "compressPath": "",
    "efsPath": "",
    "previewType": 0,
    "convertedFilePath": "",
    "recycle": false,
    "can_create": false,
    "can_share": true,
    "can_delete": true,
    "can_recover": false,
    "can_rename": false,
    "can_association": false,
    "is_curr_user_admin": false,
    "region_id": "region-01",
    "domain_id": "f7285aa58f304ef49ac6667b902e3508",
    "creator_name": "devcloud_docman_wx491639",
    "creator_id": "78de0d103af7417aa9108644f564ede3",
    "create_time": "2023-11-27 21:09:34",
    "update_time": "2023-11-27 21:09:34",
  }
]
```

```

"updater_id" : "78de0d103af7417aa9108644f564ede3",
"updater_name" : "devcloud_docman_wx491639",
"order_no" : 524288,
"parent_id" : "p4kl0r32xp",
"parent_path" : "/p4kl0r32xp",
"likePath" : "/p4kl0r32xp/1389a0e369b543789ca02c9a86e77ab8",
"old_id" : 0,
"delete_flag" : 0,
"delete_time" : null,
"operate_time" : null,
"is_public" : "0",
"is_share" : 0,
"belong_id" : "p4kl0r32xp",
"zsk_belong_id" : "",
"belong_type" : 3,
"is_favourite" : "yes",
"invite_num" : 0,
"is_publish" : false,
"extend_attribute" : { },
"primitive_creator_name" : "",
"primitive_updater_name" : "",
"tree_id" : "",
"store_path" : "",
"file_source" : "",
"file_size" : 0,
"status" : 0,
"convert_status" : 0,
"convert_type" : "",
"file_type" : "",
"convert_time" : null,
"file_tree_id" : "",
"download_times" : 0,
"version_count" : 0,
"is_count_usage" : 0
}],
"isZhishiku" : true,
"can_create" : false,
"can_share" : false,
"can_delete" : false,
"can_recover" : false,
"can_rename" : false,
"can_association" : false,
"is_curr_user_admin" : true,
"region_id" : "region-01",
"domain_id" : "f7285aa58f304ef49ac6667b902e3508",
"name" : "IPD-自运营软件/云服务 类452",
"doc_count" : 3,
"file_count" : 0,
"creator_name" : "devcloud_docman_wx491639",
"primitive_creator_name" : "",
"creator_id" : "78de0d103af7417aa9108644f564ede3",
"update_time" : "2023-06-12 20:09:15",
"role_id" : "",
"role_name" : "",
"belong_id" : "3e53557ac840483985ce0e29e321d238",
"belong_type" : 3,
"belong_name" : "",
"is_public" : "0",
"delete_flag" : 0,
"delete_time" : null,
"is_favourite" : "",
"icon" : 1,
"is_share" : 0,
"team_belong_path" : "",
"team_belong_name" : "",
"privilege_json" : null,
"privilege_belong_id" : "p4kl0r32xp",
"extend_attribute" : {
  "doc_width" : "standard",

```

```

"enable_tag": "1",
"disable_copy": "1",
"enable_thumbs": "1",
"enable_comments": "1",
"enable_watermark": "1",
"enable_attachments": "1"
},
"sort_order": 0
}

```

状态码: 400

Bad request

```

{
"error_code": "ZHISHIKU.001002",
"error_msg": "参数错误",
"error_details": "参数错误: invalid type",
"suggestion": ""
}

```

状态码

状态码	描述
200	ok
400	Bad request

错误码

请参见[错误码](#)。

4.2.2 项目上传文件接口

功能介绍

项目上传文件接口

URI

POST /v1/openapi/prj/upload

请求参数

表 4-7 FormData 参数

参数	是否必选	参数类型	描述
file	是	File	上传的文件
project_id	是	String	项目id 最小长度: 1 最大长度: 64

参数	是否必选	参数类型	描述
parent_id	是	String	父目录id 最小长度：1 最大长度：64

响应参数

状态码：400

表 4-8 响应 Body 参数

参数	参数类型	描述
error_details	String	错误描述 最小长度：8 最大长度：36
suggestion	String	错误描述详情 最小长度：2 最大长度：512
error_code	String	错误编码 最小长度：1 最大长度：64
error_msg	String	错误信息 最小长度：1 最大长度：64

请求示例

project_id 为项目id； parent_id 为父目录id； file为要上传的文件；

```
/v1/openapi/prj/upload
{
  "file": "测试.txt",
  "project_id": "d4f44ed232ea427da9b5137f8b0ce7a0",
  "parent_id": "d4f44ed652ea427da9b5137f8b0ce7a0"
}
```

响应示例

状态码：200

OK

```
{
  "file_name": "dsdsds.txt",
```

```
"file_id" : "dsadfwqdwq"  
}
```

状态码: 400

Bad request

```
{  
  "error_code" : "ZHISHIKU.001002",  
  "error_msg" : "参数错误",  
  "error_details" : "参数错误: invalid type",  
  "suggestion" : ""  
}
```

状态码

状态码	描述
200	OK
400	Bad request

错误码

请参见[错误码](#)。

4.2.3 下载文件

功能介绍

下载文件

URI

GET /v1/openapi/file/download

表 4-9 Query 参数

参数	是否必选	参数类型	描述
id	是	String	要下载的文件id 最小长度: 1 最大长度: 64

请求参数

无

响应参数

状态码: 400

表 4-10 响应 Body 参数

参数	参数类型	描述
error_details	String	错误描述 最小长度: 8 最大长度: 36
suggestion	String	错误描述详情 最小长度: 2 最大长度: 512
error_code	String	错误编码 最小长度: 1 最大长度: 64
error_msg	String	错误信息 最小长度: 1 最大长度: 64

请求示例

请求下载id为167ff210a9474349b4c70cf5641f125b的文件, 其中167ff210a9474349b4c70cf5641f125b为文件id

```
/v1/openapi/file/download?id=167ff210a9474349b4c70cf5641f125b
```

响应示例

状态码: 200

OK

```
{}
```

状态码: 400

Bad request

```
{
  "error_code": "ZHISHIKU.001002",
  "error_msg": "参数错误",
  "error_details": "参数错误: invalid type",
  "suggestion": ""
}
```

状态码

状态码	描述
200	OK
400	Bad request

错误码

请参见[错误码](#)。

5 附录

5.1 获取项目 ID

调用 API 获取项目 ID

项目ID可以通过调用IAM的查询指定条件下的项目信息API获取。

获取项目ID的接口为“GET https://{Endpoint}/v3/projects/”，其中{Endpoint}为IAM的终端节点，可以从管理员处获取。接口的认证鉴权请参见[认证鉴权](#)。

响应示例如下，其中projects下的“id”即为项目ID。

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

从控制台获取项目 ID

在调用接口的时候，部分URL中需要填入项目编号，所以需要获取到项目编号。项目编号获取步骤如下：

1. 登录管理控制台。

2. 在页面右上角单击用户名，在下拉列表中选择“我的凭证”。
3. 在页面中找到“项目ID”

5.2 获取租户 ID

在调用接口的时候，部分URL中需要填入租户ID（tenant-id），所以需要先在管理控制台上获取到租户ID。租户ID获取步骤如下：

1. 登录管理控制台。
2. 在页面右上角单击用户名，在下拉列表中选择“我的凭证”。
3. 在页面中找到账号ID。

5.3 状态码

表 5-1 状态码

状态码	编码	状态码说明
100	Continue	继续请求。 这个临时响应用来通知客户端，它的部分请求已经被服务器接收，且仍未被拒绝。
101	Switching Protocols	切换协议。只能切换到更高级的协议。 例如，切换到HTTP的新版本协议。
200	OK	请求成功。
201	Created	创建类的请求完全成功。
202	Accepted	已经接受请求，但未处理完成。
203	Non-Authoritative Information	非授权信息，请求成功。
204	NoContent	请求完全成功，同时HTTP响应不包含响应体。 在响应OPTIONS方法的HTTP请求时返回此状态码。
205	Reset Content	重置内容，服务器处理成功。
206	Partial Content	服务器成功处理了部分GET请求。
300	Multiple Choices	多种选择。请求的资源可包括多个位置，相应可返回一个资源特征与地址的列表用于用户终端（例如：浏览器）选择。
301	Moved Permanently	永久移动，请求的资源已被永久的移动到新的URI，返回信息会包括新的URI。
302	Found	资源被临时移动。

状态码	编码	状态码说明
303	See Other	查看其它地址。 使用GET和POST请求查看。
304	Not Modified	所请求的资源未修改，服务器返回此状态码时，不会返回任何资源。
305	Use Proxy	所请求的资源必须通过代理访问。
306	Unused	已经被废弃的HTTP状态码。
400	BadRequest	非法请求。 建议直接修改该请求，不要重试该请求。
401	Unauthorized	在客户端提供认证信息后，返回该状态码，表明服务端指出客户端所提供的认证信息不正确或非法。
402	Payment Required	保留请求。
403	Forbidden	请求被拒绝访问。 返回该状态码，表明请求能够到达服务端，且服务端能够理解用户请求，但是拒绝做更多的事情，因为该请求被设置为拒绝访问，建议直接修改该请求，不要重试该请求。
404	NotFound	所请求的资源不存在。 建议直接修改该请求，不要重试该请求。
405	MethodNotAllowed	请求中带有该资源不支持的方法。 建议直接修改该请求，不要重试该请求。
406	Not Acceptable	服务器无法根据客户端请求的内容特性完成请求。
407	Proxy Authentication Required	请求要求代理的身份认证，与401类似，但请求者应当使用代理进行授权。
408	Request Time-out	服务器等候请求时发生超时。 客户端可以随时再次提交该请求而无需进行任何更改。
409	Conflict	服务器在完成请求时发生冲突。 返回该状态码，表明客户端尝试创建的资源已经存在，或者由于冲突请求的更新操作不能被完成。
410	Gone	客户端请求的资源已经不存在。 返回该状态码，表明请求的资源已被永久删除。
411	Length Required	服务器无法处理客户端发送的不带Content-Length的请求信息。
412	Precondition Failed	未满足前提条件，服务器未满足请求者在请求中设置的其中一个前提条件。

状态码	编码	状态码说明
413	Request Entity Too Large	由于请求的实体过大，服务器无法处理，因此拒绝请求。为防止客户端的连续请求，服务器可能会关闭连接。如果只是服务器暂时无法处理，则会包含一个Retry-After的响应信息。
414	Request-URI Too Large	请求的URI过长（URI通常为网址），服务器无法处理。
415	Unsupported Media Type	服务器无法处理请求附带的媒体格式。
416	Requested range not satisfiable	客户端请求的范围无效。
417	Expectation Failed	服务器无法满足Expect的请求头信息。
422	Unprocessable Entity	请求格式正确，但是由于含有语义错误，无法响应。
429	TooManyRequests	表明请求超出了客户端访问频率的限制或者服务端接收到多于它能处理的请求。建议客户端读取相应的Retry-After首部，然后等待该首部指出的时间后再重试。
500	InternalServerError	表明服务端能被请求访问到，但是不能理解用户的请求。
501	Not Implemented	服务器不支持请求的功能，无法完成请求。
502	Bad Gateway	充当网关或代理的服务器，从远端服务器接收到了一个无效的请求。
503	ServiceUnavailable	被请求的服务无效。 建议直接修改该请求，不要重试该请求。
504	ServerTimeout	请求在给定的时间内无法完成。客户端仅在为请求指定超时（Timeout）参数时会得到该响应。
505	HTTP Version not supported	服务器不支持请求的HTTP协议的版本，无法完成处理。

5.4 错误码

知识库错误码

表 5-2 错误码

状态码	错误码	错误信息	描述	处理措施
417	ZHISHIKU.010031	鉴权失败	非项目成员，没有操作权限	将该用户添加到项目空间

状态码	错误码	错误信息	描述	处理措施
417	ZHISHIKU .011014	上传失败	批量上传文件数量不能超过20个	建议减少同时上传的文件数量
417	ZHISHIKU .011014	上传失败	文件大小超出限制，不允许上传！	单个文件最大不超过1GB
417	ZHISHIKU .011001	文件不存在	文件不存在	请检查文件是否被删除
417	ZHISHIKU .003014	文档预览失败	文档不存在	请检查文档是否存在
417	ZHISHIKU .019001	导入文档失败	上传文件的格式与导入菜单选择不一致	导入正确格式的文件
417	ZHISHIKU .019001	导入文档失败	没有导入权限	联系管理员在设置页面添加导入权限
417	ZHISHIKU .003002	复制失败	复制前没有发布文档	编辑文档并发布后再复制
417	ZHISHIKU .003003	删除失败	没有删除权限	给用户配置删除权限
417	ZHISHIKU .003004	发布失败	标题长度超出100，请重新输入	控制标题长度在100以内
417	ZHISHIKU .003005	移动失败	目标目录存在相同名字的文件，请检查	修改文件名，不要在相同目录下有同名文件