

# 键值存储服务

## API 参考

文档版本 01  
发布日期 2026-01-06



版权所有 © 华为云计算技术有限公司 2026。保留一切权利。

非经本公司书面许可，任何单位和个人不得擅自摘抄、复制本文档内容的部分或全部，并不得以任何形式传播。

## 商标声明



HUAWEI和其他华为商标均为华为技术有限公司的商标。

本文档提及的其他所有商标或注册商标，由各自的所有人拥有。

## 注意

您购买的产品、服务或特性等应受华为云计算技术有限公司商业合同和条款的约束，本文档中描述的全部或部分产品、服务或特性可能不在您的购买或使用范围之内。除非合同另有约定，华为云计算技术有限公司对本文档内容不做任何明示或暗示的声明或保证。

由于产品版本升级或其他原因，本文档内容会不定期进行更新。除非另有约定，本文档仅作为使用指导，本文档中的所有陈述、信息和建议不构成任何明示或暗示的担保。

# 华为云计算技术有限公司

地址：贵州省贵安新区黔中大道交兴功路华为云数据中心 邮编：550029

网址：<https://www.huaweicloud.com/>

---

# 目录

---

<b>1 使用前必读</b>	<b>1</b>
1.1 概述	1
1.2 调用说明	1
1.3 终端节点	1
1.4 基本概念	2
<b>2 API 概览</b>	<b>4</b>
<b>3 如何调用 API</b>	<b>5</b>
3.1 构造请求	5
3.2 认证鉴权	8
3.3 返回结果	8
<b>4 API</b>	<b>10</b>
4.1 仓接口	10
4.1.1 列举仓	10
4.1.2 创建表	12
4.1.3 删除表	18
4.1.4 列举表	23
4.2 表接口	25
4.2.1 查询表	25
4.3 KV 接口	30
4.3.1 上传单个 kv	30
4.3.2 查询单个 kv	34
4.3.3 更新单个 kv	36
4.3.4 删除单个 kv	41
4.3.5 扫描所有 kv	44
4.3.6 扫描分区键内 kv	50
4.3.7 批量写请求	55
<b>5 附录</b>	<b>59</b>
5.1 状态码	59
5.2 错误码	60
5.3 获取项目 ID	64
5.4 获取账号 ID	64

# 1 使用前必读

## 1.1 概述

欢迎使用键值存储服务（Key-Value Storage Service, KVS）。键值存储服务提供完全托管的键值存储及索引服务，主要用于应用的键值类数据（如：元数据、描述数据、管理参数、状态数据）的存储，提供可预测的性能和无缝扩展，无需进行分区管理、硬件预置、集群扩展等操作。

您可以使用本文档提供API对键值存储服务进行相关操作，如创建、查询、删除、更新等。支持的全部操作请参见[API概览](#)。

在调用键值存储服务API之前，请确保已经充分了解键值存储服务相关概念，详细信息请参见《用户指南》的“[产品介绍](#)”。

## 1.2 调用说明

键值存储服务提供了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](#)。

同时键值存储服务还提供多种编程语言的SDK供您使用，SDK的使用方法请参见[SDK参考](#)。

## 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，您可以从[地区和终端节点](#)中查询所有服务的终端节点。

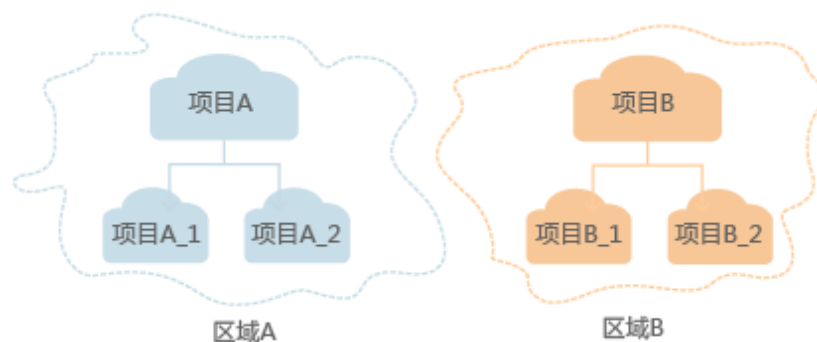
### 须知

用户购买VPC终端节点，创建内网域名，内网域名即用户使用API访问KVS过程中使用的终端节点。

## 1.4 基本概念

- 账号  
用户注册时的账号，账号对其所拥有的资源及云服务具有完全的访问权限，可以重置用户密码、分配用户权限等。由于账号是付费主体，为了确保账号安全，建议您不要直接使用账号进行日常管理工作，而是创建用户并使用用户进行日常管理工作。
- 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）。  
在[我的凭证](#)下，您可以查看账号ID和IAM用户ID。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。  
详情请参见[区域和可用区](#)。
- 可用区（AZ，Availability Zone）  
一个可用区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。
- 项目  
区域默认对应一个项目，这个项目由系统预置，用来隔离物理区域间的资源（计算资源、存储资源和网络资源），以默认项目为单位进行授权，用户可以访问您账号中该区域的所有资源。如果您希望进行更加精细的权限控制，可以在区域默认的项目中创建子项目，并在子项目中创建资源，然后以子项目为单位进行授权，使得用户仅能访问特定子项目中的资源，使得资源的权限控制更加精确。

图 1-1 项目隔离模型



同样在[我的凭证](#)下，您可以查看项目ID。

- 企业项目  
企业项目是项目的升级版，针对企业不同项目间的资源进行分组和管理，是逻辑隔离。企业项目中可以包含多个区域的资源，且项目中的资源可以迁入迁出。

关于企业项目ID的获取及企业项目特性的详细信息，请参见《[企业管理用户指南](#)》。

# 2 API 概览

表 2-1 接口说明

类型	子类型	说明
仓接口	列举仓	一个账户下可以创建最多25个仓，每个仓可以创建最多100个表，响应中一次性返回所有仓名称。
	创建表	在指定存储仓内创建表，表名在存储仓内唯一；创建表时，指定主键模板及本地二级索引模板及全局二级索引模板。
	列举表	指定存储仓列举创建的所有表。
表接口	查询表	指定存储仓查询表属性，如容量，规模，配额。
KV接口	上传单个kv	指定表，新建kv或覆盖已有kv，且满足表的key schema描述；允许指定条件执行。
	查询单个kv	下载一个kv文档的全部内容，或者部分字段的内容。
	更新单个kv	指定表，指定主键，指定更新文档的部分内容，如果是自描述文档，指定字段名；如果是二进制文档，指定偏移位置和长度；允许指定条件执行。
	删除单个kv	指定表，指定主键，删除该文档；允许指定条件执行。
	扫描所有kv	指定表，扫描表下所有kv；允许指定过滤条件。
	扫描分区键内kv	指定表及分区键，携带条件查询kv；允许指定过滤条件。
	批量写请求	批量写请求，其中可以携带一或多个表的不同kv的写操作，上传kv/删除kv。

# 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KVS服务的[创建表](#)来说明如何调用API，该API可以创建表。

### 请求 URI

请求URI由如下部分组成（方括号部分可选）：

**{URI-scheme}://[{store-name}]{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	请求使用的协议类型，如HTTP、HTTPS。HTTPS表示通过安全的HTTPS访问该资源，键值存储服务支持HTTP，HTTPS两种传输协议。
store-name	请求使用的仓名称，是可选部分，并不是每个API都有仓名称。存储仓的命名规范为{用户自定义前缀}-{KVS所在区域的区域名}-{用户的账号ID}，例如：exampleprefix-cn-north-4-exampledomainid。如何获取账号ID请参见 <a href="#">获取账号ID</a>
Endpoint	指定承载KVS服务端点的服务器域名或IP。Endpoint需要通过购买VPC终端节点，创建内网域名来获取，具体操作请参见 <a href="#">VPC终端节点对接KVS</a> 。
resource-path	资源路径，即API访问路径。从具体API的URI模块获取，例如“创建表”API接口的resource-path为“/v1/create-table”。

参数	描述
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“?store-name=exampleprefix-cn-north-4-exampldomainid”，表示仓名称为exampleprefix-cn-north-4-exampldomainid。

例如您需要在KVS的“华北-北京四”区域创建表，则需使用“华北-北京四”区域的Endpoint ( kvs.cn-north-4.myhuaweicloud.com )，并在具体API的URI部分找到resource-path ( /v1/create-table )，拼接起来如下所示。其中`{store-name}`为仓名。

`https://{store-name}.kvs.cn-north-4.myhuaweicloud.com/v1/create-table`

#### 说明

为方便查看，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），它告诉服务您正在请求什么类型的操作。

表 3-2 HTTP 方法

方法	说明
POST	请求服务器新增资源或执行特殊操作。

在具体API接口的URI部分，您可以看到其请求方法为“POST”，则其请求为：

`POST https://{store-name}.kvs.cn-north-4.myhuaweicloud.com/v1/create-table`

其中`{store-name}`为仓名。

## 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
Host	请求的服务器信息，从服务API的URL中获取。值为hostname[:port]。端口缺省时使用默认的端口，https的默认端口为443。	否 使用AK/SK认证时该字段必选。	code.test.com or code.test.com:443
Content-Type	消息体的类型（格式）。推荐用户使用默认值application/bson，有其他取值时会在具体接口中专门说明。	是	application/bson
Content-Length	请求body长度，单位为Byte。	否	3495
X-Project-Id	project id，项目编号。请参考 <a href="#">获取项目ID</a> 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。	e9993fc787d94b6c886cb aa340f9c0f4

### 说明

API同时支持使用AK/SK认证，AK/SK认证使用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[认证鉴权](#)的“AK/SK认证”。

## 请求消息体（可选）

该部分可选。请求消息体通常以Bson格式发出，与请求消息头中Content-Type对应，传递除请求消息头之外的内容。若请求消息体中的参数支持中文，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），列举仓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。

对于创建表接口，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，斜体字段需要根据实际值填写，其中\${store-name}为仓

名, \$YOUR\_TABLE\_NAME为表名, \$YOUR\_SHARD\_KEY\_NAME为分区键名称, \$YOUR\_SORT\_KEY\_NAME为排序键名称。

```
POST https://{store-name}.kvs.cn-north-4.myhuaweicloud.com/v1/create-table
Content-Type: application/bson

{
  "table_name": "$YOUR_TABLE_NAME",
  "primary_key_schema": {
    "shard_key_fields": [ {
      "name": "$YOUR_SHARD_KEY_NAME",
      "order": true
    } ],
    "sort_key_fields": [ {
      "name": "$YOUR_SORT_KEY_NAME",
      "order": true
    } ]
  }
}
```

到这里为止这个请求需要的内容就具备齐全了, 您可以直接编写代码发送请求调用 API。对于创建表接口, 返回的响应体中包含了创建表的属性信息。表创建好以后, 您就可以调用其他API执行向表中插入数据、查询表中数据等其他操作了。

## 3.2 认证鉴权

KVS现在仅支持AK/SK认证鉴权。

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

### AK/SK 认证

#### 📖 说明

AK/SK签名认证方式仅支持消息体大小在12MB以内。

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 ): 私有访问密钥。与访问密钥ID结合使用, 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。详细的签名方法和SDK使用方法请参见[API签名指南](#)。

#### 📖 说明

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。

## 3.3 返回结果

### 状态码

请求发送以后, 您会收到响应, 其中包含状态码、响应消息头和消息体。

状态码是一组从2xx（成功）到4xx或5xx（错误）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[创建表](#)接口，如果调用后返回状态码为“200”，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[创建表](#)接口，返回如[图3-1](#)所示的消息头。表创建好后，可以开始调用其他API接口执行向表中插入KV数据、查询KV数据等其他操作。

图 3-1 创建表响应消息头

```
server: UB5\r\n
Date: Wed, 22 May 2024 06:30:02 GMT\r\n
Content-Type: application/bson\r\n
> Content-Length: 577\r\n
Connection: keep-alive\r\n
X-Request-Id: 7608babc7dff0dfd16292a7750e6e0dc\r\n
```

## 响应消息体（可选）

该部分可选。响应消息体通常以Bson格式返回，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[创建表](#)接口，返回如下消息体。为篇幅起见，这里只展示部分内容。

```
{
  "table_name": "example-table",
  "primary_key_schema": {
    "shard_key_fields": [ {
      "name": "user_id",
      "order": true
    } ],
    "sort_key_fields": [ {
      "name": "user_name",
      "order": true
    } ]
  }
}
.....
```

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。

```
{
  "errorCode": KVS.00001001
  "errorMsg": table_name: string element has invalid char.
}
```

其中，errorCode表示错误码，errorMsg表示错误描述信息。

# 4 API

## 4.1 仓接口

### 4.1.1 列举仓

#### 功能介绍

一个账户下可以创建最多25个仓，每个仓可以创建最多100个store，响应中一次性返回所有仓名称。

#### URI

POST /v1/list-store

#### 请求参数

表 4-1 请求 Body 参数

参数	是否必选	参数类型	描述
cursor_name	否	String	上次返回的游标位置，本次响应包含该仓名。 <ul style="list-style-type: none"><li>长度：[16,52]</li><li>取值字符限制：[a-z0-9-]+</li></ul>
limit	否	Integer	响应返回的仓个数。

#### 响应参数

状态码：200

表 4-2 响应 Body 参数

参数	参数类型	描述
cursor_name	String	本次响应后的游标位置，下次请求时携带。 <ul style="list-style-type: none"> <li>长度：[16,52]</li> <li>取值字符限制：[a-z0-9-]+</li> </ul> <b>说明</b> 如果为空，表示后面无更多仓名。
stores	Array of strings	返回的仓名列表。

**状态码：400**

表 4-3 响应 Body 参数

参数	参数类型	描述
error_code	String	请求返回的错误码。
error_msg	String	请求返回的错误信息。

## 请求示例

列举用户账户下的仓，从指定仓名位置开始返回，限制返回仓名数量2个。

```
POST https://{endpoint}/v1/list-store
```

```
{
  "limit" : 2
}
```

## 响应示例

**状态码：200**

表示列举仓成功。

```
{
  "cursor_name" : "test-store-name-3",
  "store_name_list" : [ "test-store-name-1", "test-store-name-2" ]
}
```

## 状态码

状态码	描述
200	表示列举仓成功。
400	BadRequest

## 错误码

请参见[错误码](#)。

## 4.1.2 创建表

### 功能介绍

在指定仓内创建表，表名在仓内唯一；创建表时，指定主键模板及本地二级索引模板及全局二级索引模板。创建表时，如果仓不存在，将会自动创建仓。

### URI

POST /v1/create-table

表 4-4 Query 参数

参数	是否必选	参数类型	描述
store_name	否	String	仓名，全域唯一，不同租户的仓名不能相同。 <ul style="list-style-type: none"><li>格式：\${prefix}-\${region-code}-\${account-id}，其中 prefix 为自定义前缀，region-code 为 kvs 集群所在的区域代码，account-id 为用户的账户 id</li><li>取值字符限制：[a-z0-9-]+</li><li>长度：[16,52]</li></ul> <b>说明</b> "-"不能出现在名字头部或尾部

### 请求参数

表 4-5 请求 Body 参数

参数	是否必选	参数类型	描述
table_name	是	String	表名，仓内唯一。
bill_mode	否	String	表计费模式，可为"provisioned"或"on_demand" <ul style="list-style-type: none"><li>预置模式：provisioned</li><li>按需模式：on_demand</li></ul>
provisioned_throughput	否	<b>provisioned_throughput</b> object	预置吞吐读写请求单元数，在 bill_mode="provisioned"有效。

参数	是否必选	参数类型	描述
primary_key_schema	是	<b>primary_key_schema</b> object	主键模板, 最多1个, 必须指定主键schema。
local_secondary_index_schema	否	Array of <b>local_secondary_index</b> objects	本地二级索引模板, 可以多个。
global_secondary_index_schema	否	Array of <b>global_secondary_index</b> objects	全局二级索引模板。
pre_split_key_options	否	<b>pre_split_key_options</b> object	按照设定的分裂键值或者键值前缀进行预分裂。 <b>说明</b> range分区时该参数有效, 且只作用于主表, 其他场景无效。

表 4-6 primary\_key\_schema

参数	是否必选	参数类型	描述
shard_key_fields	是	Array of <b>field</b> objects	分区键字段名数组, 顺序组合。
shard_mode	否	String	分区模式。
sort_key_fields	否	Array of <b>field</b> objects	排序键字段名数组, 顺序组合。

表 4-7 local\_secondary\_index

参数	是否必选	参数类型	描述
index_name	是	String	二级索引名称, 表内唯一。
sort_key_fields	是	Array of <b>field</b> objects	排序键字段名数组, 顺序组合。
abstract_fields	否	Array of strings	摘要字段名数组。

表 4-8 global\_secondary\_index

参数	是否必选	参数类型	描述
index_name	是	String	二级索引名称，表内唯一。
shard_key_fields	是	Array of field objects	分区键字段名数组，顺序组合。
shard_mode	否	String	分区模式。
sort_key_fields	否	Array of field objects	排序键字段名数组，顺序组合。
abstract_fields	否	Array of strings	摘要字段名数组。
provisioned_throughput	否	provisioned_throughput object	预置吞吐读写请求单元数，在 bill_mode="provisioned"有效。

表 4-9 field

参数	是否必选	参数类型	描述
name	是	String	字段名。
order	否	Boolean	bool值预留无意义。

表 4-10 provisioned\_throughput

参数	是否必选	参数类型	描述
rcu	是	Long	预置表级读请求单元数。
wcu	是	Long	预置表级写请求单元数。

表 4-11 pre\_split\_key\_options

参数	是否必选	参数类型	描述
range_split_points	否	Array of strings	在range分区模式有效，最大10个。

## 响应参数

状态码：200

表 4-12 响应 Body 参数

参数	参数类型	描述
table_name	String	表名，仓内唯一。
bill_mode	String	计费模式，可为"provisioned"或"on_demand"。
provisioned_throughput	<b>provisioned_throughput</b> object	预置吞吐读写请求单元数，在bill_mode="provisioned"有效。
primary_key_schema	<b>primary_key_schema</b> object	主键模板，最多1个，必须指定主键schema。
local_secondary_index_schema	Array of <b>local_secondary_index</b> objects	本地二级索引模板，可以多个。
global_secondary_index_schema	Array of <b>global_secondary_index</b> objects	全局二级索引模板。
pre_split_key_options	<b>pre_split_key_options</b> object	按照设定的分裂键值或者键值前缀进行预分裂。 <b>说明</b> range分区时该参数有效，且只作用于主表，其他场景无效。

表 4-13 primary\_key\_schema

参数	参数类型	描述
shard_key_fields	Array of <b>field</b> objects	分区键字段名数组，顺序组合。
shard_mode	String	分区模式。
sort_key_fields	Array of <b>field</b> objects	排序键字段名数组，顺序组合。

表 4-14 local\_secondary\_index

参数	参数类型	描述
index_name	String	二级索引名称，表内唯一。
sort_key_fields	Array of <b>field</b> objects	排序键字段名数组，顺序组合。
abstract_fields	Array of strings	摘要字段名数组。

表 4-15 global\_secondary\_index

参数	参数类型	描述
index_name	String	二级索引名称，表内唯一。
shard_key_fields	Array of <b>field</b> objects	分区键字段名数组，顺序组合。
shard_mode	String	分区模式。
sort_key_fields	Array of <b>field</b> objects	排序键字段名数组，顺序组合。
abstract_fields	Array of strings	摘要字段名数组。
provisioned_throughput	<b>provisioned_throughput</b> object	预置吞吐读写请求单元数，在 bill_mode="provisioned"有效。

表 4-16 field

参数	参数类型	描述
name	String	字段名。
order	Boolean	bool值预留无意义。

表 4-17 provisioned\_throughput

参数	参数类型	描述
rcu	Long	预置表级读请求单元数。
wcu	Long	预置表级写请求单元数。

表 4-18 pre\_split\_key\_options

参数	参数类型	描述
range_split_points	Array of strings	在range分区模式有效，最大10个。

状态码：400

表 4-19 响应 Body 参数

参数	参数类型	描述
error_code	String	请求返回的错误码。

参数	参数类型	描述
error_msg	String	请求返回的错误信息。

**状态码：409**

**表 4-20 响应 Body 参数**

参数	参数类型	描述
error_code	String	请求返回的错误码。
error_msg	String	请求返回的错误信息。

## 请求示例

在test-store仓内，创建表，表名为test-table，分区键字段名为owner，排序键字段名称为filename

POST https://test-store.{endpoint}/v1/create-table

```
{
  "table_name": "test-table",
  "primary_key_schema": {
    "shard_key_fields": [ {
      "name": "owner",
      "order": true
    } ],
    "sort_key_fields": [ {
      "name": "filename",
      "order": true
    } ]
  }
}
```

## 响应示例

**状态码：200**

表示创建表成功。

```
{
  "table_name": "test-table",
  "primary_key_schema": {
    "shard_key_fields": [ {
      "name": "owner",
      "order": true
    } ],
    "sort_key_fields": [ {
      "name": "filename",
      "order": true
    } ]
  }
}
```

## 状态码

状态码	描述
200	表示创建表成功。
400	BadRequest
409	表已存在

## 错误码

请参见[错误码](#)。

### 4.1.3 删除表

#### 功能介绍

DeleteTable操作删除表及其所有项。在DeleteTable请求之后，指定的表一直处于DELETING状态，直到KVS完成删除。

#### URI

POST /v1/delete-table

表 4-21 Query 参数

参数	是否必选	参数类型	描述
store_name	否	String	<p>仓名，全域唯一，不同租户的仓名不能相同。</p> <ul style="list-style-type: none"> <li>格式：\${prefix}-\${region-code}-\${account-id}，其中prefix为自定义前缀，region-code为KVS集群所在的区域代码，account-id为用户的账户id</li> <li>取值字符限制：[a-z0-9-]+</li> <li>长度：[16,52]</li> </ul> <p><b>说明</b> "."不能出现在名字头部或尾部</p>

## 请求参数

表 4-22 请求 Body 参数

参数	是否必选	参数类型	描述
table_name	是	String	表名。 <ul style="list-style-type: none"> <li>长度: [3, 52]</li> <li>取值字符限制: [a-zA-Z0-9_-]+</li> </ul>

## 响应参数

状态码: 200

表 4-23 响应 Body 参数

参数	参数类型	描述
table_name	String	表名。 <ul style="list-style-type: none"> <li>长度: [3, 52]</li> <li>取值字符限制: [a-z0-9_-]+</li> </ul>
primary_key_schema	<b>primary_key_schema</b> object	主键模板, 最多1个, 必须指定主键 schema。
local_secondary_index_schema	Array of <b>local_secondary_index</b> objects	本地二级索引模板, 可以多个。
global_secondary_index_schema	Array of <b>global_secondary_index</b> objects	全局二级索引模板。
run_time_info	<b>run_time_info</b> object	运行信息。

表 4-24 primary\_key\_schema

参数	参数类型	描述
shard_key_fields	Array of <b>field</b> objects	分区键字段名数组, 顺序组合。
shard_mode	String	分区模式。
sort_key_fields	Array of <b>field</b> objects	排序键字段名数组, 顺序组合。

表 4-25 local\_secondary\_index

参数	参数类型	描述
index_name	String	二级索引名称，表内唯一。
sort_key_fields	Array of <b>field</b> objects	排序键字段名数组，顺序组合。
abstract_fields	Array of strings	摘要字段名数组。

表 4-26 global\_secondary\_index

参数	参数类型	描述
index_name	String	二级索引名称，表内唯一。
shard_key_fields	Array of <b>field</b> objects	分区键字段名数组，顺序组合。
shard_mode	String	分区模式。
sort_key_fields	Array of <b>field</b> objects	排序键字段名数组，顺序组合。
abstract_fields	Array of strings	摘要字段名数组。
provisioned_throughput	<b>provisioned_throughput</b> object	预置吞吐读写请求单元数，在 bill_mode="provisioned"有效。

表 4-27 field

参数	参数类型	描述
name	String	字段名。
order	Boolean	bool值预留无意义。

表 4-28 provisioned\_throughput

参数	参数类型	描述
rcu	Long	预置表级读请求单元数。
wcu	Long	预置表级写请求单元数。

表 4-29 run\_time\_info

参数	参数类型	描述
table_info	<b>table_info</b> object	表信息。
local_secondary_index_infos	Array of <b>secondary_index_info</b> objects	索引状态。
global_secondary_index_infos	Array of <b>global_secondary_index_info</b> objects	全局二级索引运行态。

表 4-30 table\_info

参数	参数类型	描述
table_status	String	表状态。 <ul style="list-style-type: none"><li>• "creating"</li><li>• "active"</li><li>• "deleting"</li></ul>

表 4-31 secondary\_index\_info

参数	参数类型	描述
index_name	String	索引状态。 <ul style="list-style-type: none"><li>• 长度: [1, 255]</li><li>• 取值字符限制: [a-z0-9_-]+</li></ul>
index_status	String	索引状态。 <ul style="list-style-type: none"><li>• "creating"</li><li>• "active"</li><li>• "deleting"</li></ul>

表 4-32 global\_secondary\_index\_info

参数	参数类型	描述
index_name	String	二级索引名称。

参数	参数类型	描述
index_status	String	二级索引名称。 <ul style="list-style-type: none"><li>"creating"</li><li>"active"</li><li>"deleting"</li></ul>

**状态码：404****表 4-33** 响应 Body 参数

参数	参数类型	描述
error_code	String	请求返回的错误码。
error_msg	String	请求返回的错误信息。

**状态码：409****表 4-34** 响应 Body 参数

参数	参数类型	描述
error_code	String	请求返回的错误码。
error_msg	String	请求返回的错误信息。

## 请求示例

在test-store仓内，删除表名为test-table的表

```
POST https://test-store.{endpoint}/v1/delete-table
```

```
{  
  "table_name": "test-table"  
}
```

## 响应示例

**状态码：200**

表示删除表成功。

```
{  
  "table_name": "test-table",  
  "primary_key_schema": {  
    "shard_key_fields": [ {  
      "name": "owner",  
      "order": true  
    } ],  
    "sort_key_fields": [ {
```

```
"name": "filename",  
"order": true  
}]  
}  
}
```

## 状态码

状态码	描述
200	表示删除表成功。
404	表不存在。
409	表请求冲突。

## 错误码

请参见[错误码](#)。

### 4.1.4 列举表

#### 功能介绍

指定仓列举创建的所有表。

#### URI

POST /v1/list-table

表 4-35 Query 参数

参数	是否必选	参数类型	描述
store_name	否	String	<p>仓名，全域唯一，不同租户的仓名不能相同。</p> <ul style="list-style-type: none"><li>格式：\${prefix}-\${region-code}-\${account-id}，其中 prefix 为自定义前缀，region-code 为 kvs 集群所在的区域代码，account-id 为用户的账户 id</li><li>取值字符限制：[a-z0-9-]+</li><li>长度：[16,52]</li></ul> <p><b>说明</b> "-"不能出现在名字头部或尾部</p>

## 请求参数

表 4-36 请求 Body 参数

参数	是否必选	参数类型	描述
cursor_name	否	String	上次返回游标位置，本次响应包含该table，空表示遍历完。 <ul style="list-style-type: none"><li>长度：[3, 63]</li><li>取值字符限制：[a-z0-9_-]+</li></ul> <b>说明</b> 如果为空，表示后面无更多。
limit	否	Integer	响应返回的表个数。 <ul style="list-style-type: none"><li>长度：最大100</li></ul>

## 响应参数

状态码：200

表 4-37 响应 Body 参数

参数	参数类型	描述
cursor_name	String	本次响应后的游标位置，下次请求时携带，如果为空，表示后面无更多。 <ul style="list-style-type: none"><li>长度：[3, 63]</li><li>取值字符限制：[a-z0-9_-]+</li></ul> <b>说明</b> 如果为空，表示后面无更多。
table_names	Array of strings	返回的表名列表。 <ul style="list-style-type: none"><li>长度：最大100</li></ul>

状态码：400

表 4-38 响应 Body 参数

参数	参数类型	描述
error_code	String	请求返回的错误码。
error_msg	String	请求返回的错误信息。

## 请求示例

列举指定仓中的表，限制返回表数量为2

```
POST https://{endpoint}/v1/list-table
```

```
{  
  "limit" : 2  
}
```

## 响应示例

**状态码: 200**

表示列举表请求成功

```
{  
  "cursor_name" : "test-table-name_3",  
  "table_name_list" : [ "test-table-name_1", "test-table-name_2" ]  
}
```

## 状态码

状态码	描述
200	表示列举表请求成功
400	BadRequest

## 错误码

请参见[错误码](#)。

## 4.2 表接口

### 4.2.1 查询表

#### 功能介绍

指定仓查询表属性，如容量，规模，配额。

#### URI

```
POST /v1/describe-table
```

表 4-39 Query 参数

参数	是否必选	参数类型	描述
store_name	否	String	<p>仓名，全域唯一，不同租户的仓名不能相同。</p> <ul style="list-style-type: none"> <li>格式：\${prefix}-\${region-code}-\${account-id}，其中 prefix 为自定义前缀，region-code 为 kvs 集群所在的区域代码，account-id 为用户的账户 id</li> <li>取值字符限制：[a-z0-9-]+</li> <li>长度：[16,52]</li> </ul> <p><b>说明</b> "." 不能出现在名字头部或尾部</p>

## 请求参数

表 4-40 请求 Body 参数

参数	是否必选	参数类型	描述
table_name	是	String	<p>表名。</p> <ul style="list-style-type: none"> <li>长度：[3, 52]</li> <li>取值字符限制：[a-zA-Z0-9_-]+</li> </ul>

## 响应参数

状态码：200

表 4-41 响应 Body 参数

参数	参数类型	描述
table_name	String	<p>表名。</p> <ul style="list-style-type: none"> <li>长度：[3, 52]</li> <li>取值字符限制：[a-z0-9_-]+</li> </ul>
primary_key_schema	<a href="#">primary_key_schema</a> object	主键模板，最多1个，必须指定主键 schema。
local_secondary_index_schema	Array of <a href="#">local_secondary_index</a> objects	本地二级索引模板，可以多个。

参数	参数类型	描述
global_secondary_index_schema	Array of <b>global_secondary_index</b> objects	全局二级索引模板。
run_time_info	<b>run_time_info</b> object	运行信息。

表 4-42 primary\_key\_schema

参数	参数类型	描述
shard_key_fields	Array of <b>field</b> objects	分区键字段名数组，顺序组合。
shard_mode	String	分区模式。
sort_key_fields	Array of <b>field</b> objects	排序键字段名数组，顺序组合。

表 4-43 local\_secondary\_index

参数	参数类型	描述
index_name	String	二级索引名称，表内唯一。
sort_key_fields	Array of <b>field</b> objects	排序键字段名数组，顺序组合。
abstract_fields	Array of strings	摘要字段名数组。

表 4-44 global\_secondary\_index

参数	参数类型	描述
index_name	String	二级索引名称，表内唯一。
shard_key_fields	Array of <b>field</b> objects	分区键字段名数组，顺序组合。
shard_mode	String	分区模式。
sort_key_fields	Array of <b>field</b> objects	排序键字段名数组，顺序组合。
abstract_fields	Array of strings	摘要字段名数组。
provisioned_throughput	<b>provisioned_throughput</b> object	预置吞吐读写请求单元数，在 bill_mode="provisioned"有效。

表 4-45 field

参数	参数类型	描述
name	String	字段名。
order	Boolean	bool值预留无意义。

表 4-46 provisioned\_throughput

参数	参数类型	描述
rcu	Long	预置表级读请求单元数。
wcu	Long	预置表级写请求单元数。

表 4-47 run\_time\_info

参数	参数类型	描述
table_info	<a href="#">table_info</a> object	表信息。
local_secondary_index_infos	Array of <a href="#">secondary_index_info</a> objects	索引状态。
global_secondary_index_infos	Array of <a href="#">global_secondary_index_info</a> objects	全局二级索引运行态。

表 4-48 table\_info

参数	参数类型	描述
table_status	String	表状态。 <ul style="list-style-type: none"> <li>creating (表示该表正在创建中)</li> <li>active (表示该表处于正常运行状态)</li> <li>deleting (表示该表处于删除中状态, 或者已经被删除)</li> </ul>

表 4-49 secondary\_index\_info

参数	参数类型	描述
index_name	String	索引状态。 <ul style="list-style-type: none"><li>长度: [1, 255]</li><li>取值字符限制: [a-z0-9_-]+</li></ul>
index_status	String	索引状态。 <ul style="list-style-type: none"><li>creating (表示该索引正在创建中)</li><li>active (表示该索引处于正常运行状态)</li><li>deleting (表示该索引处于删除中状态, 或者已经被删除)</li></ul>

表 4-50 global\_secondary\_index\_info

参数	参数类型	描述
index_name	String	二级索引名称。
index_status	String	二级索引名称。 <ul style="list-style-type: none"><li>creating (表示该全局二级索引正在创建中)</li><li>active (表示该全局二级索引处于正常运行状态)</li><li>deleting (表示该全局二级索引处于删除中状态, 或者已经被删除)</li></ul>

状态码: 400

表 4-51 响应 Body 参数

参数	参数类型	描述
error_code	String	请求返回的错误码。
error_msg	String	请求返回的错误信息。

## 请求示例

在test-store仓内, 查询表名为test-table的表信息

```
POST https://test-store.{endpoint}/v1/describe-table
```

```
{  
  "table_name": "test-table"  
}
```

## 响应示例

### 状态码：200

表示查询表请求成功

```
{
  "table_name": "test-table-1",
  "primary_key_schema": {
    "shard_key_fields": [ {
      "name": "owner",
      "order": true
    } ],
    "sort_key_fields": [ {
      "name": "filename",
      "order": true
    } ]
  },
  "runtime_info": {
    "table_info": {
      "table_status": "ACTIVE"
    }
  }
}
```

## 状态码

状态码	描述
200	表示查询表请求成功
400	BadRequest

## 错误码

请参见[错误码](#)。

## 4.3 KV 接口

### 4.3.1 上传单个 kv

#### 功能介绍

指定表，新建kv或覆盖已有kv，且满足表的key schema描述；允许指定条件执行。

#### URI

POST /v1/put-kv

表 4-52 Query 参数

参数	是否必选	参数类型	描述
store_name	否	String	<p>仓名，全域唯一，不同租户的仓名不能相同。</p> <ul style="list-style-type: none"> <li>格式：\${prefix}-\${region-code}-\${account-id}，其中 prefix 为自定义前缀，region-code 为 kvs 集群所在的区域代码，account-id 为用户的账户 id</li> <li>取值字符限制：[a-z0-9-]+</li> <li>长度：[16,52]</li> </ul> <p><b>说明</b> "-"不能出现在名字头部或尾部</p>

## 请求参数

表 4-53 请求 Body 参数

参数	是否必选	参数类型	描述
table_name	是	String	<p>表名，仓内唯一。</p> <ul style="list-style-type: none"> <li>长度：[3, 52]</li> <li>取值字符限制：[a-z0-9_-]+</li> </ul>
condition_expression	否	<a href="#">condition_expression</a> object	<p>条件表达式，参见 conditional 小节。</p> <ul style="list-style-type: none"> <li>长度：[8,4096]</li> </ul>
kv_doc	否	Document	用户文档。

表 4-54 condition\_expression

参数	是否必选	参数类型	描述
single_field_expression	否	<a href="#">single_field_expression</a> object	单字段条件。
multi_field_expression	否	<a href="#">multi_field_expression</a> object	多字段条件，多个相同优先级的单字段条件。
composed_expression	否	<a href="#">composed_expression</a> object	组合条件，逐层同级的子表达式嵌套。

表 4-55 composed\_expression

参数	是否必选	参数类型	描述
logic	否	String	逻辑关系，取值如"\$and", "\$or", "\$nor"。
expressions	是	Array of <a href="#">expression objects</a>	多个相同优先级且相同逻辑的单字段或多字段条件。

表 4-56 expression

参数	是否必选	参数类型	描述
single_field_expression	否	<a href="#">single_field_expression object</a>	单字段条件。
multi_field_expression	否	<a href="#">multi_field_expression object</a>	多字段条件，多个相同优先级的单字段条件。
composed_expression	否	<a href="#">composed_expression object</a>	组合条件，逐层同级的子表达式嵌套。

表 4-57 single\_field\_expression

参数	是否必选	参数类型	描述
field	是	String	条件字段。
func	是	String	条件函数，取值如"\$gt", "\$lt", "\$gte", "\$lte", "\$eq", "\$ne", "\$prefix", "\$exists"。
value	否	Document	value和value_array二选一。 <ul style="list-style-type: none"><li>value条件值，适用于除"\$in", "\$nin"外的func。</li><li>字段名无意义，可以传空，也可以传字段名。</li><li>\$exists值为true/false。</li></ul> <b>说明</b> \$prefix操作只适用于string和binary类型。

参数	是否必选	参数类型	描述
value_array	否	Array of strings	"value"和"value_array"二选一。 <ul style="list-style-type: none"> <li>"value_array" 条件值列表, 值用于"\$in", "\$nin"。</li> </ul>

表 4-58 multi\_field\_expression

参数	是否必选	参数类型	描述
logic	否	String	多字段条件, 多个相同优先级的单字段条件。
expressions	是	Array of <a href="#">single_field_expression</a> objects	多个相同逻辑操作的单字段条件。

## 响应参数

**状态码: 200**

表示上传单个kv请求成功

**状态码: 400**

表 4-59 响应 Body 参数

参数	参数类型	描述
error_code	String	请求返回的错误码。
error_msg	String	请求返回的错误信息。

## 请求示例

上传单个kv, 表名为test-table-1, 仓名为test-store。

```
POST https://test-store.{endpoint}/v1/put-kv
```

```
{
  "table_name": "test-table-1",
  "kv_doc": {
    "owner": "user-1",
    "filename": "test-file-1",
    "otherKey": "other"
  }
}
```

## 响应示例

无

## 状态码

状态码	描述
200	表示上传单个kv请求成功
400	BadRequest

## 错误码

请参见[错误码](#)。

## 4.3.2 查询单个 kv

### 功能介绍

下载一个kv文档的全部内容，或者部分字段的内容。

### URI

POST /v1/get-kv

表 4-60 Query 参数

参数	是否必选	参数类型	描述
store_name	否	String	<p>仓名，全域唯一，不同租户的仓名不能相同。</p> <ul style="list-style-type: none"><li>格式：\${prefix}-\${region-code}-\${account-id}，其中 prefix 为自定义前缀，region-code 为 kvs 集群所在的区域代码，account-id 为用户的账户 id</li><li>取值字符限制：[a-z0-9-]+</li><li>长度：[16,52]</li></ul> <p><b>说明</b> "-"不能出现在名字头部或尾部</p>

## 请求参数

表 4-61 请求 Body 参数

参数	是否必选	参数类型	描述
table_name	是	String	表名，仓内唯一。 <ul style="list-style-type: none"><li>长度：[3, 52]</li><li>取值字符限制：[a-z0-9_-]+</li></ul>
primary_key	是	Document	用户自定义的主键名及值。

## 响应参数

状态码：200

表 4-62 响应 Body 参数

参数	参数类型	描述
kv_doc	Document	对kv_doc有效。 <b>说明</b> 内容字段：主键字段+投影字段或者全部字段。

状态码：400

表 4-63 响应 Body 参数

参数	参数类型	描述
error_code	String	请求返回的错误码。
error_msg	String	请求返回的错误信息。

## 请求示例

下载一个kv，表名为test-table-1，仓名为test-store，设置分区键值为user1，排序键值为test-file-1。

```
POST https://test-store.{endpoint}/v1/get-kv
{
  "table_name": "test-table-1",
  "primary_key": {
    "owner": "user-1",
    "filename": "test-file-1"
  }
}
```

## 响应示例

### 状态码：200

表示查询单个kv请求成功

```
{
  "kv_doc": {
    "owner": "user-1",
    "filename": "test-file-1",
    "otherKey": "otherValue"
  }
}
```

## 状态码

状态码	描述
200	表示查询单个kv请求成功
400	BadRequest

## 错误码

请参见[错误码](#)。

### 4.3.3 更新单个 kv

#### 功能介绍

指定表，指定主键，指定更新文档的部分内容，如果是自描述文档，指定字段名；如果是二进制文档，指定偏移位置和长度；允许指定条件执行。

#### URI

POST /v1/update-kv

表 4-64 Query 参数

参数	是否必选	参数类型	描述
store_name	否	String	<p>仓名，全域唯一，不同租户的仓名不能相同。</p> <ul style="list-style-type: none"> <li>格式：\${prefix}-\${region-code}-\${account-id}，其中 prefix 为自定义前缀，region-code 为 kvs 集群所在的区域代码，account-id 为用户的账户 id</li> <li>取值字符限制：[a-z0-9-]+</li> <li>长度：[16,52]</li> </ul> <p><b>说明</b> "-"不能出现在名字头部或尾部</p>

## 请求参数

表 4-65 请求 Body 参数

参数	是否必选	参数类型	描述
table_name	是	String	<p>表名，仓内唯一。</p> <ul style="list-style-type: none"> <li>长度：[3, 52]</li> <li>取值字符限制：[a-z0-9_-]+</li> </ul>
primary_key	是	Document	<p>用户自定义的主键名及值。</p> <p><b>说明</b> 内容字段：主键字段名和值，组合索引多个元素。</p>
condition_expression	否	<a href="#">condition_expression</a> object	<p>条件表达式，参见 conditional 小节，禁止携带主键字段。</p> <ul style="list-style-type: none"> <li>长度：[8,4096]</li> </ul>
update_fields	否	<a href="#">update_fields</a> object	对 kv_doc 有效。

表 4-66 condition\_expression

参数	是否必选	参数类型	描述
single_field_expression	否	<a href="#">single_field_expression</a> object	单字段条件。

参数	是否必选	参数类型	描述
multi_field_expression	否	<a href="#">multi_field_expression</a> object	多字段条件，多个相同优先级的单字段条件。
composed_expression	否	<a href="#">composed_expression</a> object	组合条件，逐层同级的子表达式嵌套。

表 4-67 composed\_expression

参数	是否必选	参数类型	描述
logic	否	String	逻辑关系，取值如"\$and", "\$or", "\$nor"。
expressions	是	Array of <a href="#">expression</a> objects	多个相同优先级且相同逻辑的单字段或多字段条件。

表 4-68 expression

参数	是否必选	参数类型	描述
single_field_expression	否	<a href="#">single_field_expression</a> object	单字段条件。
multi_field_expression	否	<a href="#">multi_field_expression</a> object	多字段条件，多个相同优先级的单字段条件。
composed_expression	否	<a href="#">composed_expression</a> object	组合条件，逐层同级的子表达式嵌套。

表 4-69 single\_field\_expression

参数	是否必选	参数类型	描述
field	是	String	条件字段。
func	是	String	条件函数，取值如"\$gt", "\$lt", "\$gte", "\$lte", "\$eq", "\$ne", "\$prefix", "\$exists"。

参数	是否必选	参数类型	描述
value	否	Document	value和value_array二选一。 <ul style="list-style-type: none"> <li>value条件值，适用于除"\$in", "\$nin"外的func。</li> <li>字段名无意义，可以传空，也可以传字段名。</li> <li>\$exists值为true/false。</li> </ul> <b>说明</b> \$prefix操作只适用于string和binary类型。
value_array	否	Array of strings	"value"和"value_array"二选一。 <ul style="list-style-type: none"> <li>"value_array" 条件值列表，值用于"\$in", "\$nin"。</li> </ul>

表 4-70 multi\_field\_expression

参数	是否必选	参数类型	描述
logic	否	String	多字段条件，多个相同优先级的单字段条件。
expressions	是	Array of <a href="#">single_field_expression</a> objects	多个相同逻辑操作的单字段条件。

表 4-71 update\_fields

参数	是否必选	参数类型	描述
set	否	Document	新增字段或覆盖1个或多个字段的值。 <b>说明</b> 禁止修改sortkey的字段。
add	否	Document	对1个或多个字段做加法运算，并更新为运算后的值。
rmv	否	Array of strings	删除1个或多个字段。 <ul style="list-style-type: none"> <li>数组元素为待删除字段名。</li> </ul>
insert	否	Document	插入元素到数组中 <b>说明</b> 非数组返回失败。

参数	是否必选	参数类型	描述
delete	否	Document	从数组中删除某个元素 <b>说明</b> 值不存在则返回失败。

## 响应参数

**状态码：200**

表示更新单个kv请求成功

**状态码：400**

表 4-72 响应 Body 参数

参数	参数类型	描述
error_code	String	请求返回的错误码。
error_msg	String	请求返回的错误信息。

## 请求示例

更新单个kv，设置表名为test-table-1，仓名为test-store，设置更新的kv分区键值为user1，排序键为test-file-1，将updateKey字段的值更新为updateValue。

POST https://test-store.{endpoint}/v1/update-kv

```
{
  "table_name": "test-table-1",
  "primary_key": {
    "owner": "user-1",
    "filename": "test-file-1"
  },
  "update_fields": {
    "set": {
      "updateKey": "updateValue"
    }
  }
}
```

## 响应示例

无

## 状态码

状态码	描述
200	表示更新单个kv请求成功
400	BadRequest

## 错误码

请参见[错误码](#)。

### 4.3.4 删除单个 kv

#### 功能介绍

指定表，指定主键，删除该文档；允许指定条件执行。

#### URI

POST /v1/delete-kv

表 4-73 Query 参数

参数	是否必选	参数类型	描述
store_name	否	String	<p>仓名，全域唯一，不同租户的仓名不能相同。</p> <ul style="list-style-type: none"> <li>格式：\${prefix}-\${region-code}-\${account-id}，其中 prefix 为自定义前缀，region-code 为 kvs 集群所在的区域代码，account-id 为用户的账户 id</li> <li>取值字符限制：[a-z0-9-]+</li> <li>长度：[16,52]</li> </ul> <p><b>说明</b> "-"不能出现在名字头部或尾部</p>

#### 请求参数

表 4-74 请求 Body 参数

参数	是否必选	参数类型	描述
table_name	是	String	<p>表名，仓内唯一。</p> <ul style="list-style-type: none"> <li>长度：[3, 52]</li> <li>取值字符限制：[a-z0-9_-]+</li> </ul>
primary_key	是	Document	<p>用户自定义的主键名及值。</p> <p><b>说明</b> 内容字段为主键字段名和值，组合索引多个元素。</p>

参数	是否必选	参数类型	描述
condition_expression	否	<a href="#">condition_expression</a> object	条件表达式，参见conditional小节。 <b>说明</b> 禁止携带主键字段，参见conditional小节的定义。 <ul style="list-style-type: none"> <li>长度：[8,4096]</li> </ul>

表 4-75 condition\_expression

参数	是否必选	参数类型	描述
single_field_expression	否	<a href="#">single_field_expression</a> object	单字段条件。
multi_field_expression	否	<a href="#">multi_field_expression</a> object	多字段条件，多个相同优先级的单字段条件。
composed_expression	否	<a href="#">composed_expression</a> object	组合条件，逐层同级的子表达式嵌套。

表 4-76 composed\_expression

参数	是否必选	参数类型	描述
logic	否	String	逻辑关系，取值如"\$and", "\$or", "\$nor"。
expressions	是	Array of <a href="#">expression</a> objects	多个相同优先级且相同逻辑的单字段或多字段条件。

表 4-77 expression

参数	是否必选	参数类型	描述
single_field_expression	否	<a href="#">single_field_expression</a> object	单字段条件。
multi_field_expression	否	<a href="#">multi_field_expression</a> object	多字段条件，多个相同优先级的单字段条件。

参数	是否必选	参数类型	描述
composed_expression	否	<a href="#">composed_expression</a> object	组合条件，逐层同级的子表达式嵌套。

表 4-78 single\_field\_expression

参数	是否必选	参数类型	描述
field	是	String	条件字段。
func	是	String	条件函数，取值如"\$gt", "\$lt", "\$gte", "\$lte", "\$eq", "\$ne", "\$prefix", "\$exists"。
value	否	Document	value和value_array二选一。 <ul style="list-style-type: none"> <li>value条件值，适用于除"\$in", "\$nin"外的func。</li> <li>字段名无意义，可以传空，也可以传字段名。</li> <li>\$exists值为true/false。</li> </ul> <b>说明</b> \$prefix操作只适用于string和binary类型。
value_array	否	Array of strings	"value"和"value_array"二选一。 <ul style="list-style-type: none"> <li>"value_array" 条件值列表，值用于"\$in", "\$nin"。</li> </ul>

表 4-79 multi\_field\_expression

参数	是否必选	参数类型	描述
logic	否	String	多字段条件，多个相同优先级的单字段条件。
expressions	是	Array of <a href="#">single_field_expression</a> objects	多个相同逻辑操作的单字段条件。

## 响应参数

状态码：200

表示删除单个kv请求成功

### 状态码：400

表 4-80 响应 Body 参数

参数	参数类型	描述
error_code	String	请求返回的错误码。
error_msg	String	请求返回的错误信息。

### 请求示例

删除单个kv，删除kv的分区键为user，排序键为test-file-1，表名为test-table-1,仓名为test-store。

```
POST https://test-store.{endpoint}/v1/delete-kv
{
  "table_name": "test-table-1",
  "primary_key": {
    "owner": "user-1",
    "filename": "test-file-1"
  }
}
```

### 响应示例

无

### 状态码

状态码	描述
200	表示删除单个kv请求成功
400	BadRequest

### 错误码

请参见[错误码](#)。

## 4.3.5 扫描所有 kv

### 功能介绍

指定表，扫描表下所有kv；允许指定过滤条件。

### URI

POST /v1/scan-kv

表 4-81 Query 参数

参数	是否必选	参数类型	描述
store_name	否	String	<p>仓名，全域唯一，不同租户的仓名不能相同。</p> <ul style="list-style-type: none"> <li>格式：\${prefix}-\${region-code}-\${account-id}，其中 prefix 为自定义前缀，region-code 为 kvs 集群所在的区域代码，account-id 为用户的账户 id</li> <li>取值字符限制：[a-z0-9-]+</li> <li>长度：[16,52]</li> </ul> <p><b>说明</b> "-"不能出现在名字头部或尾部</p>

## 请求参数

表 4-82 请求 Body 参数

参数	是否必选	参数类型	描述
table_name	是	String	<p>表名，仓内唯一。</p> <ul style="list-style-type: none"> <li>长度：[3, 52]</li> <li>取值字符限制：[a-z0-9_-]+</li> </ul>
hint_index_name	否	String	<p>create_table 时指定的索引名，默认空表示主索引。</p> <ul style="list-style-type: none"> <li>长度：[3, 63]</li> <li>取值字符限制：[a-z0-9_-]+</li> </ul>
limit	否	Integer	<p>返回的文档个数，最大 100 个。</p> <p><b>说明</b> 默认 100 个文档。</p>
start_key	否	Document	<p>起始主键或索引键值。</p> <ul style="list-style-type: none"> <li>默认空，表示从头遍历，左闭。</li> </ul> <p><b>说明</b> 分页返回时，该值使用上次响应返回的 cursor_key。</p>
end_key	否	Document	<p>终止主键或索引键值。</p> <ul style="list-style-type: none"> <li>默认空，表示直到最后，右开。</li> </ul>

参数	是否必选	参数类型	描述
filter_expression	否	condition_expression object	<p>过滤表达式。</p> <p><b>说明</b></p> <ul style="list-style-type: none"> <li>允许键字段作为条件。</li> <li>在满足start_key和endkey的范围内，执行filter逻辑。</li> <li>长度：[8,4096]</li> </ul>
sample_segments_count	否	Integer	<p>对表进行采样，尽最大努力保证返回的段列表均分整张表。举例： sample_segments_count=4，返回的段列表[MinKey, KV1)、[KV1,KV2)、[KV2,KV3)和[KV3,MaxKey)。用户可以使用scan-kv对这四个分区执行并发扫描，提高遍历效率。</p> <ul style="list-style-type: none"> <li>范围：[1, 10000]。默认值：不执行采样。</li> <li>sample_segments_count仅能和table_name、start_key和end_key字段配合使用。Range分区模式下支持全表采样和范围采样；Hash分区模式仅支持全表扫描。</li> <li>仅支持对Primary key进行采样，不支持本地/全局二级索引。</li> <li>返回的段列表仅包含主键，不包含键值；且段列表是编码后的数据，不可修改。</li> </ul>
return_count_only	否	Boolean	<p>是否只返回满足条件的KV个数；</p> <ul style="list-style-type: none"> <li>true:响应rsp只返回"returned_count"、"filtered_count"和"cursor_key"（如果有"cursor_key"）。</li> <li>false:scan-kv的标准响应体定义返回消息。</li> </ul>
returned_segment_items	否	Array of returned_segment_item objects	采样段区间列表。

表 4-83 condition\_expression

参数	是否必选	参数类型	描述
single_field_expression	否	<a href="#">single_field_expression</a> object	单字段条件。
multi_field_expression	否	<a href="#">multi_field_expression</a> object	多字段条件，多个相同优先级的单字段条件。
composed_expression	否	<a href="#">composed_expression</a> object	组合条件，逐层同级的子表达式嵌套。

表 4-84 composed\_expression

参数	是否必选	参数类型	描述
logic	否	String	逻辑关系，取值如"\$and", "\$or", "\$nor"。
expressions	是	Array of <a href="#">expression</a> objects	多个相同优先级且相同逻辑的单字段或多字段条件。

表 4-85 expression

参数	是否必选	参数类型	描述
single_field_expression	否	<a href="#">single_field_expression</a> object	单字段条件。
multi_field_expression	否	<a href="#">multi_field_expression</a> object	多字段条件，多个相同优先级的单字段条件。
composed_expression	否	<a href="#">composed_expression</a> object	组合条件，逐层同级的子表达式嵌套。

表 4-86 single\_field\_expression

参数	是否必选	参数类型	描述
field	是	String	条件字段。

参数	是否必选	参数类型	描述
func	是	String	条件函数，取值如"\$gt", "\$lt", "\$gte", "\$lte", "\$eq", "\$ne", "\$prefix", "\$exists"。
value	否	Document	value和value_array二选一。 <ul style="list-style-type: none"> <li>value条件值，适用于除"\$in", "\$nin"外的func。</li> <li>字段名无意义，可以传空，也可以传字段名。</li> <li>\$exists值为true/false。</li> </ul> <b>说明</b> \$prefix操作只适用于string和binary类型。
value_array	否	Array of strings	"value"和"value_array"二选一。 <ul style="list-style-type: none"> <li>"value_array" 条件值列表，值用于"\$in", "\$nin"。</li> </ul>

表 4-87 multi\_field\_expression

参数	是否必选	参数类型	描述
logic	否	String	多字段条件，多个相同优先级的单字段条件。
expressions	是	Array of <a href="#">single_field_expression</a> objects	多个相同逻辑操作的单字段条件。

表 4-88 returned\_segment\_item

参数	是否必选	参数类型	描述
segment_min_key	否	Document	采样段区间起始值。
segment_max_key	否	Document	采样段区间终止值。

## 响应参数

状态码：200

表 4-89 响应 Body 参数

参数	参数类型	描述
returned_count	Long	返回的文档数量，为0不表示结束。 <b>说明</b> 如果filtered_count超过500仍无匹配，则返回0。
filtered_count	Integer	被过滤掉的文档数量。
cursor_key	Document	下次请求时的start_key，该值为空时，表示指定范围或者指定filter条件所有kv已经返回。
returned_kv_items	Array of <a href="#">returned_kv_item</a> objects	返回的kv列表。
returned_segment_items	Array of <a href="#">returned_segment_item</a> objects	采样段区间列表。

表 4-90 returned\_kv\_item

参数	参数类型	描述
kv_doc	Document	对kv_doc有效。

表 4-91 returned\_segment\_item

参数	参数类型	描述
segment_min_key	Document	采样段区间起始值。
segment_max_key	Document	采样段区间终止值。

**状态码：400**

Bad Request

## 请求示例

扫描所有kv，扫描条件为分区键的值为user2，表名为test-table-1，仓名为test-store。

```
POST https://test-store.{endpoint}/v1/scan-kv
{
  "table_name": "test-table-1",
  "filter_expression": {
    "single_field_expr": {
      "field": "owner",
```

```
"func": "$eq",  
  "value": {  
    "value": "user-2"  
  }  
}
```

## 响应示例

**状态码: 200**

表示扫描所有kv请求成功

```
{  
  "returned_count": 1,  
  "filtered_count": 0,  
  "returned_kv_items": [{  
    "kv_doc": {  
      "owner": "user-2",  
      "filename": "test-file-2",  
      "otherKey": "otherValue"  
    }  
  }]  
}
```

## 状态码

状态码	描述
200	表示扫描所有kv请求成功
400	Bad Request

## 错误码

请参见[错误码](#)。

### 4.3.6 扫描分区键内 kv

#### 功能介绍

指定表及分区键，携带条件查询kv；允许指定过滤条件。

#### URI

POST /v1/scan-skey-kv

表 4-92 Query 参数

参数	是否必选	参数类型	描述
store_name	否	String	<p>仓名，全域唯一，不同租户的仓名不能相同。</p> <ul style="list-style-type: none"> <li>格式：\${prefix}-\${region-code}-\${account-id}，其中 prefix 为自定义前缀，region-code 为 kvs 集群所在的区域代码，account-id 为用户的账户 id</li> <li>取值字符限制：[a-z0-9-]+</li> <li>长度：[16,52]</li> </ul> <p><b>说明</b> "-"不能出现在名字头部或尾部</p>

## 请求参数

表 4-93 请求 Body 参数

参数	是否必选	参数类型	描述
table_name	是	String	<p>表名，仓内唯一。</p> <ul style="list-style-type: none"> <li>长度：[3, 52]</li> <li>取值字符限制：[a-z0-9_-]+</li> </ul>
hint_index_name	否	String	create_table 时指定的索引名。
limit	否	Long	<p>返回的文档个数，最大 100 个。</p> <p><b>说明</b> 默认 100 个文档。</p>
shard_key	是	Document	要遍历的指定分区键下的 kv。
start_sort_key	否	Document	<p>起始排序键值，默认空表示从头遍历，左闭。</p> <p><b>说明</b> 分页返回时，该值使用上次响应返回的 cursor_sort_key</p>
end_sort_key	否	Document	终止主键或索引键值，默认空表示直到最后，右开。

参数	是否必选	参数类型	描述
filter_expression	否	<a href="#">condition_expression</a> object	过滤表达式。 <b>说明</b> <ul style="list-style-type: none"> <li>允许主键字段作为条件。</li> <li>在满足shard_key和start_sort_key和end_sortkey的范围内，执行filter逻辑。</li> <li>长度：[8,4096]</li> </ul>

表 4-94 condition\_expression

参数	是否必选	参数类型	描述
single_field_expression	否	<a href="#">single_field_expression</a> object	单字段条件。
multi_field_expression	否	<a href="#">multi_field_expression</a> object	多字段条件，多个相同优先级的单字段条件。
composed_expression	否	<a href="#">composed_expression</a> object	组合条件，逐层同级的子表达式嵌套。

表 4-95 composed\_expression

参数	是否必选	参数类型	描述
logic	否	String	逻辑关系，取值如"\$and", "\$or", "\$nor"。
expressions	是	Array of <a href="#">expression</a> objects	多个相同优先级且相同逻辑的单字段或多字段条件。

表 4-96 expression

参数	是否必选	参数类型	描述
single_field_expression	否	<a href="#">single_field_expression</a> object	单字段条件。
multi_field_expression	否	<a href="#">multi_field_expression</a> object	多字段条件，多个相同优先级的单字段条件。

参数	是否必选	参数类型	描述
composed_expression	否	<a href="#">composed_expression</a> object	组合条件，逐层同级的子表达式嵌套。

表 4-97 single\_field\_expression

参数	是否必选	参数类型	描述
field	是	String	条件字段。
func	是	String	条件函数，取值如"\$gt", "\$lt", "\$gte", "\$lte", "\$eq", "\$ne", "\$prefix", "\$exists"。
value	否	Document	value和value_array二选一。 <ul style="list-style-type: none"> <li>value条件值，适用于除"\$in", "\$nin"外的func。</li> <li>字段名无意义，可以传空，也可以传字段名。</li> <li>\$exists值为true/false。</li> </ul> <b>说明</b> \$prefix操作只适用于string和binary类型。
value_array	否	Array of strings	"value"和"value_array"二选一。 <ul style="list-style-type: none"> <li>"value_array" 条件值列表，值用于"\$in", "\$nin"。</li> </ul>

表 4-98 multi\_field\_expression

参数	是否必选	参数类型	描述
logic	否	String	多字段条件，多个相同优先级的单字段条件。
expressions	是	Array of <a href="#">single_field_expression</a> objects	多个相同逻辑操作的单字段条件。

## 响应参数

状态码：200

表 4-99 响应 Body 参数

参数	参数类型	描述
returned_count	Long	返回的文档数量，为0不表示结束。 <ul style="list-style-type: none"> <li>如果filtered_count超过500仍无匹配，则返回0。</li> <li>长度：4</li> </ul>
filtered_count	Integer	被过滤掉的文档数量。 <ul style="list-style-type: none"> <li>长度：4</li> </ul>
cursor_sort_key	Document	下次请求时的start_key。 <b>说明</b> 该值为空时，表示指定范围或者指定filter条件所有kv已经返回。
returned_kv_items	Array of returned_kv_item objects	返回的kv列表，与scan-kv接口的returned_kv_item字段相同。

表 4-100 returned\_kv\_item

参数	参数类型	描述
kv_doc	Document	对kv_doc有效。

**状态码：400**

Bad Request

## 请求示例

扫描分区键值等于user1的所有kv，表名为test-table-1，仓名为test-store。

```
POST https://test-store.{endpoint}/v1/scan-skey-kv
```

```
{
  "table_name": "test-table-1",
  "shard_key": {
    "owner": "user-1"
  }
}
```

## 响应示例

**状态码：200**

表示扫描分区键内kv请求成功

```
{
  "returned_count": 1,
  "filtered_count": 0,
  "kv_array": [{
```

```
"returned_kv_items": {  
  "kv_doc": {  
    "owner": "user-1",  
    "filename": "test-file-1",  
    "otherkey": "otherValue"  
  }  
}  
}]  
}
```

## 状态码

状态码	描述
200	表示扫描分区键内kv请求成功
400	Bad Request

## 错误码

请参见[错误码](#)。

## 4.3.7 批量写请求

### 功能介绍

批量写请求，其中可以携带一或多个表的不同kv的写操作，上传kv/删除kv。

## URI

POST /v1/batch-write-kv

表 4-101 Query 参数

参数	是否必选	参数类型	描述
store_name	否	String	仓名，全域唯一，不同租户的仓名不能相同。 <ul style="list-style-type: none"><li>格式：\${prefix}-\${region-code}-\${account-id}，其中prefix为自定义前缀，region-code为kvs集群所在的区域代码，account-id为用户的账户id</li><li>取值字符限制：[a-z0-9]+</li><li>长度：[16,52]</li></ul> <b>说明</b> "."不能出现在名字头部或尾部

## 请求参数

表 4-102 请求 Body 参数

参数	是否必选	参数类型	描述
table_opsers	是	Array of <a href="#">table_batch</a> objects	行操作数组，可以是多个表的操作。

表 4-103 table\_batch

参数	是否必选	参数类型	描述
table_name	是	String	表名，仓内唯一。 <ul style="list-style-type: none"> <li>长度：[3, 52]</li> <li>取值字符限制：[a-z0-9_-]+</li> </ul>
kv_opsers	是	Array of <a href="#">oper_item</a> objects	kv操作数组。

表 4-104 oper\_item

参数	是否必选	参数类型	描述
put_kv	否	<a href="#">put_kv</a> object	上传kv操作。
delete_kv	否	<a href="#">delete_kv</a> object	删除kv操作。

表 4-105 put\_kv

参数	是否必选	参数类型	描述
oper_id	是	Integer	请求内的操作编码，未成功的操作返回该标识。
kv_doc	否	Document	用户文档。

表 4-106 delete\_kv

参数	是否必选	参数类型	描述
oper_id	是	Integer	请求内的操作编码，未成功的操作返回该标识。

参数	是否必选	参数类型	描述
primary_key	是	Document	用户自定义的主键名及值。

## 响应参数

状态码：200

表 4-107 响应 Body 参数

参数	参数类型	描述
unprocessed_operations	Array of <a href="#">table_oper_ids</a> objects	未处理的操作列表。

表 4-108 table\_oper\_ids

参数	参数类型	描述
table_name	String	表名，仓内唯一。 <ul style="list-style-type: none"> <li>长度：[3, 52]</li> <li>取值字符限制：[a-z0-9_-]+</li> </ul>
kv_oper_ids	<a href="#">kv_oper_ids</a> object	kv操作数组。

表 4-109 kv\_oper\_ids

参数	参数类型	描述
put_kv_ids	Array of integers	上传kv操作, "oper_id"数组。 <ul style="list-style-type: none"> <li>数组元素：请求内的操作编码，未成功的操作返回该标识。</li> </ul>
delete_kv_ids	Array of integers	请求内的操作编码，未成功的操作返回该标识。 <ul style="list-style-type: none"> <li>数组元素：请求内的操作编码，未成功的操作返回该标识。</li> </ul>

状态码：400

BatchExceedLimit

## 请求示例

批量写请求，向test-store仓内的test-table-1表中插入和删除一个kv

POST https://test-store.{endpoint}/v1/batch-write-kv

```
{
  "table_ops": [ {
    "table_batch": {
      "table_name": "test-table-1",
      "kv_ops": [ {
        "put_kv": {
          "oper_id": 1,
          "kv_doc": {
            "owner": "user-1",
            "filename": "test-file-1"
          }
        }
      }, {
        "delete_kv": {
          "oper_id": 2,
          "primary_key": {
            "owner": "user-2",
            "filename": "test-file-2"
          }
        }
      }
    ]
  }
}
```

## 响应示例

无

## 状态码

状态码	描述
200	表示批量写请求成功
400	BatchExceedLimit

## 错误码

请参见[错误码](#)。

# 5 附录

## 5.1 状态码

正常状态码	描述
200	OK
201	Created
202	Accepted
204	No Content

错误状态码	描述
400	Bad Request
401	Unauthorized
403	Forbidden
404	Not Found
405	Method Not Allowed
409	Conflict
413	Request Entity Too Large
415	Unsupported Media Type
429	Too Many Requests
500	Internal Server Error
501	Not Implemented
503	Service Unavailable

## 5.2 错误码

调用接口出错后，将不会返回结果数据。调用方可根据每个接口对应的错误码来定位错误原因。当调用出错时，HTTP请求返回一个3xx, 4xx或5xx的HTTP状态码。返回的消息体中是具体的错误代码及错误信息。在调用方找不到错误原因时，可以联系华为云客服，并提供错误码，以便尽快帮您解决问题。

### 错误码说明

当您调用API时，如果遇到“APIGW”开头的错误码，请参见[API网关错误码](#)进行处理。

表 5-1 错误码

状态码	错误码	错误信息	描述	处理措施
500 Internal Server Error	KVS.000001	InternalServerError	系统遇到内部错误，请重试	请联系技术支持
501 Not Implemented	KVS.000002	ServiceNotImplemented	请求的方法服务端没有实现	当前不支持，请联系技术支持
503 Service Unavailable	KVS.000003	SlowDown	请求频率过高	请降低请求频率
400 Bad Request	KVS.000101	Invalid request parameter	请求参数非法	根据返回的错误消息体提示进行修改
403 Forbidden	KVS.0001010	AccessDenied	拒绝访问，请求没有携带日期头域或者头域格式错误	请求携带正确的日期头域
400 Bad Request	KVS.000102	InvalidStore	请求访问的Store已不存在，或者状态不正确	更换Store名
400 Bad Request	KVS.000103	InvalidTable	请求访问的Table已不存在，或者状态不正确	更换Table名

400 Bad Request	KVS.0 00010 04	EntityTooLarge	请求超过大小限制	减少大小
400 Bad Request	KVS.0 00010 05	IncompleteBody	由于网络原因或其他问题导致请求体未接受完整	重试
400 Bad Request	KVS.0 00010 06	InvalidRequest	无效请求	body体不合法，或者没有携带，根据返回的错误消息体提示进行修改
400 Bad Request	KVS.0 00010 07	TooManyStores	用户拥有的Store的数量达到了系统的上限，并且请求试图创建一个新Store	删除部分Store后重试
400 Bad Request	KVS.0 00010 08	TooManyTables	用户Store拥有的Tables的数量达到了系统的上限，并且请求试图创建一个新Tables	删除部分Table后重试，或选取其他Store
400 Bad Request	KVS.0 00010 09	TooManyKeyValues	超出单分区键的kv限制	-
403 Forbidden	KVS.0 00010 11	AccessForbidden	权限不足	检查权限配置
403 Forbidden	KVS.0 00010 12	InArrearsOrInsufficientBalance	用户欠费或余额不足而没有权限进行某种操作	充值
403 Forbidden	KVS.0 00010 13	InsufficientStorageSpace	存储空间不足	超过配额限制，增加配额或删除部分对象
404 Not Found	KVS.0 00010 14	NoSuchStore	指定的Store不存在	先创Store再操作
404 Not Found	KVS.0 00010 15	NoSuchTable	指定的Table不存在	先创Table再操作
404 Not Found	KVS.0 00010 16	NoSuchKey	指定的Key不存在	先PUT

408 Request Timeout	KVS.0001017	RequestTimeout	用户与Server之间的socket连接在超时时间内没有进行读写操作	检查网络后重试, 或联系技术支持
409 Conflict	KVS.0001018	StoreAlreadyExists	请求的Store名已经存在Store的命名空间是系统中所有用户共用的, 选择一个不同的Store名再重试一次	更换Store名
409 Conflict	KVS.0001019	StoreNotEmpty	用户尝试删除的Store不为空	先删除Store中对象, 然后再删除Store
409 Conflict	KVS.0001020	TableAlreadyExists	请求的Table名已经存在, 选择一个不同的Table名再重试一次	更换Table名
409 Conflict	KVS.0001021	TableNotEmpty	用户尝试删除的Table不为空	先删除Table中对象, 然后再删除Table
409 Conflict	KVS.0001022	OperationAborted	另外一个冲突的操作当前正作用在这个资源上, 请重试	等待一段时间后重试
503 Service Unavailable	KVS.0001023	SlowDown	请求频率过高	请降低请求频率
400 Bad Request	KVS.0005000	RegionIsNotOnService	Region不支持创建store	等待一段时间后重试
400 Bad Request	KVS.0005001	RegionNumberExceedLimit	Region数量超过限制	减少Region数量
400 Bad Request	KVS.0005002	TableAlreadySwitchOnPITR	下面的某个表已经打开了PITR	无需重复开启PITR
400 Bad Request	KVS.0005004	BackupNameConflict	备份名冲突	更换备份名

400 Bad Request	KVS.0005008	BackupNameNotAvailable	备份名无效	更换备份名
400 Bad Request	KVS.0005009	RecoveryTimeStampInvalid	恢复时间点无效	更换恢复时间点
400 Bad Request	KVS.0005010	Local2ndIndexNumberExceedLimit	本地二级索引数量超过限制	减少本地二级索引数量
400 Bad Request	KVS.0005011	Global2ndIndexNumberExceedLimit	全局二级索引数量超过限制	减少全局二级索引数量
400 Bad Request	KVS.0005012	PrimaryKeyIndexSchemaInvalid	主键索引模板定义无效	检查模板定义参数
400 Bad Request	KVS.0005013	Local2ndIndexSchemaInvalid	本地二级索引模板定义无效	检查模板定义参数
400 Bad Request	KVS.0005014	Global2ndIndexSchemaInvalid	全局二级索引模板定义无效	检查模板定义参数
400 Bad Request	KVS.0005015	StoreAlreadySwitchOnPITR	store已经打开了PITR	无需重复开启PITR
400 Bad Request	KVS.0005016	ConditionExpressionInvalid	条件表达式超过上限，或者语法不正确，或者预定义变量找不到	检查条件表达式定义
400 Bad Request	KVS.0005017	ConditionIsFalse	条件表达式非真	-
400 Bad Request	KVS.0005018	keyValueSizeExceeded	KV大小超过限制	缩减文档数据量

400 Bad Request	KVS.0005019	ReturnPartialBlobsInvalid	ReturnPartialBlob指定的偏移无效	修改偏移参数值
400 Bad Request	KVS.0005020	BatchExceedLimit	batch大小或操作个数超限	减少操作个数

## 5.3 获取项目 ID

### 操作场景

在调用接口的时候，部分URL中需要填入项目ID，所以需要获取到项目ID。有如下两种获取方式：

- [调用API获取项目ID](#)
- [从控制台获取项目ID](#)

### 调用 API 获取项目 ID

获取项目ID请参考：[查询指定条件下的项目列表](#)。

### 从控制台获取项目 ID

从控制台获取项目ID的步骤如下：

1. 登录管理控制台。
2. 鼠标悬停在右上角的用户名，选择下拉列表中的“我的凭证”。  
在“API凭证”页面的项目列表中查看项目ID。

图 5-1 查看项目 ID



## 5.4 获取账号 ID

在调用接口的时候，部分URL中需要填入账号ID，所以需要先在管理控制台上获取到账号ID。账号ID获取步骤如下：

1. 登录管理控制台。
2. 单击用户名，在下拉列表中单击“我的凭证”。  
在“API凭证”页面中查看账号ID。

图 5-2 获取账号 ID

