

## 目录

目录	1
API参考手册	4
什么是云数据库MongoDB?	4
文档概述	4
API概览	4
实例相关接口	4
安全组相关接口	4
地域相关接口	4
备份相关接口	4
服务信息	5
公共参数	5
签名机制	5
Python Demo	5
Java Demo	6
Go Demo	7
通用错误信息	8
创建mongo副本集实例 CreateMongoDBInstance	8
请求方式	8
请求参数	9
返回结果	9
错误信息	9
样例	9
DeleteMongoDBInstance 删除MongoDB实例	10
请求参数	10
请求方式	10
返回结果	10
错误信息	10
样例	10
DescribeMongoDBInstances 查询服务列表	10
请求参数	10
请求方式	11
错误信息	11
请求样例	11
返回样例	11
DescribeMongoDBInstance	12
请求参数	12
请求方式	12
错误信息	12
请求样例	12
返回样例	12
RenameMongoDBInstance 重命名MongoDB实例	13
请求参数	13
请求方式	13
错误信息	13
请求样例	13
返回样例	13
RestartMongoDBInstance 重启MongoDB实例	13
请求参数	13
请求方式	14
错误信息	14

请求样例	14
返回样例	14
DescribeMongoDBInstanceNode 查询副本集实例节点信息	14
请求参数	14
请求方式	14
错误信息	14
请求样例	14
返回样例	14
DescribeMongoDBShardNode 查询分片集群实例节点信息	15
请求参数	15
请求方式	15
增加只读节点AddSecondaryInstance	16
请求参数	16
请求方式	16
错误信息	16
样例	16
创建分片集群实例CreateMongoDBShardInstance	16
请求方式	16
请求参数	16
错误信息	17
样例	17
查询可用机房及可用区DescribeRegions	17
请求参数	17
请求方式	17
错误信息	17
样例	18
AddSecurityGroupRule 添加安全组规则	18
功能描述	18
请求参数	18
请求方式	18
返回结果	18
错误信息	18
样例	18
DeleteCacheSecurityRule	19
请求参数	19
请求方式	19
返回结果	19
错误信息	19
样例	19
ListSecurityGroupRules	20
请求参数	20
请求方式	20
返回结果	20
错误信息	20
样例	20
CreateMongoDBSnapshot 创建手动备份	21
请求参数	21
请求方式	21
返回结果	21
错误信息	21
样例	21
SetMongoDBTimingSnapshot 创建自动备份	21

请求参数	22
请求方式	22
返回结果	22
错误信息	22
样例	22
DeleteMongoDBSnapshot删除实例备份	22
请求参数	22
请求方式	22
返回结果	23
错误信息	23
样例	23
DescribeMongoDBSnapshot查看备份列表	23
请求参数	23
请求方式	23
返回结果	23
错误信息	23
样例	23
RenameMongoDBSnapshot备份重命名	24
请求参数	24
请求方式	24
返回结果	24
错误信息	24
样例	24

# API参考手册

## 什么是云数据库MongoDB?

金山云云数据库MongoDB是金山云为企业和开发者提供了100%兼容MongoDB协议的文档型NoSQL数据库服务，底层基于三节点副本集，能为客户提供故障切换，容灾迁移，在线备份等一体化的解决方案。服务提供了用于实例操作的OpenAPI，可以根据需要开发应用程序，实现实例的创建，更配、备份等多种自动化任务。

## 文档概述

本指南提供了API的说明以及使用案例。  
通用请求部分介绍了API服务的基本信息，公共参数，签名机制和报错信息等内容。  
实例管理部分介绍了实例管理操作的API和编程示例。  
备份管理部分介绍了创建备份、备份设置等相关操作以及示例。数据类型部分介绍了API自定义的数据类型。

## API概览

[实例相关接口](#)

[安全组相关接口](#)

[地域相关接口](#)

[备份相关接口](#)

## 实例相关接口

Action Name (接口名称)	Action Function (接口功能)
CreateMongoDBInstance	创建mongo副本集实例
DeleteMongoDBInstance	删除MongoDB实例
DescribeMongoDBInstances	查询服务列表
DescribeMongoDBInstance	查看MongoDB实例详情
RenameMongoDBInstance	重命名MongoDB实例
RestartMongoDBInstance	重启MongoDB实例
DescribeMongoDBInstanceNode	查询副本集实例节点信息
DescribeMongoDBShardNode	查询分片集群实例节点信息
AddSecondaryInstance	增加只读节点
CreateMongoDBShardInstance	创建分片集群实例

## 安全组相关接口

Action Name (接口名称)	Action Function (接口功能)
AddSecurityGroupRule	添加安全组规则
DeleteCacheSecurityRule	删除安全组规则
ListSecurityGroupRules	查看服务安全规则

## 地域相关接口

Action Name (接口名称)	Action Function (接口功能)
DescribeRegions	查询可用机房及可用区

## 备份相关接口

Action Name (接口名称)	Action Function (接口功能)
CreateMongoDBSnapshot	创建手动备份
SetMongoDBTimingSnapshot	创建自动备份
DeleteMongoDBSnapshot	删除实例备份
DescribeMongoDBSnapshot	查看备份列表

RenameMongoDBSnapshot 备份重命名

## 服务信息

- **服务地址**  
mongodb.api.ksyun.com
- **通讯协议**  
支持HTTP或HTTPS协议
- **请求方法**  
详见具体API接口
- **请求参数**  
每个请求都需要包含公共的请求参数和操作所特有的请求参数，详见具体API接口
- **编码方式**  
请求及结果返回都使用UTF-8字符集编码
- **返回结果**  
仅支持JSON返回结果

## 公共参数

支持GET和POST两种HTTP方法。

固定请求域名：mongodb.api.ksyun.com（不指定区域则默认为cn-beijing-6，若要指定区域：mongodb.{Region}.api.ksyun.com）

云数据库MongoDB的服务名称(Service)指定为mongodb

详情请参考：[公共参数](#)

## 签名机制

数据库的openAPI支持GET和POST两种HTTP方法，具体流程请参见[签名机制](#)。

### Python Demo

```
import hashlib
import hmac
import time
import urllib.request

import requests

# Python 3.9.6
def http_execute(methed, host, heads, data=None):
    url = 'https://%s/?' % host
    for key in data:
        url += urllib.request.quote(key, '~') + '=' + urllib.request.quote(str(data[key]), '~') + '&'
    response = requests.request(methed, url, headers=heads, data=None)
    return response;

# sign方法
def sign(params, secret_key):
    str_encode = ''
    param_keys = sorted(params.keys())
    for key in param_keys:
        str_encode += urllib.request.quote(key, '~') + '=' + urllib.request.quote(str(params[key]), '~') + '&'
    return hmac.new(bytes(secret_key, 'utf-8'), bytes(str_encode[:-1], 'utf-8'), hashlib.sha256).hexdigest()

class ksyun_iam_api_tools():
    """金山云API相关"""

    def __init__(self):
        self.AK = 'your_ak'
        self.SK = 'your_sk'
        self.SERVICE = "krds"
        self.REGION = "cn-beijing-6"
        self.HOST = "%s.%s.api.ksyun.com" % (self.SERVICE, self.REGION)
        self.additional_headers = {'Accept': 'application/json'}

    def describe_db_instances(self):
        data = {
            'Accesskey': self.AK,
```

```

        'Service': 'krds',
        'Action': 'DescribeDBInstances',
        'Version': '2016-07-01',
        'Timestamp': time.strftime("%Y-%m-%dT%H:%M:%SZ", time.gmtime()), # 使用UTC格式的时间
        'SignatureVersion': '1.0',
        'SignatureMethod': 'HMAC-SHA256',
    }
    data["Signature"] = sign(data, self.SK)
    return http_execute("GET", self.HOST, self.additional_headers, data)

if __name__ == '__main__':
    api = ksyun_iam_api_tools()
    instances = api.describe_db_instances()
    print(instances.text)

```

## Java Demo

```

import okhttp3.OkHttpClient;
import okhttp3.Request;
import okhttp3.Response;
import org.apache.commons.codec.digest.HmacAlgorithms;
import org.apache.commons.codec.digest.HmacUtils;

import java.io.IOException;
import java.io.UnsupportedEncodingException;
import java.net.URLEncoder;
import java.nio.charset.StandardCharsets;
import java.text.ParseException;
import java.text.SimpleDateFormat;
import java.util.*;
import java.util.regex.Matcher;
import java.util.regex.Pattern;

// jdk1.8.0_241
public class ram_demo {
    private static final Pattern ENCODED_CHARACTERS_PATTERN;

    static {
        StringBuilder pattern = new StringBuilder();
        pattern.append(Pattern.quote("+"))
            .append("|")
            .append(Pattern.quote("*"))
            .append("|")
            .append(Pattern.quote("%7E"))
            .append("|")
            .append(Pattern.quote("%2F"));
        ENCODED_CHARACTERS_PATTERN = Pattern.compile(pattern.toString());
    }

    /**
     * 注意空格( )会编码成+号, 所以+最后需要替换成%20 (%20为空格)
     * 在URLEncoder后需对三种字符替换: 加号(+) 替换成 %20、星号(*) 替换成 %2A、 %7E 替换回波浪号(~)
     */
    private static String urlEncode(final String value, final boolean path) {
        if (value == null) {
            return "";
        }

        try {
            String encoded = URLEncoder.encode(value, StandardCharsets.UTF_8.name());
            Matcher matcher = ENCODED_CHARACTERS_PATTERN.matcher(encoded);
            StringBuffer buffer = new StringBuffer(encoded.length());

            while (matcher.find()) {
                String replacement = matcher.group(0);
                if ("+".equals(replacement)) {
                    replacement = "%20";
                } else if ("*".equals(replacement)) {
                    replacement = "%2A";
                } else if ("%7E".equals(replacement)) {
                    replacement = "~";
                } else if (path && "%2F".equals(replacement)) {
                    replacement = "/";
                }

                matcher.appendReplacement(buffer, replacement);
            }

            matcher.appendTail(buffer);
            return buffer.toString();
        } catch (UnsupportedEncodingException ex) {
            throw new RuntimeException(ex);
        }
    }
}

```

```

    }

    private static String getCanonicalizedQueryString(Map<String, Object> params) {
        final Map<String, String> tmap = new TreeMap<>();
        for (Map.Entry<String, Object> entry : params.entrySet()) {
            String key = entry.getKey();
            Object value = entry.getValue();
            String encodedParamName = urlEncode(key, false);
            String encodedValues = urlEncode(value.toString(), false);
            tmap.put(encodedParamName, encodedValues);
        }

        final StringBuilder sb = new StringBuilder();
        for (Map.Entry<String, String> entry : tmap.entrySet()) {
            if (sb.length() > 0) {
                sb.append("&");
            }
            sb.append(entry.getKey()).append('=').append(entry.getValue());
        }
        return sb.toString();
    }

    public static String signature(Map<String, Object> params, String secretKey) {
        String canonicalizedQueryString = getCanonicalizedQueryString(params);
        return new HmacUtils(HmacAlgorithms.HMAC_SHA_256, secretKey).hmacHex(canonicalizedQueryString);
    }

    public static String utc2Local(String utcTime, String utcTimePatten, String localTimePatten) throws ParseException {
        SimpleDateFormat utcFormatter = new SimpleDateFormat(utcTimePatten);
        utcFormatter.setTimeZone(TimeZone.getTimeZone("UTC")); // 时区定义并进行时间获取
        Date gpsUTCDate = null;
        gpsUTCDate = utcFormatter.parse(utcTime);
        SimpleDateFormat localFormatter = new SimpleDateFormat(localTimePatten);
        localFormatter.setTimeZone(TimeZone.getDefault());
        String localTime = localFormatter.format(gpsUTCDate.getTime());
        return localTime;
    }

    public static void test1() throws IOException {
        String accesskey = "your_ak";
        String secretKey = "your_sk";

        SimpleDateFormat sdf = new SimpleDateFormat("yyyy-MM-dd'T'HH:mm:ss'Z'");
        TimeZone zone = TimeZone.getTimeZone("UTC");
        Calendar cal = Calendar.getInstance(zone);
        sdf.setTimeZone(zone);

        Map<String, Object> params = new HashMap<>();
        params.put("Accesskey", accesskey); // 公共参数
        params.put("Service", "krds"); // 公共参数
        params.put("Action", "DescribeDBInstances"); // 公共参数
        params.put("Version", "2016-07-01"); // 公共参数
        params.put("Timestamp", sdf.format(cal.getTime())); // 公共参数
        params.put("SignatureVersion", "1.0"); // 公共参数
        params.put("SignatureMethod", "HMAC-SHA256"); // 公共参数
        params.put("Signature", signature(params, secretKey));

        String product_name = "krds";
        String region = "cn-beijing-6";

        String url = "https://" + product_name + "." + region + ".api.ksyun.com/?" + getCanonicalizedQueryString(params);
        OkHttpClient client = new OkHttpClient().newBuilder()
            .build();
        Request request = new Request.Builder()
            .url(url)
            .method("GET", null)
            .addHeader("Accept", "application/json")
            .build();
        Response response = client.newCall(request).execute();

        System.out.println(new String(response.body().bytes()));
    }

    public static void main(String[] args) {
        try {
            test1();
        } catch (IOException e) {
            e.printStackTrace();
        }
    }
}

```

## Go Demo

```

package main

import (
    "crypto/hmac"
    "crypto/sha256"
    "encoding/hex"
    "fmt"
    "io/ioutil"
    "net/http"
    "net/url"
    "strings"
    "time"
)

// go version go1.16.5 darwin/amd64

//获取signature签名
func sign(params url.Values, sk string) string {
    strEncode := params.Encode()
    strEncode = strings.Replace(strEncode, "+", "%20", -1)
    h := hmac.New(sh256.New, []byte(sk))
    h.Write([]byte(strEncode))
    return hex.EncodeToString(h.Sum(nil))
}

func main() {

    params := url.Values{}
    params.Add("Accesskey", "your_ak")
    params.Add("Service", "krds")
    params.Add("Action", "DescribeDBInstances")
    params.Add("Version", "2016-07-01")
    params.Add("Timestamp", time.Now().UTC().Format("2006-01-02T15:04:05Z")) //通过time.Now().UTC().Format("2006-01-02T15:04:05Z")实现
    params.Add("SignatureVersion", "1.0")
    params.Add("SignatureMethod", "HMAC-SHA256")
    sk := "your_sk"
    signature := sign(params, sk)
    params.Add("Signature", signature)

    strEncode := params.Encode()
    strEncode = strings.Replace(strEncode, "+", "%20", -1)

    url := "https://krds.cn-beijing-6.api.ksyun.com/?" + strEncode
    method := "GET"
    client := &http.Client{}
    req, _ := http.NewRequest(method, url, nil)
    req.Header.Add("Accept", "application/json")
    res, _ := client.Do(req)
    defer res.Body.Close()
    body, _ := ioutil.ReadAll(res.Body)
    fmt.Println(string(body))
}

```

## 通用错误信息

Code值	解释	HTTP Status Code
MissingParameter	表示请求参数	400
MissingAction	表示没有提示出正确的Action 或 Version	400
InvalidParameter	非法的请求参数	400
RequestExpired	请求过期	400
ServiceUnavailable	服务不可用	503
ValidationError	请求验证失败	400
InvalidAction	非法的操作	400
BusinessError	程序中业务错误信息提示	400
InternalFailure	内部错误	500

## 创建mongo副本集实例 CreateMongoDBInstance

根据需求创建一个副本集实例，此接口只支持创建副本集实例。

### 请求方式



POST

## 请求参数

关于所有操作需要的通用参数，请参照通用请求-[公共参数](#)

英文名称	中文名称	类型	是否必须	备注
Action	调用接口名称	String	是	取固定值：CreateMongoDBInstance
Version	API版本号	String	是	取固定值：2017-01-01
Name	mongo服务实例名称	String	是	支持6-64个中文或者英文字符，包括汉字，大小写字母，数字，下划线和中划线
DbVersion	MongoDB引擎版本	String	否	目前仅支持WiredTiger，默认值为3.2，取值范围3.2、3.6
InstanceClass	实例规格	String	是	实例配置，客户只能购买特定规格的套餐。详见 <a href="#">实例规格</a>
Storage	磁盘大小	Integer	是	硬盘区间值，单位是GB
NetworkType	网络类型	String	是	固定值VPC，目前只支持创建VPC实例。
VpcId	虚拟专用网络	String	是	VPC网络ID，可在网络控制台获取。
VnetId	终端子网id	String	否	终端子网ID，可在网络控制台获取（注意类型必须为终端子网）。
InstanceAccount	实例管理员账号	String	否	实例管理员账号：默认账户名root
InstancePassword	实例管理员密码	String	是	实例管理员密码
PayType	计费方式：默认为byMonth	String	是	取值范围：byMonth（包年包月），byDay（按日计费），hourlyInstantSettlement（按小时实时结算）。
Duration	时长，默认值：1（单位：月）	Integer	否	时长 默认值：1（单位：月）；PayType=byMonth（包年包月）则必填，最大支持范围是（1~36月）
IamProjectId	项目ID	String	否	默认为0：默认项目
NodeNum	节点数	Integer	否	默认为3，取值范围3、5、7。为实例所有节点数，每个实例包含一个primary一个hidden节点，其余为secondary节点。
AvailabilityZone.N	可用区	String	是	目前暂不支持跨可用区，建议将实例创建在云主机的可用区内已减少网络延时。可 <a href="#">查询可用机房及可用区</a>

## 返回结果

英文名称	中文名称	类型	是否必须	备注
RequestId	请求ID	String	是	--
Data	返回数据信息	Object	否	返回列表的数据信息，具体可以查看参数列表中的返回例子
Error	错误	String	否	Code：错误状态码 Message：文字描述

## 错误信息

关于所有操作返回的错误信息，请参照通用请求-[通用错误信息](#)

## 样例

## • 请求样例

```
http://mongodb.${region}.api.ksyun.com/
api?Action=CreateMongoDBInstance
&Version=2017-01-01
&Name=SomebodyCallMe芒果
&InstancePassword=IamAdmin
```

## • 返回样例

```
{
  "RequestId": "T0E1S2T3S-AFQWOFJQ0JF-WFWFWF-FDGD",
  "MongoDBInstanceResult": {
    "UserId": "73403574",
    "Region": "cn-shanghai-3",
    "InstanceId": "e6933d66-7bae-4f43-af82-b74846d6076d",
    "Name": "XB-Test-Create-MongoDB-001"
  }
}
```

## DeleteMongoDBInstance 删除MongoDB实例

删除MongoDB服务，根据服务ID删除服务资源

### 请求参数

关于所有操作需要的通用参数，请参照通用请求—[公共参数](#)

英文名称	中文名称	类型	是否必须	备注
Action	调用接口名称	String	是	取固定值：DeleteCacheCluster
Version	API版本号	String	是	取固定值：2017-01-01
InstanceId	服务ID	String	是	

### 请求方式

DELETE

### 返回结果

英文名称	中文名称	类型	是否必须
reqId	请求流水号	String	是
code	状态： 0：SUCCESS >0：FAILED	Integer	是
message	文字描述	String	是
data	返回数据信息	Object	否

### 错误信息

关于所有操作返回的错误信息，请参照通用请求—[通用错误信息](#)

### 样例

#### • 请求样例

```
https://mongodb.${region}.api.ksyun.com/
api?Action=DeleteMongoDBInstance
&Version=2017-01-01
&InstanceId=xxxInstanceId
```

#### • 返回样例

```
{
  "RequestId": "T0E1S2T3S-AFQW0FJQ0JF-WFWWF-FDGD",
  "InstanceId": "41ed452d-6d0e-4c47-9edb-921fffab9a6a"
}
```

## DescribeMongoDBInstances 查询服务列表

查询服务列表，允许您根据特定的条件查询资源，并支持对资源排序，默认每次查出10个服务

### 请求参数

关于所有操作需要的通用参数，请参照通用请求—[公共参数](#)

英文名称	中文名称	类型	是否必须	备注
Action	调用接口名称	String	是	取固定值：DescribeMongoDBInstances
Version	API版本号	String	是	取固定值：2017-01-01
InstanceId	MongoDB服务ID	String	是	
Name	实例名称	String	否	
Vip	服务IP地址	String	否	
VpcId	虚拟专用网络ID	String	否	只适用于VPC网络下的服务

VnetId	虚拟专用网EndpointID, 或vpclbID	String	否	只适用于VPC网络下的服务
Offset	查询数据的起始位置	Integer	否	默认为0
Limit	需要从起始位置开始查询的缓存服务的个数	Sinteger	否	取值范围为[1~100], 默认为10
OrderBy	排序字段	String	否	可传值为{name, asc, name, desc, created, asc, created, desc}, 默认按照创建时间降序, 只有排序字段时, 默认按照升序排列
IamProjectId	项目ID	String	否	默认是0(默认项目), 如果查询全部项目, 需要传入所有的项目ID, ‘,’ 隔开

## 请求方式

GET

## 错误信息

关于所有操作返回的错误信息, 请参照通用请求-[通用错误信息](#)

## 请求样例

```
https://mongodb.${region}.api.ksyun.com/
api?Action=DescribeMongoDBInstances
&Version=2017-01-01
&OrderBy=name,asc
&IamProjectId=1,2,3
```

## 返回样例

HttpStatusCode=200

```
{
  "RequestId": "TOE1S2T3S-AFQWOFJQ0JF-WFWFWF-FDGD",
  "MongoDBInstancesResult": [
    {
      "IamProjectId": "22",
      "IamProjectName": "tes9t1",
      "UserId": "73403574",
      "Region": "cn-shanghai-3",
      "Name": "dhhu-test",
      "InstanceId": "468428c3-8637-4444-a11e-4b5d985e419d",
      "Status": "unknown",
      "IP": "10.0.0.198,10.0.0.202,10.0.0.151",
      "InstanceType": "HighIO",
      "Version": null,
      "InstanceClass": "1C2G",
      "Storage": 10,
      "Port": null,
      "NetworkType": "VPC",
      "VpcId": "3c12ccdf-9b8f-4d9b-8aa6-a523897e97a1",
      "VnetId": "293c16a5-c757-405c-a693-3b2a3adead50",
      "ProductId": "xxxxxxx",
      "ProductWhat": 1,
      "PayType": "byMonth",
      "CreateDate": "2017-01-20 18:13:14",
      "ExpirationDate": null
    },
    {
      "IamProjectId": "22",
      "IamProjectName": "tes9t1",
      "UserId": "73403574",
      "Region": "cn-shanghai-3",
      "Name": "lxb-test-002",
      "InstanceId": "cbc2273c-bd7e-4c8b-a2ca-43582b09ad26",
      "Status": "unknown",
      "IP": "10.0.0.167,10.0.0.175",
      "InstanceType": "HighIO",
      "Version": null,
      "InstanceClass": "1C2G",
      "Storage": 8,
      "Port": null,
      "NetworkType": "VPC",
      "VpcId": "3c12ccdf-9b8f-4d9b-8aa6-a523897e97a1",
      "VnetId": "293c16a5-c757-405c-a693-3b2a3adead50",
      "ProductId": "xxxxxxx",
      "PayType": "byMonth",
    }
  ]
}
```

```

    "ProductWhat": 1,
    "CreateDate": "2017-01-20 11:51:56",
    "ExpirationDate": null
  },
  {
    "IamProjectId": "22",
    "IamProjectName": "tes9t1",
    "UserId": "73403574",
    "Region": "cn-shanghai-3",
    "Name": "lxb-test-001",
    "InstanceId": "ce009fb8-1428-4052-839e-9c07d26e2431",
    "Status": "unknown",
    "IP": null,
    "InstanceType": "HighIO",
    "Version": null,
    "InstanceClass": "1C2G",
    "Storage": 8,
    "Port": null,
    "NetworkType": "Basic",
    "VpcId": null,
    "VnetId": null,
    "PayType": "byMonth",
    "ProductWhat": 2,
    "CreateDate": "2017-01-20 11:38:22",
    "ExpirationDate": null
  }
],
"Offset": 0,
"Limit": 10,
"Total": 3
}

```

## DescribeMongoDBInstance

查询MongoDB实例服务详情，根据服务ID获得资源的详细信息以及该服务资源的订单信息

### 请求参数

关于所有操作需要的通用参数，请参照通用请求-[公共参数](#)

英文名称	中文名称	类型	是否必须	备注
Action	调用接口名称	String	是	取固定值：DescribeMongoDBInstance
Version	API版本号	String	是	取固定值：2017-01-01
InstanceId	服务ID	String	是	

### 请求方式

GET

### 错误信息

关于所有操作返回的错误信息，请参照通用请求-[通用错误信息](#)

### 请求样例

```

https://kcs.${region}.api.ksyun.com/
api?Action=DescribeMongoDBInstance
&Version=2017-01-01
&InstanceId=07493f09-1161-4442-80e5-e8e8bb1a2ecd

```

### 返回样例

```

HttpStatusCode=200

{"RequestId": "T0E1S2T3S-AFQWOFJQ0JF-WFWFWF-FDGD",
 "MongoDBInstanceResult": {
  "UserId": "73403574",
  "Region": "cn-shanghai-3",
  "Name": "yj-normal-test-1",
  "InstanceId": "d5ba2ca1-53ce-479a-9fea-800be1b47e63",
  "Status": "running",
  "IP": "10.0.0.252, 10.0.0.150",
  "InstanceType": "HighIO",
  "Version": "mongodb 3.2.10",

```

```

"InstanceClass": "1C2G",
"Storage": 5,
"Port": 27017,
"NetworkType": "VPC",
"VpcId": "3c12ccdf-9b8f-4d9b-8aa6-a523897e97a1",
"VnetId": "293c16a5-c757-405c-a693-3b2a3adead50",
"TimingSwitch": "On",
"Timezone": "10:00-11:00",
"TimeCycle": "Mon, Tues, Wed",
"ProductId": null,
"ProductWhat": 1,
"PayType": "byMonth",
"CreateDate": "2017-03-17 15:31:07",
"ExpirationDate": null,
"IamProjectId": "22",
"IamProjectName": "tes9t1"
}
}

```

## RenameMongoDBInstance 重命名MongoDB实例

重命名MongoDB实例，更改实例服务的名称

### 请求参数

关于所有操作需要的通用参数，请参照通用请求-[公共参数](#)

英文名称	中文名称	类型	是否必须	备注
Action	调用接口名称	String	是	取固定值: RenameMongoDBInstance
Version	API版本号	String	是	取固定值: 2017-01-01
InstanceId	服务ID	String	是	
Name	服务名称	String	是	

### 请求方式

PUT

### 错误信息

关于所有操作返回的错误信息，请参照通用请求-[通用错误信息](#)

### 请求样例

```

https://mongodb.${region}.api.ksyun.com/
api?Action=RenameMongoDBInstance
&Version=2017-01-01
&InstanceId=LetMeFly
&Name=xxxxName

```

### 返回样例

```

{
"RequestId": "445b44c0-0dbe-42ee-878d-7fb4083cfdc1",
"InstanceId": "f0f5e95f-9673-4b78-9c21-381b138ffa4d"
}

```

## RestartMongoDBInstance 重启MongoDB实例

重启MongoDB实例，根据实例ID重启MongoDB实例

### 请求参数

关于所有操作需要的通用参数，请参照通用请求-[公共参数](#)

英文名称	中文名称	类型	是否必须	备注
Action	调用接口名称	String	是	取固定值: RestartMongoDBInstance
Version	API版本号	String	是	取固定值: 2017-01-01
InstanceId	服务ID	String	是	

## 请求方式

PUT

## 错误信息

关于所有操作返回的错误信息，请参照通用请求—[通用错误信息](#)

## 请求样例

```
https://mongodb.${region}.api.ksyun.com/
api?Action=RestartMongoDBInstance
&Version=2017-01-01
&InstanceId=LetMeFly
```

## 返回样例

```
{
  "RequestId": "445b44c0-0dbe-42ee-878d-7fb4083cfdc1",
  "InstanceId": "f0f5e95f-9673-4b78-9c21-381b138ffa4d"
}
```

# DescribeMongoDBInstanceNode查询副本集实例节点信息

查询MongoDB实例中的角色信息及连接信息。本接口适用于副本集实例，不适用于分片集群实例。

## 请求参数

关于所有操作需要的通用参数，请参照通用请求—[公共参数](#)

英文名称	中文名称	类型	是否必须	备注
Action	调用接口名称	String	是	取固定值：DescribeMongoDBInstanceNode
Version	API版本号	String	是	取固定值：2017-01-01
InstanceId	服务ID	String	是	

## 请求方式

GET

## 错误信息

关于所有操作返回的错误信息，请参照通用请求—[通用错误信息](#)

## 请求样例

```
Sample Request
http://mongodb.${region}.api.ksyun.com/
api?Action=DescribeMongoDBInstanceNode
&Version=2017-01-01
&InstanceId=07493f09-1161-4442-80e5-e8e8bb1a2ecd
```

## 返回样例

```
{
  "RequestId": "c1c120bf-e9b0-4447-ae85-e07426ebf7c2",
  "MongoDBInstanceNodeResult": [
    {
      "NodeId": "062e66a2-987b-4097-b039-40bee6325c57",
      "Name": "XB-for-CreateTime?--3",
      "Role": "PRIMARY",
      "IP": "10.111.47.174",
      "Port": 27017,
      "Status": "running"
    },
    {
      "NodeId": "4b8c8a31-54b4-495e-8476-6bd0caf2659d",
      "Name": "XB-for-CreateTime?--1",
      "Role": "HIDDEN",
    }
  ]
}
```

```

    "IP": "10.111.47.161",
    "Port": 27017,
    "Status": "running"
  },
  {
    "NodeId": "b29fc8c0-c8ae-4c94-93bf-dffaed706cad",
    "Name": "XB-for-CreateTime?--2",
    "Role": "SECONDARY",
    "IP": "10.111.47.169",
    "Port": 27017,
    "Status": "running"
  }
]
}

```

## DescribeMongoDBShardNode 查询分片集群实例节点信息

查询MongoDB分片集群实例中的角色信息、配置信息以及连接信息。 本接口适用于分片集群实例，不适用于副本集实例。

### 请求参数

关于所有操作需要的通用参数，请参照通用请求-[公共参数](#)

英文名称	中文名称	类型	是否必须	备注
Action	调用接口名称	String	是	取固定值: DescribeMongoDBShardNode
Version	API版本号	String	是	取固定值: 2017-01-01
Instanceid	服务ID	String	是	实例ID

### 请求方式

GET

#### • 请求样例

```

https://mongodb.api.ksyun.com/
api?Action=DescribeMongoDBShardNode
&Version=2017-01-01
&InstanceId=07493f09-1161-4442-80e5-e8e8bb1a2ecd

```

#### • 返回样例

```

{
  "RequestId": "031a9111-057e-4296-818d-74c34d3a86e2",
  "MongosNodeResult": [
    {
      "NodeId": "d7b109c9-f7e6-4293-88f5-e8988b1c990f",
      "Name": "SomebodyCallMe芒果--mongos--1",
      "Role": "MONGOS",
      "Endpoint": "null:27017",
      "Status": "creating",
      "Connections": 200,
      "InstanceClass": "1C2G"
    }
  ],
  "ShardNodeResult": [
    {
      "NodeId": "330eeffa-cbcc-40a6-b623-b18fcb13da19",
      "Name": "SomebodyCallMe芒果-configsvr",
      "Status": "unknown",
      "Disk": 1,
      "Iops": 4800,
      "InstanceClass": "2C4G"
    },
    {
      "NodeId": "fb34f692-3dd4-4aae-913b-5ecab431f1d0",
      "Name": "SomebodyCallMe芒果-shard1",
      "Status": "unknown",
      "Disk": 5,
      "Iops": 4800,
      "InstanceClass": "2C4G"
    }
  ]
}

```

}

## 增加只读节点AddSecondaryInstance

本接口用于新增mongoDB副本集实例的secondary节点数，目前支持3/5/7节点。

### 请求参数

关于所有操作需要的通用参数，请参照通用请求-[公共参数](#)

英文名称	中文名称	类型	是否必须	备注
Action	调用接口名称	String	是	取固定值：AddSecondaryInstance
Version	API版本号	String	是	取固定值：2017-01-01
InstanceId	服务ID	String	是	实例ID
NodeNum	节点数量	String	是	节点数量，取固定值，3、5、7

### 请求方式

PUT

### 错误信息

关于所有操作返回的错误信息，请参照通用请求-[通用错误信息](#)

### 样例

- 请求样例

```
https://mongodb.api.ksyun.com/
api?Action=AddSecondaryInstance
&Version=2017-01-01
&InstanceId=bb9202a1-46fb-4b60-8b45-e425e73bb453
&NodeNum=3
```

- 返回样例

```
{
  "RequestId": "445b44c0-0dbe-42ee-878d-7fb4083cfdc1",
  "InstanceId": "f0f5e95f-9673-4b78-9c21-381b138ffa4d"
}
```

## 创建分片集群实例CreateMongoDBShardInstance

根据需求创建一个分片集实例，此接口只支持创建分片集实例。

### 请求方式

POST

### 请求参数

关于所有操作需要的通用参数，请参照通用请求-[公共参数](#)

英文名称	中文名称	类型	是否必须	备注
Action	调用接口名称	String	是	取固定值：CreateMongoDBShardInstance
Version	API版本号	String	是	取固定值：2017-01-01
Name	实例名称	String	是	支持6-64个中文或者英文字符，包括汉字，大小写字母，数字，下划线和中划线
NetworkType	网络类型	String	否	固定值2，目前只支持创建VPC实例。
VpcId	虚拟专用网络	String	是	VPC网络ID，可在网络控制台获取。
VnetId	终端子网id	String	是	终端子网ID，可在网络控制台获取（注意类型必须为终端子网）。



InstanceAccount	实例管理员账号	String	否	实例管理员账号：默认账户名root
InstancePassword	实例管理员密码	String	是	实例管理员密码
PayType	计费方式：默认为byMonth	String	是	取值范围：byMonth（包年包月），byDay（按日计费），hourlyInstantSettlement（按小时实时结算）。
Duration	时长，默认值：1（单位：月）	Integer	否	时长 默认值：1（单位：月）；PayType=byMonth(包年包月)则必填，最大支持范围是(1~36月)
IamProjectId	项目ID	String	否	默认为0：默认项目
ShardNum	shard数量	Integer	是	取值范围[2-32]
MongosNum	mongos数量	Integer	是	最小为2，范围[2~32]
ShardClass	shard配置	String	是	实例配置，内核cpu配置，客户只能购买特定规格的套餐。详见 <a href="#">实例规格</a>
Storage	磁盘大小	Integer	否	硬盘区间值，单位是GB
MongosClass	mongos配置	String	是	详见 <a href="#">实例规格</a>
DbVersion	数据库版本	String	否	默认值：3.2
AvailabilityZone.N	可用区	String	是	目前暂不支持跨可用区，建议将实例创建在主机的可用区内。可 <a href="#">查询可用机房及可用区</a>

## 错误信息

关于所有操作返回的错误信息，请参照通用请求—[通用错误信息](#)

## 样例

### • 请求样例

```
http://mongodb.api.ksyun.com/
api?Action=CreateMongoDBShardInstance
&Version=2017-01-01
&Name=SomebodyCallMe芒果
&InstancePassword=IamAdmin&NodeNum=3
```

### • 返回样例

```
{
  "RequestId": "T0E1S2T3S-AFQWOFJQ0JF-WFWFWF-FDGD",
  "MongoDBInstanceResult": {
    "UserId": "73403574",
    "Region": "cn-shanghai-3",
    "InstanceId": "e6933d66-7bae-4f43-af82-b74846d6076d",
    "Name": "XB-Test-Create-MongoDB-001"
  }
}
```

# 查询可用机房及可用区DescribeRegions

可调用该接口查询机房的可用区信息。

## 请求参数

关于所有操作需要的通用参数，请参照通用请求—[公共参数](#)

英文名称	中文名称	类型	是否必须	备注
Action	调用接口名称	String	是	取固定值：DescribeRegions
Version	API版本号	String	是	取固定值：2017-01-01

## 请求方式

GET

## 错误信息

关于所有操作返回的错误信息，请参照通用请求—[通用错误信息](#)

## 样例

### • 请求样例

```
http://mongodb.api.ksyun.com/
api?Action=DescribeRegions
&Version=2017-01-01
```

### • 返回样例

```
{
  "Data": {
    "Regions": [
      {
        "Code": "cn-shanghai-3",
        "Name": "上海3区 (VPC)",
        "AvailabilityZones": [
          {
            "Code": "cn-shanghai-3a",
            "Name": "可用区A"
          },
          {
            "Code": "cn-shanghai-3b",
            "Name": "可用区B"
          },
          {
            "Code": "cn-shanghai-3c",
            "Name": "可用区C"
          }
        ]
      }
    ]
  }
}
```

# AddSecurityGroupRule添加安全组规则

## 功能描述

设置实例服务安全规则，为服务添加安全规则

## 请求参数

关于所有操作需要的通用参数，请参照通用请求-[公共参数](#)

英文名称	中文名称	类型	是否必须	备注
Action	调用接口名称	String	是	取固定值: SetCacheSecurityRules
Version	API版本号	String	是	取固定值: 2017-01-01
InstanceId	服务ID	String	是	
cidrs	安全规则IP地址	String	是	0.0.0.0/16, 0.0.0.0/24, 多个","分隔

## 请求方式

POST

## 返回结果

请求的返回信息请参照样例

## 错误信息

关于所有操作返回的错误信息，请参照通用请求-[通用错误信息](#)

## 样例

### • 请求样例

```
https://kcs.${region}.api.ksyun.com/
```

api?Action=AddSecurityGroupRule&Version=2017-01-01&IamProjectId=22&InstanceId=6b99d014-7542-4616-b482-27605364689f&cidrs=0.0.0.0/16

- 返回样例

HttpStatusCode=200

```
{
  "RequestId": "Mongodb-201801251302",
  "MongoDBSecurityGroupRule": [
    {
      "to_port": "27017",
      "cidr": "0.0.0.0/24",
      "from_port": "27017",
      "protocol": "tcp",
      "id": "44"
    },
    {
      "to_port": "27017",
      "cidr": "192.168.1.0/24",
      "from_port": "27017",
      "protocol": "tcp",
      "id": "45"
    },
    {
      "to_port": "27017",
      "cidr": "0.0.0.0/16",
      "from_port": "27017",
      "protocol": "tcp",
      "id": "57"
    }
  ]
}
```

## DeleteCacheSecurityRule

删除实例服务安全规则，根据实例服务的ID，删除对应的安全规则

### 请求参数

关于所有操作需要的通用参数，请参照通用请求-[公共参数](#)

英文名称	中文名称	类型	是否必须	备注
Action	调用接口名称			取固定值: DeleteCacheSecurityRule
Version	API版本号			取固定值: 2017-01-01
InstanceId	缓存服务ID	String	是	--
cidrs	安全规则ID	Integer	是	安全组规则id在白名单列表中获取

### 请求方式

DELETE

### 返回结果

请求的返回信息请参照样例

### 错误信息

关于所有操作返回的错误信息，请参照通用请求-[通用错误信息](#)

### 样例

- 请求样例

https://kcs.\${region}.api.ksyun.com/  
api?Action=DeleteSecurityGroupRule&Version=2017-01-01

- 返回样例

```
{
```

```
"RequestId": "Mongodb-201801251302",
"MongoDBSecurityGroupRule": [
  {
    "to_port": "27017",
    "cidr": "0.0.0.0/24",
    "from_port": "27017",
    "protocol": "tcp",
    "id": "44"
  },
  {
    "to_port": "27017",
    "cidr": "192.168.1.0/24",
    "from_port": "27017",
    "protocol": "tcp",
    "id": "45"
  },
  {
    "to_port": "27017",
    "cidr": "0.0.0.0/16",
    "from_port": "27017",
    "protocol": "tcp",
    "id": "57"
  }
]
```

## ListSecurityGroupRules

查看实例服务安全规则，根据实例ID，查看实例配置的安全规则

### 请求参数

关于所有操作需要的通用参数，请参照通用请求-[公共参数](#)

英文名称	中文名称	类型	是否必须	备注
Action	调用接口名称	String		取固定值：ListSecurityGroupRules
Version	API版本号	String		取固定值：2017-01-01
InstanceId	实例ID	String	是	—

### 请求方式

GET

### 返回结果

请求的返回信息请参照样例

### 错误信息

关于所有操作返回的错误信息，请参照通用请求-[通用错误信息](#)

### 样例

#### • 请求样例

```
https://kcs.${region}.api.ksyun.com/
api?Action=ListSecurityGroupRules&Version=2017-01-01&InstanceId=xxx
```

#### • 返回样例

```
{
  "RequestId": "Mongodb-201801251302",
  "MongoDBSecurityGroupRule": [
    {
      "to_port": "27017",
      "cidr": "0.0.0.0/24",
      "from_port": "27017",
      "protocol": "tcp",
      "id": "44"
    },
    {

```

```

    "to_port": "27017",
    "cidr": "192.168.1.0/24",
    "from_port": "27017",
    "protocol": "tcp",
    "id": "45"
  },
  {
    "to_port": "27017",
    "cidr": "0.0.0.0/16",
    "from_port": "27017",
    "protocol": "tcp",
    "id": "57"
  }
]
}

```

## CreateMongoDBSnapshot 创建手动备份

给实例创建手动快照，手动快照记录最多创建3个

### 请求参数

关于所有操作需要的通用参数，请参照通用请求-[公共参数](#)

英文名称	中文名称	类型	是否必须	备注
Action	调用接口名称	String		取固定值：ListSecurityGroupRules
Version	API版本号	String		取固定值：2017-01-01
InstanceId	实例ID	String	是	—
Name	备份名称	String	是	

### 请求方式

POST

### 返回结果

请求的返回信息请参照样例

### 错误信息

关于所有操作返回的错误信息，请参照通用请求-[通用错误信息](#)

### 样例

#### • 请求样例

```

https://mongodb.${region}.api.ksyun.com/
api?Action=CreateMongoDBSnapshot
&Version=2017-01-01
&InstanceId=LetMeGo
&Name=xxxxxxx

```

#### • 返回样例

```

{
  "RequestId": "T0E1S2T3S-AFQWOFJQ0JF-WFWFWF-FDGD",
  "MongoDBSnapshotResult": {
    "SnapshotId": "abd95c6d-c362-4349-aaff-1a3c892c9f25",
    "Name": "Wagaga",
    "InstanceId": "0e9fe286-64ff-42ae-adb3-653f68b70920",
    "Status": "creating",
    "Create": "2017-03-09 15:54:39"
  }
}

```

## SetMongoDBTimingSnapshot 创建自动备份

进行自动备份设置，可以设置是否开启自动备份，指定备份时间段以及备份周期，定时备份快照最多只可保留7天

## 请求参数

关于所有操作需要的通用参数，请参照通用请求-[公共参数](#)

英文名称	中文名称	类型	是否必须	备注
Action	调用接口名称	String	是	取固定值：ListSecurityGroupRules
Version	API版本号	String	是	取固定值： 2017-01-01
InstanceId	实例ID	String	是	--
TimingSwitch	定时任务开关	String	是	定时任务开关On / Off当TimingSwitch=On时；Timezone与TimeCycle必填当TimingSwitch=Off时；Timezone与TimeCycle将被忽略
Timezone	备份区间	String	是	指备份的时间段，格式如下：08:00-09:00
TimeCycle	备份周期	String	是	每周选择哪几天备份，可以指定多个值

## 请求方式

PUT

## 返回结果

请求的返回信息请参照样例

## 错误信息

关于所有操作返回的错误信息，请参照通用请求-[通用错误信息](#)

## 样例

- 请求样例

```
https://mongodb.${region}.api.ksyun.com/
api?Action=SetMongoDBTimingSnapshot
&Version=2017-01-01
&InstanceId=0e9fe286-64ff-42ae-adb3-653f68b70920
&TimingSwitch=On
&Timezone=10:00-11:00
&TimeCycle=Mon, Tues, Wed
```

- 返回样例

```
{
  "RequestId": "T0E1S2T3S-AFQW0FJQ0JF-WFWWF-FDGD",
  "InstanceId": "0e9fe286-64ff-42ae-adb3-653f68b70920"
}
```

## DeleteMongoDBSnapshot 删除实例备份

根据实例ID, 删除对应的备份

## 请求参数

关于所有操作需要的通用参数，请参照通用请求-[公共参数](#)

英文名称	中文名称	类型	是否必须	备注
Action	调用接口名称	String	是	取固定值：DeleteMongoDBSnapshot
Version	API版本号	String	是	取固定值： 2017-01-01
SnapshotId	备份Id	String	是	--

## 请求方式

DELETE

## 返回结果

请求的返回信息请参照样例

## 错误信息

关于所有操作返回的错误信息，请参照通用请求—[通用错误信息](#)

## 样例

- 请求样例

```
https://mongodb.${region}.api.ksyun.com/  
api?Action=DeleteMongoDBSnapshot  
&Version=2017-01-01  
&SnapshotId=xxxxxxxxxxxx
```

- 返回样例

```
{  
  "RequestId": "T0E1S2T3S-AFQW0FJQ0JF-WFWFWF-FDGD",  
  "SnapshotId": "fd59ealc-62d7-4b31-9aa3-f46843796e04"  
}
```

## DescribeMongoDBSnapshot 查看备份列表

### 请求参数

关于所有操作需要的通用参数，请参照通用请求—[公共参数](#)

英文名称	中文名称	类型	是否必须	备注
Action	调用接口名称	String	是	取固定值：DescribeMongoDBSnapshot
Version	API版本号	String	是	取固定值：2017-01-01
InstanceId	实例Id	String	是	--

### 请求方式

GET

## 返回结果

请求的返回信息请参照样例

## 错误信息

关于所有操作返回的错误信息，请参照通用请求—[通用错误信息](#)

## 样例

- 请求样例

```
https://mongodb.${region}.api.ksyun.com/  
api?Action=DescribeMongoDBSnapshot  
&Version=2017-01-01  
&InstanceId=xxxxxxxxxxxx
```

- 返回样例

```
{  
  "RequestId": "9661b6a7-faa7-4764-91da-7e30921967ad",  
  "MongoDBSnapshotResult": [  
    {  
      "SnapshotId": "466cf210-7c10-4131-ab30-2b4ae5f12576",  
      "Name": "autobackup-20170315100003",  
      "InstanceId": "0e9fe286-64ff-42ae-adb3-653f68b70920",  
    }  
  ]  
}
```

```

    "Type": "Timing",
    "Size": 12,
    "Status": "available",
    "BackupMode": "Physical",
    "CreateTime": "2017-03-15 10:00:03",
    "UpdateTime": "2017-03-15 10:00:16"
  },
  {
    "SnapshotId": "ble50623-9dd6-47be-8eb9-9ab11a7b8449",
    "Name": "autobackup-20170314100019",
    "InstanceId": "0e9fe286-64ff-42ae-adb3-653f68b70920",
    "Type": "Timing",
    "Size": 12,
    "Status": "available",
    "BackupMode": "Physical",
    "CreateTime": "2017-03-14 10:00:19",
    "UpdateTime": "2017-03-14 10:00:52"
  },
  {
    "SnapshotId": "de3ddf0f-580a-48fb-9747-9b3e4c0cacbe",
    "Name": "IamXuebing",
    "InstanceId": "0e9fe286-64ff-42ae-adb3-653f68b70920",
    "Type": "Manual",
    "Size": 12,
    "Status": "available",
    "BackupMode": "Physical",
    "CreateTime": "2017-03-13 18:05:13",
    "UpdateTime": "2017-03-13 18:05:20"
  },
  {
    "SnapshotId": "f5d90ce1-9e3e-42c6-aae3-d689e326845b",
    "Name": "IamXuebing",
    "InstanceId": "0e9fe286-64ff-42ae-adb3-653f68b70920",
    "Type": "Manual",
    "Size": 12,
    "Status": "available",
    "BackupMode": "Physical",
    "CreateTime": "2017-03-14 15:54:57",
    "UpdateTime": "2017-03-14 15:54:59"
  }
]
}

```

## RenameMongoDBSnapshot 备份重命名

### 请求参数

关于所有操作需要的通用参数，请参照通用请求-[公共参数](#)

英文名称	中文名称	类型	是否必须	备注
Action	调用接口名称	String	是	取固定值: DeleteMongoDBSnapshot
Version	API版本号	String	是	取固定值: 2017-01-01
SnapshotId	备份Id	String	是	--
Name	备份名称	String	是	--

### 请求方式

PUT

### 返回结果

请求的返回信息请参照样例

### 错误信息

关于所有操作返回的错误信息，请参照通用请求-[通用错误信息](#)

### 样例

- 请求样例



```
https://mongodb.${region}.api.ksyun.com/  
api?Action=RenameMongoDBSnapshot  
&Version=2017-01-01  
&SnapshotId=xxxxxxx  
&Name=xxxxxxx
```

- 返回样例

```
{  
  "RequestId": "T0E1S2T3S-AFQW0FJQ0JF-WFWFWF-FDGD",  
  "SnapshotId": "fd59ea1c-62d7-4b31-9aa3-f46843796e04"  
}
```