

## 目录

目录	1
SDK for PHP 使用指南	3
1. 概述	3
2. 初始化	3
3. 添加模板	3
4. 修改模板	5
5. 删除模板	6
6. 查询模板列表	6
7. 查询模板详情	6
8. 获取任务列表	6
9. 发起外网拉流	7
10. 停止外网拉流	7
11. 获取配额使用数据	7
SDK for Python 使用指南	8
安装python sdk	8
1. git 安装	8
2. pip 安装	8
3. 通过文件配置及管理密钥	8
4. 创建一个session (初始化)	8
5. 运行环境	8
SDK介绍和使用	8
接口目录说明	8
1. 创建和更新模板	9
2. 删除模板	9
3. 查询模板列表	9
4. 查询模板详情	10
5. 获取任务列表	10
6. 发起外网拉流	10
7. 停止外网拉流	10
8. 获取配额使用数据	10
SDK for Java 使用指南	10
安装java sdk	11
1. git 安装	11
2. mvn 中央仓库安装	11
3. 工程使用sdk录	11
建议使用Maven构建自己的项目，导入ksc-sdk-java, 添加ksc-sdk-java-ket的依赖。	11
新建maven项目	11
进入到pom文件下，进行如下配置	11
4. 通过文件配置及管理密钥	11
本地文件配置：	11
该文件包含下述内容：	11
如不通过本地文件读入ak和sk信息，则需要执行的类文件中增加以下代码：	11
SDK介绍和使用	11
在调用接口前，先初始化 KSCOFFJsonClient，如下用例：	11
SDK介绍和使用	11
接口目录说明	11
注：接口的传递的参数及返回值请参考《视频云直播转码接入说明》API接口说明	12
1. 创建和更新模板	12
2. 删除模板	13
3. 查询模板列表	13

---

4. 查询模板详情	13
5. 获取任务列表	14
6. 发起外网拉流	14
7. 停止外网拉流	14
8. 获取配额使用数据	15

# SDK for PHP 使用指南

## 1. 概述

此 SDK 适用于PHP 5.5 及以上版本。基于直播转码 API 构建。使用此 SDK 构建您的网络应用程序，能让您以非常便捷地方式调用金山云的直播转码服务。

## 2. 初始化

### 2.1 ak/sk配置

在金山云控制台获取到ak/sk后创建以下文件：

```
mkdir ~/.ksyun && vi ~/.ksyun/config
```

config文件内容

```
{
    "ak": "*****",
    "sk": "*****"
}
```

### 2.2 composer安装

```
mkdir test && cd test
```

```
composer require kscsdk/ksyun_sdk
```

如果需要使用最新版本，安装完之后，修改composer.json为

```
{
    "require": {
        "kscsdk/ksyun_sdk": "dev-master"
    }
}
```

然后再执行更新操作

```
composer update
```

### 2.3 调试demo

```
cp vendor/kscsdk/ksyun_sdk/examples/demo_Ket.php .
```

```
php demo_Ket.php GetPresetList
```

### 2.4 调用示例

```
<?php
require('vendor/autoload.php');
use Ksyun\Service\Ket;
$response = Ket::getInstance()->request('GetPresetList');
echo $response->getBody();
```

注：接口的传递的参数及返回值请参考《视频云直播转码接入说明》API接口说明

## 3. 添加模板

调用示例

```
<?php
require('vendor/autoload.php');
```

```
use Ksyun\Service\Ket;

$app = 'live'; // 频道名
$unique = 'test'; // 用户名
$preset = 'preset_demo'; // 模板名
$preset_data = [
    'Preset' => $preset,
    'App' => $app,
    'Description' => 'desc: preset demo',
    'Output' => [
        [
            'format' => [
                'output_format' => 256,
                'abr' => 60000,
                'vbr' => 800000,
                'fr' => 25,
                'logo_switch' => 1
            ]
        ],
        [
            'format' => [
                'output_format' => 257,
                'abr' => 70000,
                'vbr' => 700000,
                'fr' => 23,
                'logo_switch' => 0
            ]
        ]
    ],
    'Video' => [
        'logo' => [
            [
                'pic' => 'wangshuai9/ksyun.png',
                'short_side' => 480,
                'disable_scale' => 0,
                'offsetX' => 10,
                'offsetY' => -20
            ]
        ]
    ]
];

$response = Ket::getInstance()->request('Preset', ['query' => ['Uniqname' => $unique], 'json' => $preset_data]);
echo $response->getBody();
```

## 4. 修改模板

### 调用示例

```
<?php
require('vendor/autoload.php');
use Ksyun\Service\Ket;
$app = 'live'; // 频道名
$unique_name = 'test'; // 用户名
$preset = 'preset_demo'; // 模板名
$preset_data = [
    'Preset' => $preset,
    'App' => $app,
    'Description' => 'desc: preset demo',
    'Output' => [
        [
            'format' => [
                'output_format' => 256,
                'abr' => 60000,
                'vbr' => 800000,
                'fr' => 25,
                'logo_switch' => 1
            ]
        ],
        [
            'format' => [
                'output_format' => 257,
                'abr' => 70000,
                'vbr' => 700000,
                'fr' => 23,
                'logo_switch' => 0
            ]
        ]
    ],
    'Video' => [
        'logo' => [
            [
                'pic' => 'wangshuai9/ksyun.png',
                'short_side' => 480,
                'disable_scale' => 0,
                'offsetX' => 10,
                'offsetY' => -20
            ]
        ]
    ]
];
```

```
    ]
  ]
];

$response = Ket::getInstance()->request('UpdatePreset', ['query' => ['Uniqname' => $uniqname], 'json' => $preset_data]);
echo $response->getBody();
```

## 5. 删除模板

### 调用示例

```
<?php
require('vendor/autoload.php');
use Ksyun\Service\Ket;

$app = 'live'; // 频道名
$uniqname = 'test'; // 用户名
$preset = 'preset_demo'; // 模板名

$response = Ket::getInstance()->request('DelPreset', ['query' => ['Uniqname' => $uniqname, 'App' => $app, 'Preset' => $preset]]);
echo $response->getBody();
```

## 6. 查询模板列表

### 调用示例

```
<?php
require('vendor/autoload.php');
use Ksyun\Service\Ket;

$app = 'live'; // 频道名
$uniqname = 'test'; // 用户名

$response = Ket::getInstance()->request('GetPresetList', ['query' => ['Uniqname' => $uniqname, 'App' => $app]]);
echo $response->getBody();
```

## 7. 查询模板详情

### 调用示例

```
<?php
require('vendor/autoload.php');
use Ksyun\Service\Ket;

$app = 'live'; // 频道名
$uniqname = 'test'; // 用户名
$preset = 'preset_demo'; // 模板名

$response = Ket::getInstance()->request('GetPresetDetail', ['query' => ['Uniqname' => $uniqname, 'App' => $app, 'Preset' => $preset]]);
echo $response->getBody();
```

## 8. 获取任务列表

### 调用示例

```
<?php
require('vendor/autoload.php');
```

```
use Ksyun\Service\Ket;

$app = 'live'; // 频道名

$uniqname = 'test'; // 用户名

$response = Ket::getInstance()->request('GetStreamTranList', ['query' => ['Uniqname' => $uniqname, 'App' => $app]]);

echo $response->getBody();
```

## 9. 发起外网拉流

### 调用示例

```
<?php

require('vendor/autoload.php');

use Ksyun\Service\Ket;

$app = 'live'; // 频道名

$uniqname = 'test'; // 用户名

$streamid = 'stream20170101'; // 流名

$outpull_data = [

    'App' => $app,

    'StreamID' => $streamid,

    'SrcUrl' => 'rtmp://test.rtmp.live.ks-cdn.com/live/streamdemo',

    'Params' => ''

];

$response = Ket::getInstance()->request('StartStreamPull', ['query' => ['Uniqname' => $uniqname], 'json' => $outpull_data]);

echo $response->getBody();
```

## 10. 停止外网拉流

### 调用示例

```
<?php

require('vendor/autoload.php');

use Ksyun\Service\Ket;

$app = 'live'; // 频道名

$uniqname = 'test'; // 用户名

$streamid = 'stream20170101'; // 流名

$response = Ket::getInstance()->request('StopStreamPull', ['query' => ['Uniqname' => $uniqname], 'json' => ['App' => $app, 'StreamID' => $streamid]]);

echo $response->getBody();
```

## 11. 获取配额使用数据

### 调用示例

```
<?php

require('vendor/autoload.php');

use Ksyun\Service\Ket;

$uniqname = 'test'; // 用户名

$response = Ket::getInstance()->request('GetQuotaUsed', ['query' => ['Uniqname' => $uniqname]]);

echo $response->getBody();
```

# SDK for Python 使用指南

## 安装python sdk

### 1. git 安装

```
git clone https://github.com/KscSDK/ksc-sdk-python.git
```

```
cd ksc-sdk-python
```

```
python setup.py install
```

### 2. pip 安装

```
pip install ksc-sdk-python
```

### 3. 通过文件配置及管理密钥

- 所在位置: '/etc/kscore.cfg' 或 './.kscore.cfg'
- 注意: 使用相对路径时, 需与运行目录保持一致
- ks\_access\_key\_id和ks\_secret\_access\_key是金山云控制台身份与管理里面生成密钥对

```
[Credentials]
```

```
ks_access_key_id='xxxxxxxxxxxxxxxxxxxx'
```

```
ks_secret_access_key='xxxxxxxxxxxxxxxxxxxx'
```

### 4. 创建一个session (初始化)

```
from kscore.ket import getKetClient
```

第一种:

```
client = getKetClient("ket", "cn-beijing-6", use_ssl=False)
```

参数说明:

service\_name: 服务名, 填写"ket"

region\_name: 区域名, 填写"cn-beijing-6"

use\_ssl: 是否https访问, 填写False

第二种:

没有配置kscore.cfg调用方式

```
ks_access_key_id='xxxxxxxxxxxxxxxxxxxx'
```

```
ks_secret_access_key='xxxxxxxxxxxxxxxxxxxx'
```

参数: 服务service\_name, 大区region\_name

```
client = getKetClient("ket", "cn-beijing-6", use_ssl=False, ks_access_key_id=ks_access_key_id, ks_secret_access_key=ks_secret_access_key)
```

### 5. 运行环境

适用于2.6、2.7、3.3、3.4的Python版本

## SDK介绍和使用

### 接口目录说明

- [Preset](#) 创建模板
- [DelPreset](#) 删除模板
- [GetPresetList](#) 查询模板列表



- [GetPresetDetail](#) 查询模板详情
- [GetStreamTranList](#) 获取任务列表
- [StartStreamPull](#) 发起外网拉流
- [StopStreamPull](#) 停止外网拉流
- [GetQuotaUsed](#) 获取配额使用数据

注：接口的传递的参数及返回值请参考《视频云直播转码接入说明》API接口说明

## 1. 创建和更新模板

- 接口名: Preset/UpdatePreset
- 调用方式

```
presetParam = {  
  "UniqName": "test",  
  "Preset": presetname,  
  "App": app,  
  "Description": description,  
  "Output": [  
    {  
      "format": {  
        "output_format": 256,  
        "abr": 70000,  
        "vbr": 700000,  
        "fr": 23  
      }  
    },  
    {  
      "format": {  
        "output_format": 257  
      }  
    }  
  ]  
}
```

注:

1. presetParam必须是json格式数据

```
res = client.Preset(presetParam)
```

## 2. 删除模板

- 接口名: DelPreset
- 调用方式

```
res = client.DelPreset(App="live", UniqName="test", Preset=presetname)
```

## 3. 查询模板列表

- 接口名: GetPresetList

- 调用方式

```
res = client.GetPresetList(App="live",UniqName="test")
```

#### 4. 查询模板详情

- 接口名: GetPresetDetail
- 调用方式

```
res = client.GetPresetDetail(App="live",UniqName="test",Preset=presetname)
```

#### 5. 获取任务列表

- 接口名: GetStreamTranList
- 调用方式

```
res = client.GetStreamTranList(App="live",UniqName="test",StreamID="test",OutPull=-1)
```

#### 6. 发起外网拉流

- 接口名: StartStreamPull
- 调用方式

```
StartStreamPullParam = {  
    "UniqName": "test",  
    "App": "live",  
    "StreamID": "testName",  
    "SrcUrl": "test.uplive.ks-cdn.com"  
}
```

注:

1. StartStreamPullParam必须是json格式数据

```
res = client.StartStreamPull(StartStreamPullParam)
```

#### 7. 停止外网拉流

- 接口名: StopStreamPull
- 调用方式

```
StopStreamPullParam = {  
    "UniqName": "test",  
    "App": "live",  
    "StreamID": "testName"  
}
```

注:

1. StopStreamPullParam必须是json格式数据

```
res = client.StopStreamPull(StopStreamPullParam)
```

#### 8. 获取配额使用数据

- 接口名: GetQuotaUsed
- 调用方式

```
res = client.GetQuotaUsed(UniqName="test")
```

## SDK for Java 使用指南

## 安装 java sdk

### 1. git 安装

```
git clone https://github.com/KscSDK/ksc-sdk-java.git
```

```
mvn clean install
```

### 2. mvn 中央仓库安装

设置pom文件即可

### 3. 工程使用 sdk 录

建议使用Maven构建自己的项目，导入ksc-sdk-java，添加ksc-sdk-java-ket的依赖。

新建maven项目

进入到pom文件下，进行如下配置

```
<dependencies>
  <dependency>
    <groupId>com.ksyun</groupId>
    <artifactId>ksc-sdk-java-ket</artifactId>
    <version>0.3.21</version>
  </dependency>
</dependencies>
```

### 4. 通过文件配置及管理密钥

本地文件配置：

```
~/aws/credentials on Linux, OS X or unix
```

```
C:\Users\USERNAME\.aws\credentials on Windows
```

该文件包含下述内容：

```
[default]
aws_access_key_id = your_access_key_id
aws_secret_access_key = your_secret_access_key
```

如不通过本地文件读入ak和sk信息，则需要在执行的类文件中增加以下代码：

```
AWSCredentials credentials = new BasicAWSCredentials(aws_access_key_id, aws_secret_access_key);
```

## SDK介绍和使用

在调用接口前，先初始化 KSCOFFJsonClient，如下用例：

```
KSCOFFJsonClient ksc = new KSCOFFJsonClient();
//可传入AWSCredentials参数 如:KSCOFFJsonClient ksc = new KSCOFFJsonClient(credentials);
ksc.setEndpoint("http://ket.cn-beijing-6.api.ksyun.com/");
字符集: utf-8
```

## SDK介绍和使用

### 接口目录说明

- [Preset](#) 创建和更新模板
- [DelPreset](#) 删除模板
- [GetPresetList](#) 查询模板列表
- [GetPresetDetail](#) 查询模板详情
- [GetStreamTranList](#) 获取任务列表
- [StartStreamPull](#) 发起外网拉流
- [StopStreamPull](#) 停止外网拉流
- [GetQuotaUsed](#) 获取配额使用数据

**注：**接口的传递的参数及返回值请参考《视频云直播转码接入说明》API接口说明

## 1. 创建和更新模板

- 接口名: Preset和UpdatePreset
- 调用方式

```
PresetRequest presetRequest = new PresetRequest();
String data = PresetSet("preset", 2);
presetRequest.setData(data);
ErrResult presetResult = ksc.Preset(presetRequest);
System.out.println("ErrNum: " + presetResult.getErrNum() + ",ErrMsg: " + presetResult.getErrMsg());
//请求内容生成接口
private static String PresetSet(String preset, int num) throws JSONException {
    int formatSet[] = new int[] { 256, 257, 258, 259 };
    JSONObject data = new JSONObject();
    JSONArray output = new JSONArray();
    JSONArray logo = new JSONArray();
    JSONObject video = new JSONObject();
    data.put("UniqName", "mytest");
    data.put("Preset", preset);
    data.put("Description", "desc:" + preset);
    data.put("App", "live");
    for (int i = 0; i < num; i++) {
        JSONObject output_tmp = new JSONObject();
        JSONObject format = new JSONObject();
        JSONObject _switch = new JSONObject();
        format.put("output_format", formatSet[i]);
        format.put("vbr", 800000);
        _switch.put("sn", 0);
        _switch.put("sm", 0);
        _switch.put("hv", 0);
        output_tmp.put("format", format);
        output_tmp.put("switch", _switch);
    }
}
```

```
        output.put(output_tmp);
    }
    data.put("Output", output);
    video.put("orientationAdapt", 1);
    for (int i = 0; i < num; i++) {
        JSONObject logo_tmp = new JSONObject();
        logo_tmp.put("pic", "/wangshuai9/1.png");
        logo_tmp.put("short_side", 640);
        logo.put(logo_tmp);
    }
    video.put("logo", logo);
    data.put("Video", video);
    return data.toString();
}
```

## 2. 删除模板

- 接口名: DelPreset
- 调用方式

```
DelPresetRequest delpresetRequest = new DelPresetRequest();
delpresetRequest.setApp("live");
delpresetRequest.setUniqName("mytest");
delpresetRequest.setPreset("preset");
ErrResult delpresetResult = ksc.DelPreset(delpresetRequest);
System.out.println("ErrNum: " + delpresetResult.getErrNum() + ",ErrMsg: " + delpresetResult.getErrMsg());
```

## 3. 查询模板列表

- 接口名: GetPresetList
- 调用方式

```
GetPresetListRequest getPresetListRequest = new GetPresetListRequest();
getPresetListRequest.setApp("live");
getPresetListRequest.setUniqName("mytest");
GetPresetListResult getpresetlistResult = ksc.GetPresetList(getPresetListRequest);
```

## 4. 查询模板详情

- 接口名: GetPresetDetail
- 调用方式

```
GetPresetDetailRequest getPresetDetailRequest = new GetPresetDetailRequest();
getPresetDetailRequest.setApp("live");
getPresetDetailRequest.setUniqName("mytest");
getPresetDetailRequest.setPreset("preset");
GetPresetDetailResult getPresetDetailResult = ksc.GetPresetDetail(getPresetDetailRequest);
```

## 5. 获取任务列表

- 接口名: GetStreamTranList
- 调用方式

```
GetStreamTranListRequest getStreamTranListRequest = new GetStreamTranListRequest();
getStreamTranListRequest.setApp("live");
getStreamTranListRequest.setUniqName("mytest");
GetStreamTranListResult getStreamTranListResult = ksc.GetStreamTranList(getStreamTranListRequest);
```

## 6. 发起外网拉流

- 接口名: StartStreamPull
- 调用方式

```
StartStreamPullRequest startStreamPullRequest = new StartStreamPullRequest();
String data1 = StreamPullSet("test123");
startStreamPullRequest.setData(data1);
ErrResult startStreamPullResult = ksc.StartStreamPull(startStreamPullRequest);
System.out.println("ErrNum: " + startStreamPullResult.getErrNum() + ",ErrMsg: " + startStreamPullResult.getErrMsg());
//请求内容接口
private static String StreamPullSet(String StreamID) {
    JSONObject data = new JSONObject();
    data.put("UniqName", "mytest");
    data.put("StreamID", StreamID);
    if (type == 0) {
        data.put("SrcUrl", "test.uplive.ks-cdn.com");
    }
    data.put("App", "live");
    return data.toString();
}
```

## 7. 停止外网拉流

- 接口名: StopStreamPull
- 调用方式

```
StopStreamPullRequest stopStreamPullRequest = new StopStreamPullRequest();
String data2 = StreamPullSet("test123");
stopStreamPullRequest.setData(data2);
ErrResult stopStreamPullResult = ksc.StopStreamPull(stopStreamPullRequest);
System.out.println("ErrNum: " + stopStreamPullResult.getErrNum() + ",ErrMsg: " + stopStreamPullResult.getErrMsg());
//请求内容接口
private static String StreamPullSet(String StreamID) {
    JSONObject data = new JSONObject();
    data.put("UniqName", "mytest");
    data.put("StreamID", StreamID);
```

```
if (type == 0) {  
    data.put("SrcUrl", "test.uplive.ks-cdn.com");  
}  
data.put("App", "live");  
return data.toString();  
}
```

## 8. 获取配额使用数据

- 接口名: GetQuotaUsed
- 调用方式

```
GetQuotaUsedRequest getQuotaUsedRequest = new GetQuotaUsedRequest();  
getQuotaUsedRequest.setUniqName(UniqName);  
GetQuotaUsedResult getQuotaUsedResult = ksc.GetQuotaUsed(getQuotaUsedRequest);  
System.out.println("ErrNum: " + getQuotaUsedResult.getErrNum() + ", ErrMsg: " + getQuotaUsedResult.getErrMsg());
```