

创建分类

接口描述

- 1、使用子账号创建视频分类

2、接口支持https协议

接口URL

http://api.polyv.net/v3/category/add

[在线API调用](#)

请求方式

POST

接口约束

1、接口同时支持HTTP 、HTTPS ， 建议使用HTTPS 确保接口安全， 接口调用有频率限制， [详细请查看](#)

请求参数描述

参数名	必选	类型	说明
appId	true	String	子账号appId
timestamp	true	Long	当前时间的毫秒级时间戳， 3分钟内有效
sign	true	String	签名， 32位大写MD5值【详见 MD5签名生成规则 】
name	true	String	分类名

参数名	必选	类型	说明
parentId	false	String	父分类id，默认为1，表示默认分类

示例

```
http://api.polyv.net/v3/category/add
```

表单参数：

```
appId=xxxxxx&name=%E6%B5%8B%E8%AF%95%E5%88%9B%E5%BB%BA%E5%88%86%E7%B1%BB&sign=FC67B8973F35D00F9E843C380880DFFD&parentId=1617850424921&timestamp=1617850576238
```

响应参数描述

参数名	类型	说明
code	Integer	响应状态码，200为成功返回，非200为失败【详见 全局错误说明 】
status	String	响应状态文本信息
error	Object	响应成功返回null，响应失败返回错误描述信息【详见 error参数描述 】
requestId	String	每次请求的业务流水号，必须唯一，便于客户端/服务器端排查问题
data	String	响应成功返回成功创建的分类id，响应失败返回null

error参数描述

字段	类型	说明
code	Integer	错误码
desc	String	错误描述

Java请求示例

快速接入基础代码请[下载相关依赖源码](#)，[点击下载源代码](#)，下载后加入到自己的源码工程中即可。测试用例中的**HttpUtil.java** 和 **VodSignUtil.java** 都包含在下载文件中。

强烈建议您使用[点播Java SDK](#)完成API的功能对接，点播Java SDK 对API调用逻辑、异常处理、数据签名、HTTP请求线程池进行了统一封装和优化。

```
private static final Logger log = LoggerFactory.getLogger(VodSubAccountTest.class);
/**
 * 创建分类
 * @throws Exception
 * @throws NoSuchAlgorithmException
 */
@Test
public void testSubAddCategory() throws Exception, NoSuchAlgorithmException {
    //公共参数,填写自己的实际参数
    String appId = super.appId;
    String appSecret = super.appSecret;
    String timestamp = String.valueOf(System.currentTimeMillis());
    //业务参数
    String url = "http://api.polyv.net/v3/category/add";
    String name = "测试创建分类";
    String parentId = "1617850424921";

    Map<String, String> requestMap = new HashMap<>();
    requestMap.put("appId", appId);
    requestMap.put("timestamp", timestamp);
    requestMap.put("name", name);
    requestMap.put("parentId", parentId);
    requestMap.put("sign", VodSignUtil.getSignMd5(requestMap, appSecret));

    String response = HttpUtil.postFormBody(url, requestMap);
    log.debug("测试创建分类,{}", response);
    //do somethings
}
```

响应示例

系统全局错误说明详见[全局错误说明](#)

成功示例

```
{
  "requestId": "00543c3f-2d3d-4765-85ca-b332f1021362",
  "code": 200,
  "status": "success",
  "error": null,
  "data": 1617850556356
}
```

异常示例

时间戳错误

```
{
  "requestId": "5ede3d23-45f0-4661-99be-e95d07daf049",
  "code": 400,
  "status": "error",
  "error": {
    "code": 100,
    "desc": "invalid timestamp"
  },
  "data": null
}
```

appId不正确

```
{
  "requestId": "5ede3d23-45f0-4661-99be-e95d07daf049",
  "code": 400,
  "status": "error",
  "error": {
    "code": 101,
    "desc": "application not exist"
  },
  "data": null
}
```

子账号不存在

```
{
  "requestId": "5ede3d23-45f0-4661-99be-e95d07daf049",
  "code": 400,
  "status": "error",
  "error": {
    "code": 102,
    "desc": "user children not exist"
  },
  "data": null
}
```

子账号过期

```
{
  "requestId": "5ede3d23-45f0-4661-99be-e95d07daf049",
  "code": 403,
  "status": "error",
  "error": {
    "code": 103,
    "desc": "user children expired"
  },
  "data": null
}
```

签名错误

```
{
  "requestId": "5ede3d23-45f0-4661-99be-e95d07daf049",
  "code": 403,
  "status": "error",
  "error": {
    "code": 104,
    "desc": "invalid signature"
  },
  "data": null
}
```

账号权限不足

```
{
  "requestId": "5ede3d23-45f0-4661-99be-e95d07daf049",
  "code": 403,
  "status": "error",
  "error": {
    "code": 105,
    "desc": "permission limited"
  },
  "data": null
}
```

分类名称不能为空

```
{
  "requestId": "5ede3d23-45f0-4661-99be-e95d07daf049",
  "code": 400,
  "status": "error",
  "error": {
    "code": 3000,
    "desc": "没有填写栏目名称"
  },
  "data": null
}
```

父分类不能为空

```
{
  "requestId": "5ede3d23-45f0-4661-99be-e95d07daf049",
  "code": 400,
  "status": "error",
  "error": {
    "code": 3001,
    "desc": "category not exist"
  },
  "data": null
}
```

子账号没有权限在该目录下创建分类

```
{
  "requestId": "5ede3d23-45f0-4661-99be-e95d07daf049",
  "code": 400,
  "status": "error",
  "error": {
    "code": 3002,
    "desc": "子账号没有权限在该目录下创建分类"
  },
  "data": null
}
```

已存在同名的视频分类

```
{
  "requestId": "df5750d9-27b0-4390-ad7c-f0d1f86d0d14",
  "code": 400,
  "status": "error",
  "error": {
    "code": 3003,
    "desc": "已存在同名的视频分类"
  },
  "data": null
}
```