

# 1、查询单个视频信息

## 描述

通过视频id查询单个视频的信息

接口地址（仅做说明使用）：<https://api.polyv.net/v2/video/%s/get-video-msg>

## 调用约束

1、接口调用有频率限制，[详细请查看](#)，调用常见异常，[详细请查看](#)

## 单元测试

```
@Test
public void testGetVideo() throws IOException, NoSuchAlgorithmException {
    VodGetVideoRequest vodGetVideoRequest = new VodGetVideoRequest();
    VodGetVideoResponse vodGetVideoResponse = null;
    try {
        vodGetVideoRequest.setVideoId(super.getTestVideoId());
        vodGetVideoResponse = new VodInfoServiceImpl().getVideo(vodGetVideoRequest);
        Assert.assertNotNull(vodGetVideoResponse);
        if (vodGetVideoResponse != null) {
            log.debug("测试查询单个视频信息成功,{}", JSON.toJSONString(vodGetVideoResponse));
        }
    } catch (PloyvSdkException e) {
        //参数校验不合格 或者 请求服务器端500错误, 错误信息见PloyvSdkException.getMessage()
        log.error(e.getMessage(), e);
        // 异常返回做B端异常的业务逻辑, 记录log 或者 上报到ETL 或者回滚事务
        throw e;
    } catch (Exception e) {
        log.error("SDK调用异常", e);
        throw e;
    }
}
```

## 单元测试说明

1、请求正确，返回VodGetVideoResponse对象，B端依据此对象处理业务逻辑；

2、请求参数校验不合格，抛出PloyvSdkException，错误信息见PloyvSdkException.getMessage()，如 [ 输入参数 [xxx.chat.VodxxxRequest]对象校验失败，失败字段 [pic不能为空 / msg不能为空] ]

3、服务器处理异常，抛出PloyvSdkException， 错误信息见PloyvSdkException.getMessage()， 如 [ 保利威请求返回数据错误， 请求流水号： 66e7ad29fd04425a84c2b2b562d2025b， 错误原因： invalid signature. ]

请求入参描述

| 参数名     | 必选   | 类型     | 说明                  |
|---------|------|--------|---------------------|
| videoid | true | String | 视频ID【对应api文档的vid字段】 |

返回对象描述

| 参数名                | 类型       | 说明                                                |
|--------------------|----------|---------------------------------------------------|
| swfLink            | String   | 返回flash链接【对应api文档的swf_link字段】                     |
| tag                | String   | 视频标签                                              |
| mp4                | String   | MP4源文件                                            |
| playerWidth        | Integer  | 视频宽度【对应api文档的playerwidth字段】                       |
| title              | String   | 视频标题                                              |
| duration           | String   | 视频时长,如： 00:00:48                                  |
| filesize           | Long[]   | 编码后各个清晰度视频的文件大小，类型为array                          |
| firstImage         | String   | 视频首图【对应api文档的first_image字段】                       |
| times              | Integer  | 播放次数                                              |
| context            | String   | 视频描述                                              |
| originalDefinition | String   | 最佳分辨率， 如： 1280x720【对应api文档的original_definition字段】 |
| images             | String[] | 视频截图                                              |

| 参数名            | 类型       | 说明                                                     |
|----------------|----------|--------------------------------------------------------|
| playerHeight   | Integer  | 视频高度【对应api文档的 <b>playerheight</b> 字段】                  |
| uploadTime     | Date     | 上传时间，格式：yyyy-MM-dd HH:mm:ss【对应api文档的 <b>ptime</b> 字段】  |
| videoid        | String   | 视频id【对应api文档的 <b>vid</b> 字段】                           |
| previewVideoid | String   | 预览视频id【对应api文档的 <b>previewVid</b> 字段】                  |
| categoryId     | String   | 分类id，如1为根目录【对应api文档的 <b>cataid</b> 字段】                 |
| defaultVideo   | String   | 用户默认播放视频【对应api文档的 <b>default_video</b> 字段】             |
| df             | Integer  | 视频码率数                                                  |
| SDFlv          | String   | 流畅码率flv格式视频地址【对应api文档的 <b>flv1</b> 字段】                 |
| HDFlv          | String   | 高清码率flv格式视频地址【对应api文档的 <b>flv2</b> 字段】                 |
| FHDFlv         | String   | 超清码率flv格式视频地址【对应api文档的 <b>flv3</b> 字段】                 |
| SDMp4          | String   | 流畅码率mp4格式视频地址【对应api文档的 <b>mp4_1</b> 字段】                |
| HDmp4          | String   | 高清码率mp4格式视频地址【对应api文档的 <b>mp4_2</b> 字段】                |
| FHDmp4         | String   | 超清码率mp4格式视频地址【对应api文档的 <b>mp4_3</b> 字段】                |
| hls            | String[] | 索引文件，记录每个清晰度的m3u8的链接，数组第一个为流畅链接，第二个为高清，第三个为超清，数组长度为1-3 |
| SDHls          | String   | 流畅清晰度的m3u8                                             |

| 参数名            | 类型       | 说明                                                                     |
|----------------|----------|------------------------------------------------------------------------|
| HDHls          | String   | 高清清晰度的m3u8                                                             |
| FHDHls         | String   | 超清清晰度的m3u8                                                             |
| imagesBig      | String[] | 视频截图大图地址【对应api文档的 <b>images_b</b> 字段】                                  |
| seed           | Integer  | 加密视频为1，非加密为0                                                           |
| status         | Integer  | 视频状态：60/61已发布；10等待编码；20正在编码；50等待审核；51审核不通过；-1已删除；                      |
| keepSource     | Integer  | 是否为源文件，否：0,是：1【对应api文档的 <b>keepsourcesource</b> 字段】                    |
| uploader       | Uploader | 上传者信息【详见 <a href="#">Uploader参数描述</a> 】                                |
| hlsLevel       | String   | 加密等级 open:非授权加密 web:web授权 app: app授权 wxa_app:小程序授权                     |
| categoryName   | String   | 分类名称【对应api文档的 <b>cataname</b> 字段】                                      |
| imageUrls      | Array    | 视频截图小图url                                                              |
| sourceFileSize | String   | 源视频文件大小，单位为：byte【对应api文档的 <b>source_filesize</b> 字段】                   |
| md5Checksum    | String   | 上传到POLYV云平台的视频源文件的MD5值，可以用来校验是否上传错误或完整【对应api文档的 <b>md5checksum</b> 字段】 |

## Uploader参数描述

| 参数名   | 类型     | 说明    |
|-------|--------|-------|
| email | String | 上传者邮箱 |
| name  | String | 上传者名称 |

| 参数名  | 类型     | 说明                 |
|------|--------|--------------------|
| role | String | 上传者角色,如管理员,上传者,主账号 |

## 2、查询视频授权播放开关

### 描述

通过视频id查询视频授权播放开关状态

接口地址（仅做说明使用）：<https://api.polyv.net/v2/video/%s/authplay-status>

### 调用约束

- 1、接口调用有频率限制，[详细请查看](#)，调用常见异常，[详细请查看](#)

## 单元测试

```
@Test
public void testGetVideoPlayStatus() throws IOException, NoSuchAlgorithmException {
    VodGetVideoPlayStatusRequest vodGetVideoPlayStatusRequest = new VodGetVideoPlayStatusRequest();
    Boolean vodGetVideoPlayStatusResponse = null;
    try {
        vodGetVideoPlayStatusRequest
            //可通过 new VodQueryServiceImpl().queryVideoList()获取
            .setVideoId(super.getTestVideoId());
        vodGetVideoPlayStatusResponse = new
VodInfoServiceImpl().getVideoPlayStatus(vodGetVideoPlayStatusRequest);
        Assert.assertTrue(vodGetVideoPlayStatusResponse);
        if (vodGetVideoPlayStatusResponse) {
            log.debug("测试查询视频授权播放开关成功");
        }
    } catch (PloyvSdkException e) {
        //参数校验不合格 或者 请求服务器端500错误, 错误信息见PloyvSdkException.getMessage()
        log.error(e.getMessage(), e);
        // 异常返回做B端异常的业务逻辑, 记录log 或者 上报到ETL 或者回滚事务
        throw e;
    } catch (Exception e) {
        log.error("SDK调用异常", e);
        throw e;
    }
}
```

## 单元测试说明

- 1、请求正确，返回Boolean对象，B端依据此对象处理业务逻辑；
- 2、请求参数校验不合格，抛出PloyvSdkException，错误信息见PloyvSdkException.getMessage()，如 [ 输入参数 [xxx.chat.VodxxxRequest]对象校验失败，失败字段 [pic不能为空 / msg不能为空] ]
- 3、服务器处理异常，抛出PloyvSdkException，错误信息见PloyvSdkException.getMessage()，如 [ 保利威请求返回数据错误，请求流水号：66e7ad29fd04425a84c2b2b562d2025b，错误原因： invalid signature. ]

## 请求入参描述

| 参数名     | 必选   | 类型     | 说明                   |
|---------|------|--------|----------------------|
| videoid | true | String | 视频ID 【对应api文档的vid字段】 |

## 返回对象描述

true为开关开启，false为开关关闭

---

## 3、查询视频时长和大小

### 描述

通过视频id或分类id查询视频的时长和大小

接口地址（仅做说明使用）：<https://api.polyv.net/v2/video/%s/getSizeByCata>

### 调用约束

- 1、接口调用有频率限制，[详细请查看](#)，调用常见异常，[详细请查看](#)
- 2、当传了videoids时，按照videoids查询；当仅传categoryIds时，按照categoryIds查询；videoids和categoryIds不能同时为空；同时传以videoids为准

## 单元测试

```
@Test
public void testGetVideoSize() throws IOException, NoSuchAlgorithmException {
    VodGetVideoSizeRequest vodGetVideoSizeRequest = new VodGetVideoSizeRequest();
    List<VodGetVideoSizeResponse> vodGetVideoSizeResponseList = null;
    try {
        vodGetVideoSizeRequest.setVideoIds(super.getTestVideoId()).setCategoryIds("1602300731843");
        vodGetVideoSizeResponseList = new VodInfoServiceImpl().getVideoSize(vodGetVideoSizeRequest);
        Assert.assertNotNull(vodGetVideoSizeResponseList);
        if (vodGetVideoSizeResponseList != null) {
            log.debug("测试根据分类批量查询视频时长和大小成功,{}",
JSON.toJSONString(vodGetVideoSizeResponseList));
        }
    } catch (PloyvSdkException e) {
        //参数校验不合格 或者 请求服务器端500错误, 错误信息见PloyvSdkException.getMessage()
        log.error(e.getMessage(), e);
        // 异常返回做B端异常的业务逻辑, 记录log 或者 上报到ETL 或者回滚事务
        throw e;
    } catch (Exception e) {
        log.error("SDK调用异常", e);
        throw e;
    }
}
```

## 单元测试说明

- 1、请求正确，返回VodGetVideoSizeResponse对象，B端依据此对象处理业务逻辑；
- 2、请求参数校验不合格，抛出PloyvSdkException，错误信息见PloyvSdkException.getMessage()，如 [ 输入参数 [xxx.chat.VodxxxRequest]对象校验失败，失败字段 [pic不能为空 / msg不能为空] ]
- 3、服务器处理异常，抛出PloyvSdkException，错误信息见PloyvSdkException.getMessage()，如 [ 保利威请求返回数据错误，请求流水号：66e7ad29fd04425a84c2b2b562d2025b，错误原因： invalid signature. ]

## 请求入参描述

| 参数名      | 必选   | 类型     | 说明                                                             |
|----------|------|--------|----------------------------------------------------------------|
| videoids | true | String | 多个视频ID(英文逗号分割状态为半角)，例如 1b8be3,239c2e 【对应api文档的 <b>vids</b> 字段】 |



| 参数名         | 必选    | 类型     | 说明                                                                                                |
|-------------|-------|--------|---------------------------------------------------------------------------------------------------|
| categoryIds | false | String | 多个分类ID(逗号分割)，当传了vids时，按照vids查询；当仅传cataid时，按照cataid查询；vids和cataid不能同时为空【对应api文档的 <b>cataid</b> 字段】 |

## 返回对象描述

返回对象是List<VodGetVideoSizeResponse>，**VodGetVideoSizeResponse**具体元素内容如下：

| 参数名        | 类型     | 说明                                    |
|------------|--------|---------------------------------------|
| categoryId | String | 分类ID【对应api文档的 <b>cataid</b> 字段】       |
| videos     | Array  | 视频结果列表【详见 <a href="#">Video参数描述</a> 】 |

## Video参数描述

| 参数名       | 类型     | 说明                           |
|-----------|--------|------------------------------|
| videoid   | String | 视频ID【对应api文档的 <b>vid</b> 字段】 |
| duration  | String | 时长，格式为hh:mm:ss。例如00:03:11    |
| filesize1 | Long   | 编码后码率1FLV的大小，单位为Bytes：字节     |
| filesize2 | Long   | 编码后码率2FLV的大小，单位为Bytes：字节     |
| filesize3 | Long   | 编码后码率3FLV的大小，单位为Bytes：字节     |

## 4、查询微信分享页的视频信息

### 描述

通过视频id查询微信分享页的视频信息

接口地址（仅做说明使用）：<https://api.polyv.net/v2/video/wechat-share/%s/video-info>

### 调用约束

- 1、接口调用有频率限制，[详细请查看](#)，调用常见异常，[详细请查看](#)

### 单元测试

```
@Test
public void testGetWeChatShareVideoInfo() throws IOException, NoSuchAlgorithmException {
    VodGetWeChatShareVideoInfoRequest vodGetWeChatShareVideoInfoRequest = new
VodGetWeChatShareVideoInfoRequest();
    VodGetWeChatShareVideoInfoResponse vodGetWeChatShareVideoInfoResponse = null;
    try {
        vodGetWeChatShareVideoInfoRequest.setVideoId(super.getTestVideoId());
        vodGetWeChatShareVideoInfoResponse = new VodInfoServiceImpl().getWeChatShareVideoInfo(
            vodGetWeChatShareVideoInfoRequest);
        Assert.assertNotNull(vodGetWeChatShareVideoInfoResponse);
        if (vodGetWeChatShareVideoInfoResponse != null) {
            log.debug("测试查询微信分享页的视频相关信息成功,{}",
JSON.toJSONString(vodGetWeChatShareVideoInfoResponse));
        }
    } catch (PloyvSdkException e) {
        //参数校验不合格 或者 请求服务器端500错误，错误信息见PloyvSdkException.getMessage()
        log.error(e.getMessage(), e);
        // 异常返回做B端异常的业务逻辑，记录log 或者 上报到ETL 或者回滚事务
        throw e;
    } catch (Exception e) {
        log.error("SDK调用异常", e);
        throw e;
    }
}
```

### 单元测试说明

- 1、请求正确，返回VodGetWeChatShareVideoInfoResponse对象，B端依据此对象处理业务逻辑；

- 2、请求参数校验不合格，抛出PloyvSdkException，错误信息见PloyvSdkException.getMessage()，如 [ 输入参数 [xxx.chat.VodxxxRequest]对象校验失败，失败字段 [pic不能为空 / msg不能为空] ]
- 3、服务器处理异常，抛出PloyvSdkException，错误信息见PloyvSdkException.getMessage()，如 [ 保利威请求返回数据错误，请求流水号：66e7ad29fd04425a84c2b2b562d2025b，错误原因： invalid signature. ]

请求入参描述

| 参数名     | 必选   | 类型     | 说明                  |
|---------|------|--------|---------------------|
| videoid | true | String | 视频ID【对应api文档的vid字段】 |

返回对象描述

| 参数名               | 类型      | 说明       |
|-------------------|---------|----------|
| videoCoverImg     | String  | 视频封面图    |
| videoTitle        | String  | 视频微信分享标题 |
| videoDesc         | String  | 视频描述     |
| videoIcon         | String  | 视频图标     |
| originalPlayTimes | Integer | 初始播放量    |
| originalLikeNum   | Integer | 初始点赞量    |

## 5、查询视频播放预览时长

### 描述

通过视频id查询视频播放预览时长

接口地址（仅做说明使用）：<https://api.polyv.net/v2/video/%s/get-preview-duration>

### 调用约束

1、接口调用有频率限制，[详细请查看](#)，调用常见异常，[详细请查看](#)

### 单元测试

```
@Test
public void testGetVideoPreviewDuration() throws IOException, NoSuchAlgorithmException {
    VodGetVideoPreviewDurationRequest vodGetVideoPreviewDurationRequest = new
VodGetVideoPreviewDurationRequest();
    Integer vodGetVideoPreviewDurationResponse = null;
    try {
        vodGetVideoPreviewDurationRequest.setVideoId(super.getTestVideoId());
        vodGetVideoPreviewDurationResponse = new VodInfoServiceImpl().getVideoPreviewDuration(
            vodGetVideoPreviewDurationRequest);
        Assert.assertNotNull(vodGetVideoPreviewDurationResponse);
        if (vodGetVideoPreviewDurationResponse != null) {
            log.debug("测试查询视频播放预览时长成功,{}",
JSON.toJSONString(vodGetVideoPreviewDurationResponse));
        }
    } catch (PloyvSdkException e) {
        //参数校验不合格 或者 请求服务器端500错误, 错误信息见PloyvSdkException.getMessage()
        log.error(e.getMessage(), e);
        // 异常返回做B端异常的业务逻辑, 记录log 或者 上报到ETL 或者回滚事务
        throw e;
    } catch (Exception e) {
        log.error("SDK调用异常", e);
        throw e;
    }
}
```

### 单元测试说明

1、请求正确，返回Integer对象，B端依据此对象处理业务逻辑；

2、请求参数校验不合格，抛出PloyvSdkException，错误信息见PloyvSdkException.getMessage()，如 [ 输入参数 [xxx.chat.VodxxxRequest]对象校验失败, 失败字段 [pic不能为空 / msg不能为空] ]

3、服务器处理异常，抛出PloyvSdkException，错误信息见PloyvSdkException.getMessage()，如 [ 保利威请求返回数据错误，请求流水号：66e7ad29fd04425a84c2b2b562d2025b，错误原因： invalid signature. ]

请求入参描述

| 参数名     | 必选   | 类型     | 说明                  |
|---------|------|--------|---------------------|
| videoid | true | String | 视频ID【对应api文档的vid字段】 |

返回对象描述

视频播放预览时长，单位：秒

6、查询单个视频的首图

描述

通过视频id查询单个视频的首图

接口地址（仅做说明使用）：<https://api.polyv.net/v2/video/%s/get-image>

调用约束

1、接口调用有频率限制，[详细请查看](#)，调用常见异常，[详细请查看](#)

## 单元测试

```
@Test
public void testGetVideoFirstImage() throws IOException, NoSuchAlgorithmException {
    VodGetVideoFirstImageRequest vodGetVideoFirstImageRequest = new VodGetVideoFirstImageRequest();
    String vodGetVideoFirstImageResponse = null;
    try {
        vodGetVideoFirstImageRequest.setVideoId(super.getTestVideoId()).setThumbnail(1);
        vodGetVideoFirstImageResponse = new
VodInfoServiceImpl().getVideoFirstImage(vodGetVideoFirstImageRequest);
        Assert.assertNotNull(vodGetVideoFirstImageResponse);
        if (vodGetVideoFirstImageResponse != null) {
            log.debug("测试查询单个视频的首图成功,{})", vodGetVideoFirstImageResponse);
        }
    } catch (PloyvSdkException e) {
        //参数校验不合格 或者 请求服务器端500错误, 错误信息见PloyvSdkException.getMessage()
        log.error(e.getMessage(), e);
        // 异常返回做B端异常的业务逻辑, 记录log 或者 上报到ETL 或者回滚事务
        throw e;
    } catch (Exception e) {
        log.error("SDK调用异常", e);
        throw e;
    }
}
```

## 单元测试说明

- 1、请求正确，返回String对象，B端依据此对象处理业务逻辑；
- 2、请求参数校验不合格，抛出PloyvSdkException，错误信息见PloyvSdkException.getMessage()，如 [ 输入参数 [xxx.chat.VodxxxRequest]对象校验失败，失败字段 [pic不能为空 / msg不能为空] ]
- 3、服务器处理异常，抛出PloyvSdkException，错误信息见PloyvSdkException.getMessage()，如 [ 保利威请求返回数据错误，请求流水号：66e7ad29fd04425a84c2b2b562d2025b，错误原因： invalid signature. ]

## 请求入参描述

| 参数名     | 必选   | 类型     | 说明                  |
|---------|------|--------|---------------------|
| videold | true | String | 视频ID【对应api文档的vid字段】 |

| 参数名       | 必选    | 类型      | 说明                                                  |
|-----------|-------|---------|-----------------------------------------------------|
| thumbnail | false | Integer | 是否为视频首图的缩略图，值为1：是；值为0：否。默认为0：非视频首图的缩略图【对应api文档的t字段】 |

返回对象描述

首图地址

---

7、查询视频密码

描述

通过视频id查询视频密码  
接口地址（仅做说明使用）：<https://api.polyv.net/v2/video/%s/video-setting-page>

调用约束

1、接口调用有频率限制，[详细请查看](#)，调用常见异常，[详细请查看](#)

## 单元测试

```
@Test
public void testQueryVideoPassword() throws IOException, NoSuchAlgorithmException {
    VodQueryVideoPasswordRequest vodQueryVideoPasswordRequest = new VodQueryVideoPasswordRequest();
    VodQueryVideoPasswordResponse vodQueryVideoPasswordResponse = null;
    try {
        vodQueryVideoPasswordRequest.setVideoId(super.getTestVideoId());
        vodQueryVideoPasswordResponse = new
VodInfoServiceImpl().queryVideoPassword(vodQueryVideoPasswordRequest);
        Assert.assertNotNull(vodQueryVideoPasswordResponse);
        if (vodQueryVideoPasswordResponse != null) {
            log.debug("测试查询视频密码成功,{})", JSON.toJSONString(vodQueryVideoPasswordResponse));
        }
    } catch (PloyvSdkException e) {
        //参数校验不合格 或者 请求服务器端500错误, 错误信息见PloyvSdkException.getMessage()
        log.error(e.getMessage(), e);
        // 异常返回做B端异常的业务逻辑, 记录log 或者 上报到ETL 或者回滚事务
        throw e;
    } catch (Exception e) {
        log.error("SDK调用异常", e);
        throw e;
    }
}
```

## 单元测试说明

- 1、请求正确，返回VodQueryVideoPasswordResponse对象，B端依据此对象处理业务逻辑；
- 2、请求参数校验不合格，抛出PloyvSdkException，错误信息见PloyvSdkException.getMessage()，如 [ 输入参数 [xxx.chat.VodxxxRequest]对象校验失败，失败字段 [pic不能为空 / msg不能为空] ]
- 3、服务器处理异常，抛出PloyvSdkException，错误信息见PloyvSdkException.getMessage()，如 [ 保利威请求返回数据错误，请求流水号：66e7ad29fd04425a84c2b2b562d2025b，错误原因：invalid signature. ]

## 请求入参描述

| 参数名     | 必选   | 类型     | 说明                   |
|---------|------|--------|----------------------|
| videoid | true | String | 视频ID【对应api文档的vids字段】 |



返回对象描述

| 参数名            | 类型      | 说明          |
|----------------|---------|-------------|
| isShowPassword | Boolean | 是否显示密码，默认为否 |
| password       | String  | 视频密码        |
| videoid        | String  | 视频id        |
| title          | String  | 视频标题        |

8、批量查询视频播放次数

描述

通过视频id批量查询视频播放次数  
接口地址（仅做说明使用）：<https://api.polyv.net/v2/data/%s/play-times>

调用约束

- 1、接口调用有频率限制，[详细请查看](#)，调用常见异常，[详细请查看](#)

## 单元测试

```
@Test
public void testGetVideosPlayTimes() throws IOException, NoSuchAlgorithmException {
    VodGetVideosPlayTimesRequest vodGetVideosPlayTimesRequest = new VodGetVideosPlayTimesRequest();
    List<VodGetVideosPlayTimesResponse> vodGetVideosPlayTimesResponseList = null;
    try {
        vodGetVideosPlayTimesRequest.setVideoIds(
            "1b448be3234406608b7838c7ef6b597c_1,1b448be323a146649ad0cc89d0faed9c_1").setRealTime(0);
        vodGetVideosPlayTimesResponseList = new VodInfoServiceImpl().getVideosPlayTimes(
            vodGetVideosPlayTimesRequest);
        Assert.assertNotNull(vodGetVideosPlayTimesResponseList);
        if (vodGetVideosPlayTimesResponseList != null) {
            log.debug("测试批量查询视频播放次数成功,{})", JSON.toJSONString(vodGetVideosPlayTimesResponseList));
        }
    } catch (PloyvSdkException e) {
        //参数校验不合格 或者 请求服务器端500错误, 错误信息见PloyvSdkException.getMessage()
        log.error(e.getMessage(), e);
        // 异常返回做B端异常的业务逻辑, 记录log 或者 上报到ETL 或者回滚事务
        throw e;
    } catch (Exception e) {
        log.error("SDK调用异常", e);
        throw e;
    }
}
```

## 单元测试说明

- 1、请求正确，返回VodGetVideosPlayTimesResponse对象，B端依据此对象处理业务逻辑；
- 2、请求参数校验不合格，抛出PloyvSdkException，错误信息见PloyvSdkException.getMessage()，如 [ 输入参数 [xxx.chat.VodxxxRequest]对象校验失败，失败字段 [pic不能为空 / msg不能为空] ]
- 3、服务器处理异常，抛出PloyvSdkException，错误信息见PloyvSdkException.getMessage()，如 [ 保利威请求返回数据错误，请求流水号：66e7ad29fd04425a84c2b2b562d2025b，错误原因： invalid signature. ]

## 请求入参描述

| 参数名      | 必选   | 类型     | 说明                                                                      |
|----------|------|--------|-------------------------------------------------------------------------|
| videoids | true | String | 多个视频ID(英文逗号分割状态为半角)，建议不超过200个，例如1b8be3,239c2e 【对应api文档的 <b>vids</b> 字段】 |

| 参数名      | 必选    | 类型      | 说明                            |
|----------|-------|---------|-------------------------------|
| realTime | false | Integer | 是否实时,1表示实时, 0表示非实时, 默认为0: 非实时 |

### 返回对象描述

返回对象是List<VodGetVideosPlayTimesResponse>, **VodGetVideosPlayTimesResponse**具体元素内容如下:

| 参数名     | 类型      | 说明                           |
|---------|---------|------------------------------|
| videoid | String  | 视频ID【对应api文档的 <b>vid</b> 字段】 |
| times   | Integer | 播放次数                         |

## 9、查询视频信息

### 描述

|                                                                                                                                                |
|------------------------------------------------------------------------------------------------------------------------------------------------|
| 通过视频id查询视频信息<br>接口地址 (仅做说明使用): <a href="https://api.polyv.net/v2/video/%s/get-video-info">https://api.polyv.net/v2/video/%s/get-video-info</a> |
|------------------------------------------------------------------------------------------------------------------------------------------------|

### 调用约束

1、接口调用有频率限制, [详细请查看](#), 调用常见异常, [详细请查看](#)

## 单元测试

```
@Test
public void testGetVideoInfo() throws IOException, NoSuchAlgorithmException {
    VodAccountQueryVideoInfoRequest vodAccountQueryVideoInfoRequest = new VodAccountQueryVideoInfoRequest();
    List<VodAccountQueryVideoInfoResponse> vodAccountQueryVideoInfoResponse;
    try {
        vodAccountQueryVideoInfoRequest.setVideoIds(super.getTestVideoId());
        vodAccountQueryVideoInfoResponse = new
VodInfoServiceImpl().getVideoInfo(vodAccountQueryVideoInfoRequest);
        Assert.assertNotNull(vodAccountQueryVideoInfoResponse);
        if (vodAccountQueryVideoInfoResponse != null) {
            //to do something .....
            log.debug("测试查询视频信息成功 {}", JSON.toJSONString(vodAccountQueryVideoInfoResponse));
        }
    } catch (PloyvSdkException e) {
        //参数校验不合格 或者 请求服务器端500错误, 错误信息见PloyvSdkException.getMessage()
        log.error(e.getMessage(), e);
        // 异常返回做B端异常的业务逻辑, 记录log 或者 上报到ETL 或者回滚事务
        throw e;
    } catch (Exception e) {
        log.error("SDK调用异常", e);
        throw e;
    }
}
```

## 单元测试说明

- 1、请求正确，返回VodAccountQueryVideoInfoResponse对象，B端依据此对象处理业务逻辑；
- 2、请求参数校验不合格，抛出PloyvSdkException，错误信息见PloyvSdkException.getMessage()，如 [ 输入参数 [xxx.chat.VodxxxRequest]对象校验失败，失败字段 [pic不能为空 / msg不能为空] ]
- 3、服务器处理异常，抛出PloyvSdkException，错误信息见PloyvSdkException.getMessage()，如 [ 保利威请求返回数据错误，请求流水号：66e7ad29fd04425a84c2b2b562d2025b，错误原因： invalid signature. ]

## 请求入参描述

| 参数名      | 必选   | 类型     | 说明                                           |
|----------|------|--------|----------------------------------------------|
| videoids | true | String | 视频id（可以支持多个，用英文逗号隔开）【对应api文档的 <b>vid</b> 字段】 |

## 返回对象描述

返回对象是List<VodAccountQueryVideoInfoResponse>，**VodAccountQueryVideoInfoResponse**具体元素内容如下：

| 参数名            | 类型           | 说明                                                |
|----------------|--------------|---------------------------------------------------|
| videoid        | String       | 视频id【对应api文档的 <b>vid</b> 字段】                      |
| basicInfo      | BasicInfo    | 查询的视频基本信息【详见 <a href="#">BasicInfo参数描述</a> 】      |
| transcodeInfos | Array        | 查询的视频转码信息【详见 <a href="#">TranscodeInfos参数描述</a> 】 |
| metaData       | MetaData     | 查询的视频元数据【详见 <a href="#">MetaData参数描述</a> 】        |
| snapshotInfo   | SnapshotInfo | 查询的视频截图【详见 <a href="#">SnapshotInfo参数描述</a> 】     |

### BasicInfo参数描述

| 参数名          | 类型      | 说明                                                       |
|--------------|---------|----------------------------------------------------------|
| title        | String  | 视频标题                                                     |
| description  | String  | 视频描述                                                     |
| duration     | Integer | 源视频时长，单位：秒                                               |
| coverURL     | String  | 首图地址，大图                                                  |
| creationTime | Date    | 创建时间，格式：yyyy-MM-dd HH:mm:ss                              |
| updateTime   | Date    | 更新时间，格式：yyyy-MM-dd HH:mm:ss                              |
| size         | Long    | 源文件大小，单位：Bytes                                           |
| status       | Integer | 视频状态码;60/61:已发布;10:等待编码;20:正在编码;50:等待审核;51:审核不通过;-1:已删除; |

| 参数名          | 类型     | 说明                                      |
|--------------|--------|-----------------------------------------|
| categoryId   | String | 分类id, 如1为根目录 【对应api文档的 <b>catId</b> 字段】 |
| categoryName | String | 分类名称 【对应api文档的 <b>cateName</b> 字段】      |
| tags         | String | 标签                                      |
| uploader     | String | 上传者                                     |

TranscodeInfos参数描述

| 参数名        | 类型      | 说明                                     |
|------------|---------|----------------------------------------|
| playUrl    | String  | 播放地址                                   |
| definition | String  | 清晰度，SOURCE:原清晰度,LD:普清,SD:标清,HD:高清      |
| duration   | Integer | 时长，秒                                   |
| encrypt    | Boolean | 加密视频为true，非加密为false                    |
| format     | String  | 转码格式，如mp4、flv、pdx、hls                  |
| fps        | Integer | 视频帧率                                   |
| bitrate    | Integer | 码率kbps                                 |
| height     | Integer | 分辨率高，单位：px                             |
| width      | Integer | 分辨率宽，单位：px                             |
| status     | String  | 视频状态, normal:可以正常播放,unavailable:不能正常播放 |
| fileSize   | Long    | 编码后视频大小，单位：字节                          |

MetaData参数描述

| 参数名      | 类型      | 说明               |
|----------|---------|------------------|
| size     | Long    | 源文件大小，单位：Bytes   |
| format   | String  | 视频容器类型，如mp4、flv等 |
| duration | Integer | 源视频时长，单位：秒       |
| bitrate  | Integer | 视频码率，单位：bps      |
| fps      | Integer | 视频帧率             |
| height   | Integer | 分辨率高，单位：px       |
| width    | Integer | 分辨率宽，单位：px       |
| codec    | String  | 编码格式，如h264、h265等 |

SnapshotInfo参数描述

| 参数名      | 类型    | 说明      |
|----------|-------|---------|
| imageUrl | Array | 截图url数组 |