

讲师文档关系管理

接口描述

- 1、开启了“讲义库”功能后（如需开通请联系售后），通过此接口将讲师和公共讲义库中的文档关联起来，从而实现在不需要重复上传的情况下，多个讲师共用同一份文档

2、包括新增、解除讲师文档关系

3、接口支持https协议

接口URL

https://api.polyv.net/live/v4/channel/doc/teacher/update-relation

在线API调用

请求方式

POST

接口约束

- 1、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	账号appId【详见 获取密钥 】
timestamp	true	Long	当前13位毫秒级时间戳，3分钟内有效
sign	true	String	签名，为32位大写的MD5值【详见 签名生成规则 】

参数名	必选	类型	说明
operation	true	Integer	操作， 1：新增绑定关系，若绑定关系已经存在，则不再新增，只更新关联时间，查询列表时默认会根据此关联时间倒序排序 2：解除绑定关系
teacherId	true	String	讲师ID，32个以内ASCII码可见字符，系【 创建频道接口 】或【 更新频道接口 】所传入的customTeacherId
fileIds	true	String	文档ID，多个文档ID以,分隔，一次最多传入100个

示例

```
https://api.polyv.net/live/v4/channel/doc/teacher/update-relation&appId=frlr1zazn3&operation=1&sign=FC2296E942D2DB118C64C11AD941445D&teacherId=p001&timestamp=1661754430000
```

表单参数，不参与签名，Content-Type：application/x-www-form-urlencoded

```
fileIds=xxxxxxxxe674c952815ad1a09c66e10xxxxxxxxcommon
```

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
message	String	响应描述信息，当code为400或者500的时候，辅助描述错误原因

参数名	类型	说明
data	Object	暂无

Data参数描述

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 **LiveSignUtil.java** 都包含在下载文件中。

强烈建议您使用[直播Java SDK](#)完成API的功能对接，直播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。

```

/**
 * 更新讲师文档关系
 */
@Test
public static void main(String[] args) throws IOException, NoSuchAlgorithmException {
    // 公共参数，填写自己的实际参数，需放置于QueryString
    String appId = super.appId;
    String appSecret = super.appSecret;
    String userId = super.userId;
    String timestamp = String.valueOf(System.currentTimeMillis());
    // 业务参数
    Integer operation = 1;
    String teacherId = "tid";
    String fileIds =
"xxxxxxxxxxxxxjg112hhs71zza888axxxxxxxxxcommon,xxxxxxxxxxx123451a30dd21919999xxxxxxxxcommon";

    String url = "https://api.polyv.net/live/v4/channel/doc/teacher/update-relation";
    // QueryString中参数需要参与生成签名
    Map<String,String> queryStringMap = new HashMap<>();
    // 公共参数
    queryStringMap.put("appId", appId);
    queryStringMap.put("timestamp", timestamp);
    // 业务参数
    queryStringMap.put("operation", String.valueOf(operation));
    queryStringMap.put("teacherId", teacherId);
    // 生成签名
    String sign = LiveSignUtil.getSign(requestMap, appSecret);
    queryStringMap.put("sign", sign);

    // 将参数放置于QueryString
    url = HttpUtil.appendUrl(url, requestMap);
    System.out.println(url);

    // 请求体参数，不参与签名
    Map<String,String> formBodyMap = new HashMap<>();
    formBodyMap.put("fileIds", fileIds);
    // 发起 post application/x-www-form-urlencoded 请求获取到响应数据 response
    String response = HttpUtil.postFormBody(url, formBodyMap);
    System.out.println(response);
}

```

响应示例

系统全局错误说明详见[全局错误说明](#)

成功示例

```
{
  "code":200,
  "status":"success",
  "message": "",
  "data":null
}
```

异常示例

```
{
  "code": 400,
  "status": "error",
  "message": "invalid signature.",
  "data": ""
}
```