

修改研讨会会议类型

接口描述

- 1、修改研讨会会议类型

2、(channelId, timestamp, appId) 参与sign签名, 并和sign一起通过url传递, 请求体参数不参与签名, 通过post请求体传递【请设置请求头contentType:application/json】

3、接口支持https协议

接口URL

https://api.polyv.net/meet/v1/channel/live-type/update

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 】
channelId	true	String	频道号

请求体参数描述

参数名	必选	类型	说明
seminarLiveType	true	String	会议类型 private 对内研讨 public 对外直播

示例

```
https://api.polyv.net/meet/v1/channel/live-type/update?
appId=fr1r1zazn3&sign=B6424CE5E5D94BB795BC7B3AA6058949&channelId=3589029&timestamp=1677033522629
```

请求体参数：

```
{
  "seminarLiveType":"public"
}
```

响应参数描述

参数名	类型	说明
code	Integer	状态码，与 http 状态码相同，用于确定基本的响应状态
status	String	响应结果，由业务决定，成功返回 success，失败返回error
success	Boolean	是否成功响应
requestId	String	请求ID，每次请求生成的唯一的UUID，仅可用于排查、调试，不应该和业务挂上钩
error	Object	状态码非200时的错误信息【详见 Error字段描述 】
data	Object	失败原因，失败时返回。成功时为空【详见 data字段说明 】

Error参数描述

参数名	类型	说明
code	Integer	错误代码，用于确定具体的错误原因
desc	String	错误描述，与 error.code 对应

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。

```
private static final Logger log = LoggerFactory.getLogger(LiveTypeUpdate.class);
/**
 * 修改研讨会会议类型
 * @throws IOException
 */
@Test
public void testUpdateLiveType() throws IOException, NoSuchAlgorithmException {
    //公共参数,填写自己的实际参数
    String appId=super.appId;
    String appSecret=super.appSecret;
    String userId = super.userId;
    String timestamp=String.valueOf(System.currentTimeMillis());
    //业务参数
    String url = "https://api.polyv.net/meet/v1/channel/live-type/update";
    String channelId = "3589029";
    String body = "{\n" +
        "    \"seminarLiveType\": \"public\"\n" +
        "}";

    //http 调用逻辑
    Map<String,String> requestMap = new HashMap<>();
    requestMap.put("appId", appId);
    requestMap.put("timestamp",timestamp);
    requestMap.put("channelId",channelId);

    requestMap.put("sign",LiveSignUtil.getSign(requestMap, appSecret));
    url = HttpUtil.appendUrl(url, requestMap);
    String response = HttpUtil.postJsonBody(url, body, null);
    log.info("测试修修改研讨会会议类型接口返回值: {}",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

```
{
  "code": 200,
  "status": "success",
  "requestId": "82c0b2fba8a940419c75540f3fb6a635.64.16652796585214411",
  "data": null,
  "success": true
}
```

异常示例

```
{
  "code": 400,
  "status": "error",
  "requestId": "82c0b2fba8a940419c75540f3fb6a635.60.16652797835524421",
  "error": {
    "code": 10001,
    "desc": "研讨会直播类型不正确"
  },
  "data": null,
  "success": false
}
```

```
{
  "code": 400,
  "status": "error",
  "requestId": "6fa092a476ba43f3962c62de38557321.74.16652797988577027",
  "error": {
    "code": 30005,
    "desc": "非法频道ID: 2665820"
  },
  "success": false
}
```

```
{
  "code": 400,
  "status": "error",
  "requestId": "c4920bc4c27e40e18a8a9241d462450a.64.16653728084970005",
  "error": {
    "code": 125001,
    "desc": "研讨中不允许修改配置"
  },
  "data": null,
  "success": false
}
```