

查询时间内直播场次数据

接口描述

- 1、查询时间内直播场次数据
- 2、(timestamp, appId) 参与sign签名, 并和sign一起通过url传递
- 3、接口支持https协议

接口URL

http://api.polyv.net/live/v4/statistics/session-stats/list

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](#)
- 2、不填写查询时间段时, 默认查询当前时间到前一个小时之间产生的场次报表数据; 查询时间段最大间隔24小时;
- 3、查询的数据范围: startTime=<直播场次开始时间 直播场次结束时间<=endTime
- 4、直播中的场次信息不会在数据中返回

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 】

参数名	必选	类型	说明
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，生成签名的appSecret密钥作为通信数据安全的关键信息，严禁保存在客户端直接使用，所有API都必须通过客户自己服务器中转调用POLYV服务器获取响应数据【详见 签名生成规则 】
startTime	false	Long	开始时间，13位时间戳
endTime	false	Long	结束时间，13位时间戳
pageNumber	false	Integer	当前页数
pageSize	false	Integer	每页数据大小，默认10条数据，最大值不能超过1000

示例

```
http://api.polyv.net/live/v4/statistics/session-stats/list?
pageNumber=1&appId=frlr1zazn3&sign=EE807FB4A859A23E5B2EFA6CFC1FE7FC&pageSize=10&startTime=1634517072111&endTime=
1634599872222&timestamp=1634788009047
```

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

参数名	类型	说明
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	【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

data参数描述

参数名	类型	说明
pageNumber	Integer	当前页码
pageSize	Integer	每页数据量
totalPages	Long	页面总数
totalItems	Long	总数据量
contents	Array	当前页内容【详见 contents字段说明 】

contents参数描述

参数名	类型	说明
channelId	Integer	频道ID
sessionId	String	直播场次ID

参数名	类型	说明
name	String	场次名称
startTime	Long	直播开始时间，13位时间戳
endTime	Long	直播结束时间，13位时间戳
userId	String	直播账号ID

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(getClass());
/**
 * 获取时间内直播场次数据
 * @throws IOException
 * @throws NoSuchAlgorithmException
 */
@Test
public void testGetLiveSession() 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 timestamp = String.valueOf(System.currentTimeMillis());

    //业务参数
    String url = "http://api.polyv.net/live/v4/statistics/session-stats/list";
    String startTime = "1634517072111";
    String endTime = "1634599872222";
    String pageNumber = "1";
    String pageSize = "10";

    //http 调用逻辑
    Map<String, String> requestMap = new HashMap<>();
    requestMap.put("appId", appId);
    requestMap.put("timestamp", timestamp);
    requestMap.put("startTime", startTime);
    requestMap.put("endTime", endTime);
    requestMap.put("pageNumber", pageNumber);
    requestMap.put("pageSize", pageSize);
    requestMap.put("sign", LiveSignUtil.getSign(requestMap, appSecret));

    url = HttpUtil.appendUrl(url, requestMap);
    String response = HttpUtil.postFormBody(url, null);
    log.info("测试获取时间内直播场次数据成功: {}", 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": "b38bdd97dc404213a2436e7a086b9db9.80.16347880125725671",
  "data": {
    "pageNumber": 1,
    "pageSize": 10,
    "totalPages": 1,
    "totalItems": 2,
    "contents": [
      {
        "channelId": 2272655,
        "sessionId": "g3e0fasiw0",
        "name": "JAVA大师无延迟直播课-纯视频",
        "startTime": 1634520672000,
        "endTime": 1634520742000,
        "userId": "1b448be323"
      },
      {
        "channelId": 2477096,
        "sessionId": "g3eb7xbpn4",
        "name": "JAVA 知识精讲(误删)",
        "startTime": 1634544160000,
        "endTime": 1634544232000,
        "userId": "1b448be323"
      }
    ]
  },
  "success": true
}
```

异常示例

```
{
  "code": 400,
  "status": "error",
  "requestId": "d310b70bc329403f87f77f9203d50f89.128.16360831552223589",
  "error": {
    "code": 20001,
    "desc": "application not found."
  },
  "success": false
}
```