

根据签到ID查询所有签到记录

接口描述

- 1、通过签到id查询签到记录（包括已签到与未签到记录）

2、接口支持https协议

接口URL

http://api.polyv.net/live/v3/channel/chat/get-checkins

在线API调用

请求方式

GET

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	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

参数名	必选	类型	说明
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 】
channelId	true	String	频道号
checkinId	true	String	签到id，一场签到一个id，该参数获取自【 查询频道签到结果 】

示例

```
http://api.polyv.net/live/v3/channel/chat/get-checkins?checkinId=db14ef80-81b8-11eb-b114-e7477b&appId=fr1r1zazn3&sign=F0163ADFB54D9BB31BC8DD411252B1FD&channelId=2191569&timestamp=1621845214778
```

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
data	Array	签到记录的数据【详见 data字段说明 】

data字段说明

参数名	类型	说明
accountId	String	对应直播账号的userId
checkinid	String	签到id
nickname	String	签到学员昵称
checked	String	签到Y，未签到N
indate	String	签到日期，yyyy-MM-dd格式
id	Integer	签到记录主键
sessionId	String	场次号
time	Long	签到时间戳
userid	String	签到学员id
roomId	String	房间号
param4	String	在外部授权、直接（独立）授权情况下传过来的自定义参数
param5	String	在外部授权、直接（独立）授权情况下传过来的自定义参数

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(InteractionCheckinTest.class);
/**
 * 根据发起签到记录查询签到结果
 * @throws IOException
 */
@Test
public void testGetCheckinByCheckinId() 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 = "https://api.polyv.net/live/v3/channel/chat/get-checkins";
    String channelId = "2191569";
    String checkinId = "db14ef80-81b8-11eb-b114-e7477b";

    //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("checkinId",checkinId);
    requestMap.put("sign",LiveSignUtil.getSign(requestMap, appSecret));
    String response = HttpUtil.get(url, requestMap);
    log.info("测试根据发起签到记录查询签到结果接口返回值: {}",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": [
    {
      "accountId": "1b448be323",
      "param5": "",
      "checkinid": "db14ef80-81b8-11eb-b114-e7477b",
      "nickname": "卉",
      "checked": "Y",
      "indate": "2021-03-10",
      "id": 48010860,
      "sessionId": "",
      "time": 1615391652916,
      "param4": "oi_ubswUnFt8QQFAAEJbKmd8W9BE",
      "userid": "ovt19t_RxnrTdqkXqkT5Q51nxp2A",
      "roomid": "2191569"
    }
  ]
}
```

异常示例

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