

获取用户实时并发接口

接口URL

https://api.polyv.net/live/v3/user/statistics/realtime-concurrence

接口说明

- 1、查询用户的实时并发数
- 2、接口支持https

支持格式

JSON

请求方式

GET

请求数限制

TRUE

请求参数

参数名	必选	类型	说明
appId	是	string	从API设置中获取，在直播系统登记的appId

参数名	必选	类型	说明
timestamp	是	long	当前13位毫秒级时间戳，3分钟内有效
sign	是	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 】

响应成功JSON示例：

```
{
  "code": 200,
  "status": "success",
  "message": "",
  "data": 100
}
```

响应失败JSON示例：

未输入appId

```
{
  "code": 400,
  "status": "error",
  "message": "appId is required.",
  "data": ""
}
```

appId不正确

```
{
  "code": 400,
  "status": "error",
  "message": "application not found.",
  "data": ""
}
```

时间戳错误

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

签名错误

```
{
  "code": 403,
  "status": "error",
  "message": "invalid signature.",
  "data": ""
}
```

字段说明

参数名	类型	说明
code	int	响应代码，成功为200，失败为400，签名错误为401，异常错误500
status	string	成功为success，失败为error
message	string	错误时为错误提示消息
data	object	响应结果集，用户的实时并发数，整型

Java请求示例

```
public static void testGetUserRealTimeConcurrency() {
    String url = "https://api.polyv.net/live/v3/user/statistics/realtime-concurrency";
    // 用户对应的appId和加密串
    String appId = "xxxxxxxx";
    String appSecret = "xxxxxxxxxxxxxxxxxxxxxxxx";
    Map<String, String> params = new HashMap<>();
    PolyvTool.setLiveSign(params, appId, appSecret);
    // 调用Polyv的httpClient工具类发送请求
    String content = HttpClientUtil.getInstance()
        .sendHttpPost(url, params);
    System.out.println(content);
}
```