

修改任务奖励活动

接口描述

- 1、修改任务奖励活动

2、接口支持https协议

接口URL

https://api.polyv.net/live/v4/channel/task-reward-activity/update

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](#)

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 】

请求体参数描述

参数名	必选	类型	说明
channelId	true	Integer	频道id
activityId	true	Long	活动id
activityName	false	String	活动名称，最大长度64个字符
taskRule	false	Integer	任务规则：1：解锁任务，2：并行任务
startTime	false	Long	活动开始时间
endTime	false	Long	活动结束时间
tasks	true	Array	任务规则列表【详见 tasks字段说明 】

tasks参数描述

参数名	必选	说明	类型
sort	true	排序	Integer
reachCondition	true	达成条件设置【详见 reachCondition 】	Object

参数名	必选	说明	类型
rewardSetting	true	奖励设置 【详见 rewardSetting 】	Object

reachCondition参数描述

参数名	必选	类型	说明
type	true	String	达成条件（sign：签到，online：在线时长，questionnaire：答题，custom：自定义）
amount	true	Integer	达成数值
templateId	false	Long	题目ID，达成条件为questionnaire时必传
customEvent	false	String	自定义任务事件，达成条件为custom时必传
customEventName	false	String	自定义任务名
customIcon	false	String	自定义任务图标
customUnit	false	String	自定义任务单位
customButtonText	false	String	自定义任务按钮文案
customReachText	false	String	自定义任务达成文案，eg： {customEventName}完成：{amount} {customUnit}

rewardSetting参数描述

参数名	必选	类型	说明
type	true	String	cash:现金,custom:自定义,nothing:无奖励,externalPoint:外部积分

参数名	必选	类型	说明
amount	false	Integer	奖励数值（type为cash时，现金单位是分；type为cash、externalPoint时需要传）
limit	false	Integer	奖励限制份额（-1不限制）
customReward	false	String	自定义奖励内容，非必填

示例

```
https://api.polyv.net/live/v4/channel/task-reward-activity/update?
appId=frrlr1zazn3&sign=012332FBD8DBCFC068AFCB484A2ADEC8&timestamp=1670550787318
```

请求体json参数：

```
{
  "activityId": 5,
  "channelId": 4854808,
  "activityName": "修改测试活动",
  "activityMode": 1,
  "taskRule": 1,
  "startTime": 1717569984074,
  "endTime": 1717569994074,
  "tasks": [
    {
      "reachCondition": {
        "type": "sign",
        "amount": 11
      },
      "rewardSetting": {
        "type": "cash",
        "amount": 10,
        "limit": 10
      }
    },
    {
      "reachCondition": {
        "type": "sign",
        "amount": 21
      },
      "rewardSetting": {
        "type": "cash",
        "amount": 10,
        "limit": 10
      }
    }
  ]
}
```

响应体json:

```
{
  "code": 200,
  "status": "success",
  "requestId": "e9eb12f0-5d55-482a-8a47-62863d04772c",
  "data": true,
  "success": true
}
```

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
status	String	响应结果，由业务决定，成功返回success，失败返回error
success	Boolean	响应结果，由业务决定，成功返回true，失败返回false
data	Boolean	任务奖励活动是否修改成功
error	Object	状态码非200时的错误信息【详见 Error字段说明 】
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 final Logger log = LoggerFactory.getLogger(getClass());
/**
 * 修改任务奖励活动
 * @throws IOException
 * @throws NoSuchAlgorithmException
 */
@Test
public void createTaskRewardTest() 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());

    String url = "https://api.polyv.net/live/v4/channel/task-reward-activity/update";
    //业务参数
    Integer channelId = 5004544;
    String activityId = 1134;
    String activityName = "demoData";
    Integer taskRule = 1;
    Long startTime = 1722035312584;
    Long endTime = 1722036312584;
    List<Task> tasks = new ArrayList();

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

    Map<String, Object> jsonMap = new HashMap<>();
    jsonMap.put("channelId", channelId);
    jsonMap.put("activityId", activityId);
    jsonMap.put("activityName", activityName);
    jsonMap.put("taskRule", taskRule);
    jsonMap.put("startTime", startTime);
    jsonMap.put("endTime", endTime);
    jsonMap.put("tasks", tasks);

    requestMap.put("sign", LiveSignUtil.getSign(requestMap, appSecret));

    url = HttpUtil.appendUrl(url, requestMap);
    String response = HttpUtil.postJsonBody(url, JSON.toJSONString(jsonMap), null);

    log.info("测试修改活动结果: {}", 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": "e9eb12f0-5d55-482a-8a47-62863d04772c",
  "data": true,
  "success": true
}
```

异常示例

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