

创建优惠券

接口描述

- 1、创建优惠券

2、接口支持https协议

接口URL

https://api.polyv.net/live/v4/user/coupon/create

[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](#)

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
name	true	String	优惠券名称，最多50个字符
receiveStartTime	true	Long	领券开始时间，13位毫秒时间戳
receiveEndTime	true	Long	领券结束时间，13位毫秒时间戳
useTimeType	true	String	用券时间类型，RANGE-时间范围，DAY-天数
useStartTime	false	Long	useTimeType为RANGE的参数：用券开始时间，13位毫秒时间戳
useEndTime	false	Long	useTimeType为RANGE的参数：用券结束时间，13位毫秒时间戳
dayOfUse	false	Integer	useTimeType为DAY的参数：领取后多少天内可用
type	true	String	优惠券类型: MAX_OUT-满减券、DISCOUNT-折扣券
availableAmount	true	Integer	发放数量，大于等于0

参数名	必选	类型	说明
rule	true	UserCouponRule	优惠券规则【详情见 UserCouponRule 】

UserCouponRule

参数名	必选	类型	说明
condition	true	String	优惠券使用条件 (UNCONDITIONAL-无门槛、FULL_REDUCE-满减)
unconditional	false	Unconditional	无门槛：unconditional和fullReduce，二选一必填，【详情见 Unconditional 】
fullReduce	false	FullReduce	满减：unconditional和fullReduce，二选一必填，【详情见 FullReduce 】
limitPerPerson	true	Integer	每人限领数量: -1-不限制

Unconditional

参数名	必选	类型	说明
enable	true	boolean	是否可用: true-开, false-关
value	true	double	数值
unit	true	String	单位: MONEY-元、DISCOUNT-折

FullReduce

参数名	必选	类型	说明
enable	true	boolean	是否可用: true-开, false-关

参数名	必选	类型	说明
full	true	double	满多少
reduce	true	double	减多少
unit	true	String	单位: MONEY-元、 DISCOUNT-折

示例

```
https://api.polyv.net/live/v4/user/coupon/create?
appId=frlr1zazn3&sign=012332FBD8DBCFC068AFCB484A2ADEC&timestamp=1670550787318
```

请求体json参数：

```
{
  "availableAmount": 999,
  "dayOfUse": 30,
  "description": "买一送一",
  "name": "优惠券2号",
  "receiveEndTime": 1756102918736,
  "receiveStartTime": 1748326918736,
  "rule": {
    "condition": "UNCONDITIONAL",
    "limitPerPerson": -1,
    "unconditional": {
      "enable": true,
      "unit": "MONEY",
      "value": 1.0
    }
  },
  "type": "MAX_OUT",
  "useEndTime": 1756102918737,
  "useStartTime": 1748326918737,
  "useTimeType": "RANGE"
}
```

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

参数名	类型	说明
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	Object	标签数据 【[[Data字段说明]](#Data参数描述)】
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

Data参数描述

参数名	类型	说明
Data自身	String	优惠券ID

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());
/**
 * 新增优惠券
 * @throws IOException
 * @throws NoSuchAlgorithmException
 */
@Test
public void createCouponTest() 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/v4/user/coupon/create";

    //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("name", "优惠券2号");
    jsonMap.put("useTimeType", "RANGE");
    jsonMap.put("dayOfUse", 30);
    jsonMap.put("receiveStartTime", 1756102918736L);
    jsonMap.put("receiveStartTime", 1748326918736L);
    jsonMap.put("useStartTime", 1748326918737L);
    jsonMap.put("useEndTime", 1756102918737L);
    Map<String, Object> rule = new HashMap<>();
    rule.put("condition", "UNCONDITIONAL");
    rule.put("limitPerPerson", -1);
    Map<String, Object> unconditional = new HashMap<>();
    unconditional.put("enable", true);
    unconditional.put("unit", "MONEY");
    unconditional.put("value", 1.0);
    rule.put("unconditional", unconditional);
    jsonMap.put("rule", rule);

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

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

    log.info("测试新建优惠券结果: {}", 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": "78de516b69364aeabe7d080e7228cc5c.77.16705507903976231",
  "data": "caqfi6rqutunvofo5jrqbinswh14p2g", // 优惠券ID
  "success": true,
}
```

异常示例

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