

平台内容库分组列表

接口描述

1、平台内容库分组列表
2、接口支持https协议

接口URL

https://api.polyv.net/live/v4/global/robot/label/list

[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 】
type	true	String	标签类型: script-内容库, robot-成员库

请求体参数描述

参数名	必选	类型	说明
-----	----	----	----

示例

```
https://api.polyv.net/live/v4/global/robot/label/list?
appId=frrlr1zazn3&sign=012332FBD8DBCFC068AFCB484A2ADECB&timestamp=1670550787318&type=robot
```

请求体json参数：

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

参数名	类型	说明
data	Array	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
id	Long	分组(labelId或标签)ID
name	String	分组名称
type	String	类型: SYS-系统素材，USER-个人素材
updateTime	Long	最后更新时间

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());
/**
 * 平台内容库分组列表
 * @throws IOException
 * @throws NoSuchAlgorithmException
 */
@Test
public void listTest() 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/global/robot/label/list";
    String type = "robot";

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

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

    url = HttpUtil.appendUrl(url, requestMap);
    String response = HttpUtil.get(url, null);

    log.info("测试平台内容库分组列表结果: {}", 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,
  "data": [
    {
      "id": 900000000,
      "name": "通用组",
      "type": "SYS",
      "updateTime": 4102416000000
    },
    {
      "id": 900000001,
      "name": "中老年组",
      "type": "SYS",
      "updateTime": 4102416000000
    },
    {
      "id": 900000002,
      "name": "中文名组",
      "type": "SYS",
      "updateTime": 4102416000000
    },
    {
      "id": 900000009,
      "name": "英文名组",
      "type": "SYS",
      "updateTime": 4102416000000
    },
    {
      "id": 5300,
      "name": "111",
      "type": "USER",
      "updateTime": 1747015055000
    },
    {
      "id": 4961,
      "name": "默认分类",
      "type": "USER",
      "updateTime": 1742785082000
    }
  ],
  "requestId": "92dca09e-5a7d-4afe-a950-d352ac60f0ed",
  "status": "success",
  "success": true
}
```

异常示例

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