

快速接入

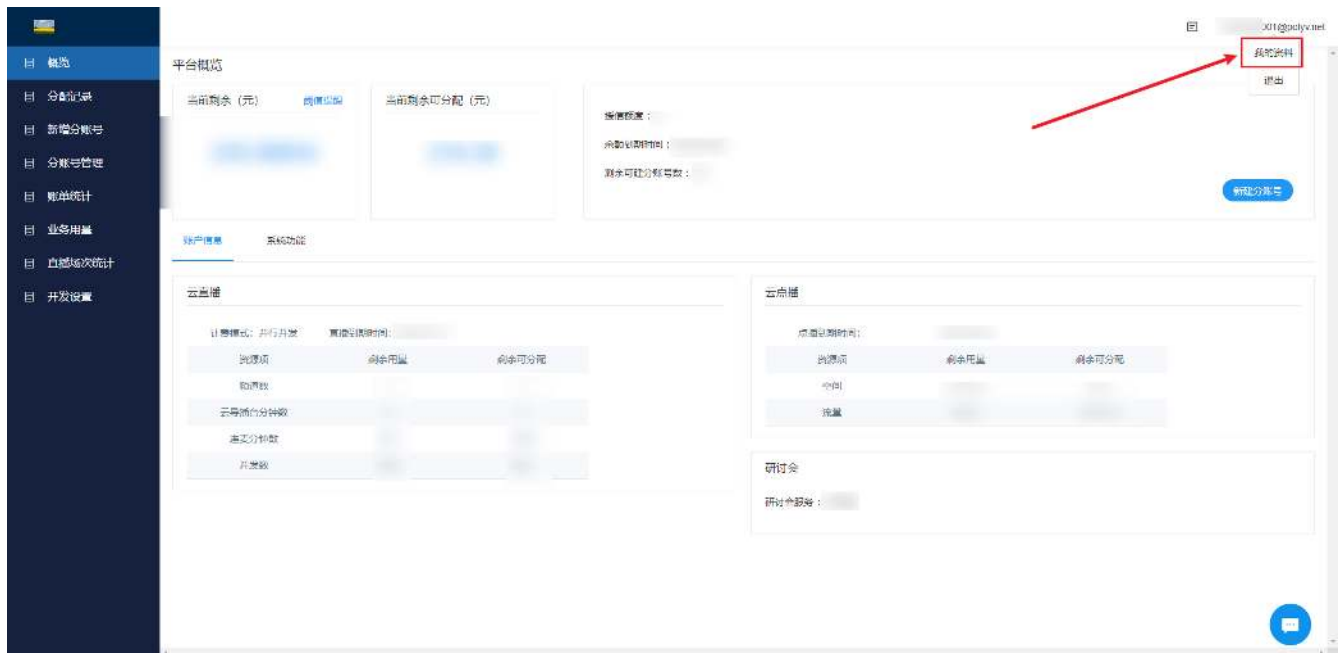
前提条件

在安装和使用保利威集团账号Java SDK功能前，确保B端已经具备如下环境：

- 安装Java环境。

保利威集团账号Java SDK支持1.8或更高版本。

- 拥有保利威集团账号并获取访问密钥信息（Appld 、 AppSecret）。



目 概览

目 分配记录

目 新增分账号

目 分账号管理

目 账单统计

目 业务用量

目 直播场次统计

目 开发设置

账号信息

集团账号邮箱: 001@polyv.net

账号密码: **** [重置密码](#)

联系人:

手机号码:

机构网址:

groupId:

AppID:

AppSecret:

保存



1.添加Maven依赖

将如下依赖加入到项目的POM文件中:

注意: 受 Maven 中央仓库网络限制, 阿里云云效 Maven 中央代理仓库可能会出现依赖查找不到的情况, 建议使用中央仓库: <https://repo1.maven.org/maven2/>

```
<dependency>
  <groupId>net.polyv</groupId>
  <artifactId>polyv-java-group-sdk</artifactId>
  <version>2.2.8</version>
</dependency>
```

注意: 为了消除Java的冗长代码, SDK使用了jdk1.8的最新特性lombok, 请在IntelliJ或者Eclipse 中配置对lombok的支持

```
<dependency>
  <groupId>org.projectlombok</groupId>
  <artifactId>lombok</artifactId>
  <version>1.18.10</version>
</dependency>
```

2.初始化系统

在执行测试代码之前，需要B端先初始化系统配置，包括AppId \ AppSecret，如没有以上信息，请参考本文[前提条件](#)部分获取，初始化示例代码如下，任选一种：

```
/**
 * 初始化配置,请配置自己的账号信息
 */
public static void initPolyvGroup(){
    String appId = "xxx";
    String appSecret = "xxx";
    GroupGlobalConfig.init(appId, appSecret);
    log.debug("--初始化完成--");
}
```

```
/**
 * 初始化配置并初始化 HTTP CLIENT 连接池超时时间和最大连接数配置,请配置自己的账号信息
 */
public static void initPolyvGroup(){
    String appId = "xxx";
    String appSecret = "xxx";
    Integer timeOut = 20000; //HTTP CLIENT 连接池超时时间
    Integer maxClientNum = 100; //HTTP CLIENT 最大连接数
    GroupGlobalConfig.init( appId, appSecret, timeOut , maxClientNum);
    log.debug("--初始化完成--");
}
```

```
/**
 * 初始化配置并初始化 HTTP CLIENT 代理信息，一般使用在正式服务器不能连接外网，必须通过代理连接,请配置自己的账号信息
 */
public static void initPolyvGroup(){
    String appId = "xxx";
    String appSecret = "xxx";
    String hostName = '127.0.0.1'; //代理的ip
    int port = 8888; //代理的端口
    String scheme = 'http'; // 代理的协议
    GroupGlobalConfig.init( appId, appSecret, hostName, port, scheme);
    log.debug("--初始化完成--");
}
```

以上代码一般配置于随系统启动执行一次的全局初始化中，如用spring框架，可以参考如下：

```
package net.polyv.live.config;

import org.springframework.beans.BeansException;
import org.springframework.context.ApplicationContext;
import org.springframework.context.ApplicationContextAware;
import org.springframework.stereotype.Component;

import lombok.extern.slf4j.Slf4j;

/**
 * 默认启动配置类
 * @author: thomas
 */
@Slf4j
@Component
public class StartupListener implements ApplicationContextAware {

    @Override
    public void setApplicationContext(ApplicationConteXt applicationConteXt) throws BeansException {
        String appId = "xxx";
        String appSecret = "xxx";
        GroupGlobalConfig.init(appId,appSecret);
        log.debug("--初始化完成--");
    }

}
```

3.执行测试代码

测试创建一个直播频道，单元测试代码如下：

```

import org.junit.Assert;
import com.alibaba.fastjson.JSON;
import lombok.extern.slf4j.Slf4j;
import net.polyv.common.v1.exception.PloyvSdkException;
import net.polyv.group.v1.config.GroupGlobalConfig;
import net.polyv.group.v1.entity.user.GroupCreateUserRequest;
import net.polyv.group.v1.entity.user.GroupCreateUserResponse;
import net.polyv.group.v1.service.user.impl.UserServiceImpl;

/**
 * @author: sadboy
 */
@Slf4j
public class GroupChannelDemo {

    /**
     * 调用demo，必须处理PloyvSdkException。
     *
     * 参数合法性校验：SDK采用自定义验证框架对输入参数进行校验，如有参数不合格，将抛出PloyvSdkException异常，exception的
     message
     * 包括具体校验不通过的字段信息，此异常是运行时异常，必须捕获处理相关业务逻辑；
     *
     * 解析返回数据：解析返回数据，如SDK调用正常成功，将封装响应对象，正常返回，如服务器返回错误信息，SDK将抛出
     PloyvSdkException异常，exception的message
     * 包括具体服务器执行错误信息，此异常是运行时异常，必须捕获处理相关业务逻辑；
     * @param args
     */
    public static void main(String[] args) {
        //全局初始化，此处应该全局执行一次
        String appId = "xxx";
        String appSecret = "xxx";
        GroupGlobalConfig.init(appId, appSecret);
        log.debug("--初始化完成--");
        try {
            //封装API请求对象
            GroupCreateUserRequest groupCreateUserRequest = new GroupCreateUserRequest();
            groupCreateUserRequest.setEmail("XXXX@polyv.net")
                .setPassword("Random08")
                .setContacts("XX")
                .setPhone("17600002222")
                .setMaxChannels(1)
                .setConcurrent(10)
                .setRemainSpace(0)
                .setRemainFlow(0);
            //调用SDK请求保利威服务器
            GroupCreateUserResponse groupCreateUserResponse = new UserServiceImpl().create(groupCreateUserRequest);
            Assert.assertNotNull(groupCreateUserResponse);
            //正常返回做B端正常的业务逻辑
            if (groupCreateUserResponse != null) {
                //to do something .....
                log.debug("账号创建成功{}", JSON.toJSONString(groupCreateUserResponse));
                log.debug("登录地址: https://my.polyv.net/v3/login/, 登录邮箱: {} , 登录密码:
                {} ", groupCreateUserRequest.getEmail(), groupCreateUserRequest.getPassword());
            }
        } catch (PloyvSdkException e) {

```

```

//参数校验不合格 或者 请求服务器端500错误, 错误信息见PloyvSdkException.getMessage()
log.error(e.getMessage(), e);
// 异常返回做B端异常的业务逻辑, 记录log 或者 上报到ETL 或者回滚事务
} catch (Exception e) {
    log.error("SDK调用异常", e);
}
}
}

```

执行代码后, 控制台应有如下关键输出, 表示整合完成:

```

2023-07-11 14:42:27:438 +0800 [main] DEBUG net.polyv.common.v1.base.HttpUtil - http 请求 url:
https://api.polyv.net/live/v4/group/user/create?
requestId=b5bbd26f9d464c49b73ecb3b9b0212dc&appId=ga1ayl6dh2&sign=501DD89E7E4818ADFC0773CF5B84FA2F&timestamp=1689
057747104 , 请求参数:
{"appId":"XXX","concurrent":0,"contacts":"XX","email":"XXX@polyv.net","maxChannels":0,"password":"Random08","pho
ne":"XXXX","remainFlow":0,"remainSpace":0,"requestId":"XXXX","sign":"XXX","timestamp":"XXX"}
2023-07-11 14:42:28:301 +0800 [main] DEBUG net.polyv.common.v1.base.HttpUtil - HTTP请求耗时分析, 请求URL:
https://api.polyv.net/live/v4/group/user/create?requestId=XXX&appId=XXX&sign=XXX&timestamp=XXX , 请求头信息:
[{"elements":[{"name":"JAVA_GROUP_SDK","parameterCount":0,"parameters":[]}],"name":"User-
Agent","value":"JAVA_GROUP_SDK"}, {"elements":[{"name":"JAVA_GROUP_SDK","parameterCount":0,"parameters":
[]},"name":"source","value":"JAVA_GROUP_SDK"}, {"elements":[{"name":"XXX","parameterCount":0,"parameters":
[]},"name":"java-sdk-app-id","value":"XXX"}, {"elements":[{"name":"2.1.5","parameterCount":0,"parameters":
[]},"name":"version","value":"2.1.5"}] , 耗时: 829 ms
2023-07-11 14:42:28:303 +0800 [main] DEBUG net.polyv.common.v1.base.HttpUtil - http 请求结果:
{"code":200,"status":"success","requestId":"0b2411850bf24bbf9006de5b0fe59a1d.16334.16810577479026979","data":
{"appId":"XXX","appSecret":"XXX","userId":"XXX"},"success":true}
2023-07-11 14:42:28:352 +0800 [main] DEBUG net.polyv.live.v1.service.channel.GroupChannelDemo - 账号创建成功
{"appId":"XXX","appSecret":"XXX","userId":"XXX"}
2023-07-11 14:42:28:353 +0800 [main] DEBUG net.polyv.live.v1.service.channel.GroupChannelDemo - 登录地址:
https://my.polyv.net/v3/login/, 登录邮箱: XXXX@polyv.net , 登录密码: Random08

```

还可以官网登录验证创建是否成功;



至此，B端已经完成**保利威集团账号Java SDK**安装配置，可以使用集团账号SDK进行其他功能开发和测试，如B端接入过程有任何问题，直接使用右下角**在线客服**找到售后技术支持提问，请将问题的运行环境、操作步骤、错误反馈信息、联系方式同步反馈，便于问题的快速定位和解决。