

# 快速接入示例

## 示例代码接入前提条件

强烈建议您使用[点播Java SDK](#)完成API的功能对接，点播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。

在对接保利威点播API功能前，确保B端已经具备如下环境：

- 安装Java环境。

示例代码支持JDK 1.8或更高版本。

- 获取密钥

到保利威官网注册[保利威账号](#)并获取[访问密钥信息](#)（userid、writetoken、readtoken、secretkey）。





```

<properties>
    <maven.compiler.source>8</maven.compiler.source>
    <maven.compiler.target>8</maven.compiler.target>
    <httpClient.version>4.5.13</httpClient.version>
    <slf4j.version>1.7.30</slf4j.version>
    <lombok.version>1.18.16</lombok.version>
    <fastjson.version>1.2.70</fastjson.version>
    <junit.version>4.12</junit.version>
</properties>

<dependencies>
    <!-- 基本工具 -->
    <dependency>
        <groupId>org.apache.httpcomponents</groupId>
        <artifactId>httpClient</artifactId>
        <version>${httpClient.version}</version>
    </dependency>
    <dependency>
        <groupId>org.apache.httpcomponents</groupId>
        <artifactId>httpmime</artifactId>
        <version>${httpClient.version}</version>
    </dependency>
    <dependency>
        <groupId>org.slf4j</groupId>
        <artifactId>slf4j-api</artifactId>
        <version>${slf4j.version}</version>
    </dependency>
    <dependency>
        <groupId>org.slf4j</groupId>
        <artifactId>slf4j-simple</artifactId>
        <version>${slf4j.version}</version>
    </dependency>
    <dependency>
        <groupId>junit</groupId>
        <artifactId>junit</artifactId>
        <version>${junit.version}</version>
        <scope>test</scope>
    </dependency>
</dependencies>

```

## 2.系统基础代码

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 **VodSignUtil.java** 都包含在下载文件中。

强烈建议您使用[点播Java SDK](#)完成API的功能对接，点播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。

### 3.执行测试代码

测试依据userid获取用户空间及流量API，单元测试代码如下：

```
package net.polyv.vod.account;

import java.security.NoSuchAlgorithmException;
import java.util.HashMap;
import java.util.Map;

import org.junit.Test;
import org.slf4j.Logger;
import org.slf4j.LoggerFactory;

import net.polyv.VodBaseTest;
import net.polyv.util.HttpUtil;
import net.polyv.util.LiveSignUtil;
import net.polyv.util.VodSignUtil;

/**
 * @author: thomas
 */

public class VodAccountTest extends VodBaseTest {

    private static final Logger log = LoggerFactory.getLogger(VodAccountTest.class);
    /**
     * 点播DEMO：获取用户空间及流量情况
     */
    @Test
    public void testGetAccountSpaceFlow() throws Exception, NoSuchAlgorithmException {
        //公共参数,填写自己的实际参数
        String secretKey = super.secretKey;
        String userId = super.userId;
        String ptime = String.valueOf(System.currentTimeMillis());
        //业务参数
        String url = "http://api.polyv.net/v2/user/"+userId+"/main";
        String date = "2020-10-13";

        Map<String, String> requestMap = new HashMap<>();
        requestMap.put("userid", userId);
        requestMap.put("ptime", ptime);
        requestMap.put("requestId", LiveSignUtil.generateUUID());
        requestMap.put("date", date);
        requestMap.put("sign", VodSignUtil.getSign(requestMap, secretKey));
        String response = HttpUtil.get(url, requestMap);
        log.debug("测试获取用户空间及流量情况成功,{", response);
        //do somethings
    }

}
```

执行代码后，控制台应有如下关键输出，表示接入成功：

```
测试获取用户空间及流量情况成功,{ "code":200,"status":"success","message":"success","data":  
{"videoCount":47,"totalFlow":21474836480,"usedSpace":0,"usedFlow":0,"totalSpace":21474836480,"userId":"xxxxxx","  
email":"xxxxx"}}
```