

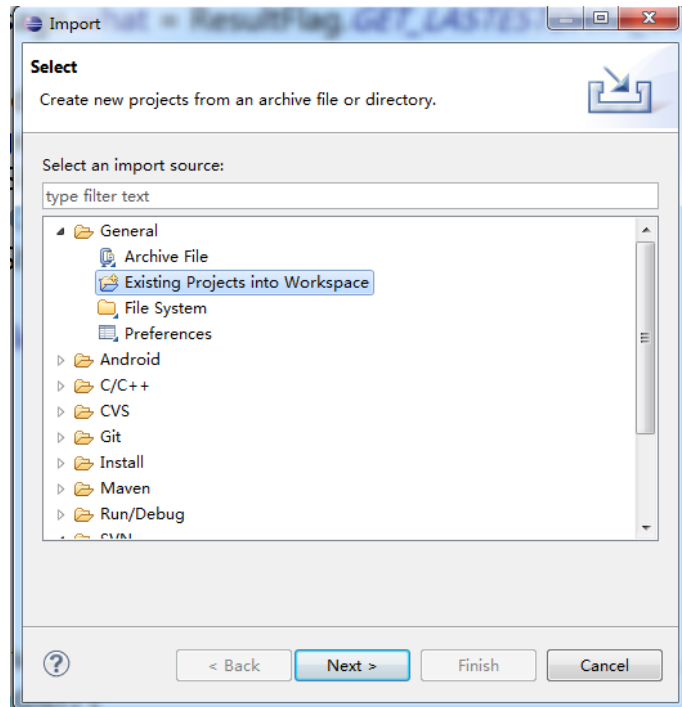
【 PunchBoxUcSDK v0.1 】

【 Android 开发人员指南】

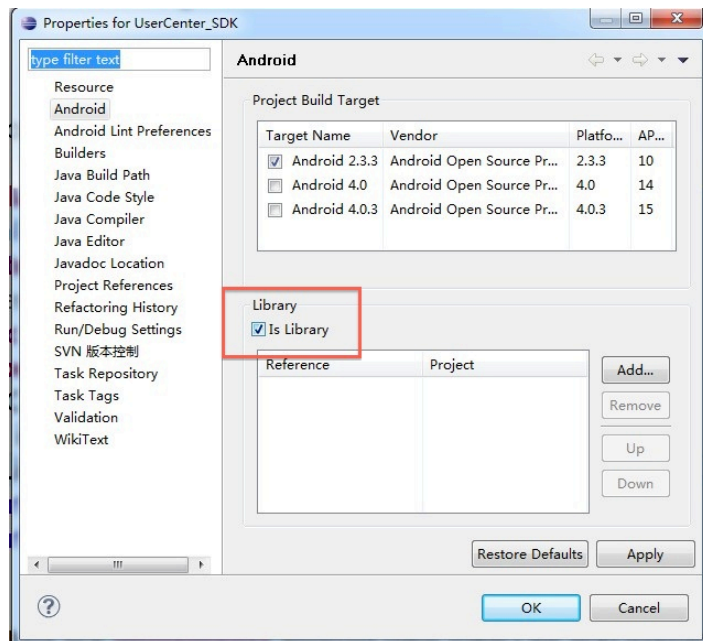
版本 0.3

一、 导入 SDK

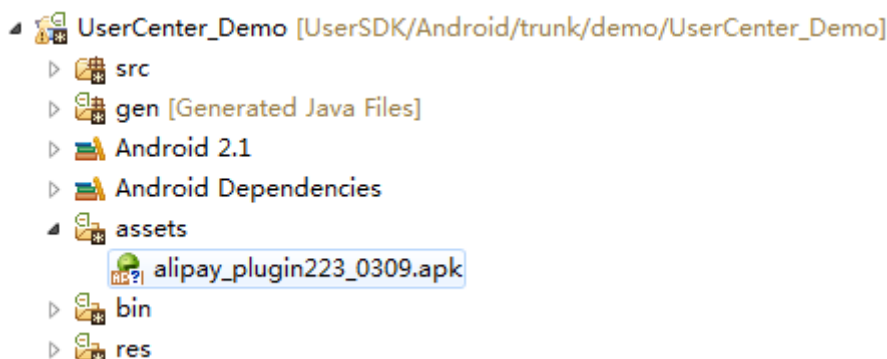
1.打开 Eclipse，选择 **File->Import->General->Existing Project into Workspace**，找到本地存放的 SDK 工程。



2.将 SDK 工程定义为一个 **Library**。



3. 将 SDK 工程 **assets** 目录下的 **alipay_plugin223_0309.apk** 文件拷贝到开发者自己工程的同级目录下。



4. 开发者需要为自己的工程添加以下权限。

```
<uses-permission android:name="android.permission.INTERNET"/>
<uses-permission android:name="android.permission.ACCESS_NETWORK_STATE"/>
<uses-permission android:name="android.permission.READ_PHONE_STATE"/>
<uses-permission android:name="android.permission.ACCESS_WIFI_STATE"/>
<uses-permission android:name="android.permission.ACCESS_COARSE_LOCATION"/>
<uses-permission android:name="android.permission.ACCESS_FINE_LOCATION"/>
```

二、API 接口

1. 初始化

声明 InitHelper 对象，参数为当前上下文，然后使用声明的对象调用 initSDK() 方法。该方法的三个参数分别是 App_key、Secret_key 以及回调 Handler(用来获取初始化方法返回的结果，以下相同)，其中 App_key 和 Secret_key 需要向触控用户系统管理平台申请，**开发者必须确认初始化方法执行成功后才能去调用别的 API 方法。**

```
InitHelper init = new InitHelper(this);
init.initSDK("91496745-DEAF-4617-A888-568EB6291AC", "ou[redacted]2332", initResultHandler);
```

2. 注册

声明 RegisterHelper 对象，参数为当前上下文，然后使用声明的对象调用 register() 方法。该方法的两个参数分别是用户信息 userinfo、回调 Handler，userinfo 必填的参数有 user_name, password, platform, 其中 platform 在 0.1 版本中固定为 0。

```
UserInfo userInfo = new UserInfo();
userInfo.setUser_name(mRegisterName);
userInfo.setPassword(mRegisterPassWord);
userInfo.setPlatform("0");
RegisterHelper registerHelper = new RegisterHelper(RegisterActivity.this);
registerHelper.register(userInfo, registerHandler);
```

3. 登陆

声明 LoginHelper 对象，参数为当前上下文，然后使用声明的对象调用 login()方法。该方法两个参数分别是用户信息 userinfo、回调 Handler。

```
UserInfo userInfo = new UserInfo();
userInfo.setUser_name(mName);
userInfo.setPassword(mPassword);
userInfo.setPlatform("0");
LoginHelper loginHelper = new LoginHelper(LoginDemoActivity.this);
loginHelper.login(userInfo, mLoginHandler);
```

4. 注销

声明 LoginHelper 对象，参数为当前上下文，然后使用声明的对象调用 logout()方法。该方法只需要回调 Handler 一个参数。

```
LoginHelper loginHelper = new LoginHelper(PayActivity.this);
loginHelper.Logout(logoutResultHandler);
```

5. 支付

声明 AliPayHelper 对象，参数为当前上下文，然后使用声明的对象调用 startPay()方法。该方法需要商品对象 productInfo，回调 Handler 两个参数，创建商品信息对象时需要游戏币数目，钱数，以及商品名称。

```
ProductInfo productInfo = new ProductInfo(coinnum, money,
    "支付测试");
AliPayHelper payHelper = new AliPayHelper(PayActivity.this);
payHelper.startPay(productInfo, payResultHandler);
```

6. 数据

声明 DataHelper 对象，参数为当前上下文，该对象包含以下方法，所有方法通过回调 Handler 返回结果，所有通过异步请求得到的数据均使用 Message.obj 返回给开发者：

```
DataHelper dataHelper = new DataHelper(PayActivity.this);
```

6.1 获取数据列表（getDataList）

获取某个 key 所对应的数据列表

```
dataHelper.getDataList(key, payResultHandler);
```

6.2 存储数据（saveData）

通过 key 和 version 存储数据

```
dataHelper.saveData(key, version, data, payResultHandler);
```

6.3 获取数据(getData)

通过 key 和 version 获取数据

```
dataHelper.getData(key, version, payResultHandler);
```

6.4 删除数据(deleteData)

通过 key 和 version 删除数据

```
dataHelper.deleteData(key, version, payResultHandler);
```

6.5 获取最新数据(getLatestData)

获取某个 key 下最新保存的数据

```
dataHelper.getLatestData(key, payResultHandler);
```

6.6 归档数据(achiveData)

通过 key 和 version 将数据进行归档，归档后的数据不能够再进行其他操作

```
dataHelper.achriveData(key, version, data, payResultHandler);
```

7. 获取交易记录

声明 RecordHelper 对象，参数为当前上下文，通过该对象调用 getRecordUrl()方法，参数为回调 Handler，方法执行完毕后通过 message.obj 将结果返回给开发者，该结果为一个 Url 地址，开发者需要自己使用 webview 去加载该地址。

```
RecordHelper getRecordHelper=new RecordHelper(PayActivity.this);  
getRecordHelper.getRecordUrl(payResultHandler);
```

8. 无登陆支付

声明 SingleGamePay 对象，参数为当前上下文，然后使用声明的对象调用 startNologinPay()方法。该方法需要商品对象 productInfo，回调 Handler 两个参数，创建商品信息对象时需要游戏币数目，钱数，以及商品名称。

```
SingleGamePay singleGamePay = new SingleGamePay(
    UserSDKTestActivity.this);
mProductInfo = new ProductInfo("600", "0.01", "600金币");
singleGamePay.startNologinPay(mProductInfo, payResultHandler);
```

三、回调结果对照表

所有回调结果均存放在通过 handleMessage 获得的 message.what 中，使用方法如下：

```
private Handler initResultHandler = new Handler() {
    public void handleMessage(android.os.Message msg) {
        switch (msg.what) {
            case ResultFlag.NO_NET:
                Toast.makeText(UserSDKTestActivity.this, "无网络连接",
                    Toast.LENGTH_SHORT).show();
                break;
            case ResultFlag.INIT_SUCCESS:
                Toast.makeText(UserSDKTestActivity.this, "初始化成功",
                    Toast.LENGTH_SHORT).show();
                break;
            case ResultFlag.INIT_FAILED:
                Toast.makeText(UserSDKTestActivity.this, "初始化失败",
                    Toast.LENGTH_SHORT).show();
                break;
        }
    }
};
```

初始化	INIT_SUCCESS	初始化成功
	INIT_FAILED	初始化失败
支付	RQF_PAY_SUCCEED	支付成功
	RQF_PAY_FAILED	支付失败
	RQF_PAY_CANCEL	支付取消
	RQF_PAY_KEEP	正在支付（注释）

		①)
登陆	LOGIN_SUCCEED	登陆成功
	LOGIN_FAILED	登陆失败
注册	REGISTER_SUCCESS	注册成功
	REGISTER_FAILED	注册失败
注销	LOGOUT_SUCCESS	注销成功
	LOGOUT_FAILED	注销失败
交易记录	GET_RECORDURL_SUCCESS	获取交易记录成功
	GET_RECORDURL_FAILED	获取交易记录失败
数据	GET_DATA_LIST_SUCCESS	获取数据列表成功
	GET_DATA_LIST_FAILED	获取数据列表失败
	GET_DATA_SUCCESS	获取数据成功
	GET_DATA_FAILED	获取数据失败
	SAVE_DATA_SUCCESS	保存数据成功
	SAVE_DATA_FAILED	保存数据失败
	DELETE_DATA_SUCCESS	删除数据成功
	DELETE_DATA_FAILED	删除数据失败
	GET_LASTESTDATA_SUCCESS	获取最新数据成功

		功
	GET_LASTESTDATA_FAILED	获取最新数据失败
	ARCHIVE_SUCESS	归档数据成功
	ARCHIVE_FAILED	归档数据失败
公共返回值(注释②)	GENERAL_SWITCH_OFF	总开关关闭
	HEARTBEAT_SWITCH_OFF	心跳开关关闭
	STORAGE_SWITCH_OFF	数据开关关闭
	NO_NET	无网络

注释①：返回正在支付是由于支付宝回调延迟造成，开发者可将该种情况按照支付失败处理，后续有客服解决。

注释②：公共返回值是指所有方法在监听回调时都能够监听到的返回结果。

四、Server 端返回错误对照表

开发者调用 API 失败时，可以从 message.obj 中取得 server 端返回的错误信息

错误	错误编号	说明
token_no_match	10000	token验证失败
sign_no_match	10001	签名不匹配
jsondata_empty	10002	jsondata数据为空
params_no_exists	10003	jsondata有必需参数不存在
email_error	10004	邮箱验证错误
username_exists	10005	通行证账号已经存在
register_fail	10006	通行证注册失败
username_no_exists	10007	通行证账号不存在
username_password_error	10008	通行证密码错误
username_no_activate	10009	通行证还没有激活
username_forbidden	10010	通行证已被禁止访问
user_login_expie	10011	通行证登陆已过期
login_token_error	10012	通行证登陆token错误

username_email_exists	10013	通行证邮箱已经绑定
email_binding	10014	邮箱已经被其他账号绑定
send_email_fail	10015	找回密码的邮件发送失败
udid_no_exists	10016	jsondata中udid不存在
money_error	10017	所要充值钱数错误，为零或负数
order_fail	10018	订单生成失败
userid_udid_error	10019	通行证错误或udid错误
coins_error	10020	用户拥有的钱数获取失败
Coins_less	10021	用户拥有的钱数比扣除的少
order_no_exists	10022	订单号不存在
userid_empty	10023	用户id为空
userid_error	10024	用户id错误
Ukid_no_exists	10025	Ckid不存在