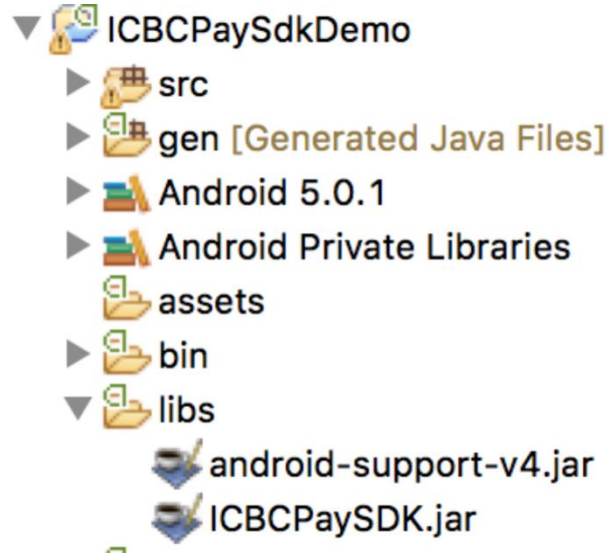


## 移动端线上支付产品 SDK 接入文档（安卓版）

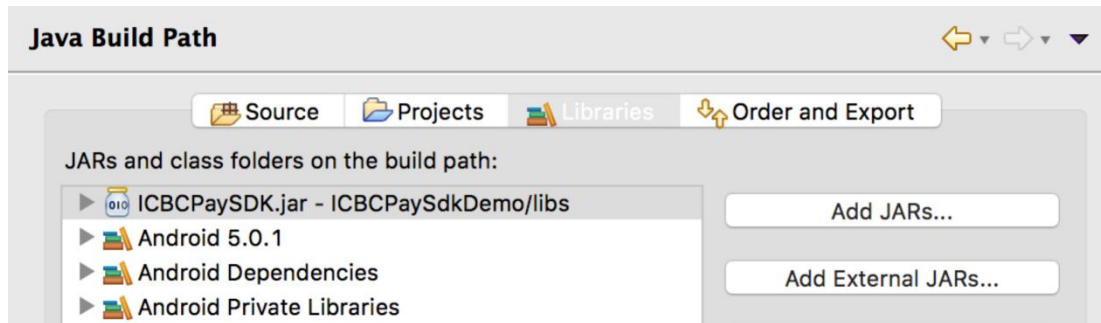
### 前期准备：

#### 一、导入开发资源

(1) 将开发包 ICBCPaySDK.jar 导入商户应用工程的 libs 目录下，如下图。



(2) 进入商户应用工程的 Java Build Path，将 libs 目录下的 ICBCPaySDK.jar 导入，如下图所示。



#### 二、修改 Manifest

(1) 使用工行支付 SDK 的 APP 需要在工程的 AndroidManifest.xml 中加入以下声明：

```
<activity
    android:name=".icbcPay.PayResultHandler"
    android:exported="true"
    android:launchMode="singleTask">
<intent-filter>
    <category android:name="android.intent.category.DEFAULT"/>
```

```

        <action android:name="com.icbc.pay.PayResultHandler.SHOW_ACTIVITY"/>
    </intent-filter>

    </activity>
</activity>
<activity
    android:name="com.icbc.paysdk.webview.PayWebView"
    android:theme="@android:style/Theme.Light.NoTitleBar"
    android:alwaysRetainTaskState="true"
    android:configChanges="orientation|keyboardHidden|screenSize"
    android:imeOptions="actionSend|flagNoEnterAction"
    android:screenOrientation="portrait"
    android:windowSoftInputMode="adjustResize" >
</activity>

```

(2) 添加权限声明:

```

<uses-permission android:name="android.permission.INTERNET"/>
<uses-permission android:name="android.permission.ACCESS_NETWORK_STATE"/>
<uses-permission android:name="android.permission.ACCESS_WIFI_STATE"/>

```

## 支付流程:

### 一、获取支付数据

商户服务器生成支付订单，先调用工行支付 API，获取支付数据。

返回支付数据包括的字段如下:

- (1) interfaceName  
接口名
- (2) interfaceVersion  
接口版本号
- (3) tranData  
交易信息
- (4) merSignMsg  
交易信息签名
- (5) merCert  
商户公钥文件信息

### 二、调起支付

调用 SDK API 发起支付。

```

PayReq req = new PayReq();
req.setInterfaceName(interfaceName);
req.setInterfaceVersion(interfaceVersion);
req.setTranData(tranData);
req.setMerSignMsg(merSignMsg);
req.setMerCert(merCert);

```

```
ICBCAPI.getInstance().sendReq(MainActivity.this,req);
```

#### 注意:

SDK的默认连接地址为生产地址:

支付平台: <https://b2c.icbc.com.cn>

用户若要修改地址可在调用sendReq()前, 加入如下方法:

```
Constants.PAY_LIST_IP = "新支付平台地址";
```

用户若需在测试环境调试, 地址如下:

```
Constants.PAY_LIST_IP = "https://b2c3.dccnet.com.cn";
```

### 三、支付结果回调

在 `com.icbc.pay` 包路径中实现 `PayResultHandler` 类(包名或类名不一致会造成无法回调), 在 `PayResultHandler` 类中实现 `onResp` 函数, 支付完成后, 会返回到 `PayResultHandler` 并回调 `onResp` 函数, 开发者需要在该函数中接收通知, 判断返回错误码, 如果支付成功则去后台查询支付结果再展示用户实际支付结果。注意一定不能以客户端返回作为用户支付的结果, 应以服务器端的接收的支付通知或查询 API 返回的结果为准。实现 `onErr` 函数, 可处理支付过程中出现的网络问题。代码示例如下:

```
@Override
public void onErr(ReqErr err) {
    // TODO Auto-generated method stub
    String errorType = err.getErrorType();
}

@Override
public void onResp(PayResp resp) {
    // TODO Auto-generated method stub
    String tranCode = resp.getTranCode();

    if("1".equals(tranCode)){
        result_text.setText("支付成功");
    }else if("2".equals(tranCode)){
        result_text.setText("交易失败");
    }else if("3".equals(tranCode)){
        result_text.setText("交易可疑");
    }else if("4".equals(tranCode)){
        result_text.setText("用户中止交易");
    }
}
```

#### 四、补充说明：

- (1) 具体使用方法，请参考 ICBCPayDemo。
- (2) 该版本 sdk 支持 Android4.0 及以上版本。