

WiseCashier Java SDK 使用文档

1. 获取SDK

- 单击此处[下载](#)
- 请配合[API文档](#)使用

2. AgentNo 和 MerchantNo

使用SDK需要MerchantNo，由WiseCashier提供的。

3. 依赖

- gons = 2.7
- commons-codec = 1.9
- jdk > 1.6

4. 设置请求签名密钥

密钥需要你自己生成，请将私钥放入放入项目文件夹，并将公钥发送给WiseCashier。

生成公钥

```
openssl genrsa -out private.key 2048
```

生成私钥

```
openssl rsa -pubout -in private.key -out public.key
```

5. 实例化WiseCashier

```
import com.transfereasy.wisecashier.WiseCashier;  
# 第三个参数为True时为测试环境， 为False时为正式环境  
WiseCashier wisecashier = new WiseCashier(  
    "MerchantNo", "Path_to_private.key", True or Flase );
```

6. 获得结果

SDK的返回值是一个Map，可以用以下方式取值：

```
Map params = new HashMap();
Map response = null;
params.put("currency", "{Currency}");
params.put("merchant_no", "{MerchantNo}");
try {
    response = wisecashier.wxQueryRate(params);
} catch (exception e) {
    e.printStackTrace();
    retrue;
}
Map meta = (Map) response.get("meta");
Map data = (Map) response.get("data");
if((Double)meta.get("status_code") == 200) {
    Double rate = (Double) data.get("rate");
    String currency = data.get("currency").toString();
    System.out.println(String.format("%s rate is %s.",currency,rate));
} else {
    System.out.println("error_code:" + meta.get("status_code").toString());
    System.out.println("message:" + meta.get("message").toString());
}
```

7. 示例

7.1 微信扫一扫收款

API说明

```
Map params = new HashMap();
params.put("merchant_no", "{MerchantNo}"); //商户号
params.put("currency", "{currency}"); //币种
params.put("amount", "{amount}"); //金额
params.put("product_info", "{product_info}"); //产品信息
params.put("ip", "{ip}"); //终端ip
params.put("auth_code", "{auth_code}"); //客户二维码内容
params.put("agent_order_id", "{agent_order_id}"); //商户订单号，可选
params.put("product_id", "{product_id}"); //产品编号，可选
params.put("device_info", "{device_info}"); //设备信息，可选
try {
    response = wisecashier.wxScanPay(params);
} catch (exception e) {
```

```
e.printStackTrace()  
}
```

7.2 微信二维码收款

API说明

```
Map params = new HashMap();  
params.put("merchant_no", "{MerchantNo}"); // 商户号  
params.put("currency", "{currency}"); // 币种  
params.put("amount", "{amount}"); // 金额  
params.put("product_info", "{product_info}"); // 产品信息  
params.put("ip", "{ip}"); // 客户二维码内容  
params.put("notify_url", "{notify_url}"); // 支付结果通知地址, 可选  
params.put("agent_order_id", "{agent_order_id}"); // 商户订单号, 可选  
params.put("product_id", "{product_id}"); // 产品编号, 可选  
params.put("device_info", "{device_info}"); // 设备信息, 可选  
try {  
    response = wisecashier.wxScanndepay(params);  
} catch (exception e) {  
    e.printStackTrace()  
}
```

7.3 微信退款

API说明

```
Map params = new HashMap();  
params.put("order_no", "{order_no}"); // 订单编号  
params.put("refund_amount", "{refund_amount}"); // 退款金额, 可选  
try {  
    response = wisecashier.wxRefund(params);  
} catch (exception e) {  
    e.printStackTrace()  
}
```

7.4 微信查询支付单

API说明

```
Map params = new HashMap();  
params.put("order_no", "{order_no}"); // 支付订单编号  
try {
```

```
        response = wisecashier.wxQueryOrder(params);
    } catch (exception e) {
        e.printStackTrace()
    }
```

7.5 微信查询汇率

API说明

```
Map params = new HashMap();
params.put("merchant_no", "{MerchantNo}"); // 商户号
params.put("currency", "{Currency}"); // 币种
try {
    response = wisecashier.wxQueryRate(params);
} catch (exception e) {
    e.printStackTrace()
}
```

7.6 支付宝二维码支付

API说明

```
Map params = new HashMap();
params.put("merchant_no", "{MerchantNo}"); // 商户号
params.put("currency", "{currency}"); // 币种
params.put("amount", "{amount}"); // 金额
params.put("subject", "{subject}"); // 产品信息
params.put("ip", "{ip}"); // 客户ip, 可选
params.put("notify_url", "{notify_url}"); // 支付结果通知地址, 可选
params.put("out_order_no", "{out_order_no}"); // 商户订单号, 可选
try {
    response = wisecashier.aliScannedPay(params);
} catch (exception e) {
    e.printStackTrace()
}
```

7.7 支付宝WAP支付

API说明

```
Map params = new HashMap();
params.put("merchant_no", "{MerchantNo}"); // 商户号
params.put("currency", "{currency}"); // 币种
```

```

params.put("amount", "{amount}"); //金额
params.put("subject", "{subject}"); //产品信息
params.put("ip", "{ip}"); //客户ip, 可选
params.put("notify_url", "{notify_url}"); //支付结果通知地址, 可选
params.put("out_order_no", "{out_order_no}"); //商户订单号, 可选
params.put("return_url", "{return_url}"); //支付完成后返回的地址, 可选
try {
    response = wisecashier.aliWapPay(params);
} catch (exception e) {
    e.printStackTrace()
}

```

7.8 支付宝查询订单

API说明

```

Map params = new HashMap();
params.put("payment_no", "{payment_no}"); //Wisecasheir 订单号
try {
    response = wisecashier.aliQueryOrder(params);
} catch (exception e) {
    e.printStackTrace()
}

```

7.9 支付宝退款

API说明

```

Map params = new HashMap();
params.put("payment_no", "{payment_no}"); //Wisecasheir 订单号
params.put("refund_amount", "{refund_amount}"); //退款金额, 不填为全额退款
try {
    response = wisecashier.aliRefund(params);
} catch (exception e) {
    e.printStackTrace()
}

```

7.10 异步通知验签

```

# Map params 为请求的所有参数
# sgin和timestamp 从请求头中获取
# pubKey 由wisecashier提供, 区分生产环境和测试环境
Boolean result = wisecashier.verify(params, sign, timestamp, pubKeyPath)

```

