

中国银行

微信 APP 支付接口文档

版本号：1.0.0

版本记录：

编号	时间	版本号	修改说明
1	2016.11.07	V1.0.0	✧ 接口文档的创建。

目 录

1. 微信 APP 支付简介.....	1
1.1. 场景介绍.....	1
1.2. 业务流程.....	3
1.3. APP 端开发步骤说明.....	4
1.3.1 IOS 开发要点说明.....	4
1.3.2 Android 开发要点说明.....	6
2. 接口说明.....	8
2.1. 接口调用规则.....	8
2.2. SIGN 签名生成方法.....	9
2.3. 代码实例.....	10
3. API 列表.....	12
3.1. 统一支付接口.....	12
3.1.1 应用说明:	12
3.1.2 接口地址:	12
3.1.3 输入参数:	12
3.1.4 返回结果:	14
3.1.5 错误码:	15
3.2. 调起支付接口.....	16
3.2.1 应用说明:	16
3.2.2 接口地址:	16
3.2.3 请求参数:	16
3.2.4 返回结果:	17
3.3. 支付结果通知接口.....	17
3.3.1 应用说明:	17
3.3.2 接口地址:	17
3.3.3 通知参数:	18
3.3.4 返回结果:	19
3.4. 查询订单接口.....	19
3.4.1 应用说明:	19
3.4.2 接口地址:	20
3.4.3 输入参数:	20
3.4.4 返回结果:	21
3.4.5 错误码:	23
3.5. 关闭订单接口.....	23
3.5.1 应用说明:	23
3.5.2 接口地址:	24
3.5.3 输入参数:	24
3.5.4 返回结果:	24
3.5.5 错误码:	25
3.6. 申请退款接口.....	25
3.6.1 应用说明:	25
3.6.2 接口地址:	26
3.6.3 输入参数:	26
3.6.4 返回结果:	27
3.6.5 错误码:	28
3.7. 查询退款接口.....	28
3.7.1 应用说明:	28
3.7.2 接口地址:	28
3.7.3 输入参数:	29
3.7.4 返回结果:	29
3.7.5 错误码:	31
3.8. 对账单接口.....	31

3.8.1 应用说明:	31
3.8.2 接口地址:	32
3.8.3 输入参数:	32
3.8.4 返回结果:	32
3.8.5 错误码:	34

1. 微信 APP 支付简介

1.1. 场景介绍

适用于商户在移动端 APP 中集成微信支付功能。

商户 APP 调用微信提供的 SDK 调用微信支付模块，商户 APP 会跳转到微信中完成支付，支付完后跳回到商户 APP 内，最后展示支付结果。

目前微信支付支持手机系统有：IOS（苹果）、Android（安卓）和 WP（Windows Phone）。

交互细节如下：

步骤 1： 用户进入商户 APP，选择商品下单、确认购买，进入支付环节。商户服务后台生成支付订单，签名后将数据传输到 APP 端。以微信提供的 DEMO 为例，见图 1.1.1。

步骤 2： 用户点击后发起支付操作，进入到微信界面，调起微信支付，出现确认支付界面，见图 1.1.2。

步骤 3： 用户确认收款方和金额，点击立即支付后出现输入密码界面，可选择零钱或银行卡支付见图 1.1.3。



图 1.1.1



图 1.1.2



图 1.1.3

步骤 4: 输入正确密码后，支付完成，用户端微信出现支付详情页面。见图 1.1.4。

步骤 5: 回跳到商户 APP 中，商户 APP 根据支付结果个性化展示订单处理结果。见图 1.1.5。

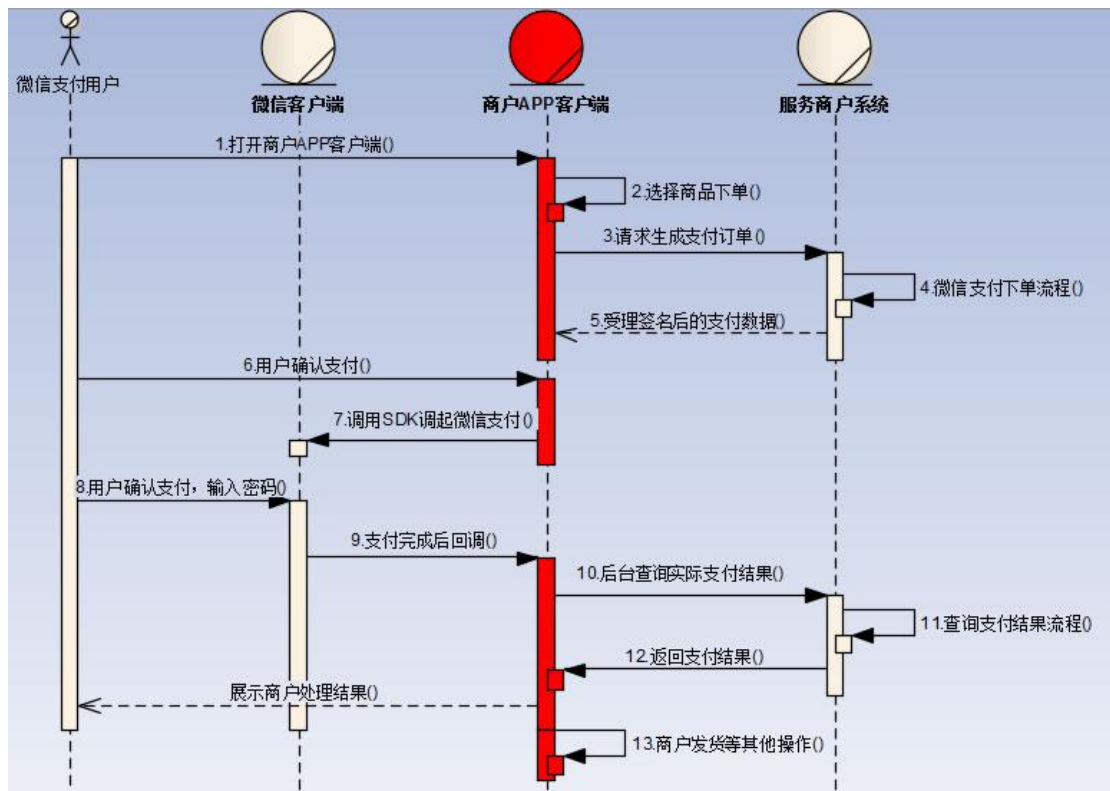


图 1.1.4



图 1.1.5

1.2. 业务流程



商户 APP 调用支付的主要交互说明：

- 1) 用户在商户 APP 中选择商品，选择微信支付，提交订单，如图中步骤 1-3 所示。
- 2) 调用中行服务商系统提供的【统一下单 API】，中行服务商系统后台收到下单请求，会返回签好名的订单数据，用于商户 APP 里面调起微信支付，如图中步骤 3-5 所示。
- 3) 用户确认支付，输入密码，支付完成，如图中步骤 6-8 所示。
- 4) 支付完成后，微信返回商户 APP，回调 APP 实现的回调函数，此时需要根据单号调用中行服务商系统提供的查询结果，查询后台实际支付结果，或通过【支付结果通知 API】接收到实际支付结果，然后再作用户页面展示和发货操作。如图步骤 9-13。

1.3. APP 端开发步骤说明

1.3.1 IOS 开发要点说明

以下项目开发环境以 Xcode10.0，运行环境为 IOS7.0 为例，说明其开发中需要的操作。

1) 项目设置 APPID

商户在微信开放平台申请开发 APP 应用后，微信开放平台会生成 APP 的唯一标识 APPID。在 Xcode 中打开项目，设置项目属性中的 URL Schemes 为商户的 APPID。如图 1.3.1.1 标红位置所示。

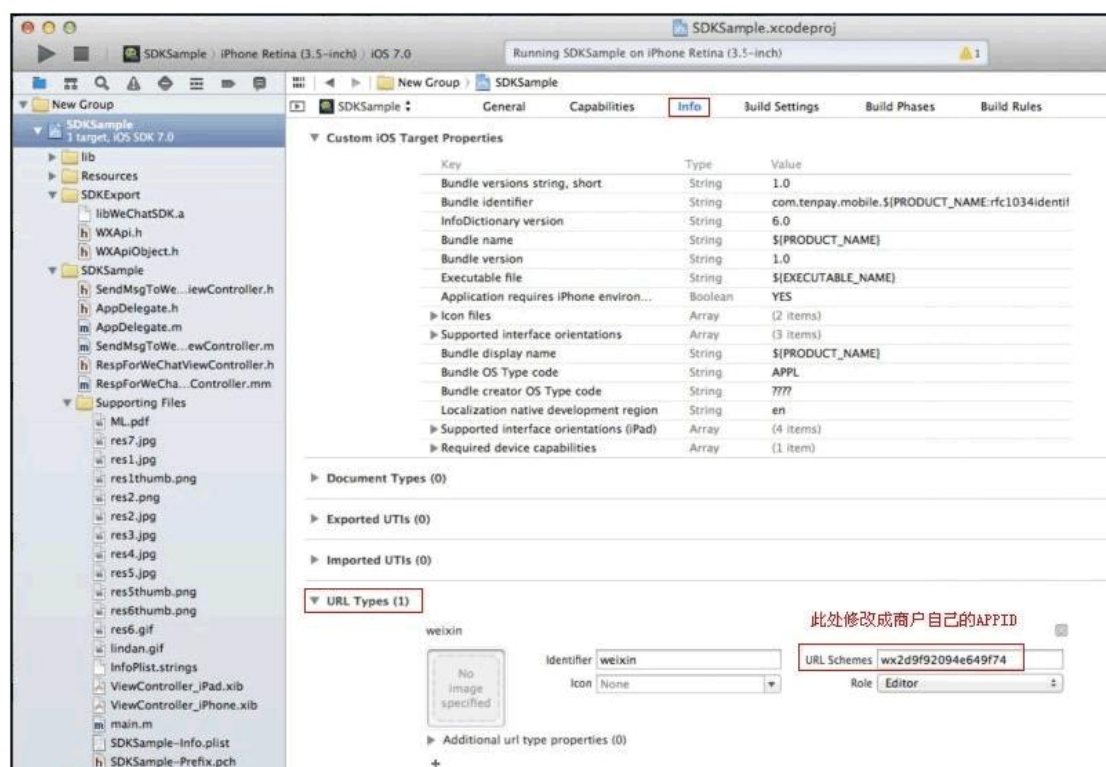


图 1.3.1.1

2) 注册 APPID

APP 工程中引入微信 lib 库和头文件，调用 API 前，需要先向微信注册商户的 APPID，代码如下：

```
[WXApi registerApp : @"wxd930ea5d5a258f4f" withDescription : @"demo 2.0"];//
```

3) 调起支付

商户调用【统一下单 API】返回支付数据，在 APP 发起支付。以下是调起

微信支付的关键代码：

```
PayReq *request = [[[PayReq alloc] init] autorelease];
request.partnerId = @"10000100";//商户的商户号
request.prepayId= @"1101000000140415649af9fc314aa427";
request.package = @"Sign=WXPAY";
request.nonceStr= @"a462b76e7436e98e0ed6e13c64b4fd1c";
request.timeStamp= @"1397527777";
request.sign= @"582282D72DD2B03AD892830965F428CB16E7A256";
[WXApi sendReq: request];
```

4) 支付结果回调

照微信 SDK Sample，在类实现 onResp 函数，支付完成后，微信 APP 会返回到商户 APP 并回调 onResp 函数，开发者需要在此函数中接收，判断返回错误码，再调用中行服务商系统提供接口查询支付结果，最后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，应以后台的接收的支付通知或查询 API 返回的结果为准。

代码示例如下：

```
-(void)onResp: (BaseResp*)resp{
    if ([resp isKindOfClass: [PayRespclass]]){
        PayResp*response=(PayResp*)resp;
        switch(response.errCode){
            case WXSuccess:
                //服务器端查询支付通知或查询 API 返回的结果再提示成功
                NSLog(@"支付成功");
                break;
            default:
                NSLog(@"支付失败，retcode=%d",resp.errCode);
                break;
        }
    }
}
```

回调中 errCode 值列表：

名称	描述	解决方案
0	成功	展示成功页面
-1	错误	可能的原因：签名错误、未注册 APPID、项目设置 APPID 不正确、注册的 APPID 与设置的不匹配、其他异常等。
-2	用户取消	无需处理。发生场景：用户不支付了，点击取消，返回 APP。

1.3.2 Android 开发要点说明

1) 后台设置

商户在微信开放平台申请开发应用后，微信开放平台会生成 APP 的唯一标识 APPID。由于需要保证支付安全，需要在开放平台绑定商户应用包名和应用签名，设置好后才能正常发起支付。设置界面在【开放平台】中的栏目【管理中心 / 修改应用 / 修改开发信息】里面，如图 1.3.2.1 红框内所示。

应用平台 请至少选择一个平台

☒ iOS 应用

AppStore下载地址 (选填) Apple AppStore中的下载地址，如应用还未上线，可置空，待应用上线后再行修改

☒ Android 应用

应用下载地址 (选填) Android应用商店中的下载页面的地址，不允许直接使用apk包的下载地址，如应用还未上线，可置空，待应用上线后再行修改

应用签名 用于对当前应用进行二次身份校验，开发者可以使用签名生成工具直接从安装当前应用的手机中获取。应用签名由开发者签名该应用的keystore文件决定。

应用包名 应用在一台设备上的唯一标识，在manifest文件里面声明，该包名应和正式发布应用的包名一致。例如，微信的包名为com.tencent.mm

☐ WP8 应用

图 1.3.2.1

应用包名：是在 APP 项目配置文件 AndroidManifest.xml 中声明的 package 值，例如 DEMO 中的 package="net.sourceforge.simcpux"。

应用签名：根据项目的应用包名和编译使用的 keystore，可由签名工具生成一个 32 位的 md5 串，在调试的手机上安装签名工具后，运行可生成应用签名串，如图 1.3.2.1 所示，绿色串即应用签名。签名工具下载地址

https://open.weixin.qq.com/zh_CN/htmledition/res/dev/download/sdk/Gen_Signature_Android.apk



图 1.3.2.2

2) 注册 APPID

商户 APP 工程中引入微信 JAR 包，调用 API 前，需要先向微信注册子商户的 APPID，代码如下：

```
final IWXAPI msgApi = WXAPIFactory.createWXAPI(context, null);
// 将该 app 注册到微信
msgApi.registerApp("wxd930ea5d5a258f4f");
```

3) 调起支付

子商户调用【统一下单 API】返回支付数据，在 APP 发起支付。以下是调起微信支付的关键代码：

```
IWXAPI api;
PayReq request = new PayReq();
request.appId = "wxd930ea5d5a258f4f";//商户 appid
request.partnerId = "1900000109";//商户号
request.prepayId = "1101000000140415649af9fc314aa427",;
request.packageValue = "Sign=WXPay";
request.nonceStr = "1101000000140429eb40476f8896f4c9";
request.timeStamp = "1398746574";
request.sign = "7FFECB600D7157C5AA49810D2D8F28BC2811827B";
api.sendReq(req);
```

4) 支付结果回调

参照微信 SDK Sample，在 net.sourceforge.simcpux.wxapi 包路径中实现

WXPayEntryActivity 类 (包名或类名不一致会造成无法回调)，在 WXPayEntryActivity 类中实现 onResp 函数，支付完成后，微信 APP 会返回到商户 APP 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，支付成功则调用“服务商提供的查询接口”获取到支付状态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。代码示例如下：

```
public void onResp(BaseResp resp) {
    if (resp.getType() == Constants.API.COMMAND_PAY_BY_WX) {
        Log.d(TAG, "onPayFinish, errCode=" + resp.errCode);
        AlertDialog.Builder builder = new AlertDialog.Builder(this);
        builder.setTitle(R.string.app_tip);
    }
}
```

回调中 errCode 值列表：

名称	描述	解决方案
0	成功	展示成功页面
-1	错误	可能的原因：签名错误、未注册 APPID、项目设置 APPID 不正确、注册的 APPID 与设置的不匹配、其他异常等。
-2	用户取消	无需处理。发生场景：用户不支付了，点击取消，返回 APP。

2. 接口说明

2.1. 接口调用规则

- ◆ 采用 HTTPS 方式访问
- ◆ 请求采用 POST 方式
- ◆ 提交和返回结果采用 XML 格式
- ◆ 字符集默认使用 UTF-8，请勿使用其它字符集
- ◆ 商户与中国银行微信 APP 支付平台之间的交互，都需要验证签名。
- ◆ 处理返回时先判断协议返回错误码，再判断业务返回错误码，最后判断交易状态

2.2. Sign 签名生成方法

微信支付中将对数据里面的内容进行鉴权，确定携带的信息是真实、有效、合理的。因此，这里将定义生成 sign 字符串的方法。

a. 对所有传入参数按照字段名的 ASCII 码从小到大排序（字典序）后，使用 URL 键值对的格式（即 key1=value1&key2=value2...）拼接成字符串 string1，注意：值为空的参数不参与签名；

b. 在 string1 最后拼接上 key=Key(商户支付密钥)得到 stringSignTemp 字符串，并对 stringSignTemp 进行 md5 运算，再将得到的字符串所有字符转换为大写，得到 sign 值 signValue。

下面定义了一段生成 sign 字符串的示范过程：

假设以下为 package 传入参数：

```
appid=wx930ea5d5a258f4f
auth_code=123456
body=test
device_info=123
mch_id=1900000109
nonce_str=960f228109051b9969f76c82bde183ac
out_trade_no=1400755861
spbill_create_ip=127.0.0.1
total_fee=1
key=8934e7d15453e97507ef794cf7b0519d
```

i: 经过 a 过程 URL 键值对字典序排序后的字符串 string1 为：

```
appid=wx930ea5d5a258f4f&auth_code=123456&body=test&device_info=123
&mch_id=1900000109&nonce_str=960f228109051b9969f76c82bde183ac&out_trade_no=1400755861&spbill_create_ip=127.0.0.1&total_fee=1
```

ii: 经过 b 过程后得到 sign 为：

```
Sign
=md5(string1&key=8934e7d15453e97507ef794cf7b0519d).toUpperCase
=md5(appid=wx930ea5d5a258f4f&auth_code=123456&body=test&device_info=123&mch_id=1900000109&nonce_str=960f228109051b9969f76c82bde183ac&o
```

```
ut_trade_no=1400755861&spbill_create_ip=127.0.0.1&total_fee=1&key=8934e7d1
5453e97507ef794cf7b0519d).toUpperCase()
="729a68ac3de268dbd9ade442382e7b24".toUpperCase()
="729A68AC3DE268DBD9ADE442382E7B24"
```

2.3. 代码实例

接口调用实例，以开发为例说明数据提交和返回的具体格式：

提交数据：

```
<xml>
<appid>wx2421b1c4370ec43b</appid>
<attach><![CDATA[`store_appid=xx#store_name=xx#op_user=xx`]]></attach>
>
<body><![CDATA[JSAPI 支付测试]]></body>
<device_info>1000</device_info>
<mch_id>10000100</mch_id>
<nonce_str>b927722419c52622651a871d1d9ed8b2</nonce_str>
<notify_url>http://wxpay.weixin.qq.com/pub_v2/pay/notify.php</notify_url>
<out_trade_no>1405713376</out_trade_no>
<spbill_create_ip>127.0.0.1</spbill_create_ip>
<total_fee>1</total_fee>
<trade_type>JSAPI</trade_type>
<sign><![CDATA[3CA89B5870F944736C657979192E1CF4]]></sign>
</xml>
```

协议级错误返回：

```
<xml>
<return_code><![CDATA[FAIL]]></return_code>
<return_msg><![CDATA[SYSERR]]></return_msg>
</xml>
```

正确返回数据：

```
<xml>
<return_code><![CDATA[SUCCESS]]></return_code>
<return_msg><![CDATA[OK]]></return_msg>
```

```

<appid><![CDATA[wx2421b1c4370ec43b]]></appid>
<mch_id><![CDATA[10000100]]></mch_id>
<device_info><![CDATA[1000]]></device_info>
<nonce_str><![CDATA[FvYsnPuFFPkAr77M]]></nonce_str>
<sign><![CDATA[63238039D6E43634297CF2A6EB5F3B72]]></sign>
<result_code><![CDATA[SUCCESS]]></result_code>
<openid><![CDATA[oUpF8uN95-Ptaags6E_roPHg7AG0]]></openid>
<is_subscribe><![CDATA[Y]]></is_subscribe>
<trade_type><![CDATA[MICROPAY]]></trade_type>
<bank_type><![CDATA[CCB_CREDIT]]></bank_type>
<total_fee>1</total_fee>
<coupon_fee>0</coupon_fee>
<fee_type><![CDATA[CNY]]></fee_type>
<transaction_id><![CDATA[10084507402014072200000058756]]></transactio
n_id>
<out_trade_no><![CDATA[1406033828]]></out_trade_no>
<attach><![CDATA[store_appid=xx#store_name=xx#op_user=xx]]></attach
>
<time_end><![CDATA[20140722160655]]></time_end>
</xml>

```

业务级错误返回：

```

<xml>
<return_code><![CDATA[SUCCESS]]></return_code>
<return_msg><![CDATA[OK]]></return_msg>
<appid><![CDATA[wx2421b1c4370ec43b]]></appid>
<mch_id><![CDATA[10000100]]></mch_id>
<device_info><![CDATA[1000]]></device_info>
<nonce_str><![CDATA[sthBJ9QyUG6vkrjJ]]></nonce_str>
<sign><![CDATA[6277A96D7875D4FF23AA7B6A4C3046AB]]></sign>
<result_code><![CDATA[FAIL]]></result_code>
<err_code><![CDATA[PAYERROR]]></err_code>
<err_code_des><![CDATA[支付错误]]></err_code_des>
</xml>

```

3. API 列表

3.1. 统一支付接口

3.1.1 应用说明:

商户系统先调用该接口在微信支付服务后台生成预支付交易单, 返回正确的预支付交易回话标识后再在 APP 里面调起支付。

3.1.2 接口地址:

<https://api.boc.dcorepay.com/pay/unifiedorder>

3.1.3 输入参数:

字段名	变量名	必填	类型	说明
版本信息	version	否	String(8)	若不填写, 则默认为 1.0.4
终端类型	device_type	否	String(10)	商户 APP 所在的终端操作系统类型, 填写 ANDROID 或 IOS。trade_type 等于 APP 时需要填写。
应用 ID	appid	是	String(32)	中国银行分配的 APPID。 注意: 此中国银行分配的 APPID 仅用于在各 API 接口调用中使用, 商户在本接口文档中 1.3 章节的开发配置中应使用微信开放平台中申请生成的 APPID。
商户号	mch_id	是	String(32)	中国银行分配的商户号
微信 APPID	wx_appid	否	String(32)	商户在微信开放平台上申请的 appid。trade_type 等于 APP 时需要填写。
设备号	device_info	否	String(32)	终端设备号(门店号或收银设备 ID), 默认请传"WEB"。
随机字符串	nonce_str	是	String(32)	随机字符串, 不长于 32 位。推荐随机数生成算法
签名	sign	是	String(32)	签名, 详细签名方法见 2.2 节

商品描述	body	是	String(32)	商品或支付单的简要描述
商品详情	detail	否	String(4000)	商品或支付单详情，如明细列表等
附加数据	attach	是	String(127)	<p>附加数据。用于识别该笔支付单的发起门店及操作员。</p> <p>格式为： `store_appid=xx#store_name=xx#op_user=xx。</p> <p>各参数以符号#分隔，参数前增加`符号，为标准键盘 1 左边键的字符，参数顺序与格式描述一致。其中， store_appid 和 store_name 的值必须填写，此两个值需与商户管理中心的门店信息一致，否则该支付单将无法在商户管理中心查询到； op_user 可填写为空，若有收银员，请填写上收银员标识（工号）。</p> <p>例如： `store_appid=s2015053015010100001#store_name=永辉超市福建省福州市鼓山苑店#op_user=0000001。</p>
商户订单号	out_trade_no	是	String(32)	商户系统内部的订单号,32 个字符内、可包含字母。 注意：请务必确保该订单号在商户系统的唯一性。
货币类型	fee_type	否	String(16)	货币类型。默认为人民币：CNY
总金额	total_fee	是	Int	订单总金额，单位为分，不能带小数点
终端 IP	spbill_create_ip	是	String(16)	提交用户端 IP
交易起始时间	time_start	否	String(14)	订单生成时间， 格式为 yyyyMMddHHmmss，如 2009 年 12 月 25 日 9 点 10 分 10 秒表示为 20091225091010。时区为 GMT+8 beijing。该时间取自商户服务器
交易结束时间	time_expire	否	String(14)	订单失效时间， 格式为 yyyyMMddHHmmss，如 2009 年 12 月 27 日 9 点 10 分 10 秒表示为 20091227091010。时区为 GMT+8 beijing。该时

				间取自商户服务器
商品标记	goods_tag	否	String(32)	商品标记，优惠券或者立减优惠功能参数。
通知地址	notify_url	是	String(256)	接收微信支付成功通知
交易类型	trade_type	是	String(16)	调用接口提交的交易类型，取值如下：JSAPI，NATIVE，APP
指定支付方式	limit_pay	否	String(32)	no_credit--指定不能使用信用卡支付

3.1.4 返回结果：

字段名	变量名	必填	类型	说明
版本信息	version	否	String(8)	若不填写，则默认为 1.0.4
返回状态码	return_code	是	String(16)	SUCCESS/FAIL 此字段是通信标识，非交易标识，交易是否成功需要查 result_code 来判断
返回信息	return_msg	否	String(128)	返回信息，如非空，为错误原因 签名失败 参数格式校验错误
以下字段在 return_code 为 SUCCESS 的时候有返回				
应用 ID	appid	是	String(32)	调用接口提交的 APPID
商户号	mch_id	是	String(32)	调用接口提交的商户号
微信 APPID	wx_appid	是	String(32)	商户在微信开放平台上申请的 appid
设备号	device_info	否	String(32)	调用接口提交的终端设备号
随机字符串	nonce_str	是	String(32)	中国银行返回的随机字符串，不长于 32 位
签名	sign	是	String(32)	中国银行返回的签名, 详细签名方法见 2.2 节
业务结果	result_code	是	String(16)	SUCCESS/FAIL
错误代码	err_code	否	String(32)	详见错误代码列表
错误代码描述	err_code_des	否	String(128)	错误信息描述

以下字段在 return_code 和 result_code 都为 SUCCESS 的时候有返回				
交易类型	trade_type	是	String(16)	调用接口提交的交易类型,取值如下: JSAPI, NATIVE, APP
预支付 ID	prepay_id	是	String(64)	微信生成的预支付 ID, 用于后续接口调用中使用, 该值有效期为 2 小时
以下字段在 return_code 和 result_code 都为 SUCCESS, 并且 trade_type 为 APP 的时候有返回				
app 调起支付中的 appid 参数	req_appid	是	string(32)	调起支付中的 appid 参数
app 调起支付中的 partnerid 参数	req_partnerid	是	string(32)	调起支付中的 partnerid 参数
app 调起支付中的 prepayid 参数	req_prepayid	是	string(32)	调起支付中的 prepayid 参数
app 调起支付中的 package 参数	req_package	是	string(128)	调起支付中的 package 参数
app 调起支付中的 noncestr 参数	req_noncestr	是	string(32)	调起支付中的 noncestr 参数
app 调起支付中的 timestamp 参数	req_timestamp	是	string(10)	调起支付中的 timestamp 参数
app 调起支付中的 sign 参数	req_sign	是	string(32)	调起支付中的 sign 参数

3.1.5 错误码:

名称	描述	原因	解决方案
NOAUTH	商户无此接口权限	商户未开通此接口权限	请商户前往申请此接口权限
NOTENOUGH	余额不足	用户帐号余额不足	用户帐号余额不足, 请用户充值或更换支付卡后再支付
ORDERPAID	商户订单已支付	商户订单已支付, 无需重复操作	商户订单已支付, 无需更多操作
ORDERCLOSED	订单已关闭	当前订单已关闭, 无法支付	当前订单已关闭, 请重新下单
SYSTEMERROR	系统错误	系统超时	系统异常, 请用相同参数重新调用
APPID_NOT_EXIST	APPID 不存在	参数中缺少 APPID	请检查 APPID 是否正确
MCHID_NOT_EXIST	MCHID 不存在	参数中缺少 MCHID	请检查 MCHID 是否正确
APPID_MCHID_NOT_MATCH	appid 和 mch_id 不匹配	appid 和 mch_id 不匹配	请确认 appid 和 mch_id 是否匹配
LACK_PARAMS	缺少参数	缺少必要的请求参数	请检查参数是否齐全
OUT_TRADE_NO_USED	商户订单号重复	同一笔交易不能多次提交	请核实商户订单号是否重复提交
SIGNERROR	签名错误	参数签名结果不正确	请检查签名参数和方法是否都符合签名算法要求
XML_FORMAT_ERROR	XML 格式错误	XML 格式错误	请检查 XML 参数格式是否正确
REQUIRE_POST_METHOD	请使用 post 方	未使用 post 传递参数	请检查请求参数是否通过 post 方法

	法		提交
POST_DATA_EMPTY	post 数据为空	post 数据不能为空	请检查 post 数据是否为空
NOT_UTF8	编码格式错误	未使用指定编码格式	请使用 NOT_UTF8 编码格式

3.2. 调起支付接口

3.2.1 应用说明：

在 APP 端调起支付的参数列表，注意：在 APP 端调起支付前应先调用【统一支付接口 API】获取到调起支付需要使用到的所有参数。

3.2.2 接口地址：

无。

3.2.3 请求参数：

字段名	变量名	必填	类型	说明
app 调起支付中的 appid 参数	appid	是	string(32)	调起支付中的 appid 参数
app 调起支付中的 partnerid 参数	partnerid	是	string(32)	调起支付中的 partnerid 参数
app 调起支付中的 prepayid 参数	prepayid	是	string(32)	调起支付中的 prepayid 参数
app 调起支付中的 packagevalue 参数	package	是	string(128)	调起支付中的 packagevalue 参数
app 调起支付中的 noncestr 参数	noncestr	是	string(32)	调起支付中的 noncestr 参数
app 调起支付中的 timestamp 参数	timestamp	是	string(10)	调起支付中的 timestamp 参数
app 调起支付中的 sign 参数	sign	是	string(32)	调起支付中的 sign 参数

3.2.4 返回结果：

名称	描述	解决方案
0	成功	展示成功页面
-1	错误	可能的原因：签名错误、未注册 APPID、项目设置 APPID 不正确、注册的 APPID 与设置的不匹配、其他异常等。
-2	用户取消	无需处理。发生场景：用户不支付了，点击取消，返回 APP。

3.3. 支付结果通知接口

3.3.1 应用说明：

支付完成后，中国银行把微信相关支付和用户信息发送到该 URL，商户需要接收处理信息。

对后台通知交互时，如果中国银行收到商户的应答不是成功或超时，中国银行认为通知失败，中国银行会通过一定的策略定期重新发起通知，尽可能提高通知的成功率，但中国银行不保证通知最终能成功。

由于存在重新发送后台通知的情况，因此同样的通知可能会多次发送给商户系统。**商户系统必须能够正确处理重复的通知。**

推荐的做法是：当收到通知进行处理时，首先检查对应业务数据的状态，判断该通知是否已经处理过，如果没有处理过再进行处理，如果处理过直接返回结果成功。在对业务数据进行状态检查和处理之前，要采用数据锁进行并发控制，以避免函数重入造成的数据混乱。

特别提醒：商户系统对于支付结果通知的内容一定要做签名验证，防止数据泄漏导致出现“假通知”，造成资金损失。

3.3.2 接口地址：

该链接是通过【统一下单接口】中提交的参数 notify_url 设置，如果链接无法访问，商户将无法接收到通知。

3.3.3 通知参数:

字段名	变量名	必填	类型	说明
版本信息	version	否	String(8)	若不填写, 则默认为 1.0.4
返回状态码	return_code	是	String(16)	SUCCESS/FAIL 此字段是通信标识, 非交易标识, 交易是否成功需要查看 result_code 来判断
返回信息	return_msg	否	String(128)	返回信息, 如非空, 为错误原因 签名失败 参数格式校验错误
以下字段在 return_code 为 SUCCESS 的时候有返回				
应用 ID	appid	是	String(32)	中国银行分配的 APPID
商户号	mch_id	是	String(32)	中国银行分配的商户号
微信 APPID	wx_appid	否	String(32)	商户在微信开放平台上申请的 appid
设备号	device_info	否	String(32)	终端设备号
随机字符串	nonce_str	是	String(32)	随机字符串, 不长于 32 位
签名	sign	是	String(32)	签名, 详细签名方法见 2.2 节
业务结果	result_code	是	String(16)	SUCCESS/FAIL
错误代码	err_code	否	String(32)	详见错误码列表
错误代码描述	err_code_des	否	String(128)	错误信息描述
以下字段在 return_code 和 result_code 都为 SUCCESS 的时候有返回				
用户标识	openid	是	String(128)	用户在商户 appid 下的唯一标识
是否关注公众账号	is_subscribe	是	String(1)	用户是否关注公众账号, Y-关注, N-未关注, 仅在公众账号类型支付有效
交易类型	trade_type	是	String(16)	APP
付款银行	bank_type	是	String(16)	银行类型, 采用字符串类型的银行标识
总金额	total_fee	是	Int	订单总金额, 单位为分

货币种类	fee_type	否	String(8)	货币类型，符合 ISO4217 标准的三位字母代码，默认人民币：CNY
代金券或立减优惠金额	coupon_fee	否	Int	代金券或立减优惠金额<=订单总金额，订单总金额-代金券或立减优惠金额为现金支付金额
微信支付订单号	transaction_id	是	String(32)	微信支付订单号
通道订单号	pass_trade_no	否	String(32)	通道的统一订单号，即微信支付通知中的“商户单号”。
商户订单号	out_trade_no	是	String(32)	商户系统的订单号，与请求一致。
商家数据包	attach	否	String(128)	商家数据包，原样返回
支付完成时间	time_end	是	String(14)	支付完成时间，格式为 yyyyMMddhhmmss，如 2009 年 12 月 27 日 9 点 10 分 10 秒 表示为 20091227091010。时区为 GMT+8 beijing。该时间取自微信支付服务器

3.3.4 返回结果：

字段名	变量名	必填	类型	说明
版本信息	version	否	String(8)	若不填写，则默认为 1.0.4
返回状态码	return_code	是	String(16)	SUCCESS/FAIL SUCCESS表示商户接收通知成功并校验成功
返回信息	return_msg	否	String(128)	返回信息，如非空，为错误原因： 签名失败，参数格式校验错误

3.4. 查询订单接口

3.4.1 应用说明：

该接口提供所有微信支付订单的查询，商户可以通过该接口主动查询订单状

态，完成下一步的业务逻辑。

需要调用查询接口的情况：

- ◆ 当商户后台、网络、服务器等出现异常，商户系统最终未接收到支付通知；
- ◆ 调用支付接口后，返回系统错误或未知交易状态情况；
- ◆ 调用刷卡支付 API，返回 USERPAYING 的状态；
- ◆ 调用关单或撤销接口 API 之前，需确认支付状态。

3.4.2 接口地址：

<https://api.boc.dcorepay.com/pay/orderquery>

3.4.3 输入参数：

字段名	变量名	必填	类型	说明
版本信息	version	否	String(8)	若不填写，则默认为 1.0.4
应用 ID	appid	是	String(32)	中国银行分配的 APPID
商户号	mch_id	是	String(32)	中国银行分配的商户号
微信 APPID	wx_appid	否	String(32)	商户在微信开放平台上申请的appid。trade_type 等于 APP 时需要填写。
微信订单号	transaction_id	否	String(32)	微信订单号，优先于通道订单号和商户订单号。
通道订单号	pass_trade_no	否	String(32)	通道的统一订单号，即微信支付通知中的“商户单号”，优先于商户订单号。
商户订单号	out_trade_no	否	String(32)	商户订单号。商户订单号、通道订单号和微信订单号必须传入其中一个订单号，如果三个订单号同时传入，优先顺序为：微信订单号、通道订单号、商户订单号
随机字符串	nonce_str	是	String(32)	随机字符串，不长于 32 位
签名	sign	是	String(32)	签名

举例：

```
<xml>
  <appid>a20150609000000138</appid>
  <mch_id>m20150609000000138</mch_id>
  <nonce_str>6e9feef0d48ead6730ce88ecc22b34cc</nonce_str>
  <out_trade_no>201501010000000001</out_trade_no>
  <transaction_id>1008450740201411110005820873</transaction_id>
  <sign><![CDATA[50A8800F0B58411968045BF620852860]]></sign>
</xml>
```

3.4.4 返回结果:

字段名	变量名	必填	类型	说明
版本信息	version	否	String(8)	若不填写, 则默认为 1.0.4
返回状态码	return_code	是	String(16)	SUCCESS/FAIL 此字段是通信标识, 非交易标识, 交易是否成功需要查看 result_code 来判断
返回信息	return_msg	否	String(128)	返回信息, 如非空, 为错误原因 签名失败 参数格式校验错误
以下字段在 return_code 为 SUCCESS 的时候有返回				
应用 ID	appid	是	String(32)	中国银行分配的应用 ID
商户号	mch_id	是	String(32)	中国银行分配的商户 ID
微信 APPID	wx_appid	否	String(32)	商户在微信开放平台上申请的appid。
随机字符串	nonce_str	是	String(32)	随机字符串, 不长于 32 位
签名	sign	是	String(32)	签名, 签名方式详见 1.6 节
业务结果	result_code	是	String(16)	SUCCESS/FAIL
错误代码	err_code	否	String(32)	详见错误列表
错误代码描述	err_code_des	否	String(128)	错误信息描述
以下字段在 return_code 和 result_code 都为 SUCCESS 的时候有返回				
交易状态	trade_state	是	String(32)	SUCCESS--支付成功 REFUND--转入退款 NOTPAY--未支付 CLOSED--已关闭 REVOKED--已撤销 (刷卡支付)

				USERPAYING--用户支付中 NOPAY--未支付(确认支付超时) PAYERROR--支付失败(其他原因, 如银 行返回失败)
交易状态描述	trade_state_desc	否	String(256)	对当前查询订单状态的描述和下一步操作的指引
以下字段在 trade_state 为 SUCCESS 或者 REFUND 的时候有返回				
设备号	device_info	否	String(32)	终端设备号
用户标识	openid	是	String(128)	用户在商户 APPID 下的唯一标识
是否关注公众号	is_subscribe	是	String(1)	用户是否关注公众账号,Y-关注,N-未关注 注: 仅在公众账号类型支付有效
交易类型	trade_type	是	String(16)	JSAPI、NATIVE、MICROPAY、APP
付款银行	bank_type	是	String(16)	银行类型, 采用字符串类型的银行标识
货币种类	fee_type	否	String(8)	货币类型, 符合 ISO 4217 标准的三位字母代码, 默认人民币: CNY
总金额	total_fee	是	Int	订单总金额, 单位为分
代金券或立减优惠金额	coupon_fee	否	Int	代金券或立减优惠金额<=订单总金额, 订单总金额-代金券或立减优惠金额=现金支付金额
微信订单号	transaction_id	是	String(32)	微信订单号
通道订单号	pass_trade_no	否	String(32)	通道的统一订单号, 即微信支付通知中的“商户单号”。
商户订单号	out_trade_no	是	String(32)	商户系统的订单号, 与请求一致
商家数据包	attach	否	String(128)	商家数据包, 原样返回
支付完成时间	time_end	是	String(14)	支付完成时间, 格式为 yyyyMMddhh mmss, 如 2009 年 12 月 27 日 9 点 10 分 10 秒表示为 20091227091010。 时区为 GMT+8 beijing。该时间取自微信支付服务器

举例:

```
<xml>
  <return_code><![CDATA[SUCCESS]]></return_code>
  <return_msg><![CDATA[OK]]></return_msg>
  <appid><![CDATA[a20150609000000138]]></appid>
  <mch_id><![CDATA[m20150609000000138]]></mch_id>
  <device_info><![CDATA[1000]]></device_info>
  <nonce_str><![CDATA[TN55wO9Pba5yENl8]]></nonce_str>
  <sign><![CDATA[BDF0099C15FF7BC6B1585FBB110AB635]]></sign>
  <result_code><![CDATA[SUCCESS]]></result_code>
  <openid><![CDATA[oUpF8uN95-Ptaags6E_roPHg7AG0]]></openid>
  <is_subscribe><![CDATA[Y]]></is_subscribe>
  <trade_type><![CDATA[MICROPAY]]></trade_type>
  <bank_type><![CDATA[CCB_DEBIT]]></bank_type>
  <total_fee>1</total_fee>
  <fee_type><![CDATA[CNY]]></fee_type>
  <transaction_id><![CDATA[1008450740201411110005820873]]></transaction_id>
  <out_trade_no><![CDATA[1415757673]]></out_trade_no>
  <attach>'store_appid=s20150609000000138#store_name=测试门店#op_user=000001</attach>
  <time_end><![CDATA[20141111170043]]></time_end>
  <trade_state><![CDATA[SUCCESS]]></trade_state>
</xml>
```

3.4.5 错误码：

名称	描述	原因	解决方案
ORDERNO TEXIST	此交易订单号不存在	查询系统中不存在此交易订单号	该 API 只能查提交支付交易返回成功的订单，请商户检查需要查询的订单号是否正确
SYSTEME RROR	系统错误	后台系统返回错误	系统异常，请再调用发起查询

3.5. 关闭订单接口

3.5.1 应用说明：

以下情况需要调用关单接口：商户订单支付失败需要生成新单号重新发起支付，要对原订单号调用关单，避免重复支付；系统下单后，用户支付超时，系统退出不再受理，避免用户继续，请调用关单接口。

3.5.2 接口地址:

<https://api.boc.dcorepay.com/pay/closeorder>

3.5.3 输入参数:

字段名	变量名	必填	类型	说明
版本信息	version	否	String(8)	若不填写, 则默认为 1.0.4
应用 ID	appid	是	String(32)	中国银行分配的 APPID
商户号	mch_id	是	String(32)	中国银行分配的商户号
微信 APPID	wx_appid	否	String(32)	商户在微信开放平台上申请的 <code>appid</code> 。 <code>trade_type</code> 等于 APP 时需要填写。
商户订单号	out_trade_no	是	String(32)	商户系统内部的订单号
随机字符串	nonce_str	是	String(32)	随机字符串, 不长于 32 位。建议采用随机数算法。
签名	sign	是	String(32)	签名, 详细签名方法见 2.2 小节

3.5.4 返回结果:

字段名	变量名	必填	类型	说明
版本信息	version	否	String(8)	若不填写, 则默认为 1.0.4
返回状态码	return_code	是	String(16)	SUCCESS/FAIL
返回信息	return_msg	否	String(128)	返回信息, 如非空, 为错误原因 签名失败 参数格式校验错误
以下字段在 <code>return_code</code> 为 SUCCESS 的时候有返回				
应用 ID	appid	是	String(32)	中国银行分配的 APPID
商户号	mch_id	是	String(32)	中国银行分配的商户号
微信 APPID	wx_appid	否	String(32)	商户在微信开放平台上申请的 <code>appid</code>

随机字符串	nonce_str	是	String(32)	随机字符串，不长于 32 位
签名	sign	是	String(32)	签名, 详细签名方法见 3.2 节
业务结果	result_code	是	String(16)	SUCCESS/FAIL SUCCESS 表示关单成功, 此笔订单不能再发起支付; FAIL 其它表示关单接口异常, 可再次发起关单操作
错误代码	err_code	否	String(32)	详见错误码列表
错误代码描述	err_code_des	否	String(128)	错误信息描述

3.5.5 错误码:

名称	描述	原因	解决方案
ORDERPAID	订单已支付	订单已支付, 不能发起关单	订单已支付, 不能发起关单, 请当作已支付的正常交易
SYSTEMERROR	系统错误	系统错误	系统异常, 请重新调用该 API
ORDERNOTEXIST	订单不存在	订单系统不存在此订单	不需要关单, 当作未提交的支付的订单
ORDERCLOSED	订单已关闭	订单已关闭, 无法重复关闭	订单已关闭, 无需继续调用
SIGNERROR	签名错误	参数签名结果不正确	请检查签名参数和方法是否都符合签名算法要求
REQUIRE_POST_METHOD	请使用 post 方法	未使用 post 传递参数	请检查请求参数是否通过 post 方法提交
XML_FORMAT_ERROR	XML 格式错误	XML 格式错误	请检查 XML 参数格式是否正确

3.6. 申请退款接口

3.6.1 应用说明:

当交易发生之后一段时间内, 由于买家或者卖家的原因需要退款时, 卖家可以通过退款接口将支付款退还给买家, 微信支付将在收到退款请求并且验证成功之后, 按照退款规则将支付款按原路退到买家帐号上。

注意:

1. 交易时间超过一个月的订单无法提交退款;
2. 退款金额必须等于支付订单金额, 即仅支持全额退款, 不支持单笔交易分多次退款。退款失败后重新提交, 要采用原来的退款单号。

3.6.2 接口地址:

<https://api.boc.dcorepay.com/pay/refund>

3.6.3 输入参数:

字段名	变量名	必填	类型	说明
版本信息	version	否	String(8)	若不填写, 则默认为 1.0.4
应用 ID	appid	是	String(32)	中国银行分配的 APPID
商户号	mch_id	是	String(32)	中国银行分配的商户号
微信 APPID	wx_appid	否	String(32)	商户在微信开放平台上申请的 appid。trade_type 等于 APP 时 需要填写。
设备号	device_info	否	String(32)	设备号, 与下单一致
微信订单号	transaction_id	否	String(32)	微信订单号, 优先于通道订单 号和商户订单号。
通道订单号	pass_trade_no	否	String(32)	通道的统一订单号, 即微信 支付通知中的“商户单 号”, 优先于商户订单号。
商户订单号	out_trade_no	否	String(32)	商户订单号。商户订单号、通 道订单号和微信订单号必须传 入其中一个订单号, 如果三个 订单号同时传入, 优先顺序为: 微信订单号、通道订单号、商 户订单号
随机字符串	nonce_str	是	String(32)	随机字符串, 不长于 32 位
签名	sign	是	String(32)	签名
商户退款单号	out_refund_no	是	String(32)	商户系统内部退款单号, 商户 系统内部唯一, 同一笔退款单 号多次请求只退一笔
总金额	total_fee	是	Int	订单总金额, 单位为分
退款金额	refund_fee	是	Int	退款总金额, 单位为分, 必须 等于总金额
货币类型	refund_fee_type	否	String(8)	货币类型。默认为人民币: CNY
操作员	op_user_id	是	String(32)	操作员账号, 默认为商户号

3.6.4 返回结果:

字段名	变量名	必填	类型	说明
版本信息	version	否	String(8)	若不填写, 则默认为 1.0.4
返回状态码	return_code	是	String(16)	SUCCESS/FAIL
返回信息	return_msg	否	String(128)	返回信息, 如非空, 为错误原因 签名失败 参数格式校验错误
以下字段在 return_code 为 SUCCESS 的时候有返回				
业务结果	result_code	是	String(16)	SUCCESS/FAIL SUCCESS 退款申请接收成功, 结果通过退款查询接口查询 FAIL 提交业务失败
错误代码	err_code	是	String(32)	详见错误列表
错误代码描述	err_code_des	否	String(128)	错误信息描述
应用 ID	appid	是	String(32)	中国银行分配的应用 ID
商户号	mch_id	是	String(32)	中国银行分配的商户 ID
微信 APPID	wx_appid	否	String(32)	商户在微信开放平台上申请的 appid
设备号	device_info	否	String(32)	设备 ID, 与下单一致
随机字符串	nonce_str	是	String(32)	随机字符串, 不长于 32 位
签名	sign	是	String(32)	签名
微信订单号	transaction_id	是	String(32)	微信订单号
通道订单号	pass_trade_no	否	String(32)	通道的统一订单号, 即微信支付通知中的“商户单号”。
商户订单号	out_trade_no	是	String(32)	商户系统的订单号, 与请求一致
商户退款单号	out_refund_no	是	String(32)	商户退款单号
通道退款单号	pass_refund_no	否	String(32)	通道的统一退款单号。

微信退款单号	refund_id	是	String(32)	微信退款单号
退款渠道	refund_channel	否	String(16)	ORIGINAL—原路退款 BALANCE—退回到余额
退款金额	refund_fee	是	Int	退款总金额，单位为分，等于 订单总金额
代金券或立减优惠退款金额	coupon_refund_fee	否	Int	代金券或立减优惠退款金额=订单金额-现金退款金额，注意：部分代金券或立减优惠退款金额不会退回，如满立减

3.6.5 错误码：

名称	描述	原因	解决方案
SYSTEMERROR	接口返回错误	系统超时	请用相同参数再次调用 API
INVALID_TRANSACTIONID	无效 transaction_id	请求参数未按指引进行填写	请求参数错误，检查原交易号是否存在或发起支付交易接口返回失败
PARAM_ERROR	参数错误	请求参数未按指引进行填写	请求参数错误，请重新检查再调用退款申请
APPID_NOT_EXIST	APPID 不存在	参数中缺少 APPID	请检查 APPID 是否正确
MCHID_NOT_EXIST	MCHID 不存在	参数中缺少 MCHID	请检查 MCHID 是否正确
APPID_MCHID_NOT_MATCH	appid 和 mch_id 不匹配	appid 和 mch_id 不匹配	请确认 appid 和 mch_id 是否匹配
REQUIRE_POST_METHOD	请使用 post 方法	未使用 post 传递参数	请检查请求参数是否通过 post 方法提交
SIGNERROR	签名错误	参数签名结果不正确	请检查签名参数和方法是否都符合签名算法要求
XML_FORMAT_ERROR	XML 格式错误	XML 格式错误	请检查 XML 参数格式是否正确

3.7. 查询退款接口

3.7.1 应用说明：

提交退款申请后，通过调用该接口查询退款状态。退款有一定延时，用零钱支付的退款 20 分钟内到账，银行卡支付的退款 3 个工作日后重新查询退款状态。

3.7.2 接口地址：

<https://api.boc.dcorepay.com/pay/refundquery>

3.7.3 输入参数:

字段名	变量名	必填	类型	说明
版本信息	version	否	String(8)	若不填写, 则默认为 1.0.4
应用 ID	appid	是	String(32)	中国银行分配的 APPID
商户号	mch_id	是	String(32)	中国银行分配的商户号
微信 APPID	wx_appid	否	String(32)	商户在微信开放平台上申请的 appid。trade_type 等于 APP 时 需要填写。
设备号	device_info	否	String(32)	设备号, 与下单一致
微信订单号	transaction_id	否	String(32)	微信订单号。
通道订单号	pass_trade_no	否	String(32)	通道的统一订单号, 即微信 支付通知中的“商户单 号”。
商户订单号	out_trade_no	否	String(32)	商户订单号
随机字符串	nonce_str	是	String(32)	随机字符串, 不长于 32 位
签名	sign	是	String(32)	签名
商户退款单号	out_refund_no	否	String(32)	商户系统内部退款单号, 商户 系统内部唯一, 同一笔退款单 号多次请求只退一笔
通道退款单号	pass_refund_no	否	String(32)	通道统一退款单号
微信退款单号	refund_id	否	String(32)	微信退款单号 refund_id、 pass_refund_no、out_refund_ no、out_trade_no、 pass_trade_no、transaction_id 六个参数必填一个, 如果同时 存在优先级为: refund_id>pass_refund_no>out _refund_no>transaction_id>pass trade_no>out trade_no

3.7.4 返回结果:

字段名	变量名	必填	类型	说明
版本信息	version	否	String(8)	若不填写, 则默认为 1.0.4

返回状态码	return_code	是	String(16)	SUCCESS/FAIL
返回信息	return_msg	否	String(128)	返回信息,如非空,为错误原因: 签名失败 参数格式校验错误
以下字段在 return_code 为 SUCCESS 的时候有返回				
业务结果	result_code	是	String(16)	SUCCESS/FAIL SUCCESS 退款申请接收成功, 结果通过退款查询接口查询 FAIL
错误代码	err_code	是	String(32)	详见错误码
错误代码描述	err_code_des	否	String(128)	错误信息描述
应用 ID	appid	是	String(32)	中国银行分配的应用 ID
商户 ID	mch_id	是	String(32)	中国银行分配的商户 ID
微信 APPID	wx_appid	否	String(32)	商户在微信开放平台上申请的 appid
设备号	device_info	否	String(32)	商户定义的设备号
随机字符串	nonce_str	是	String(32)	随机字符串, 不长于 32 位
签名	sign	是	String(32)	签名
微信订单号	transaction_id	是	String(32)	微信订单号
通道订单号	pass_trade_no	否	String(32)	通道的统一订单号, 即微信 支付通知中的“商户单 号”。
商户订单号	out_trade_no	是	String(32)	商户订单号
退款笔数	refund_count	是	Int	退款记录数 (v1.0.3 版本起, 若有退款, 则退款记录数均为 1)
商户退款单号	out_refund_no_\$n	是	String(32)	商户退款单号
通道退款单号	pass_refund_no_\$n	否	String(32)	通道统一退款单号
微信退款单号	refund_id_\$n	是	String(28)	微信退款单号
退款渠道	refund_channel_\$n	否	String(16)	ORIGINAL-原路退款 BALANCE-退回余额
退款金额	refund_fee_\$n	是	Int	退款总金额, 单位为分
货币类型	fee_type_\$n	否	String(8)	货币类型。默认为人民币:CNY

代金券或立减优惠退款金额	coupon_refund_fee_\$n	否	Int	代金券或立减优惠退款金额<=退款金额, 退款金额-代金券或立减优惠退款金额为现金
退款状态	refund_status_\$n	是	String(16)	退款状态: SUCCESS--退款成功 FAIL--退款失败 PROCESSING--退款处理中 NOTSURE--未确定, 需要商户原退款单号重新发起 CHANGE--转入代发, 退款到银行发现用户的卡作废或者冻结了, 导致原路退款银行卡失败, 资金回流到商户的现金帐号, 需要商户人工干预, 通过线下或者财付通转账的方式进行退款。

\$n 表示记录的序号, 取值为 0~(\$ refund_count -1), 例如 refund_count 指示返回的退款记录有 2 条。第一条序号为“0”, 第二条序号为“1”。

3.7.5 错误码:

名称	描述	原因	解决方案
SYSTEMERROR	接口返回错误	系统超时	请尝试再次掉调用 API。
INVALID_TRANSACTIONID	无效 transaction_id	请求参数未按指引进行填写	请求参数错误, 检查原交易号是否存在或发起支付交易接口返回失败
PARAM_ERROR	参数错误	请求参数未按指引进行填写	请求参数错误, 请检查参数再调用退款申请
APPID_NOT_EXIST	APPID 不存在	参数中缺少 APPID	请检查 APPID 是否正确
MCHID_NOT_EXIST	MCHID 不存在	参数中缺少 MCHID	请检查 MCHID 是否正确
APPID_MCHID_NOT_MATCH	appid 和 mch_id 不匹配	appid 和 mch_id 不匹配	请确认 appid 和 mch_id 是否匹配
REQUIRE_POST_METHOD	请使用 post 方法	未使用 post 传递参数	请检查请求参数是否通过 post 方法提交
SIGNERROR	签名错误	参数签名结果不正确	请检查签名参数和方法是否都符合签名算法要求
XML_FORMAT_ERROR	XML 格式错误	XML 格式错误	请检查 XML 参数格式是否正确

3.8. 对账单接口

3.8.1 应用说明:

商户可以通过该接口下载历史交易清单。比如掉单、系统错误等导致商户侧和微信侧数据不一致, 通过对账单核对后可校正支付状态。

注意:

- 1、微信侧未成功下单的交易不会出现在对账单中。支付成功后撤销的交易会出现在对账单中，跟原支付单订单号一致，bill_type 为 REVOKED；
- 2、微信在次日 9 点启动生成前一天的对账单，建议商户 10 点后再获取；
- 3、对账单中涉及金额的字段单位为“元”。

3.8.2 接口地址：

<https://api.boc.dcorepay.com/pay/downloadbill>

3.8.3 输入参数：

字段名	变量名	必填	类型	说明
版本信息	version	否	String(8)	若不填写，则默认为 1.0.4
应用 ID	appid	是	String(32)	中国银行分配的 APPID
商户号	mch_id	是	String(32)	中国银行分配的商户号
微信 APPID	wx_appid	否	String(32)	商户在微信开放平台上申请的appid。trade_type 等于 APP 时需要填写。
设备号	device_info	否	String(32)	终端设备号，填写此字段，只下载该设备号的对账单
随机字符串	nonce_str	是	String(32)	随机字符串，不长于 32 位
签名	sign	是	String(32)	签名
对账单日起	bill_date	是	String(8)	下载对账单的日期，格式：20140603
账单类型	bill_type	否	String(8)	ALL，返回当日所有订单信息，默认值；SUCCESS，返回当日成功支付的订单；REFUND，返回当日退款订单；REVOKED，已撤销的订单。

3.8.4 返回结果：

字段名	变量名	必填	类型	说明
版本信息	version	否	String(8)	若不填写，则默认为 1.0.4

返回状态码	return_code	是	String(32)	FAIL
返回信息	return_msg	否	String(32)	返回信息，如非空，为错误原因： 签名失败 参数格式校验错误 该日期订单未生成

成功时，数据以文本表格的方式返回，第一行为表头，后面各行为对应的字段内容，字段内容跟查询订单或退款结果一致，具体字段说明可查阅相应接口。

第一行为表头，根据请求下载的对账单类型不同而不同(由 **bill_type** 决定)，目前有：

当日所有订单

交易时间,应用 ID,商户 ID,设备号,微信订单号,商户订单号,用户标识, 交易类型,交易状态,付款银行,货币种类,总金额,代金券或立减券优惠金额,微信退款单号,商户退款单号,退款金额,代金券或立减券退款金额, 退款类型, 退款状态,商品名称,商户数据包,手续费,费率

当日成功支付的订单

交易时间,应用 ID,商户 ID,设备号,微信订单号,商户订单号,用户标识, 交易类型,交易状态,付款银行,货币种类,总金额, 代金券或立减券优惠金额,商品名称,商户数据包,手续费,费率

当日退款的订单

交易时间,应用 ID,商户 ID,设备号,微信订单号,商户订单号,用户标识, 交易类型,交易状态,付款银行,货币种类,总金额, 代金券或立减券优惠金额,退款申请时间,退款成功时间,微信退款单号,商户退款单号,退款金额, 代金券或立减券优惠退款金额,退款类型,退款 状态,商品名称,商户数据包,手续费,费率

从第二行起，为数据记录，各参数以逗号分隔，参数前增加`符号，为标准键盘 1 左边键的字符，字段顺序与表头一致。其中，应用 ID 和商户 ID 指中国银行分配给商户的应用号和商户号。

倒数第二行为订单统计标题，最后一行为统计数据：

总交易单数，总交易额，总退款金额，总代金券或立减券优惠退款金额，手续费总金额

3.8.5 错误码：

名称	描述	原因	解决方案
SYSTEMERROR	接口返回错误	系统超时	请尝试再次查询。
INVALID_TRANSACTIONID	无效 transaction_id	请求参数未按指引进行填写	参数错误，请重新检查
PARAM_ERROR	参数错误	请求参数未按指引进行填写	参数错误，请重新检查