

# API 接口下单 POST 参数生成与签名教程

## 一、准备工作

### 1. 了解 API 文档

首先，你需要仔细阅读 API 文档，了解以下信息：

- 请求 URL
- 请求方式 (POST)
- 必要参数和可选参数
- 签名算法 (MD5)
- 字符编码 (UTF-8)
- 测试账号信息

### 2. 获取测试账号

根据文档，测试账号信息如下：

- PID: 1003
- KEY: fnv5Xf0BnV5n5bGzFf7V7Fvn9tVtzn9v

## 二、生成 POST 参数

### 1. 确定必要参数

根据 API 文档，统一下单支付接口需要以下必要参数：

- `pid`：商户 ID
- `type`：支付方式 (alipay:支付宝, wxpay:微信支付)
- `out\_trade\_no`：商户订单号 (唯一)
- `notify\_url`：异步通知地址

- `return\_url`: 跳转通知地址
- `name`: 商品名称
- `money`: 商品金额 (单位: 元, 最大 2 位小数)
- `clientip`: 用户 IP 地址
- `device`: 设备类型
- `sign`: 签名字符串
- `sign\_type`: 签名类型 (MD5)

## 2. 填写参数值

根据实际情况填写参数值, 例如:

- `pid`: 1003 (测试账号)
- `type`: alipay
- `out\_trade\_no`: 20240303123456 (使用当前日期时间生成)
- `notify\_url`: http://www.example.com/notify\_url.php
- `return\_url`: http://www.example.com/return\_url.php
- `name`: 测试商品
- `money`: 0.01 (小额测试)
- `clientip`: 192.168.1.100
- `device`: jump (返回自适应支付页面)
- `sign\_type`: MD5

## 三、计算签名

### 1. 排序参数

将所有参数 (除了 `sign` 和 `sign\_type`) 按照参数名 ASCII 码从小到大排序 (a-z)。

排序后的参数:

```
plain text
clientip=192.168.1.100&device=jump&money=0.01&name=测试商品
&notify_url=http://www.example.com/notify_url.php&out_trade_no=202
40303123456&pid=1003&return_url=http://www.example.com/return_url.
```

```
php&type=alipay
```

## 2. 拼接商户密钥

将排序后的参数字符串与商户密钥拼接，注意中间没有连接符：

```
plain text
clientip=192.168.1.100&device=jump&money=0.01&name=测试商品
&notify_url=http://www.example.com/notify_url.php&out_trade_no=202
40303123456&pid=1003&return_url=http://www.example.com/return_url.
php&type=alipayf5Xf0BnV5n5bGzFf7V7Fvn9tVtzn9v
```

## 3. 计算 MD5 哈希值

使用 MD5 算法对拼接后的字符串进行加密，结果转为小写。

### 计算方法：

使用 **PowerShell** 计算：

```
powershell
$params = 'clientip=192.168.1.100&device=jump&money=0.01&name=测试
商品
&notify_url=http://www.example.com/notify_url.php&out_trade_no=202
40303123456&pid=1003&return_url=http://www.example.com/return_url.
php&type=alipayf5Xf0BnV5n5bGzFf7V7Fvn9tVtzn9v';
$md5 = New-Object -TypeName
System.Security.Cryptography.MD5CryptoServiceProvider;
$utf8 = New-Object -TypeName System.Text.UTF8Encoding;
$hash =
[System.BitConverter]::ToString($md5.ComputeHash($utf8.GetBytes($p
arams)));
$hash = $hash.ToLower() -replace '-', '';
Write-Host 'MD5 签名值:' $hash
```

使用在线 **MD5** 工具计算：

打开在线 MD5 加密工具（如 <https://md5jiami.51240.com/>），输入拼接后的字符串，点击加密即可。

计算结果：

```
plain text
```

```
8e1c35a7d599801bb723a6f9ee3ef3f9
```

## 四、组装最终请求参数

将所有参数（包括计算得到的`sign`）按照`application/x-www-form-urlencoded`格式组装：

```
plain text
pid=1003&type=alipay&out_trade_no=20240303123456&notify_url=http://
/www.example.com/notify_url.php&return_url=http://www.example.com/
return_url.php&name=测试商品
&money=0.01&clientip=192.168.1.100&device=jump&param=&sign_type=MD
5&sign=8e1c35a7d599801bb723a6f9ee3ef3f9
```

## 五、发送 POST 请求

使用 POST 工具（如 Postman、curl 等）发送请求：

### 1. 使用 Postman：

- 选择 POST 方法
- 输入请求 URL：https://api.payqixiang.cn/mapi.php
- 在 Body 选项卡中选择 x-www-form-urlencoded
- 粘贴或逐个添加参数
- 点击 Send 按钮发送请求

### 2. 使用 curl：

```
bash
curl -X POST "https://api.payqixiang.cn/mapi.php" \
  -d "pid=1003" \
  -d "type=alipay" \
  -d "out_trade_no=20240303123456" \
  -d "notify_url=http://www.example.com/notify_url.php" \
  -d "return_url=http://www.example.com/return_url.php" \
  -d "name=测试商品" \
  -d "money=0.01" \
  -d "clientip=192.168.1.100" \
```

```
-d "device=jump" \  
-d "param=" \  
-d "sign_type=MD5" \  
-d "sign=8e1c35a7d599801bb723a6f9ee3ef3f9"
```

## 六、注意事项

### 1. 参数排序

- 必须按照参数名 ASCII 码从小到大排序 (a-z)
- `sign`和`sign\_type`不参与签名计算
- 空值参数不参与签名计算

### 2. 签名计算

- 拼接商户密钥时，中间没有连接符
- MD5 结果必须转为小写
- 确保参数值没有被 URL 编码

### 3. 订单号

- 商户订单号必须唯一
- 微信支付和支付宝支付需要使用不同的订单号，不可混用

### 4. 测试金额

- 测试金额应为小额，建议单笔测试交易 0.01 元以内
- 测试交易完成后，如需退款应当日通过 API 提交订单退款

### 5. 测试账号

- 测试账号仅供测试使用，请勿用在正式运营的网站上

### 6. 设备类型

- `pc`：返回二维码链接，需要自行转换成二维码
- `mobile`：返回 H5 调起支付/APP 支付链接

- `wechat`：返回微信内支付链接
- `jump`：返回自适应支付页面链接（推荐）

## 7. 通知地址

- 通知地址必须是可访问的公网地址
- 异步通知地址（`notify_url`）用于服务器端接收支付结果
- 跳转通知地址（`return_url`）用于用户支付完成后跳转

## 8. 其他

- 如果对接过程中出现异步参数名被转义成"`ify_url`"这样的情况，请自行百度解决
- APP 对接建议先跳转到手机自带的浏览器，通过浏览器唤起支付
- 收到异步通知后，需返回 `success` 以表示服务器接收到了订单通知
- 注意支付状态的参数返回来的如果是大写就要保持大写，不要转成小写
- 回调回来的参数如果没有 `name` 就不用参加签名，如有就参加

## 七、示例代码

### PHP 示例：

```
php
<?php
// 商户信息
$pid = '1003'; // 商户 ID
$key = 'fnv5Xf0BnV5n5bGzFf7V7Fvn9tVtzn9v'; // 商户密钥

// 订单信息
$type = 'alipay'; // 支付方式
$out_trade_no = date('YmdHis'); // 商户订单号
$notify_url = 'http://www.example.com/notify_url.php'; // 异步通知地址
$return_url = 'http://www.example.com/return_url.php'; // 跳转通知地址
$name = '测试商品'; // 商品名称
$money = '0.01'; // 商品金额
$clientip = '192.168.1.100'; // 用户 IP
```

```
$device = 'jump'; // 设备类型
$param = ''; // 业务扩展参数

// 组装参数
$params = array(
    'pid' => $pid,
    'type' => $type,
    'out_trade_no' => $out_trade_no,
    'notify_url' => $notify_url,
    'return_url' => $return_url,
    'name' => $name,
    'money' => $money,
    'clientip' => $clientip,
    'device' => $device,
    'param' => $param
);

// 排序参数
ksort($params);

// 拼接参数
$sign_str = '';
foreach ($params as $k => $v) {
    if ($v) {
        $sign_str .= $k . '=' . $v . '&';
    }
}
$sign_str = rtrim($sign_str, '&') . $key;

// 计算签名
$sign = md5($sign_str);

// 组装最终参数
$params['sign'] = $sign;
$params['sign_type'] = 'MD5';

// 发送 POST 请求
$ch = curl_init();
curl_setopt($ch, CURLOPT_URL,
    'https://api.payqixiang.cn/mapi.php');
curl_setopt($ch, CURLOPT_POST, 1);
curl_setopt($ch, CURLOPT_POSTFIELDS, http_build_query($params));
curl_setopt($ch, CURLOPT_RETURNTRANSFER, 1);
```

```
$response = curl_exec($ch);
curl_close($ch);

// 处理响应
$result = json_decode($response, true);
if ($result['code'] == 1) {
    // 成功, 获取支付链接
    $payurl = $result['payurl'];
    echo '支付链接: ' . $payurl;
} else {
    // 失败
    echo '错误: ' . $result['msg'];
}
?>
```

## Python 示例:

```
python
import hashlib
import requests

# 商户信息
pid = '1003' # 商户 ID
key = 'fnv5Xf0BnV5n5bGzFf7V7Fvn9tVtzn9v' # 商户密钥

# 订单信息
type_pay = 'alipay' # 支付方式
out_trade_no = '20240303123456' # 商户订单号
notify_url = 'http://www.example.com/notify_url.php' # 异步通知地址
return_url = 'http://www.example.com/return_url.php' # 跳转通知地址

name = '测试商品' # 商品名称
money = '0.01' # 商品金额
clientip = '192.168.1.100' # 用户 IP
device = 'jump' # 设备类型
param = '' # 业务扩展参数

# 组装参数
params = {
    'pid': pid,
```

```
'type': type_pay,
'out_trade_no': out_trade_no,
'notify_url': notify_url,
'return_url': return_url,
'name': name,
'money': money,
'clientip': clientip,
'device': device,
'param': param
}

# 排序参数
sorted_params = sorted(params.items(), key=lambda x: x[0])

# 拼接参数
sign_str = ''
for k, v in sorted_params:
    if v:
        sign_str += f'{k}={v}&'
sign_str = sign_str.rstrip('&') + key

# 计算签名
sign = hashlib.md5(sign_str.encode('utf-8')).hexdigest()

# 组装最终参数
params['sign'] = sign
params['sign_type'] = 'MD5'

# 发送 POST 请求
url = 'https://api.payqixiang.cn/mapi.php'
response = requests.post(url, data=params)

# 处理响应
result = response.json()
if result['code'] == 1:
    # 成功, 获取支付链接
    payurl = result['payurl']
    print(f'支付链接: {payurl}')
else:
    # 失败
    print(f'错误: {result['msg']}')
```

## 八、常见问题

### 1. 签名错误

- 检查参数排序是否正确
- 检查商户密钥是否正确
- 检查参数值是否与签名计算时一致
- 检查是否包含了`sign`或`sign\_type`参数

### 2. 支付失败

- 检查订单号是否唯一
- 检查支付方式是否正确
- 检查商品金额格式是否正确
- 检查通知地址是否可访问

### 3. 通知不回调

- 检查异步通知地址是否可访问
- 检查服务器是否返回了 `success`
- 检查防火墙是否阻止了回调请求

### 4. 退款失败

- 检查退款金额是否小于等于原订单金额
- 检查当天收款是否大于等于退款金额
- 检查是否在商户后台开启了订单退款 API 接口开关

## 九、总结

生成 POST 参数和签名的流程如下：

1. 准备必要参数
2. 按照 ASCII 码排序参数
3. 拼接商户密钥
4. 计算 MD5 签名

5. 组装最终请求参数
6. 发送 POST 请求

遵循以上步骤和注意事项，你应该能够成功调用 API 接口进行支付测试。如果遇到问  
题，请仔细检查每一步的操作是否正确或联系客服咨询