

0.整体接入步骤

第一步:先找运维解析域名

第二步:创建应用并获取 appKey, 配置回调地址, 获取访问密钥;

第三步:用户授权注册, 获取三方商户code;

第四步:通过code获取开放平台反回token;

第五步:访问开放的资源:

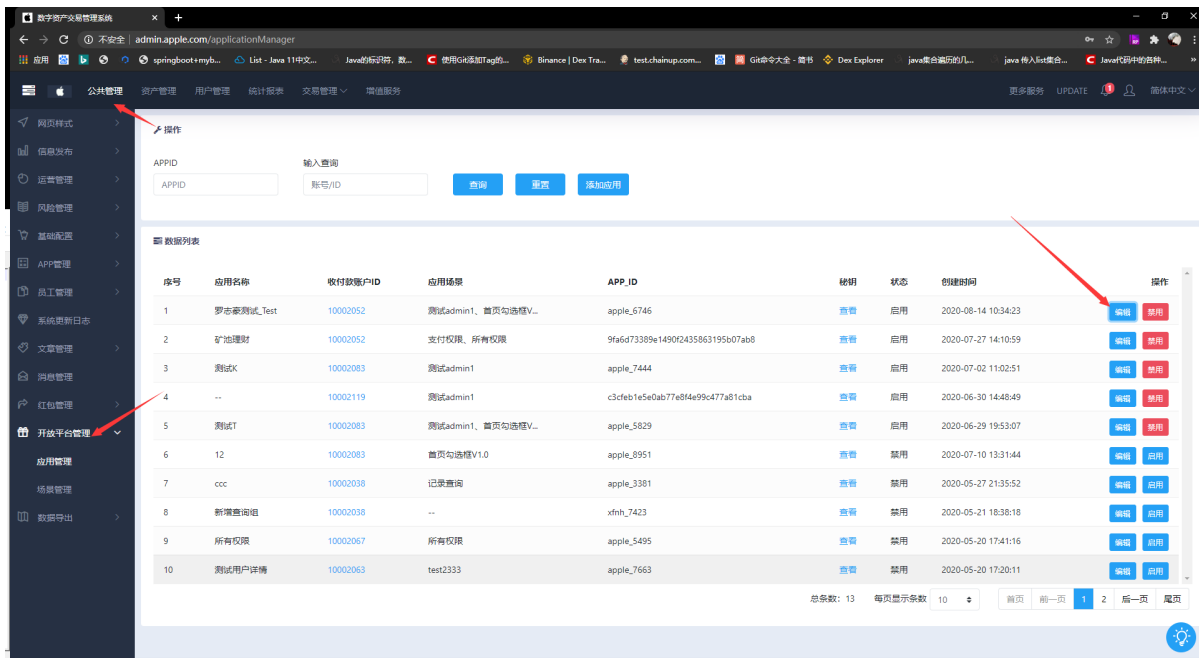
1.准备工作 – 获取 appKey 和设置redirectUrl(回调地址)

开放平台是基于OAuth2.0协议标准构建的OAuth2.0授权登录系统。 在进行开放平台授权登录接入之前, 需要通过管理后台准备以下接入材料:

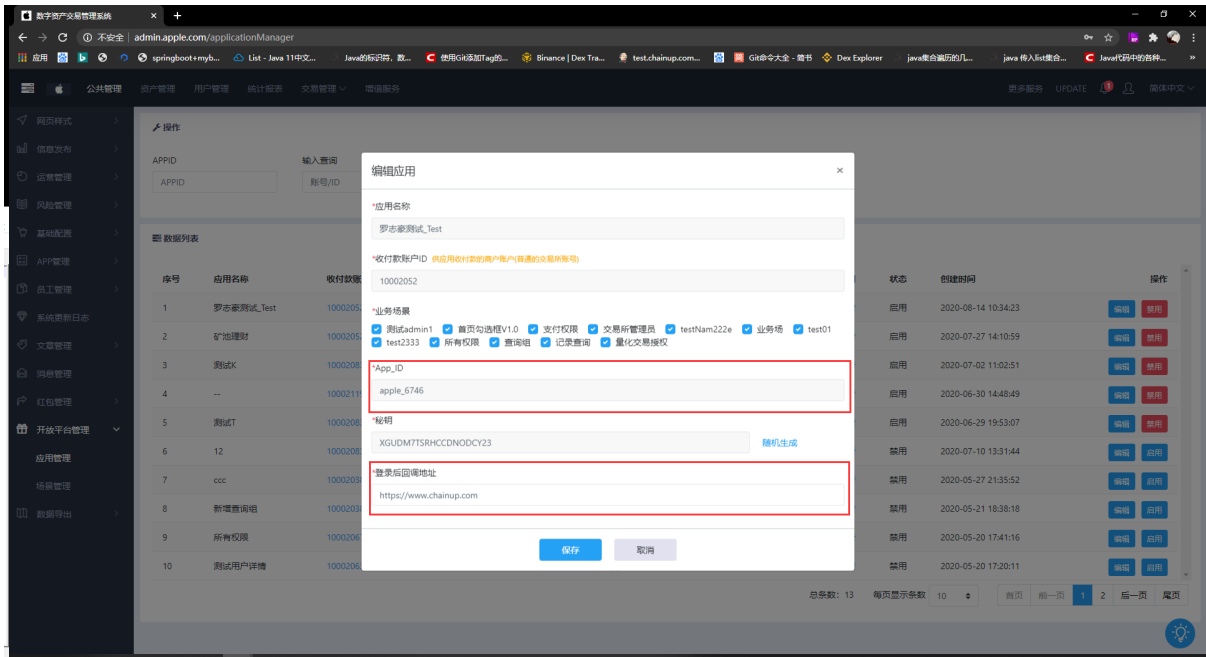
- 1) appKey: 用于和开通开放平台的商户UID进行一对一绑定, 获取开放平台登陆权限的必要条件;
- 2) redirectUrl: 用于完成开放平台开通授权帐号成功后的跳转地址; 该地址根据实际业务需求填写, 如不清楚业务需求, 可默认填写交易所主页;

管理后台地址: <http://admin.apple.com/>

登陆管理后台: 公共管理-- > 开放平台管理-- > 应用管理



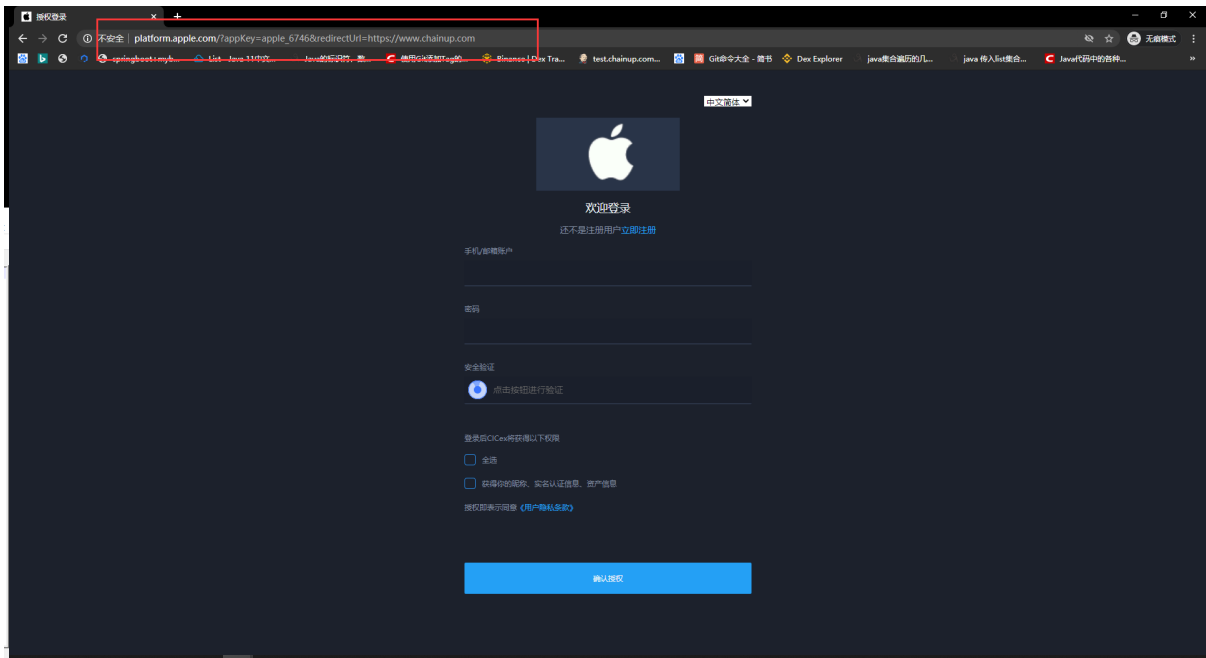
根据红框中内容, 获取appKey和redirectUrl



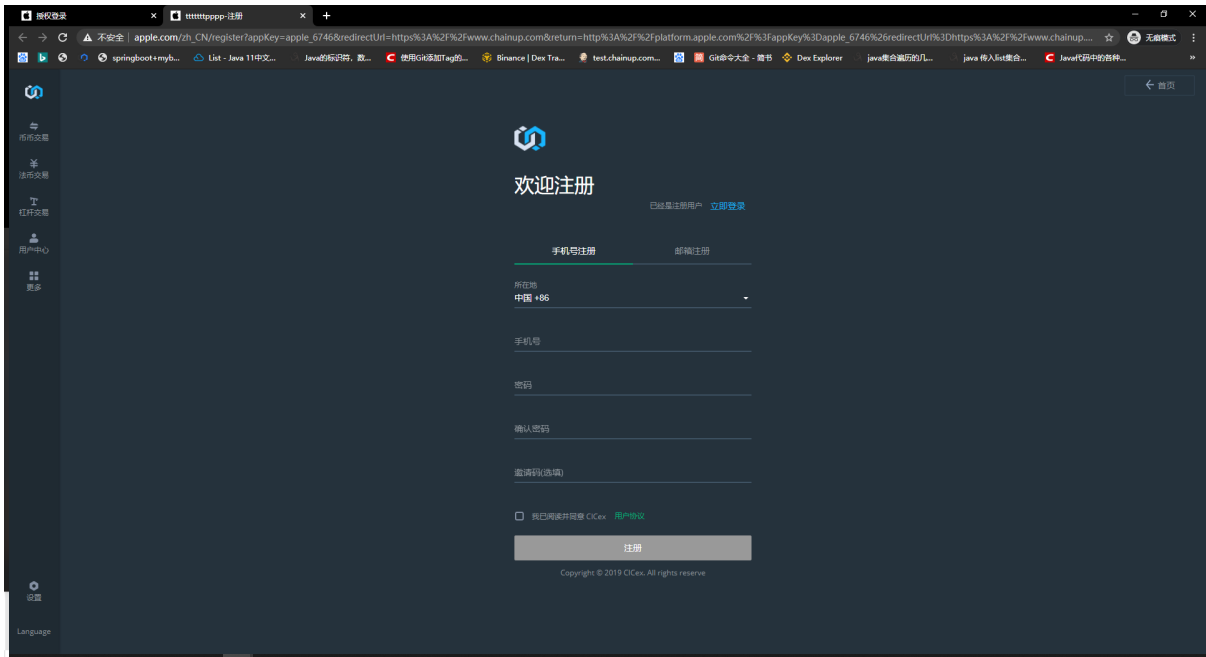
2. 用户注册授权 – OAuth授权登录页

2.1) 带上正确的appkey和redirectUrl点击注册

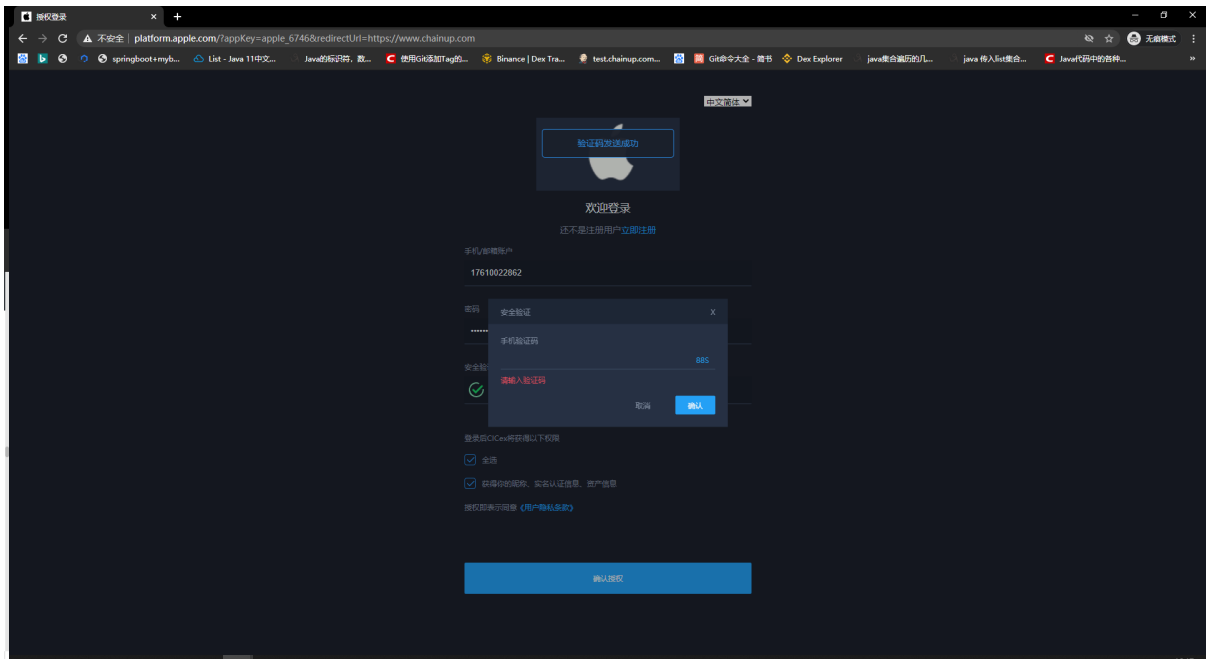
浏览器访问地址: <http://service.xfnh.com/platform/login.html?appKey=xxx&redirectUrl=https:xxxx>



如果没有注册, 先进行注册; 如果已经是注册用户, 则直接点击上一张图片中的"确认授权" 按钮即可;



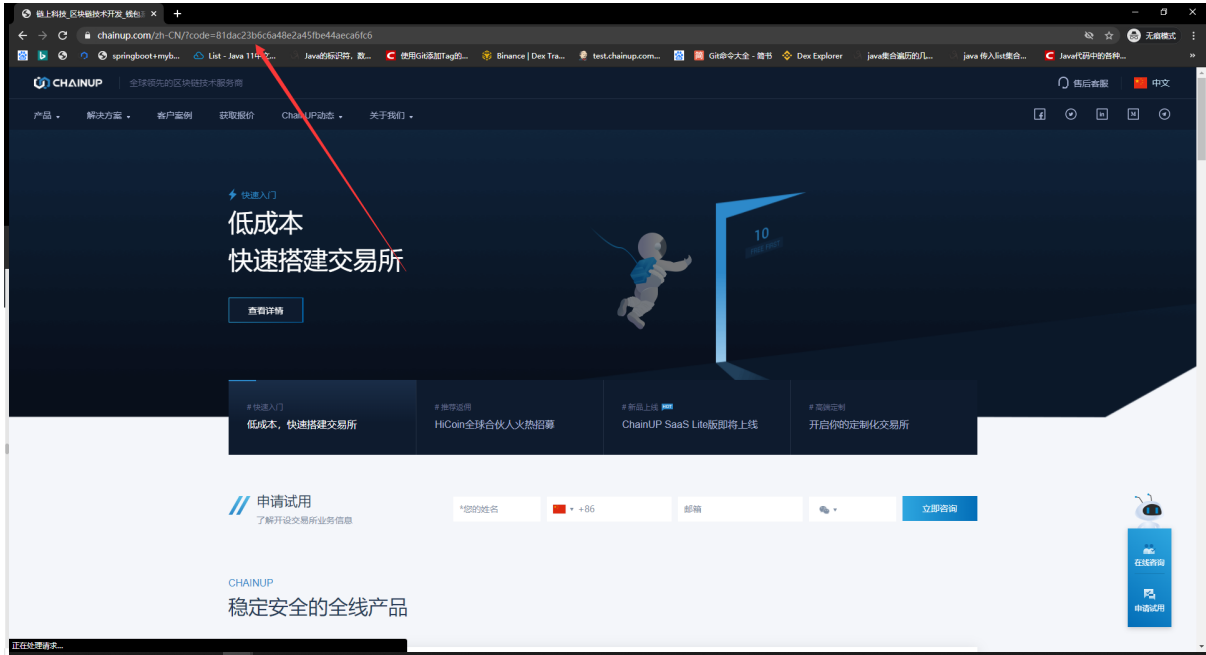
系统会发送验证码进行授权;



2.2) 注册(或登陆)成功获取code

如果上述步骤操作成功;

且在第一步中填写的注册回调地址为交易所首页;
则可以看到下述图片, **箭头处则为当前获取的Code**



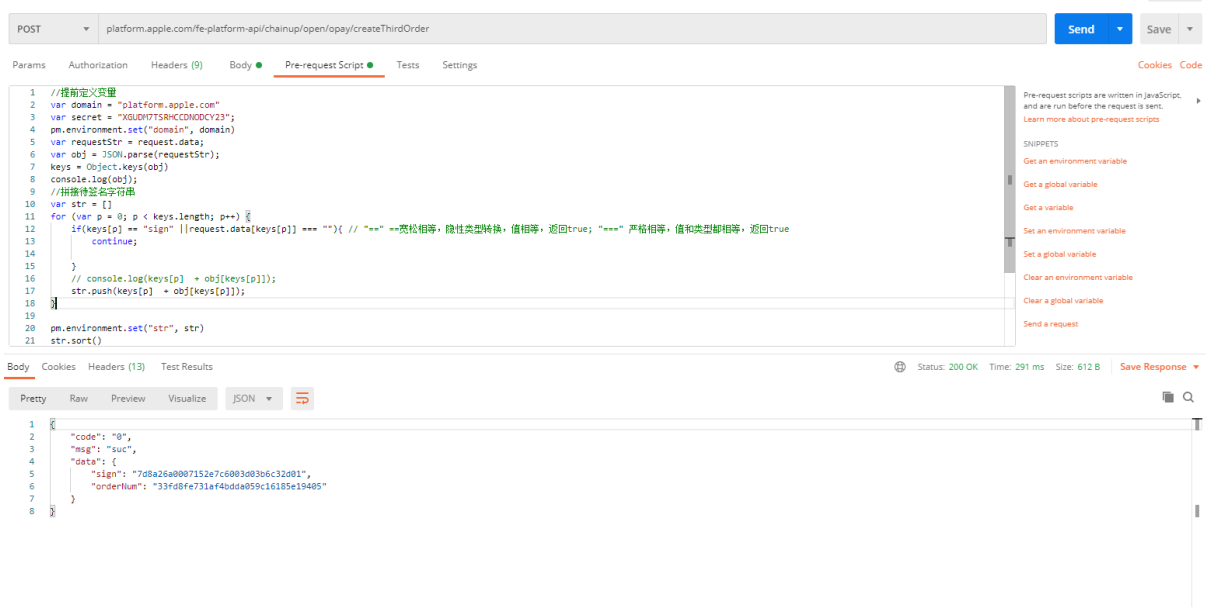
3.获取token – 使用code换取token、openId

3.1) 生成签名脚本

签名脚本配置方式:

在postman工具下图所示位置进行配置脚本内容;

请求参数中带入 **"sign":"{{sign}}"** 即可自动生成签名(参考后续步骤请求参数截图);



脚本内容

```
1 //提前定义变量
2 var domain = "service.xxx.com"
3 var secret = "XGUDM7TSRHCCDN0DCY23";
4 pm.environment.set("domain", domain)
5 var requestStr = request.data;
```

```

6  var obj = JSON.parse(requestStr);
7  keys = Object.keys(obj)
8  console.log(obj);
9  //拼接待签名字符串
10 var str = []
11 for (var p = 0; p < keys.length; p++) {
12     if(keys[p] == "sign" || request.data[keys[p]] === ""){ // "==" ==宽松相等, 隐性类
13         continue;
14     }
15     // console.log(keys[p] + obj[keys[p]]);
16     str.push(keys[p] + obj[keys[p]]);
17 }
18
19
20 pm.environment.set("str", str)
21 str.sort()
22 str.push(secret)
23 var s = str.join("")
24 console.log("准备签名");
25 console.log(s);
26
27 //MD5加密签名规格, 并赋值给环境变量`sign`
28 var sign =CryptoJS.MD5(CryptoJS.enc.Latin1.parse(s)).toString();
29 pm.environment.set("sign", sign);

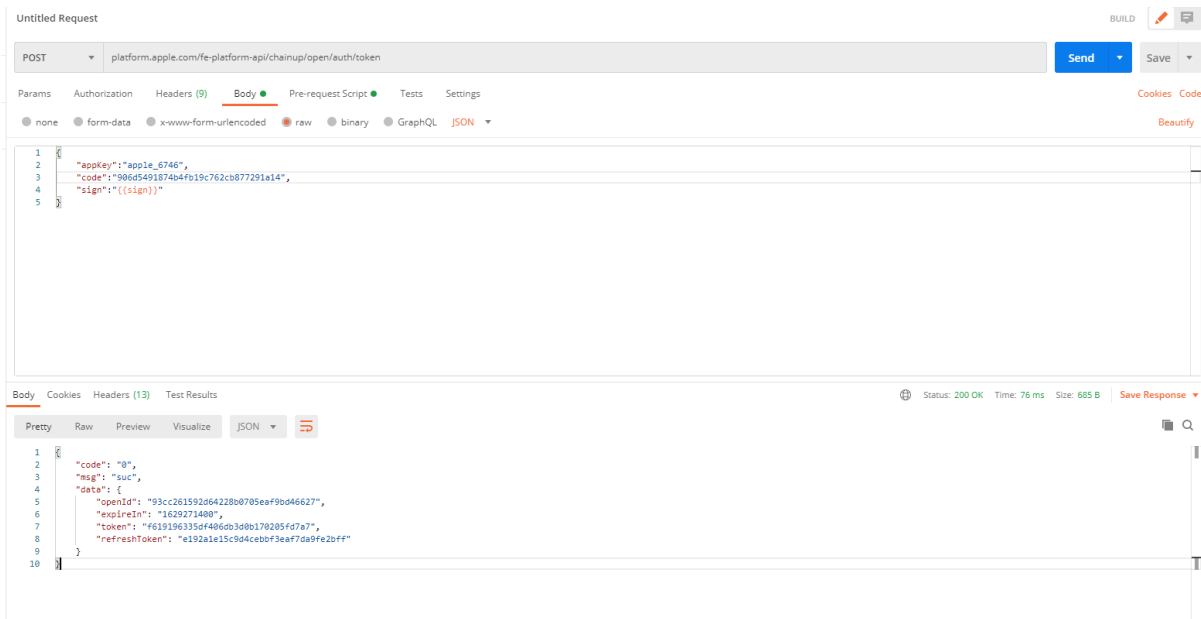
```

3.2) 调用接口换取token或openId

接口地址: <http://service.xfnh.com/platformapi/chainup/open/auth/token>

接口文档地址:[使用code换取token、openId接口](#)

POSTMAN 工具调用接口示意图:



如上图所示;
接口调用成功后, 返回信息会携带token和openId;

4.通过Token访问开放平台资源 – 创建订单/支付/查询订单

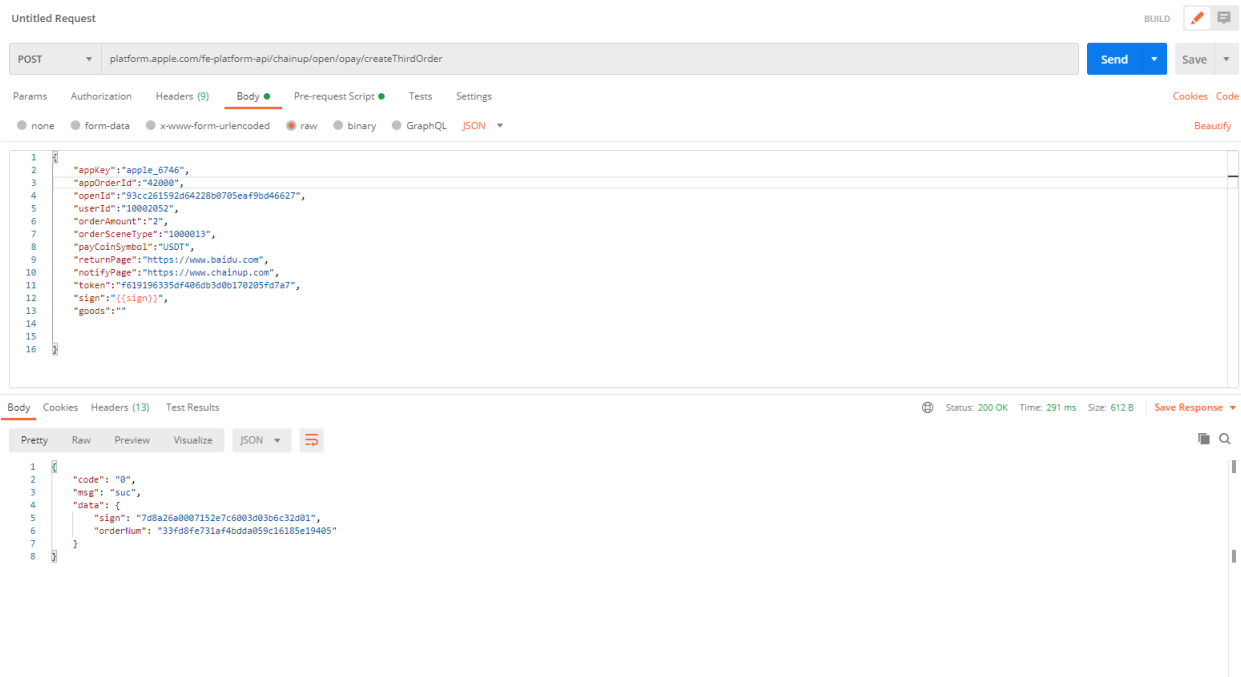
4.1) 创建支付订单

接口地址: <http://service.x.fnhcom/platformapi/chainup/open/opay/createThirdOrder>

接口文档地址:[创建支付订单](#)

注意!!! openId 和userId 只能存在一个,如果都同时存在 优先选择openId

POSTMAN 工具调用接口示意图:



返回结果如上图所示为创建订单成功;

4.1.1)支付页面

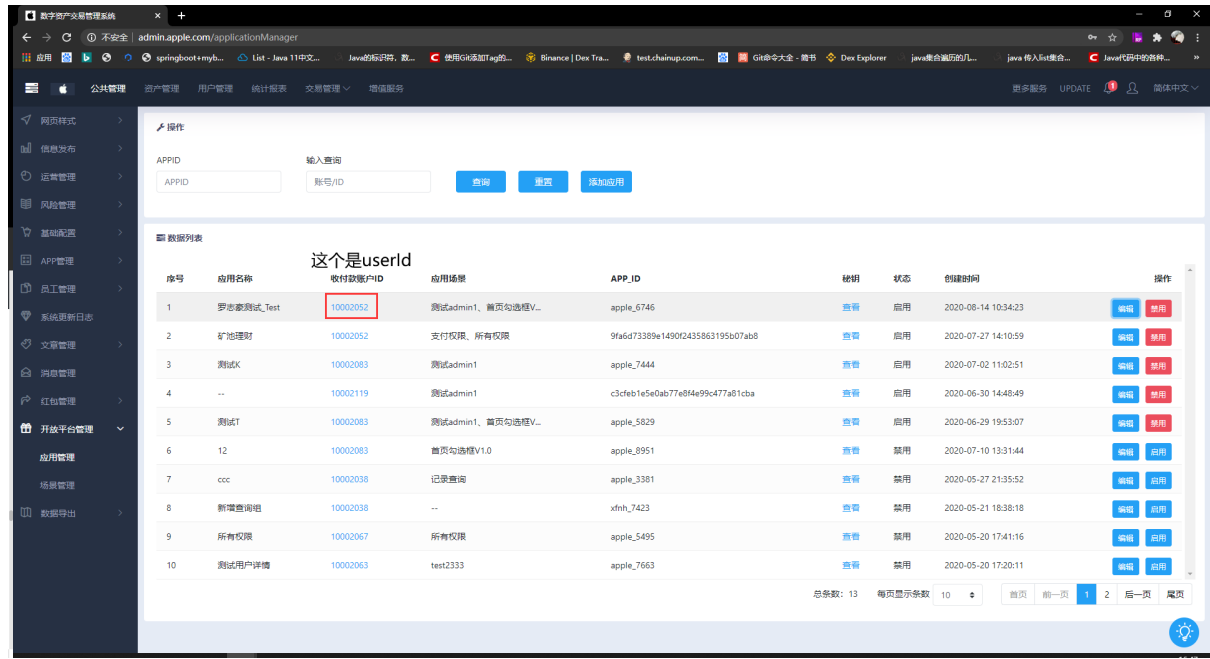
接口地址: <http://service.xfnh.com/pay.html>

接口文档地址: [支付页面](#)

4.1.2) 获取请求参数:

登陆管理后台: 公共管理-- > **开放平台**管理-- > 应用管理

根据图片所示获取请求参数中的userId

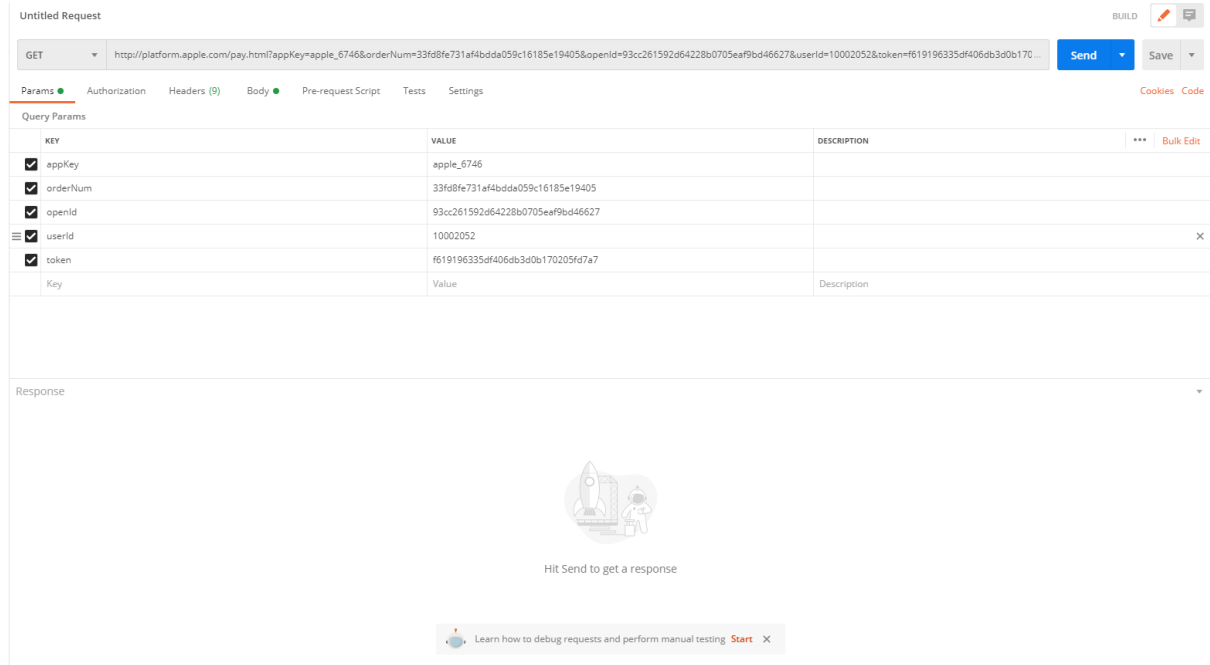


使用POSTMAN工具获取接口调用地址;

拼装完参数后的接口调用地址: [https://service.xxx.com/pay.html?](https://service.xxx.com/pay.html?appKey=xxx&openId=xxx&token=xxx&orderNum=xxx)

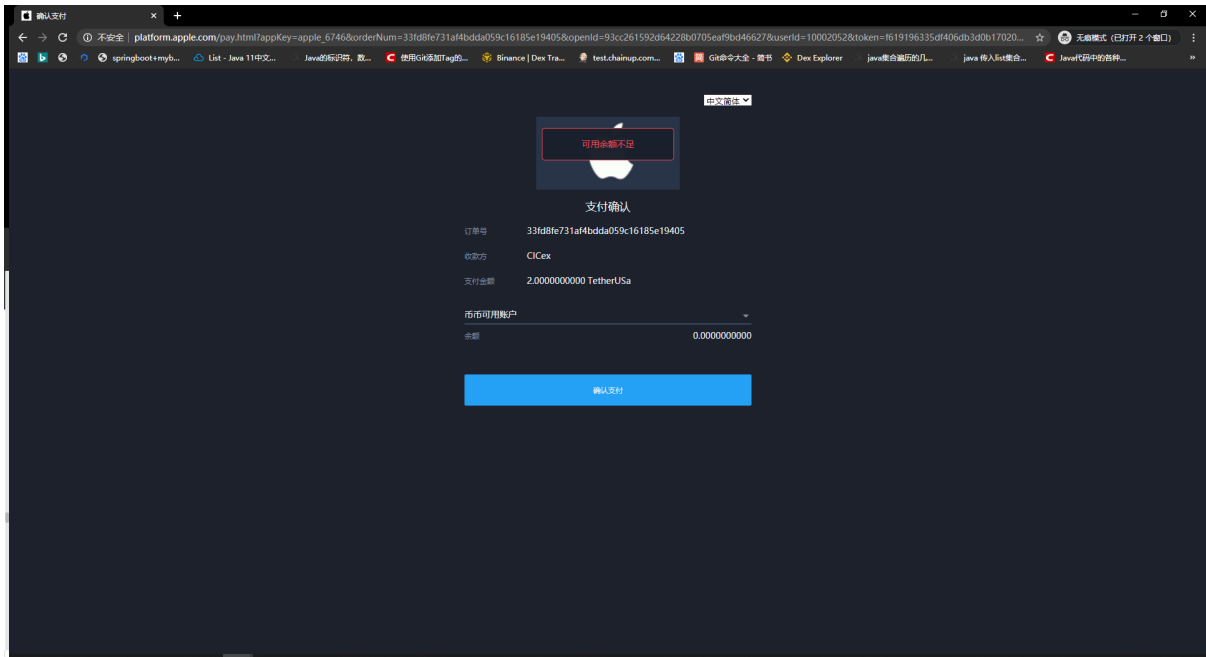
[appKey=xxx&openId=xxx&token=xxx&orderNum=xxx](#)

如果熟悉GET请求的工程师可以直接按照参数拼装地址访问;



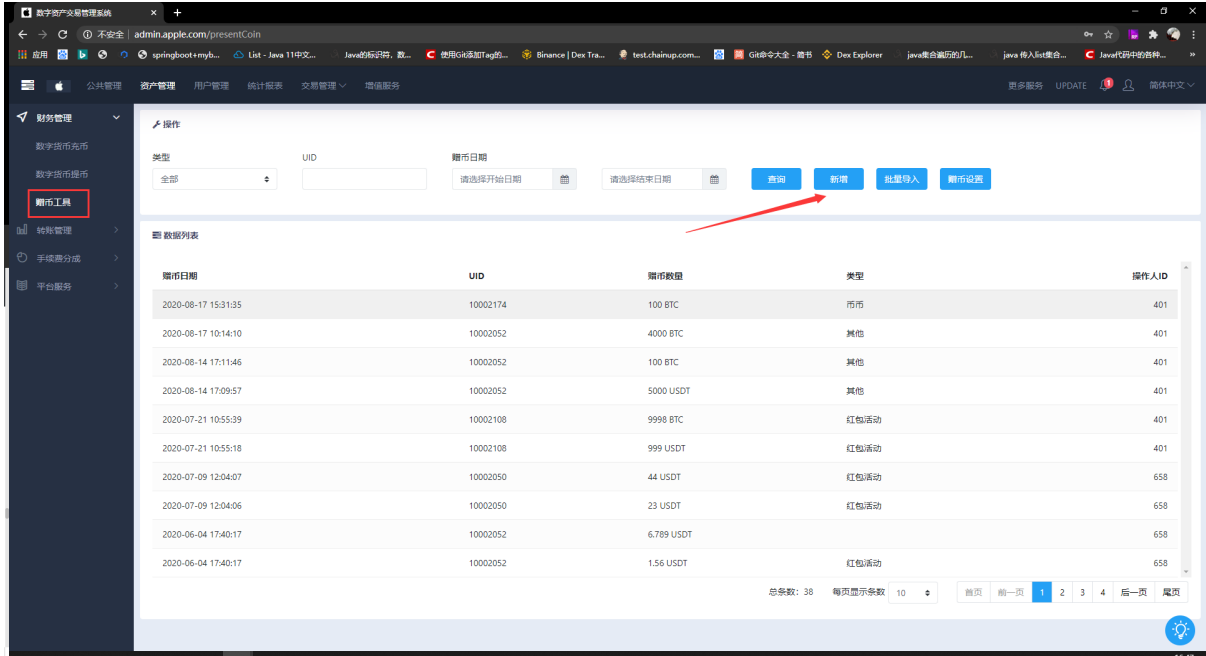
4.1.3) 调用请求支付页面:

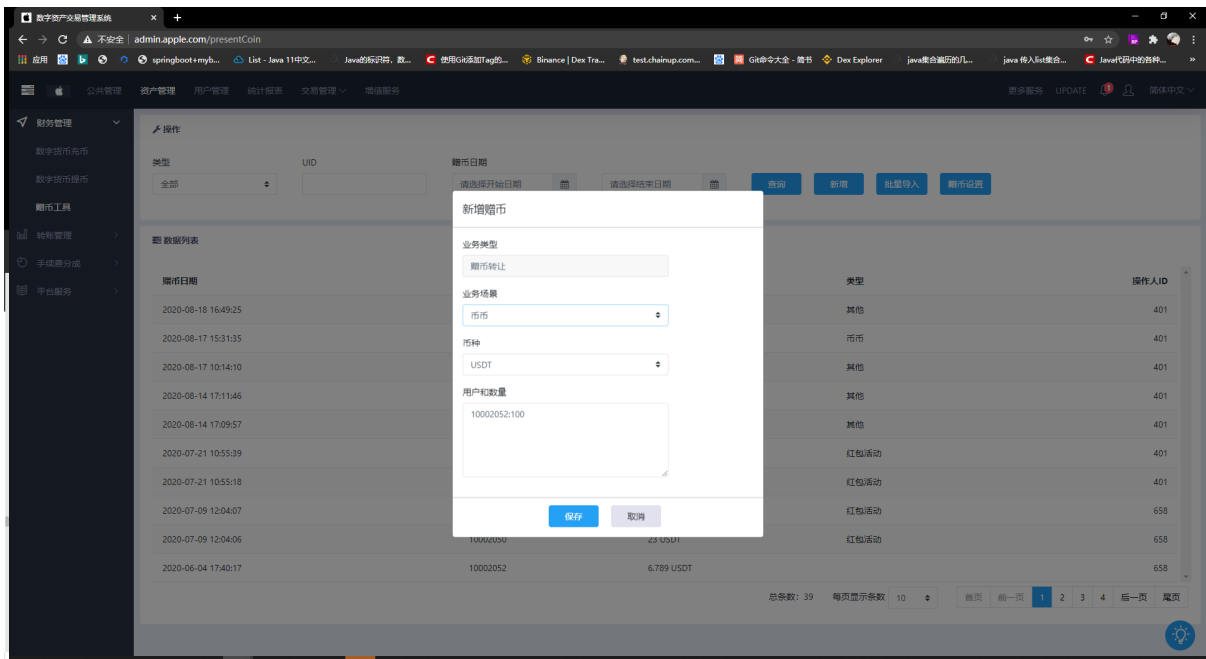
请求支付页面示意图:



4.1.4) 余额不足在管理平台充值增币（如果余额充足可忽略该步）

资产管理-->财务管理 --> 赠币工具





4.1.5) 支付成功后跳转的页面(该页面为第一步配置的 redirectUrl)



至此完成订单创建和支付流程;

4.2) 查询支付订单

接口地址: <http://service.xfnh.com/platformapi/chainup/open/opay/orderDetail>

接口文档地址:[查询支付订单](#)

使用POSTMAN工具获取接口调用地址;

