**企业微信内部应用获取用户信息**

1、构造页面授权链接 获取code参数

文档链接：[https://work.weixin.qq.com/api/doc#90000/90135/91022](https://work.weixin.qq.com/api/doc" \l "90000/90135/91022)  
授权链接：<https://open.weixin.qq.com/connect/oauth2/authorize?appid=CORPID&redirect_uri=REDIRECT_URI&response_type=code&scope=snsapi_base&state=STATE#wechat_redirect>



Appid: 登录企业微信管理后台->我的企业->企业id（一般以wx开头的一串字符如wx0078xxxxx）  
Redirect\_url: 授权后的项目链接地址，如:http://www.yijushch.com/nebula，URL需要使用urlencode编码处理再填写  
State：自己随机填入a-zA-Z0-9的参数值  
测试应用链接：<https://open.weixin.qq.com/connect/oauth2/authorize?appid=ww007675df259cad12&redirect_uri=http%3a%2f%2fwww.yijushch.com%2fnebula&response_type=code&scope=snsapi_base&state=sdfasd#wechat_redirect>

链接配置好了，放入对应项目的工作台对应主页的链接，前端在http://www.yijushch.com/nebula被访问时监听code这参数值，取得改参数值就是授权访问的code 需要传递给后台：



**2、获取应用的接口凭证 access\_token**

文档链接：[https://work.weixin.qq.com/api/doc#90000/90135/91039](https://work.weixin.qq.com/api/doc" \l "90000/90135/91039)  
获取access\_token地址：<https://qyapi.weixin.qq.com/cgi-bin/gettoken?corpid=ID&corpsecret=SECRET>  
Corpid: 企业id，已wx开头的一串字符串，登录企业微信管理后台->我的企业->企业id  
Corpsecret:应用秘钥，登录企业微信管理后台->应用于小程序->相应应用->secret  
Access\_token 有时间限制7200秒，该接口不宜频繁调用，获取一次后应该缓存一小时左右或者7000秒这样，再获取新的access\_token

1. **获取访问用户身份信息获取UserId**

文档链接: [https://work.weixin.qq.com/api/doc#90000/90135/91023](https://work.weixin.qq.com/api/doc" \l "90000/90135/91023)

获取UserId地址: <https://qyapi.weixin.qq.com/cgi-bin/user/getuserinfo?access_token=ACCESS_TOKEN&code=CODE>

Access\_token: 第二步获取的access\_token

Code: 第一步由前端获取的code传递给到后台

在返回结果信息中获取UserId

1. **通过UserId获取成员信息，即访问用户详情**

文档链接：[https://work.weixin.qq.com/api/doc#90000/90135/90196](https://work.weixin.qq.com/api/doc" \l "90000/90135/90196)

获取用户信息地址: <https://qyapi.weixin.qq.com/cgi-bin/user/get?access_token=ACCESS_TOKEN&userid=USERID>

Access\_token: 第二步获取的access\_token

Userid: 第三部获取的UserId

**结果集参数说明请移步文档！**

**后端代码逻辑：**

**配置文件**

*#WeChat***wx**:  
 **config**:  
 *#企业id* **corpId** : ww007675df259cad12  
 *#应用的密钥* **secret** : Z-BQPW4cH5YWEmwkCECIC3Dna8FClfFMq6hwGbS8gn4  
 *#获取access\_token是调用企业微信API接口* **access-token-url** : https://qyapi.weixin.qq.com/cgi-bin/gettoken?corpid={0}&corpsecret={1}  
 *#获取用户userid* **user-info-url** : https://qyapi.weixin.qq.com/cgi-bin/user/getuserinfo?access\_token={0}&code={1}  
 *#获取用户详情信息地址* **user-details-url** : https://qyapi.weixin.qq.com/cgi-bin/user/get?access\_token={0}&userid={1}

实现代码

private String accessToken = "access\_token";  
private String userId = "UserId";  
private String mobile = "mobile";  
private String wxAccessToken = "wx\_access\_token";  
  
@Autowired  
private QYWXConfig qywxConfig;  
@Autowired  
private RedisUtils redisUtils;

@Override  
public String getUserMobile(String code) throws IOException {  
 JSONObject userInfo = getUserInfo(code);  
 String userMobile = userInfo.getString(mobile);  
 AssertUtil.*isBlank*(userMobile, "手机号码错误,请检查您的企业微信设置");  
 return userMobile;  
}

// 通过 UserId 获取用户详细信息  
private JSONObject getUserInfo(String code) throws IOException {  
 // 判断wx\_access\_token 是否存在还有效，如果存在有效就取缓存，否者重新获取  
 String access\_token;  
 if (redisUtils.isExpire(wxAccessToken)) {  
 access\_token = redisUtils.get(wxAccessToken);  
 } else {  
 access\_token = getAccessToken();  
 }  
 // 获取用户id  
 String userId = getUserInfoOfUserID(code, access\_token);  
 // 拼接获取用户信息URL  
 String url = StrUtil.*indexedFormat*(qywxConfig.getUserDetailsURL(), access\_token, userId);  
 *log*.info("获取用户详细信息:" + url);  
 JSONObject jsonObject = HttpUtil.*doGetStr*(url);  
 AssertUtil.*isNull*(jsonObject, CodeMsg.*WX\_CONFIG\_ERR*, "获取用户详细信息返回空信息,url:" + url);  
 return jsonObject;  
}  
  
// 获取access\_token是调用企业微信API接口, 需要缓存 access\_token 7200秒过期 失效后再从新获取  
private String getAccessToken() throws IOException {  
 // 取出企业微信的企业id和内部应用的秘钥  
 String corpid = qywxConfig.getCorpID();  
 String corpsecret = qywxConfig.getSecret();  
 String url = StrUtil.*indexedFormat*(qywxConfig.getAccessTokenURL(),corpid, corpsecret);  
 *log*.info("获取调用接口凭证access\_token:" + url);  
 JSONObject jsonObject = HttpUtil.*doGetStr*(url);  
 AssertUtil.*isNull*(jsonObject, CodeMsg.*WX\_CONFIG\_ERR*, "获取AccessToken返回空信息,url:" + url);  
 String access\_token = jsonObject.getString(accessToken);  
 // 需要进行数据缓存access\_token,以及过期时间 定7000秒后过期  
 redisUtils.set(wxAccessToken, access\_token, 7000);  
 return access\_token;  
}  
  
// 通过接口凭证access\_token 获取用户信息中的 UserId  
private String getUserInfoOfUserID(String code, String access\_token) throws IOException {  
 String url = StrUtil.*indexedFormat*(qywxConfig.getUserInfoURL(), access\_token, code);  
 *log*.info("获取用户信息URL:" + url);  
 JSONObject jsonObject = HttpUtil.*doGetStr*(url);  
 AssertUtil.*isNull*(jsonObject, CodeMsg.*WX\_CONFIG\_ERR*, "获取user\_id返回空信息,url:" + url);  
 return jsonObject.getString(userId);  
}