目录

[微信支付DLL](#_Toc14912_WPSOffice_Level1) [2](#_Toc14912_WPSOffice_Level1)

[对程序添加引用WxTenpay.dll](#_Toc21675_WPSOffice_Level1) [2](#_Toc21675_WPSOffice_Level1)

[DLL 所用Framework版本 .NET Framework4.0](#_Toc3970_WPSOffice_Level1) [2](#_Toc3970_WPSOffice_Level1)

[WXconfig类配置公众号支付](#_Toc27069_WPSOffice_Level1) [2](#_Toc27069_WPSOffice_Level1)

[APP\_Iconfig类配置IOS APP](#_Toc7678_WPSOffice_Level1) [2](#_Toc7678_WPSOffice_Level1)

[APP\_Aconfig类配置安卓APP](#_Toc20176_WPSOffice_Level1) [2](#_Toc20176_WPSOffice_Level1)

[先给配置类配置参数后调用所需方法](#_Toc21675_WPSOffice_Level2) [2](#_Toc21675_WPSOffice_Level2)

[WXconfig类](#_Toc3970_WPSOffice_Level2) [2](#_Toc3970_WPSOffice_Level2)

[APP\_Iconfig类与APP\_Aconfig类相同](#_Toc27069_WPSOffice_Level2) [2](#_Toc27069_WPSOffice_Level2)

[WeChatPayment类](#_Toc7678_WPSOffice_Level2) [3](#_Toc7678_WPSOffice_Level2)

[WechatPublic类](#_Toc20176_WPSOffice_Level2) [3](#_Toc20176_WPSOffice_Level2)

[Wx\_pay类](#_Toc20548_WPSOffice_Level2) [7](#_Toc20548_WPSOffice_Level2)

[Gerenxinxi类](#_Toc23068_WPSOffice_Level2) [7](#_Toc23068_WPSOffice_Level2)

[前端页面引用实例](#_Toc21675_WPSOffice_Level3) [9](#_Toc21675_WPSOffice_Level3)

[微信扫一扫](#_Toc20548_WPSOffice_Level1) [9](#_Toc20548_WPSOffice_Level1)

[微信上传头像](#_Toc23068_WPSOffice_Level1) [10](#_Toc23068_WPSOffice_Level1)

[微信自定义分享](#_Toc29739_WPSOffice_Level1) [11](#_Toc29739_WPSOffice_Level1)

[微信支付有2个调起微信公众号支付的模式](#_Toc6313_WPSOffice_Level1) [13](#_Toc6313_WPSOffice_Level1)

[1.直接调起支付（旧版）](#_Toc18147_WPSOffice_Level1) [13](#_Toc18147_WPSOffice_Level1)

[1. 配置wx.config后调起支付（新版）](#_Toc19975_WPSOffice_Level1) [14](#_Toc19975_WPSOffice_Level1)

[说明：wx.config所需参数由Weixinjsapi\_ticket（）方法提供](#_Toc29739_WPSOffice_Level2) [14](#_Toc29739_WPSOffice_Level2)

微信支付DLL

对程序添加引用WxTenpay.dll

DLL 所用Framework版本 .NET Framework4.0

WXconfig类配置公众号支付

APP\_Iconfig类配置IOS APP

APP\_Aconfig类配置安卓APP

先给配置类配置参数后调用所需方法

### WXconfig类

|  |  |  |
| --- | --- | --- |
| **参数** | **类型** | **描述** |
| **appid** | string | 公众号ID |
| **secret** | string | 微信公众号的secret |
| **url** | String | 回调url |
| **mch\_id** | String | 商户号 |
| **paysignkey** | String | 证书密匙 |

### APP\_Iconfig类与APP\_Aconfig类相同

|  |  |  |
| --- | --- | --- |
| **参数** | **类型** | **描述** |
| **appid** | string | 微信公众号的secret |
| **url** | String | 回调url |
| **partnerid** | String | 商户号 |
| **paysignkey** | String | 证书密匙 |

### WeChatPayment类

|  |  |  |
| --- | --- | --- |
| **方法** | **返回类型** | **描述** |
| **NATIVEPayMent（）** | string | 微信扫码支付 |
| **GetPayMent\_result（）** | string | 查询扫码订单情况 |
| **PayMent\_result（）** | String | 获取支付订单状态（微信回调） |
| **JSAPIPayMent（）** | String | 微信公众号支付（返回所需参数） |
| **APP\_PayMent（）** | Wx\_pay类型 | 微信APP支付 |
| **CashPayMent（）** | string | 微信提现 |

### WechatPublic类

|  |  |  |
| --- | --- | --- |
| **方法** | **返回类型** | **描述** |
| **GetPersonal（）** | Gerenxinxi类型 | code获取个人信息 |
| **GetOpenid（）** | string | code返回openid |
| **Weixinjsapi\_ticket（）** | String | 获取signature,appid, nonceStr, timeStamp参数 |
| **GetToken（）** | string | 获取Token |
| **Head\_portrait（）** | String | 返回上传头像路劲 |
| **OpenidGetHead（）** | String | 返回上传头像路劲 |
| **Download\_Image（）** | String | 返回上传头像路劲 |
| **WeiXinKeFu（）** | string | 微信文本客户消息 |
| **WeiXinTemplate（）** | string | 发送微信模板消息 |

=======WechatPublic类(方法)说明===========================

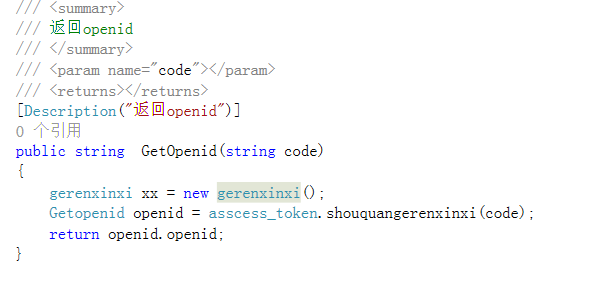
## GetPersonal(string code)方法

|  |  |  |
| --- | --- | --- |
| **参数** | **类型** | **描述** |
| **Code** | string | 微信给予的Code |

## 

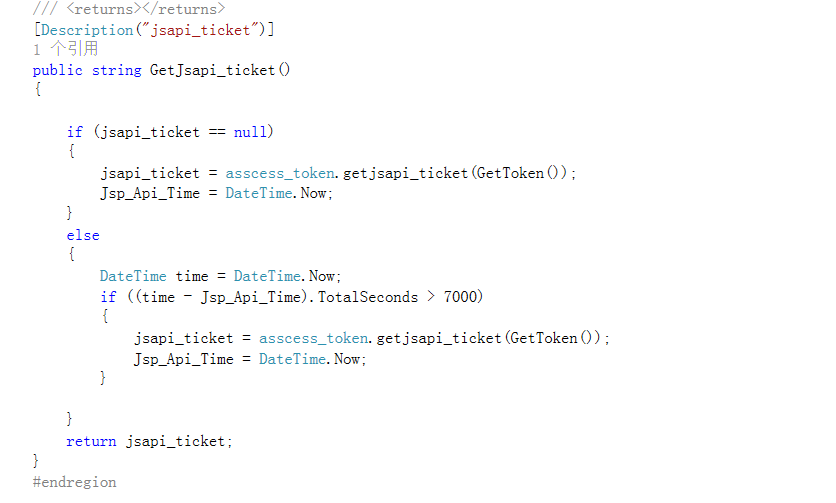
## GetOpenid(string code)方法

|  |  |  |
| --- | --- | --- |
| **参数** | **类型** | **描述** |
| **Code** | string | 微信给予的Code |

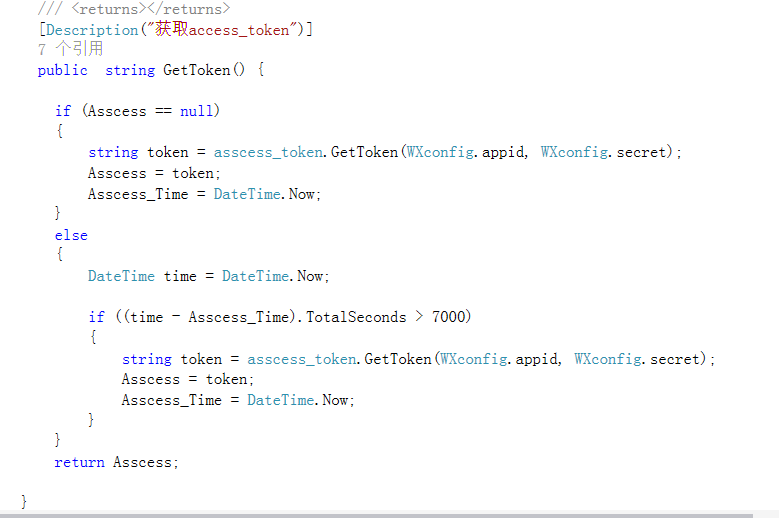


## Weixinjsapi\_ticket(string url)方法

|  |  |  |
| --- | --- | --- |
| **参数** | **类型** | **描述** |
| **URL** | string | 需要配置前端wx.Config的路径如 var url= location.href.split('#')[0] |



## GetToken()方法



### Head\_portrait(string serverId,string path,string Name)方法

|  |  |  |
| --- | --- | --- |
| **参数** | **类型** | **描述** |
| **serverId** | string | 微信上传图片返回的id |
| **Path** | String | 需要下载图片的路劲以/结尾 |
| **Name** | String | 图片的名称以.jpg，.png 结尾 |



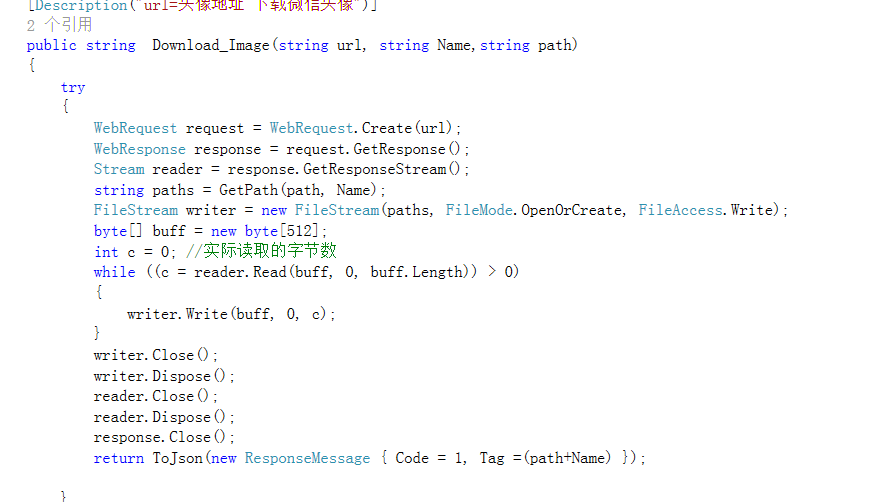
## OpenidGetHead(string openid, string path, string Name)方法

|  |  |  |
| --- | --- | --- |
| **参数** | **类型** | **描述** |
| **openid** | string | 用户的openid |
| **Path** | String | 需要下载图片的路劲以/结尾 |
| **Name** | String | 图片的名称已.jpg，.png 结尾 |



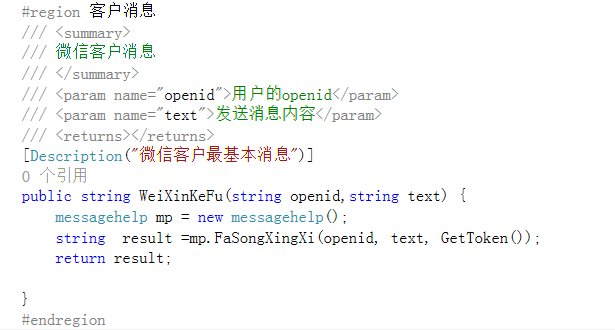
## Download\_Image(string url, string Name,string path)方法

|  |  |  |
| --- | --- | --- |
| **参数** | **类型** | **描述** |
| **url** | string | 需要下载图片的url |
| **Path** | String | 需要下载图片的路劲以/结尾 |
| **Name** | String | 图片的名称已.jpg，.png 结尾 |



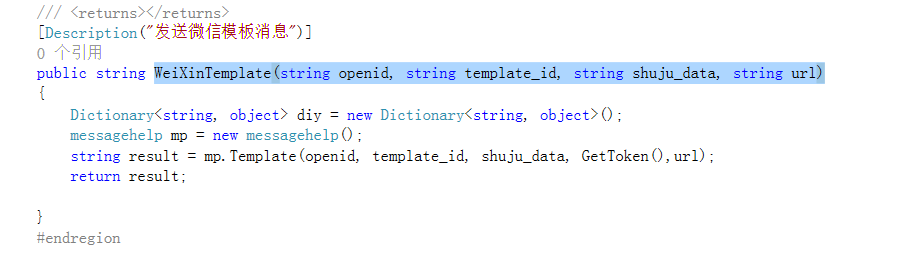
**WeiXinKeFu(string openid,string text)**

|  |  |  |
| --- | --- | --- |
| **参数** | **类型** | **描述** |
| **openid** | string | 用户的openid |
| **text** | String | 所要提示的信息 |

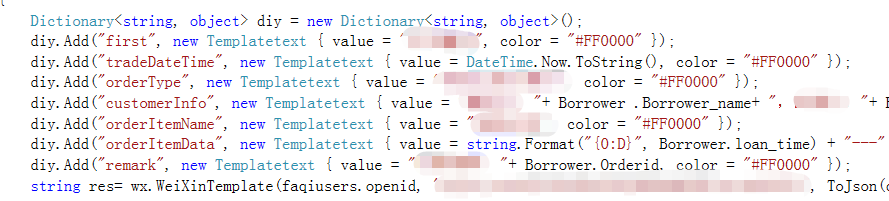


**WeiXinTemplate(string openid, string template\_id, string shuju\_data, string url)**

|  |  |  |
| --- | --- | --- |
| **参数** | **类型** | **描述** |
| **openid** | string | 用户的openid |
| **shuju\_data** | string | 发送模板消息的内容 |
| **url** | string | 模板点击的地址 |
| **template\_id** | String | 模板的ID |



**Shuju\_data 说明：**

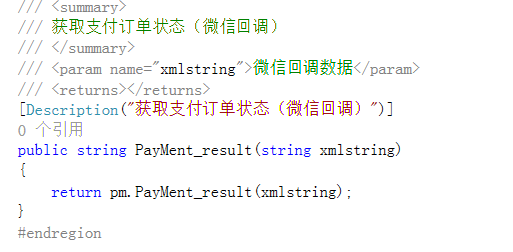


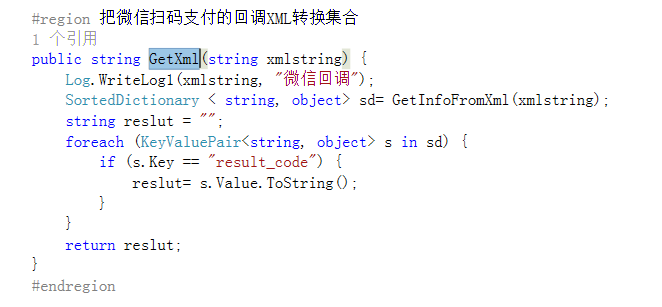
//----------------------------WechatPublic结束--------------------------

//------------------WeChatPayment类(方法)说明-------------------------

### PayMent\_result（string xmlstring）方法

|  |  |  |
| --- | --- | --- |
| **参数** | **类型** | **描述** |
| **xmlstring** | string | 微信回调数据 |





### NATIVEPayMent(string boby, string attach,string spbill\_create\_ip, Double total\_fee, string out\_trade\_no, string product\_id)方法

|  |  |  |
| --- | --- | --- |
| **参数** | **类型** | **描述** |
| **boby** | string | 商品描述 |
| **spbill\_create\_ip** | string | 终端IP |
| **total\_fee** | Double | 金额，实际金额 |
| **out\_trade\_no** | string | 商户订单号 |
| **attach** | string | 附加数据 |
| **product\_id** | String | 二维码中包含的商品ID |

# 

# APP\_PayMent(string boby, string attach, string spbill\_create\_ip, Double total\_fee, string out\_trade\_no, int type)方法

|  |  |  |
| --- | --- | --- |
| **参数** | **类型** | **描述** |
| **boby** | string | 商品描述 |
| **spbill\_create\_ip** | string | 终端IP |
| **total\_fee** | Double | 金额，实际金额 |
| **out\_trade\_no** | string | 商户订单号 |
| **attach** | string | 附加数据 |
| **type** | Int | APP类型1.安卓，2.IOS |

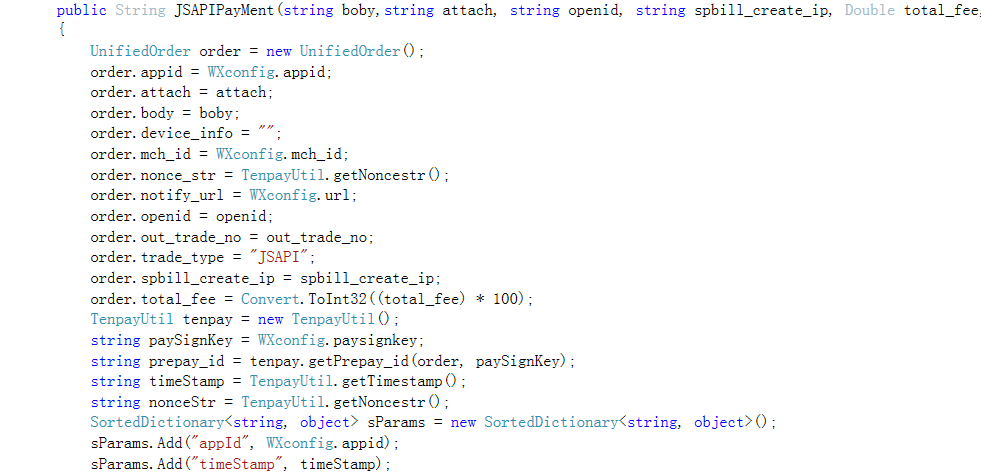
# 

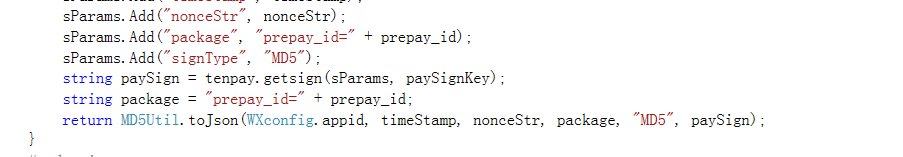
# 



# JSAPIPayMent(string boby, string attach, string openid, string spbill\_create\_ip, int total\_fee, string out\_trade\_no)方法

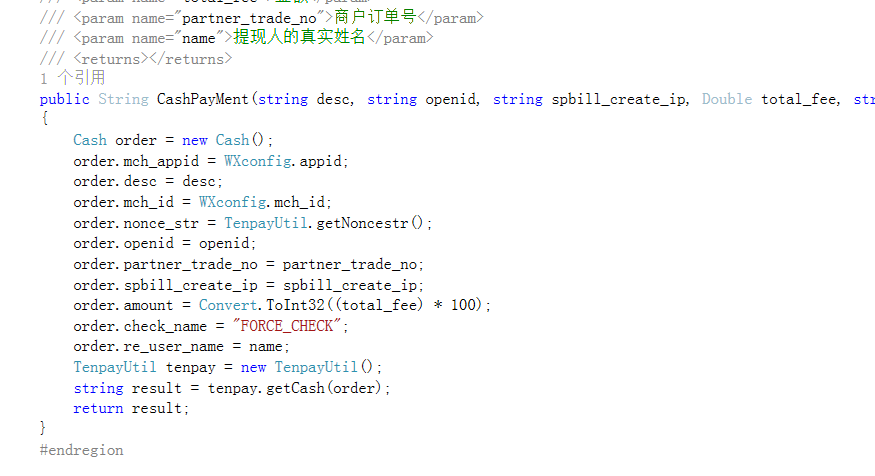
|  |  |  |
| --- | --- | --- |
| **参数** | **类型** | **描述** |
| **boby** | string | 商品描述 |
| **spbill\_create\_ip** | string | 终端IP |
| **total\_fee** | Double | 金额，实际金额 |
| **out\_trade\_no** | string | 商户订单号 |
| **attach** | string | 附加数据 |
| **Opened** | String | 用户的openid |





**CashPayMent(string desc, string openid, string spbill\_create\_ip, Double total\_fee, string partner\_trade\_no, string name)**

|  |  |  |
| --- | --- | --- |
| **参数** | **类型** | **描述** |
| **desc** | string | 企业付款描述信息 |
| **openid** | String | 用户的openid |
| **spbill\_create\_ip** | String | 用户的ip地址 |
| **total\_fee** | double | 提现的金额 |
| **partner\_trade\_no** | String | 商户的订单号 |
| **name** | string | 提现人的真实姓名 |



**说明：微信提现到零钱（所需要求：相关安全证书（去商户号上下载），商户号入驻90天以上，商户号有连续30的交易流水）**

### GetPayMent\_result(string out\_trade\_no)方法

|  |  |  |
| --- | --- | --- |
| **参数** | **类型** | **描述** |
| out\_trade\_no | string | 商户订单ID |





//--------------- WeChatPayment类结束----------------

### Wx\_pay类

|  |  |  |
| --- | --- | --- |
| **参数** | **类型** | **描述** |
| **appid** | string | 应用ID |
| **partnerid** | string | 商户号 |
| **prepayid** | String | 预支付交易会话ID |
| **wxpackage** | string | 扩展字段 |
| **noncestr** | string | 随机字符串 |
| **timestamp** | String | 时间戳 |
| **sign** | String | 签名 |

### Gerenxinxi类

|  |  |  |
| --- | --- | --- |
| **参数** | **类型** | **描述** |
| **subscribe** | string | 用户是否订阅该公众号标识，值为0时，代表此用户没有关注该公众号，拉取不到其余信息 |
| **openid** | string | 用户的标识，对当前公众号唯一 |
| **nickname** | String | 用户的昵称 |
| **sex** | Int | 用户的性别，值为1时是男性，值为2时是女性，值为0时是未知 |
| **city** | string | 用户所在城市 |
| **country** | String | 用户所在国家 |
| **province** | String | 用户所在省份 |
| **language** | String | 用户的语言，简体中文为zh\_CN |
| **headimgurl** | String | 用户头像，最后一个数值代表正方形头像大小（有0、46、64、96、132数值可选，0代表640\*640正方形头像），用户没有头像时该项为空。若用户更换头像，原有头像URL将失效。 |
| **subscribe\_time** | String | 用户关注时间，为时间戳。如果用户曾多次关注，则取最后关注时间 |
| **unionid** | String | 只有在用户将公众号绑定到微信开放平台帐号后，才会出现该字段。 |
| **remark** | String | 公众号运营者对粉丝的备注，公众号运营者可在微信公众平台用户管理界面对粉丝添加备注 |
| **groupid** | String | 用户所在的分组ID（兼容旧的用户分组接口 |
| **tagid\_list** | String[] | 用户被打上的标签ID列表 |
| **errcode** | String | 错误编号 |
| **errmsg** | String | 错误原因 |

# 前端页面引用实例

微信扫一扫

$.get("url", date, function (data) {

var da = $.parseJSON(data);

var da1 = eval("(" + da.Tag + ")");

wx.config({

debug: false, // 开启调试模式,调用的所有api的返回值会在客户端alert出来，若要查看传入的参数，可以在pc端打开，参数信息会通过log打出，仅在pc端时才会打印。

appId: da1.appId, // 必填，公众号的唯一标识

timestamp: da1.timeStamp, // 必填，生成签名的时间戳

nonceStr: da1.nonceStr, // 必填，生成签名的随机串

signature: da1.signature,// 必填，签名，见附录1

jsApiList: ["scanQRCode", "checkJsApi"] // 必填，需要使用的JS接口列表，所有JS接口列表见附录2

});

wx.error(function (res) {

alert("wx.config failed.");

//alert(res);

// config信息验证失败会执行error函数，如签名过期导致验证失败，具体错误信息可以打开config的debug模式查看，

// 也可以在返回的res参数中查看，对于SPA可以在这里更新签名。

});

});

//微信扫一扫

function QeCode(id) {

wx.scanQRCode({

needResult: 1,

desc: 'scanQRCode desc',

success: function (res) {

//扫到的参数

var result = res.resultStr;

var url = "";

$.get(url,function (result) {

})

}

});

}

微信上传头像

wx.config({

debug: false, // 开启调试模式,调用的所有api的返回值会在客户端alert出来，若要查看传入的参数，可以在pc端打开，参数信息会通过log打出，仅在pc端时才会打印。

appId: da1.appId, // 必填，公众号的唯一标识

timestamp: da1.timeStamp, // 必填，生成签名的时间戳

nonceStr: da1.nonceStr, // 必填，生成签名的随机串

signature: da1.signature,// 必填，签名，见附录1

jsApiList: ["chooseImage", "uploadImage", "downloadImage"] // 必填，需要使用的JS接口列表，所有JS接口列表见附录2

});

wx.ready(function () {

//chooseImage();

//alert("wx.config success.");

// config信息验证后会执行ready方法，所有接口调用都必须在config接口获得结果之后，config是一个客户端的异步操作，所以如果需要在页面加载时就调用相关接口，则须把相关接口放在ready函数中调用来确保正确执行。对于用户触发时才调用的接口，则可以直接调用，不需要放在ready函数中。

});

wx.error(function (res) {

alert("wx.config failed.");

// alert(res);

// config信息验证失败会执行error函数，如签名过期导致验证失败，具体错误信息可以打开config的debug模式查看，

// 也可以在返回的res参数中查看，对于SPA可以在这里更新签名。

});

//=============================================

function chooseImage() {

wx.chooseImage({

count: 1, // 默认9

sizeType: ['original', 'compressed'], // 可以指定是原图还是压缩图，默认二者都有

sourceType: ['album', 'camera'], // 可以指定来源是相册还是相机，默认二者都有

success: function (res) {

var localIds = res.localIds; // 返回选定照片的本地ID列表，localId可以作为img标签的src属性显示图片

$("#imgs").attr("src", localIds[0]);

images = localIds[0];

}

});

}

function uploadImage(localId) {

wx.uploadImage({

localId: localId, // 需要上传的图片的本地ID，由chooseImage接口获得

isShowProgressTips: 1, // 默认为1，显示进度提示

success: function (res) {

var serverId = res.serverId; // 返回图片的服务器端ID

//get中url指后台处理的url，

$.get(url, data, function (r) {

var result=eval("("+r+")");

if (result.Code ==1) {

alert("保存成功！");

} else

alert(result.Message);

})

//downloadImage(serverId);

}

});

}

function downloadImage(serverId) {

wx.downloadImage({

serverId: serverId, // 需要下载的图片的服务器端ID，由uploadImage接口获得

isShowProgressTips: 1, // 默认为1，显示进度提示

success: function (res) {

var localId = res.localId; // 返回图片下载后的本地ID

}

});

}

-----------------------------------------------------------

微信自定义分享

wx.config({

debug: false, // 开启调试模式,调用的所有api的返回值会在客户端alert出来，若要查看传入的参数，可以在pc端打开，参数信息会通过log打出，仅在pc端时才会打印。

appId: da1.appId, // 必填，公众号的唯一标识

timestamp: da1.timeStamp, // 必填，生成签名的时间戳

nonceStr: da1.nonceStr, // 必填，生成签名的随机串

signature: da1.signature,// 必填，签名，见附录1

jsApiList: ["onMenuShareTimeline", "onMenuShareQZone", "onMenuShareAppMessage"] // 必填，需要使用的JS接口列表，所有JS接口列表见附录2

});

wx.ready(function () {

var title = '<%=name%>';

var link = '<%=url%>';

var imgUrl = '<%=pathurl%>';

var desc = '<%=describe%>';

//获取“分享到朋友圈”按钮点击状态及自定义分享内容接口

wx.onMenuShareTimeline({

title: title, // 分享标题

link: link, // 分享链接，该链接域名或路径必须与当前页面对应的公众号JS安全域名一致

imgUrl: imgUrl, // 分享图标

success: function () {

alert("分享成功！");

// 用户确认分享后执行的回调函数

},

cancel: function () {

alert("您取消了分享");

// 用户取消分享后执行的回调函数

}

});

//获取“分享到QQ空间”按钮点击状态及自定义分享内容接口

wx.onMenuShareQZone({

title: title, // 分享标题

desc: desc, // 分享描述

link: link, // 分享链接

imgUrl: imgUrl, // 分享图标

success: function () {

alert("分享成功！");

// 用户确认分享后执行的回调函数

},

cancel: function () {

alert("您取消了分享");

// 用户取消分享后执行的回调函数

}

});

//获取“分享给朋友”按钮点击状态及自定义分享内容接口

wx.onMenuShareAppMessage({

title: title, // 分享标题

desc: desc, // 分享描述

link: link, // 分享链接，该链接域名或路径必须与当前页面对应的公众号JS安全域名一致

imgUrl: imgUrl, // 分享图标

type: '', // 分享类型,music、video或link，不填默认为link

dataUrl: '', // 如果type是music或video，则要提供数据链接，默认为空

success: function () {

alert("分享成功！");

// 用户确认分享后执行的回调函数

},

cancel: function () {

alert("您取消了分享");

// 用户取消分享后执行的回调函数

}

});

});

wx.error(function (res) {

alert("wx.config failed.");

alert(res);

// config信息验证失败会执行error函数，如签名过期导致验证失败，具体错误信息可以打开config的debug模式查看，

// 也可以在返回的res参数中查看，对于SPA可以在这里更新签名。

});

微信支付有2个调起微信公众号支付的模式

1.直接调起支付（旧版）

//支付功能

function jsApiCall(r) {

WeixinJSBridge.invoke(

'getBrandWCPayRequest', r,

function (res) {

// alert(res.err\_code + res.err\_desc);

// alert(res.err\_msg);

if (res.err\_msg == "get\_brand\_wcpay\_request:ok") {

alert("支付成功");

} else {

alert("用户取消购买");

}

}

);

}

function callpay(r) {

if (typeof WeixinJSBridge == "undefined") {

if (document.addEventListener) {

document.addEventListener('WeixinJSBridgeReady', jsApiCall, false);

} else if (document.attachEvent) {

document.attachEvent('WeixinJSBridgeReady', jsApiCall);

document.attachEvent('onWeixinJSBridgeReady', jsApiCall);

}

} else {

jsApiCall(r);

}

}

说明：r是配置的参数 = JSAPIPayMent（）方法返回的参数

1. 配置wx.config后调起支付（新版）

## 说明：wx.config所需参数由Weixinjsapi\_ticket（）方法提供

微信回调方法调用说明

#region 微信回调

/// <summary>

/// 微信url回调

/// </summary>

/// <returns></returns>

public void NATIVEPayMent\_Result()

{

// 接受来自微信的信息

if (System.Web.HttpContext.Current.Request.HttpMethod.ToUpper() == "POST")

{

PayMent pay = new PayMent();

using (Stream stream = System.Web.HttpContext.Current.Request.InputStream)

{

byte[] postbyte = new byte[stream.Length];

stream.Read(postbyte, 0, (Int32)stream.Length);

string poststring = Encoding.UTF8.GetString(postbyte);

string result= pay.PayMent\_result(poststring);

if (result == "SUCCESS")

{

// string rest = @"< xml >

//< return\_code >< ![CDATA[SUCCESS]] ></ return\_code >

// < return\_msg >< ![CDATA[OK]] ></ return\_msg >

// </ xml >";

System.Web.HttpContext.Current.Response.AddHeader("ContentType", "text/xml");

System.Web.HttpContext.Current.Response.Write("SUCCESS");

System.Web.HttpContext.Current.Response.End();

}

}

}

System.Web.HttpContext.Current.Response.AddHeader("ContentType", "text/xml");

System.Web.HttpContext.Current.Response.Write("SUCCESS");

System.Web.HttpContext.Current.Response.End();

}

#endregion