

——微信与移动Web开发之

# 第5讲 自定义菜单

# 本次课程目录

---

- 一 . 使用微信测试号
- 二 . CURL与JSON
- 三 . 自定义菜单

# 课程概要

- 了解什么是微信测试号，并配置测试号URL。
- 微信接口的格式是JSON数据，并且发起请求是使用curl，先介绍PHP的CURL扩展以及JSON格式化数据
- 在以上基础上，讲解微信接口调用流程，获取并使用access\_token，然后使用access\_token调用微信自定义菜单创建接口创建菜单。

# 第一节

一 . 使用微信测试号

二 . CURL与JSON

三 . 自定义菜单

# 什么是微信测试号

- 未认证的服务号以及已认证和未认证的订阅号接口权限有限，很多接口不能调用。微信公众平台提供了测试号可以体验所有的微信接口，用于开发测试之用。
- 在微信公众平台开发者工具里面有公众平台测试账号选项，微信测试号使用微信客户端扫码既可登录。
- 测试号可以使用所有微信公众平台提供的接口，推出的目的就是针对开发者测试使用。

# 如何申请微信测试号

- 登录微信公众号，之后打开左侧导航下边的开发者工具，页面有‘公众平台测试账号’选项，点击进入。之后弹出的页面，直接点击登录会显示二维码，使用手机微信扫码即可登录。

---

微信号扫一扫登录  
免注册，方便快捷



若你已注册手机帐号，请[点此登录](#)

# 关注测试号

登陆后使用自己的微信号关注测试号

## 测试号二维码



请用微信扫码关注测试公众号

## 用户列表 (最多100个)

序号	昵称	微信号	操作
1			移除

# 测试号与URL验证

- 测试号URL验证与普通公众号一致，但是没有加密选项。
- 测试号没有加密功能，不能配置加密密钥与加密模式。



# 第二讲

一．使用微信测试号

二．CURL与JSON

三．自定义菜单

# PHP扩展CURL

- CURL能够连接通讯各种服务器、使用各种协议。目前支持的协议有 http、https、ftp、telnet等。
- PHP使用curl发起http请求。
- PHP调用微信接口需要使用curl扩展或是基于curl的第三方库。

# 如何使用CURL

- 原生CURL的使用流程：
    - curl\_init初始化连接
    - curl\_setopt设置选项值
    - curl\_exec运行连接结果
    - curl\_close关闭连接。
- 输出结果。

# 原生CURL示例

```
//初始化连接
$c=curl_init('http://www.baidu.com');
//设置连接选项
//返回的内容作为变量储存，而不是直接输出
curl_setopt($c,CURLOPT_RETURNTRANSFER,true);
//禁止 cURL 验证对等证书，
//https连接设置为false免去证书验证过程，直接返回数据
curl_setopt($c,CURLOPT_SSL_VERIFYPEER,false);
//执行请求，获取数据
$ret = curl_exec($c);
//获取数据后关闭当前连接
curl_close($c);
echo $ret;
```

- 运行此示例会显示百度搜索页面。

# 使用第三方扩展库

- 使用anlutro/curl
  - composer require anlutro/curl
- 示例代码：

```
use anlutro\cURL\cURL;
$url = 'http://php.net/';
$curl = new anlutro\cURL\cURL;
$request = $curl->newRequest('get', $url, [])
    ->setHeader('Accept-Charset', 'utf-8')
    ->setOption(CURLOPT_RETURNTRANSFER, true)
    ->setOption(CURLOPT_SSL_VERIFYPEER, false);
$response = $request->send();
echo $response;
```
- 运行此代码显示PHP官方网站页面

# JSON基础

- JSON 是存储和交换文本信息的语法。类似 XML。
- JSON 比 XML 更小、更快，更易解析。
- JSON 使用 Javascript语法来描述数据对象，但是 JSON 仍然独立于语言 and 平台。JSON 解析器和 JSON 库支持许多不同的编程语言。
- JSON 语法是 JavaScript 语法的子集。
- JSON 语法是 JavaScript 对象表示法语法的子集：

数据在名称/值对中

数据由逗号分隔

大括号保存对象

中括号保存数组

示例：

```
{  
  "name":"AlbertWang",  
  "age":"27",  
  "info":{  
    "sex":"male",  
    "native_place":"Hebei"  
  },  
  "skill":[  
    "PHP",  
    "Python",  
    "Linux"  
  ]  
}
```

# JSON格式解析处理

- PHP处理：
  - `json_encode($data)`;把数组转换成JSON格式字符串。
  - `json_decode($data,true)`;把JSON格式字符串转换成PHP数组。

# JSON解析示例

- PHP处理：
  - `json_encode($data,$options)`;把数组转换成JSON格式字符串。
  - `json_decode($data,true)`;把JSON格式字符串转换成PHP数组。
- 通常可以直接使用`json_encode($data)`;后面的参数选项可以没有，网站开发过程中前后端显示数据不会有问题，但是微信服务器会检测数据编码，如果没有参数，在传递`json_encode`编码后的JSON数据会返回错误信息。这种情况要使用`json_encode($data, JSON_UNESCAPED_UNICODE)`;处理数据。



# JSON解析对比

```
$json_test = [  
    'name'=>'河北师范大学',  
    'address'=>'河北省石家庄市裕华区南二环'  
];
```

```
echo json_encode($json_test) . '<br><br>';  
echo json_encode($json_test,JSON_UNESCAPED_UNICODE);
```

- 返回数据：

```
{"name": "\u6cb3\u5317\u5e08\u5927\u5b66", "address": "\u6cb3\u5317\u77f3\u5bb6\u5e84\u88d5\u534e\u533a\u5357\u4e8c\u73af"}
```

```
{"name": "河北师范大学", "address": "河北石家庄裕华区南二环"}
```

# 第三讲

一．使用微信测试号

二．CURL与JSON

三．自定义菜单

# IP白名单

- 调用接口以前需要先把开发者服务器的IP地址加入白名单列表
- 如果在本地测试，则把本地所在公网IP也加入到白名单
- 不加入白名单，调用接口会失败
- 白名单机制是为了安全性考虑

# access\_token

- access\_token是公众号的全局唯一接口调用凭据，公众号调用各接口时都需使用access\_token。
- access\_token的存储至少要保留512个字符空间。
- access\_token的有效期目前为2个小时，需定时刷新。
- 重复获取将导致上次获取的access\_token失效。

# 调用微信接口的凭据

- 开发者ID ( AppId ) : 申请微信公众号产生的唯一ID标识
- 开发者密码 ( AppSecret ) : 开发者密码是校验公众号开发者身份的密码
- access\_token : 每次由获取access\_token接口获取后保存并用于其他接口的调用
- 除access\_token接口以外, 其他接口的调用流程:
  1. 先使用AppId与AppSecret获取有效的access\_token
  2. 之后要通过access\_token与其他参数调用接口
- 微信为何要使用access\_token, 而不是直接使用appid与appsecret调用接口?

# 获取access\_token

- 获取access\_token接口

[https://api.weixin.qq.com/cgi-bin/token?grant\\_type=client\\_credential&appid=APPID&secret=APPSECRET](https://api.weixin.qq.com/cgi-bin/token?grant_type=client_credential&appid=APPID&secret=APPSECRET)

参数	是否必须	说明
grant_type	是	获取access_token填写client_credential
appid	是	第三方用户唯一凭证
secret	是	第三方用户唯一凭证密钥，即appsecret

- 成功调用返回：  
{"access\_token":"ACCESS\_TOKEN","expires\_in":7200}
- 出错：  
{"errcode":40013,"errmsg":"invalid appid"}

# 小练习

- 封装一个access\_token管理的类，用于快速获取access\_token。要求：每次获取从已获得的access\_token读取，如果没有或access\_token过期则重新获取，否则直接返回已获取的access\_token。

# 创建自定义菜单

- 在以上代码基础上实现自定义菜单功能：[https://api.weixin.qq.com/cgi-bin/menu/create?access\\_token=ACCESS\\_TOKEN](https://api.weixin.qq.com/cgi-bin/menu/create?access_token=ACCESS_TOKEN)是创建自定义菜单的接口，把之前获取到的access\_token作为get参数，JSON格式的文本数据使用POST传递。
- 返回值为JSON格式的文本数据：  
正确返回数据格式：`{"errcode":0,"errmsg":"ok"}`  
错误返回数据格式：`{"errcode":40018,"errmsg":"invalid button name size"}`



# 获取自定义菜单

- 获取access\_token

- 调用接口：

[https://api.weixin.qq.com/cgi-bin/menu/get?access\\_token=ACCESS\\_TOKEN](https://api.weixin.qq.com/cgi-bin/menu/get?access_token=ACCESS_TOKEN)

# 获取自定义菜单输出

```
menu=>[
  button=>[
    0=>[
      name=>Tech Web
      sub_button=>[
        0=>[
          type=>view
          name=>
          url=>http://blog.jobbole.com/
          sub_button=>[
            ]
          ]
        ]
        1=>[
          type=>view
```

# 删除自定义菜单

- 删除自定义菜单接口：

`https://api.weixin.qq.com/cgi-bin/menu/delete?access_token=ACCESS_TOKEN`

- 对应创建接口，正确的Json返回结果:

```
{"errcode":0,"errmsg":"ok"}
```

# 感谢聆听！

---

THANK YOU FOR YOUR ATTENTION