



从今天开始微信公众平台教程开写，前面部分会比较浅，已经在玩的朋友可以忽略。

## 第一章 微信公众账号注册、设置、登陆

### 一、微信公众平台注册

- 1、一个没有注册过公众账号的邮箱，如果是 **qq** 邮箱那么对应的 **qq** 号也要没有注册过公众账号。
- 2、身份证扫描件，每个身份证可以注册 **5** 个公众账号
- 3、手机，用来接受注册验证码

4、想好公众账号名称，非常重要，一旦申请成功名称不能修改，并且该名称最好与已获得认证的腾讯微博名称相同，等公众号到 **500** 粉丝后可以自助认证。

准备工作做好后开始进行注册，在浏览器地址栏输入 <http://mp.weixin.qq.com>，进入微信公众平台。



点击注册按钮后进入注册界面

The image shows the registration form on the WeChat Public Platform. The form is divided into four steps: 1. 基本信息 (Basic Information), 2. 邮箱激活 (Email Activation), 3. 信息登记 (Information Registration), and 4. 公众号信息 (Public Account Information). The first step is currently active. It contains the following fields: "邮箱" (Email) with the value "weixin.course@qq.com", "密码" (Password) with a masked value "\*\*\*\*\*", "确认密码" (Confirm Password) with a masked value "\*\*\*\*\*", and "验证码" (Verification Code) with the value "yutb". Below the verification code field, there's a small image of the code "YUTB" and a link "看不清，换一张" (Can't see, change one). At the bottom, there's a checkbox labeled "我同意并遵守《微信公众平台服务协议》" (I agree and comply with the WeChat Public Platform Service Agreement).

填写注册的邮箱（现在没有 qq 号注册了，个人觉得更好，新注册的 qq 号太长太难记）和设置密码，记得协议选项打钩，协议里有一段公众账号所有权其实是归微信所有，so 这个霸王条款大家都懂的，免费的嘛。

点击注册后会发送一封邮件到你注册邮箱要求激活



点击邮件连接后跳转至第三步，公众账号所有者的信息填写，根据要求填写就好了，上传身份证扫描件，手机验证一下。这里要注意的是如果是公司账号，请填写完整，方便后期公司申请一些接口使用，比如自定义菜单接口。

姓名

如果名字包含分隔号“”，请勿省略。

身份证号码

证件照片  证件正面头像及面部区域清晰可见。  
支持 jpg jpeg bmp gif 格式图片，大小不超过 5M。  
[重新上传](#) [已上传](#)

参考示例 

---

手机号码

短信验证码  [发送验证码](#)

城市

固定电话

单位名称

职务

个人住址(可选)

单位地址(可选)

然后就是最重要的一步，微信账号名称设置

1 基本信息 2 邮箱激活 3 信息登记 4 公众号信息

**公众号信息**

帐号名称

(2-16个字)名称一经设置无法更改。如果有同名的微博认证帐号，粉丝满500后可进行自助认证。

功能介绍

(4-120个字)介绍此公众帐号功能与特色，该信息可以更改，将展示给订阅本帐号的用户。

运营地区

语言

类型

[完成](#)

帐号名称 编辑资料



公众号名称

公众号助手

功能介绍: 关注微信官方助手，随时获取微信官方问题答疑、抢先知晓官方平台最新资讯。

关注

这里需要注意的有三点：

- 1、账号名称一旦设定是不能再更改的。
- 2、公众账号的自助认证必须要用与该名称相同的已认证微博来辅助认证。如果两个名称不同的话需要通过邮件方式人工认证，需要提供的资料会比较多。
- 3、在微信客户端里用户搜索公众账号有两个途径，一个是通过微信号直接搜索，一个是通过账号名称搜索。由于微信号通常是英文字符加数字等组合，对国人来说记忆性不强，因此通过中文搜索公众账号是一个重要途径，企业的公众账号中文名称要取的辨识度高，可搜索性强。

功能介绍这块可以先随便写个，后期可以随时修改，账号类型选择普通公众账号。

至此微信公众账号就注册成功了。

## 二、公众账号设置

注册成功以后会直接进入到公众账号后台的设置页面



公众号名称: 公众平台教程

类型: 普通公众帐号

微信号: 设置微信号

申请认证

地区: 中国 浙江 城市

功能介绍: 学习公众平台教程 修改

二维码: 

更多尺寸

首先是设置头像，企业的可以直接拿自己微博上的头像上传，应用类或者个人类的可以根据自己公众账号定位来设计一个头像。

头像需要注意的是微信公众账号头像会有两个样子，一个是方的一个是圆的，圆的那个头像很容易切掉图像或者文字，在设计的时候就要考虑好

功能介绍根据账号定位来设置，建议文字不要超过 40 个字，以账号服务内容为主，力求让用户在关注你前就了解到你的账号是干什么的，不要写公司介绍。



设置微信号

weixincourse

设置微信号后将无法更改。  
可以使用6~20个字母、数字、下划线和减号，必须以字母开头。

确定 取消

然后设置公众账号的微信号，长度必须在 6 位以上，填写后也是不能修改的，大小写没有关系，用户搜索时都是按照小写字母来搜索的。要注意的是尽量少用下划线、减号和数字，减少用户切换键盘的动作，另外下划线和减号用户容易输错。

### 三、公众账号登陆

公众账号登陆还是从 <http://mp.weixin.qq.com> 进入，点击右上角的登陆后弹出窗口，共有三种方式登陆，QQ 号、微信号和注册邮箱，但是目前新注册的只能用邮箱登陆了，以前注册的可以使用 qq 和微信账号登陆，不知道将来是否会有改变。

×

请使用已注册的帐号登录。

帐号：

密码：

登录

忘记密码？

登陆成功以后就进入了微信公众平台后台首页了



请认真阅读公众平台的运营公告，不要推送垃圾广告、色情信息、暴力违法违规内容，或者强制、诱导分享与账号无关的推广信息，也就是说只能做与你公众账号定位相对应的，或者是与你品牌相对应的内容，一些什么增高、丰胸的广告就别搞了，会被封号的。

今天就先到这里了，下一期是微信公众平台功能介绍



# 《微信公众平台入门到精通》Vol.2

## 第二章 微信公众平台后台介绍

### 一、微信公众平台后台欢迎页面

登陆公众平台后首先进入的就是欢迎页，跟大部分 CMS 后台一样，这里会提供的是公众账号的一些运营数据，如下图：



1. 在上次点击用户管理和实时消息两个导航选项后新增加的用户数和消息数，这话有点拗口，但实际就是这样，点击“查看”会分别跳转到用户管理页面和实时消息页面，然后这些数据就分别清零，直到再有新的用户关注和新的消息回复。

2. 最近 7 天的每日新增订阅用户数（每天新增用户数=每天实际新增用户数 - 每天退订用户数）。
3. 最近 7 天的每日公众平台接受到的用户消息数，包括自定义回复能够回复的和不能回复的。消息数的波动一般是根据群发内容的好坏或者活动调查发起等互动产生起伏。
4. 官方关于微信公众平台功能调整的通知，一定要每天瞟一眼，以免遗漏一些重大通知，比如 3 月 26 日起用户关注公众平台的接口消息将由之前推送一条“Hello2BizUser”文本，变化为推送一条“subscribe”。

公众平台后台只保留最新 7 天的新增订阅用户数和接收消息数，可以做一个 **excel** 表格手工来做个记录，方便做一些数据分析或者日志记录，格式非常简单，如下图：

	A	B	C	D
1	日期	新增用户	回复数	事件
2	1.29	53	123	
3	1.30	64	130	
4	1.31	75	189	
5	2.1	79	156	
6	2.2	79	112	
7	2.3	51	233	
8	2.4	123	351	
9	2.5	133	381	
10	2.6	129	201	
11	2.7	67	131	
12	2.8	1521	3329	fenng推荐
13	2.9	541	751	玩转微信微信推荐
14	2.10	178	621	
15	2.11	159	417	
16	2.12	198	486	
17	2.13	475	951	李开复微博推荐
18	2.14	124	267	
19	2.15	96	358	
20	2.16	84	491	
21	2.17	95	257	
22	2.18	87	247	
23	2.19	75	289	
24	2.20	65	533	

## 二、微信公众平台实时消息页面

点击公众平台后台导航的“实时消息”，或者在欢迎页里点击新消息数后面的“查看”就可以进入，该页面用于展现用户通过手机向公众平台发送的即时消息，这是与用户用心互动的重要阵地，老贼我就经常在这个页面和朋友们互动，做人工客服。如下图：



1. 实时消息页面菜单栏。全部消息指的是 7 天内用户发送给公众平台的消息数据；今天、昨天、前天分别对应不同日期的用户当天消息；5 天内消息指的是除去最新三天以后 4 天的消息；星标消息是用来查看重要的被标注为星标的消息。
2. 新消息提醒。这个功能主要是让大家在后台跟用户交流时不必频繁去刷新页面查看是否有新的消息，有用户发送新消息来的时候系统会自动提示。
3. 消息搜索。这个功能有点问题，有很多内容是搜索不到，即使是在 7 天以内，搜索到的话会出现如下页面：

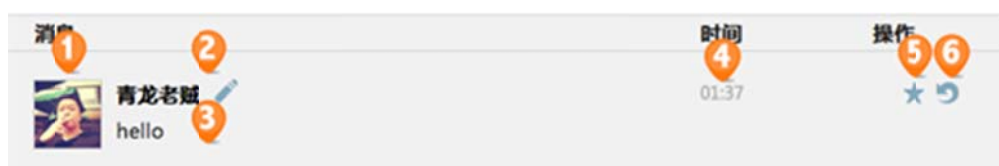


4. 消息列表。根据顺序将用户消息排序显示，每页 50 个，翻页使用的是 ajax，所以无法跳转页面只能一页页翻。

想查 7 天之前的消息怎么办？先点击“5 天内消息”，然后将浏览器地址栏的 URL “day=3”改成“day=10”，就可以查看 10 天以前的消息，如下图：



## 单条消息详解



### 1. 用户头像。

鼠标移动到头像上会显示用户信息卡，有昵称、性别、备注名、地区、个性签名、用户分组等。用户信息里没有用户的微信账号，在公众平台里与用户进行信息交互都通过唯一的 **openid**，主要是为了保护用户不被骚扰。如下图：



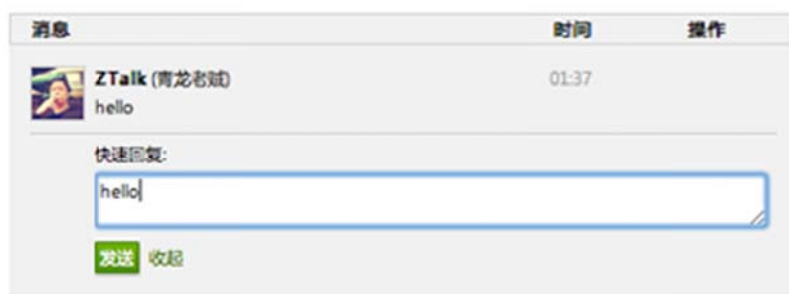
点击用户头像会跳转到与用户一对一的聊天页面，除了可以发文字外，还可以发送表情、语音、图片、视频、图文消息等，对于一些企业客服来说，方便发送一些预设好的咨询回复。该页面只显示最新的 **20** 条互动消息，包括用户发送和公众平台回复消息。如下图：



2. 用户昵称。点击旁边铅笔图标可以设置该用户的备注，方便对用户进行辨识。
3. 消息内容。
4. 消息发送时间。
5. 星标消息。点击可将消息设置为星标消息，标注成功后图标会变为橙色五角星，并且该消息将永久保存在星标消息栏中，适用于一些重要的用户消息保存。如下图



6. 回复消息。点击该图标会弹出快捷消息回复输入框，只能输入文字或者用"[愉快]"这类文字方式发送表情。如下图：



一对一的聊天页面消息数只显示最近 **20** 条，那么更多的消息如何看到呢？聪明的朋友应该可以猜到，对了，跟查询 **7** 天以前消息的方法差不多，只是页面 **URL** 里修改的参数是“**count**”，默认是“**count=20**”，可以改到比如 **100**。

### 三、微信公众平台用户管理页面

点击公众平台后台导航的“用户管理”，或者在欢迎页里点击新朋友数后面的“查看”就可以进入，该页面用于对关注了该公众账号用户的管理，点击用户头像与消息如下图：



1. 用户管理菜单。



菜单是根据用户分组来显示的，默认有未分组、黑名单、星标组三个分组。公众平台将不接收被拉入黑名单的用户消息，但是可以接收公众平台的消息，包括群发消息，这个规则非常奇葩，比如上次那个偷我文字的朋友，我虽然拉他进了黑名单，但是他还是可以关注我公众号接收我的群发内容.....

点击新建分组选项可以新建自定义的分组。



2. 用户分组。点击分组下拉框选择组别，然后点击放入就可以将用户进行分组，被分组的用户将消失在当前组别列表中。用户将出现在另外一个分组。



3. 修改备注。同实时消息介绍，备注修改后用户名称将由备注名+（用户昵称）组成

4. 批量分组。将需要分组的用户头像前面的 **check** 框选中，然后统一放到一个组别。

公众平台后台的用户管理没有用户搜索这个功能，因此想找个人是非常苦逼甚至不可能的事情，比如 **ZTalk** 已经快 **15000** 用户了，靠翻页找人怎么找？解决的方法也很简单，还是改 **URL** 参数，需要修改的参数是“**pagesize**”，默认是“**pagesize=10**”，咱们可以改大些，比如我直接改成 **14000**，那么所有用户都会在一个页面里出现，然后 **ctrl+f** 就可以查找了。

今天就先到这里了，下一期是微信公众平台素材管理

## 《微信公众平台入门到精通》Vol.3

2013-04-03



微信公众平台最常用到的就是群发消息，这种一对多、几乎百分百到达的传播方式，取代了短信群发，并且具有多媒体形态，还可以直接引导转化。因此编辑一条好的内容，挑选恰当的时候发送，既可以给用户带来有价值的信息，又有机会给公众账号带来新的关注和收益。本期将重点为大家讲解素材管理该怎么玩，群发消息需要注意些什么。

### 第三章 微信公众平台群发消息

#### 一、微信公众平台后台素材管理

点击公众平台后台导航的“素材管理”，或者在群发消息或者与用户聊天页面时，选择发送多媒体内容时快捷进入。这个页面主要是用来管理公众平台的图片、声音、视频和图文消息内容的。如下图：



1. 素材类型。微信公众平台可以群发的消息类型为文字、语音、图片、视频、图文消息，除了文字可以在群发消息时直接生成，其他四种类型都需要先在素材管理里生成原始素材。
2. 图文消息类型。图文消息有两种形式，一种是单图文消息，一种是多图文消息，适合推送单篇文章或者是多个文章列表。
3. 多图文消息。头条有大图展现，其余是标题配小图，灵活使用可以有很多玩法，如图例中就用多图文消息生成一个微信的欢迎菜单。
4. 单图文消息。主要由标题、封面、提要、时间组成。

## 二、新建单条图文消息

前面已经提过图文消息有两种形式，先来说单篇文章的，点击单图文消息图标后进入消息编辑器，如下图：



1. 图文消息预览区。可以让作者看到最终发到用户手机时的效果，但是封面图片的展示和最终效果可能会有一些差别。
2. 文章标题。文章标题力求抓人眼球，因为当消息推送到用户手机上，用户的对话框里最先显示的是文章标题。  
**ZTalk 秘籍** 长度尽量不要超过 13 个汉字，否则手机显示会折行影响显示效果。

3. **封面图片上传**。选图尽量与文章主题相关，同时要符合公众账号定位，少用卡通漫画等抽象形式。

**ZTalk 秘籍** 图片要选择横方图，即宽度大于高度的，或者用多张竖长图做拼图也可以。如果图片里有人物，要避免人的头部被切去，否则就会出现这种情况：



4. **文章摘要**。关于文章的一些简要说明，可以挑选文章的中某段比较精辟的话，长度不能超过 120 个汉字。

**ZTalk 秘籍** 摘要一定要重视，因为在没有 wifi 环境下用户图片显示比较慢，了解文章主要靠标题和摘要，一段好的摘要一定会提高用户的阅读率。

5. **编辑器工具栏**。从左到右的功能分别为：加粗、斜体、下划线、编号列表、符号列表、图片上传、样式清除、字体大小、字体颜色、字体背景。使用方式跟其他的后台编辑器一样。

6. **内容编辑框**。编辑的内容所见即所得，图片会自动缩放至合适宽度，鼠标右键会有菜单出现，可以调整段落左对齐、居中、右对齐，还可以插入表格。

**ZTalk 秘籍** 用户阅读图文消息时是在手机上，因此内容排版上要考虑手机阅读习惯，主要是 1、样式要简单，不要有太多颜色突出会分散注意力；2、文章配图不宜多，会受到网速影响无法打开，无特殊情况三张足矣；3、多分段多留白，不要出现大量文字堆积的情况；4、文字 500-1000 为宜，用户阅读都是利用碎片时间，太长了没这个耐心。



7. **原文链接**。这是一个微信群发消息里唯一可以加外链的地方，很多公众账号现在都是在微信的模板里放部分内容，全部内容必须点击原文链接才可以阅读，是个导流量的好方法。

**ZTalk 秘籍** 很多生活类或者电商类账号在通过原文链接卖货，这里要小心的是不要卖假货或者做垃圾广告，淘宝链接的话尽量和自己账号定位相符或者品牌相同，否则容易被封号。

8. **发送预览**。这个按钮非常有用，点击后会出现一个对话框，输入自己的微信号，点击确定后手机会收到一条图文消息，内容格式与群发消息时用户收到的一样，可以用来检查标题、图片等格式是否正确，检查内容排版上是否有问题。由于微信的缓存机制，有时候会发现预览的消息和之前一样，过段时间再发次预览就刷新了。

**ZTalk 秘籍** 如果是在微信后台上直接编辑文章，由于微信后台没有做自动保存功能，时不时点下发送预览可以用来保存文章。



9. **完成按钮**。保存文章退出编辑模式。

### 三、新建多条图文消息

多条图文消息顾名思义就是一次群发消息可以发送多个文章，编辑器也就和单条图文消息有所不同，如下图：



1. 头条图文消息预览。跟单条图文消息预览非常像，不同的是没有文章摘要区域，文章标题层叠在封面图片上。当鼠标移动到封面图片时会出现遮罩层，同时有一个笔的 icon 出现，点击后右边的编辑器会指向头条位置，可以进行头条消息编辑。



2. 列表图文消息预览。这里的形式就跟网站的文章列表一样，只显示缩略图和文章标题。当鼠标移动到该区域时，会出现遮罩层，除了笔的 icon 还有回收站的 icon，点击后分别执行编辑该条消息和删除该条消息。





3. **添加一条图文消息。**点击后在尾部添加一条新的图文消息，最多可添加 7 条，加上头条图文消息一共是 8 条。
4. **多条图文消息编辑器。**少了摘要输入框，其他与单条图文编辑器是一样的，在使用中同样要注意标题长度、图片尺寸、文字内容等。编辑器会根据左边的选择，自动切换消息内容同时箭头指向相应的图文消息。



### 三、新建多媒体消息

除了图文消息以外，公众平台还支持图片、语音和视频的群发，同时此类素材也可以在用户与公众平台之间的互动中使用，比如用户发送某个关键字时，公众平台自动回复一张图片、一段语音或者一段视频，增加趣味性，还可以检测用户的活跃度以及做一些用户兴趣的分析。

**新建图片素材。**这块很简单，就是一个图片管理模块，图文消息的封面图片会自动存放到这里，支持的图片类型为 bmp, png, jpeg, jpg, gif，图片大小限制：2M。

**ZTalk 秘籍** 1、GIF 动画图片是不能上传的。2、有时候会发生图片上传出现上传失败显示红色 **error** 字样的事情，用美图秀秀这类工具重新保存下就可以了。

**新建语音、视频素材。**语音和视频的上传管理和图片管理一样，也是直接可以将准备好的文件上传到后台。

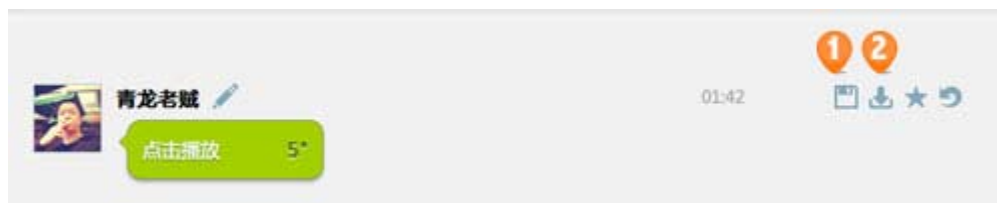
语音支持的格式为 mp3, wma, wav, amr，大小限制 5M，时间长度限制 60s。

视频支持的格式为 rm, rmvb, wmv, avi, mpg, mpeg, mp4，大小限制 20M。

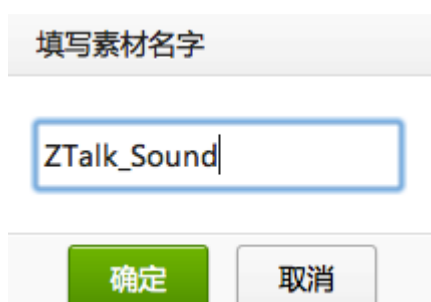
**ZTalk 秘籍** 制作语音素材看起来是一件很麻烦的事，特别是播报文字的话，需要麦克风录到电脑里，然后还得控制语音的长度和大小。但是青龙老贼给大家一个好办法，就是掏出你的手机把你要播报的文字直接通过微信说给公众平台听。



然后到公众平台后台会看到那条消息, 选中该条消息后会发现和纯文字消息不一样, 多了两个新图标。



1. 点击该按钮会出现一个填写素材名称的对话框，填上素材名称（注意名称要规范否则查找麻烦）点确定后这个语音消息就自动保存到语音素材库了。



不信到语音素材库看看。现在就不用担心什么大小和时间长度了，微信语音消息最多也只能发 60S，我很怀疑是官方故意设置的，然后看谁摸索出这个秘密。



2. 这个 icon 很明显了，就是下载文件到本地。

视频素材新建也可以这样玩，拿手机拍一段视频通过公众平台上传，然后保存到视频素材库，或者将已经拍好的视频放到照片目录，然后点击照片选择视频发送，微信会将自动将视频压缩。（有部分朋友说公众平台后台上传视频不好使可以试试这个）



## 四、群发消息

素材都准备好以后，剩下就是群发给关注公众平台的用户了，**在群发前最好再发送预览到自己手机上检查一遍！因为发出去以后就没有办法撤销，而且用户接收到的消息是存放在用户本地的，也就是在公众平台修改也无法刷新用户手里的消息包括分享，因此群发请谨慎！**

群发页面很简单，如下图：



1. 群发消息菜单。一个是新建群发消息，一个是查看已经群发的消息历史记录。
2. 群发消息对象。可以根据用户的分组、性别、地区来设置群发消息的用户范围。**注意：虽然可以选择用户范围群发，但是每次发送都将消耗一次群发机会，也就是如果一天只有一条群发的公众平台，发完以后就变成 0 次消息不能再给其他用户群发了。**
- 3.

**群发消息类型。**可选择文字、语音、图片、视频和图文消息类型。

**ZTalk 秘籍** 当选错群发内容时，可以点击选项卡重新选择。

4. **群发消息内容框**。文字可直接输入在框内，其他则根据不同的素材有不同的显示格式。

5. **剩余群发条数**。目前新注册的公众平台每天都只有一条群发消息的配额，以前注册的一些媒体账号有三条群发消息配额。跟是否认证无关，估计将来这个可以成为微信收费的一个渠道。

“已发送”主要是用来记录已经群发的消息，状态一项会显示正在群发和发送成功，现在群发消息经常会延迟很久，两三个小时都算正常的，因此在群发消息时要掌握好时间，不要凌晨吵醒用户。

根据 **ZTalk** 上次做的用户调查来看，用户比较喜欢的是晚上 **6 点到 10 点** 接收消息，因为这个时候处在下班路上或者在家休息，状态比较放松，适合阅读消息。



**ZTalk 秘籍** 可能很多朋友不知道“公众号手机助手”这个东西，它隐藏在导航栏的设置选项里。如图：

The image shows a web interface for setting up a 'Public Account Mobile Assistant' (公众号手机助手). On the left is a sidebar with '帐号信息' (Account Information) and '公众号手机助手' (Public Account Mobile Assistant). The main area has a title '公众号手机助手' and a subtitle '如何设置微信公众号手机助手?'. Below this is a paragraph explaining that a private WeChat account must be bound to the assistant (微信号mphelper) for messages to be sent to all subscribers. There is a text input field labeled '绑定到微信号:' and a green '确定' (Confirm) button.

设置
帐号信息
公众号手机助手

### 公众号手机助手

如何设置微信公众号手机助手?

绑定一个私人微信号后，通过该私人微信号向公众号助手（微信号mphelper，你需要添加它为好友）发送的消息将被视为此公众号向所有粉丝群发的消息，系统会自动进行群发，这同样会消耗群发限额。

绑定到微信号:

[确定](#)

“公众号手机助手”的使用很简单，将自己的微信号填到输入框内，然后微信搜索“mphelper”关注“公众号助手”，就完成了绑定。

绑定成功后主要可以实现两个功能，一个是群发消息成功后“公众号助手”会向绑定的微信发送一条成功发送的提示信息，二是如果出门在外没有电脑的时候，可以向“公众号助手”发送文字、语音、图片和视频，由它来完成群发消息的功能。

今天的文章很长.....本来想分开两篇写的，但是为了更完整还是放到了一起，写的很累，估计大家也会看得很累， anywhere！



## 《微信公众平台入门到精通》Vol.4

2013-04-10



微信公众平台的核心是与用户互动，在互动中为用户带去服务，从服务中产生价值，微信的运营提倡使用人工回复来和用户交流沟通，但是有很多信息其实是可以设置一些关键字让用户自助获取的，在这方面微信公众平台也提供了很强大的工具，其中编辑模式是针对没有开发能力的公众账号运营者的，今天就给大家带来编辑模式的介绍。

特别鸣谢：**BiaNews**（微信号：**bianews8**）提供自定义菜单模式体验！

### 第四章 微信公众平台编辑模式介绍

#### 一、编辑模式开启

点击公众平台后台导航的“高级功能”就可以进入，该页面有两个模式选择，分别为“编辑模式”和“开发模式”，如下图：





默认两个模式都是关闭状态，两种模式不能同时开启，我们这次先点击编辑模式区域进入编辑模式设置页面，如下图：



1. 模式总开关。点击开启后，下面两个区域才会有设置和启用按钮出现。

2. 自动回复开关。点击启用后可以使用用户关注自动回复、用户默认回复和用户关键字回复功能。
3. 自定义菜单开关。点击启用后可以使用自定义菜单，利用自定义菜单给用户更好体验。

## 二、自动回复介绍

自动回复是微信官方为没有开发能力的公众账号的强大工具，灵活使用自动回复功能不但可以引导用户进行自助信息获取，还可以提升用户使用体验，甚至可以完成一些复杂的交互功能，启用自动回复后点击“设置”按钮进入自动回复设置页面，如下图：



1. 自动回复类型。被添加自动回复是指用户关注公众账号时，公众账号自动发送的欢迎词；消息自动回复是指用户发送消息时关键字自动回复不能匹配时的默认回复；关键词自动回复是指用户发送消息符合设定的规则时自动回复相应内容；
2. 回复消息设置框。被添加自动回复和消息自动回复的消息设置框是一样的，只支持文字、语音、图片和视频回复，文字最多 300 个字符，英文汉字一样计算，语音、图片和视频从素材库选取。

## 三、关键词自动回复介绍



1. 添加新规则按钮。点击后出现新规则编辑表单。
2. 新规则名称。这个取名规则主要是便于记忆该规则设定的。
3. 关键字列表。当用户输入那些字符时自动返回消息，这里要注意的是记得将**全匹配**打钩，否则只要用户输入的一串文字中带有这个关键字时，就会发送消息。
4. 添加关键字按钮。点击后弹出关键字添加框。
5. 回复消息内容选择。有三种文字、文件、图文，文字不能超过**300**字，文件是指语音、图片和视频等多媒体内容，图文是指图文消息。
6. 回复消息内容有多个时，可以选择发送全部，这样用户会同时收到多个回复，否则用户会随机收到其中一个回复。

#### 四、自定义菜单介绍

自定义菜单目前官方还处在内测阶段，只有少量媒体类和企业类公众账号获得使用资格，ZTalk 没有申请到只有上别人的后台截图介绍了。

介绍前先说个丑事，为了演示我直接把 bianews 的菜单全删除了，本以为没预览和保存应该没事，结果发现删除就是直接没了，后来是对照手机上的菜单设置重新搞了一遍，这里也顺便给大家提个醒，删除菜单请谨慎.....自定义菜单的主设置页面如下：



1. 添加主菜单按钮，即公众平台最底下那一篇菜单，最多只能创建三个。  
排序可以对菜单重新排序，直接拖动菜单即可，如下图：

## 编辑

你可以创建最多 3 个一级菜单，每个一级菜单下可以创建最多 5 个二级菜单。编辑中的菜单不会马上被用户看到，可以放心调试。



2. 菜单动作设置框。自定义菜单目前只有两个动作，一个是点击显示子菜单，一个是点击发送消息。
3. 预览和发布按钮，发布即同步到公众账号手机端，用户在 24 小时内都将更新。

预览会弹出一个模拟手机框让编辑看效果，如下图：



点击添加菜单按钮，在弹出窗口输入菜单名，就可以添加一个新主菜单，如下图：



1. 主菜单设置。加号为添加子菜单，铅笔为修改主菜单名，废纸篓为删除主菜单，删除时连同下面的子菜单一并删除。

2. 主菜单动作设置。

在未添加任何子菜单时主菜单也可以添加发送消息的动作，即在公众账号点击主菜单时发送一个图文消息或者多媒体信息；添加了子菜单后会变成下图，即主菜单自动赋予了点击显示子菜单功能，无法再绑定发送消息功能：



点击主菜单的加号，在弹出窗口输入菜单名，就可以添加一个新的子菜单，如下图：



1. 子菜单选项。铅笔为修改菜单名，废纸篓为删除该子菜单，子菜单下无法再建子菜单，每个主菜单最多只能新建 5 个子菜单。
2. 子菜单动作设置。子菜单动作只能绑定发送消息，消息类型有文字、语音、图片、视频和图文消息。

公众平台的编辑模式介绍基本就是这些，比较简单，但越是简单的东西只要你肯用心思考就能玩出不同的东西，下一期为大家带来实例操作，由于大部分公众账号都没有自定义菜单，因此实例主要是用自动回复功能实现。



## 《微信公众平台入门到精通》Vol.5



实例为资讯应用，主要以首页、导航、列表、搜索四块组成，所有设置都是在公众平台后台的高级功能-》编辑模式-》自动回复页面下完成。

### 一、使用被关注自动回复制作首页

当用户第一次关注公众账号的时候，是一个公众账号最难能可贵的资源，如果用户无法在几秒钟之内得知公众账号的功能，他很有可能就取消关注。

你必须迅速地告诉用户为什么要在你的公众账号上花费时间。三秒内让用户知道你是做什么的，并且能够准确的指引用户他想要去的地方。太复杂的操作步骤，只会让用户觉得这个公众账号很没劲。

所以说好的欢迎首页既可以让用户更好的了解公众账号，又可以快速引导用户进行互动，因此设置被关注自动回复非常重要。

先说下实际操作，选择“被添加自动回复”，可以看到右边设置框可以选择文字、语音、图片、视频四种方式，推荐使用文字，语音不够直观，图片打开很慢，视频基本属于扯淡，但是不知道微信官方为什么没有图文消息栏，本来有这个的话可以用多图文消息做一个菜单出来。

如下图为简单设置的一个欢迎首页：





当用户关注该公众账号时，首先是感谢关注，然后是告诉用户这个公众账号是可以学公众平台使用的，最后告诉用户输入“跟我学”就可以进行互动。

## 秘籍

好的欢迎词可以给用户带来深刻印象：

普通欢迎词：“欢迎您关注 XXXX”；

卖萌欢迎词：“亲，你怎么才来呢？感谢关注 XXXX，亲会收到每天的问候和精选文章”；

幽默欢迎词：“我们灰常严肃地告诉你，你已经成为 XXXX 的一员，必须严守组织纪律，争当 XXXX 的优秀用户！”

文艺欢迎词：“唯有美食与爱不可辜负，亲爱的，听我聊聊厨房与爱吧！”

二逼欢迎词：此处省略 100 字……

简单的互动提示有利于激发用户马上体验功能，可以根据公众账号定位来设置，比如“跟我学”、“开始吧”、“试一试”等，或者用问候语，比如“Hi”、“Hello”、“你好”等，或者使用计

算机特定词“dir”、“ls”等。

加入一些表情符号，让文字活跃起来，带点情感。

分段中间要有空行，这样可以避免欢迎、引导文字挤在一堆，用户看不清楚。

## 二、用关键词自动回复制作一个导航

导航并不是每个资讯应用都需要的，特别是自媒体的公众账号，由于文章量较少也不分类，可以直接进入列表。导航设置可以参考自己网站的栏目设置。

规则名是用来辨识该规则的，一般可以直接用功能名称来命名。然后在左栏的添加关键字“跟我学”，并选择“已全匹配”，右栏输入回复内容，回复内容使用文字，当用户输入该关键字时就会回复右栏的内容。如下图：

新规则

收起

规则名：

导航

规则名最多60个字

关键字

跟我...

编辑

已全匹配

删除选中

添加关键字

回复

发送全部回复

文字(1) 文件(0) 图文(0)

公众平台教程导航，回复相应数字编码，了解更多：

1.公众平台入门篇

2.公众平台中级篇

3.公众平台高级篇

4.公众平台案例

编辑

删除选中

文字

文件

图文

三、用关键词自动回复制作一个列表

设置完导航后就可以做列表了，列表的关键词要根据前面导航的回复内容来设置，比如数字编码等于 1 的时候，就回复“公众平台入门篇”的文章列表，如下图：

规则1: 公众平台入门篇

收起

规则名：

公众平台入门篇

规则名最多60个字

关键字

1

编辑

已全匹配

删除选中

添加关键字

回复

发送全部回复

文字(1) 文件(0) 图文(0)

公众平台教程入门篇，回复相应数字编码查看教程：

11.微信公众平台注册、设置、登陆

12.微信公众平台后台介绍

13.微信公众平台群发消息

14.微信公众平台编辑模式介绍

编辑

删除选中

文字

文件

图文

保存

删除

秘籍

关键字命名规则建议首字母为导航栏数字编码，比如前面导航的“入门篇”数字编码是 1，那么该文章列表的数字编码第一个字母就设置为 1，后面为该列表的顺序。

由于文字版最多只能输入 300 个字，当该文章列表内容字符超过 300 个时，可以再设置“p2”的规则来翻页，如下图：



这样就可以实现翻页了，是不是很简单？

注意规则名如果有英文字母的时候，请将大小写字母的关键字都设置进去，特别是第一个为英文字母时，手机键盘一般会自动变为大写，比如上图这个翻页关键字是“p2”，我设置了两个“p2”和“P2”，这样用户不管输入那个都可以查询了。

#### 四、用关键词自动回复返回详细消息

用户可以通过微信公众账号查看文本、语音、图片、视频、图文消息等富媒体信息，列表设置完成后用户就可以根据每篇文章的数字编码来查看详细信息。

1、根据数字回复图文消息。先标注规则名，然后在左栏设置关键字“11”，然后点击右栏右下角的图文按钮，选择要发送的图文消息，保存即可。用户发送“11”时将会收到该图文消息。如下图：



2、根据数字回复文字消息。先标注规则名，然后在左栏设置关键字“12”，然后点击右栏右下角的文字按钮，在弹出的文本消息框里输入文字，保存即可。用户发送“12”时将会收到该文本消息。如下图：



3、根据数字回复语音、图片、视频消息。先标注规则名，然后在左栏设置关键字“12”，然后点击右栏右下角的文件按钮，在弹出素材管理页面选择相应素材，保存即可。用户输入“13”时将会随机收到其中一条内容，如下图：



## 秘籍

在这个例子里我放了三个素材，当用户输入关键字“13”的时候公众账号会随机发送一个给用户（测试发现第一个素材命中率特别高，超过 60%，不知道再多几个素材会不会有改变？）。那么如何一下子把三个素材一起发给用户呢？很简单，选中左上角的“发送全部回复”。应用场景：美女主播照片+语音播报……

## 五、用关键词自动回复做语义搜索

语义搜索是个很 NB 的技术，谷歌、百度这些大公司都在这个上面花了很多力气，语义搜索技术越强大，搜索结果就会越接近用户想要的内容，但实际上使用**关键字不全匹配**这个设置，咱们也可以在公众平台上实现一些简单的语义搜索。如下图：



根据文章的关键字，将一些有相同 TAG 的文章归类到一个新规则下，用这些 TAG 来做关键字，不选择全匹配，当用户发送的消息里包含有这些关键字时就会返回相应文章。如上图两篇文章都是有关“微信”和“教程”的，当用户发送的消息里含有微信或者教程时，就会收到这两篇文章。

**莫言：**文章页面不是可以加图片吗，图片放在自己服务器上，看请求次数就简单估算出 pv 了

**ZTalk：**这位朋友有心了！我测试过可行，不过一般人还是没法用，我到有个好 IDEA，有没有开发者做个类似网站计数器的东西？让公众账号可以自助生成图片（比如做成签名），然后拷贝到文章末尾，第三方平台可以帮助计数。谁要是做出来了说下哦，我帮推广！

**小道消息：**互联网行业里面，以个人名义运营的微信帐号，目前订阅量最大的是「小道消息」。骑自行车的也是互联网？

**ZTalk：**feng 急了……哈哈哈哈！！

**湛旭东：**微信加上如 safari 将 URL 添加至主屏幕的功能，将公众帐号或普通好友的图标加到主屏幕，可好？可能会有吗？

**奔跑的树叶：**老贼您好，请教个问题，目前我们的微信公众号(ebsorg)，支持在线通过“企业名称”，“网站名称”，“营业执照”，“法人”等字段查询，自动回复深圳企业执照信息，验证信息，接口数据库约 1 百万的数据量，经常查询超时接口没响应，想请问下有那些好的解决办法？ 十分感谢！

**ZTalk:** 这个解决方法还是要优化你们自己的后台，数据库优化、缓存优化、搜索语句优化等，1 百万量级的数据其实也不算太多吧，缓存、索引、分表查询这些搞搞呗。如果学的 12306 那基本可以歇了。

**顾里:** 想请教资深前辈，我每天的工作是做微博。还要顾及公司的威信平台。说好听的也许叫编辑。不好听的就是发微博的。。语言混乱了。我的未来在哪里。这样算是一份工作吗？没压力。都散漫的不行了。要怎么做到底

**ZTalk:** 编辑是当然是一份工作！好的微博微信运营很稀缺啊！关键是你把它当自己的工作还是自己的事业？你把用户当朋友还是陌生人？你是用心在工作还是只是用手在工作？

**吃肉的杨:** 前辈你好，微信搞抽奖活动该如何搞？有无案例分享一下？

**ZTalk:** 现场一般用摇一摇，和主持人一起摇手机，排在前 N 位的中奖；公众账号里都可以做，可以用 H5 页面来做转盘摇奖、刮刮卡这些。

**熊可:** 公众号设置了消息自动回复后，若与微信私人号聊天，会不停发送自动回复。请问老贼，可有两全法？

**ZTalk:** 没有，这个确实是一个很困扰人的问题。如果使用开发接口倒是可以避免这个问题，比如在 ZTalk 里输入“error”就可以关闭自动回复。

**ZTalk 推荐:** 微小说（微信号：weixinxiaoshuo），专注于推荐微信上好玩的，有趣的应用。由国内首部微信小说作者，某上市公司前社会化媒体总监：nbc 二当家的操刀。



ZTalk 每日一曲，输入“V”欣赏，历史欣赏请输入“VM”：

Klaus and Kinski 的《Susto de Muerte》，西班牙语的民谣调子

=====关于 ZTalk@青龙老贼=====

青龙老贼，音乐、咖啡和烟是我最好的伴侣，喜欢电影和阅读。

传媒梦工场战略发展总监、互联网从业者，移动互联网资深分析师。

创业家、钛媒体、虎嗅网、雷锋网、艾瑞、速途网、新浪科技、搜狐科技等科技媒体专栏作家。

微信搜索“ZTalk”就可以关注我

查看历史文章请在公众账号内输入“m”

## 《微信公众平台入门到精通》Vol.6

2013-04-15



先更正一个昨天的错误：关键字自动回复设置现在不区分大小写了，所以只设置一个 **p2** 或 **P2** 即可。感谢崔广斌、馋死你不偿命两位朋友的提醒。

今天带来公众账号自动回复操作案例，不过没有实际运用过，属于本人 YY 的产品逻辑，欢迎朋友们尝试使用，有问题的请及时反馈。

### 《外婆家》菜谱

很多餐馆现在都在使用 **ipad** 菜谱，这种菜谱的好处在于图文并茂，浏览方便，可以直观的展现菜品给顾客，同时维护菜品简单，省去很多重复印刷纸质菜谱的麻烦，个别复杂的菜谱还具备了在线下单、结算、付款等功能。

但是 **ipad** 菜谱价格较贵，一般连硬件加软件起码 6000-8000 一套，我看到过很多小商家聪明的把菜品做成照片，放在 **ipad** 图片浏览里让用户点单，也是个不错的 **IDEA**，不过 **ipad** 还是太贵。

现在智能手机那么普及，让顾客扫描二维码关注餐馆公众账号后，就可以在自己手机里看菜谱了，会不会很好玩？另外这些顾客也就成为你公众账号的用户了，然后.....不说了，自己想象去。

一、设置被添加自动回复，如下图：



二、设置消息自动回复，如下图：

**消息自动回复** 公众平台如何设置消息自动回复？



三、把所有菜品上传到公众账号：

这里有三种方案：

1. 每个菜品是一个图文消息，标题为菜品名称，封面图片为菜品图片，菜品介绍和价格放摘要，如下图：

## 茶香鸡

2013-04-15



价格：¥45

描述：招牌菜。——选用大小刚好，肥瘦均匀的三黄鸡，配上他家的秘方，放入烤箱烤制金黄色，取出装盘，浓郁的肉香飘来，看上去有点油油的，吃起来却一点儿也不油腻，肉香里还有淡淡的茶香，妙不可言。



2. 多图文消息展现，封面图文消息用来做菜品类别，添加六条菜品信息格式为菜品名称+价格，最后一条用来做翻页提示。如下图：

	
外婆菜	
开心土豆 ( 16元 )	
铁板虾 ( 36元 )	
酸辣汤 ( 16元 )	
鱼香茄子 ( 12元 )	
麻辣豆腐 ( 16元 )	
茶香鸡 ( 36元 )	
下一页请输入 "w2" , 返回请输 入 "菜谱"	

- 每个菜品做一条关键词自动回复，将菜品的图片、价格、描述分解成三个回复，根据关键字全部发送。如下图：

规则1: 茶香鸡

收起

规则名: 茶香鸡

规则名最多60个字

关键字

□ w11

编辑

✓ 已全匹配

回复

发送全部回复

文字(3) 文件(1) 图文(0)

□ Snip20130415\_76.png



□ 价格: ¥45

编辑

□ 食材: 鸡、茶叶、糖

编辑

□ 描述: 招牌菜。——选用大小刚好, 肥瘦均匀的三黄鸡, 配上他家的秘方, 放入烤箱烤制金黄色, 取出装盘, 浓郁的肉香飘来, 看上去有点油油的, 吃起来却一点儿也不油腻, 肉香里还有淡淡的茶香, 妙不可言。

编辑

第一种和第三种的好处是图片较大, 展现内容直观, 添加删除灵活, 并且第三种的介绍内容可以多一些, 第二种的好处是关键字设置较少, 菜单层级少, 公众平台关键字回复可设置最多条数只有 **200** 条, 每条规则下的回复最多 **5** 个, 如果菜品较多可能就不够用。

四、关键字自动回复做菜谱类目层级, 如下图:

新规则

收起

规则名: 菜谱

规则名最多60个字

关键字

□ 菜谱

编辑

✓ 已全匹配

回复

发送全部回复

文字(1) 文件(0) 图文(0)

□ 请输入字母编码浏览相应菜品:

lc冷菜

hb杭帮菜

sk烧烤

wp外婆烧

ml麻辣

rc热菜

tp甜品

编辑

采用第一种和第三种菜品上传方案的, 要做 **2** 级菜单, 如下图:

新规则

收起

规则名：

冷菜

规则名最多60个字

关键字

□ lc

编辑

全匹配

回复

□ 发送全部回复

文字(1)

文件(0)

图文(0)

L1.白芷豆腐

L3.我爱凤爪

L4.琥珀桃浆

L5.五香花生

L6.手拍黄瓜

L7.口味凉皮

L8.爽口海带结

L9.手剥皮蛋

L10.蜂窝豆干

L11.长寿菜

L12.有机泡菜

L13.爽口拌海草

L14.吉祥三宝

L15.蓝莓山药

L16.香拌木耳

L17.红柱香笋

如果一个菜品类别下数量比较多，单条文字消息字数超过的话，也可以设置多条文字消息，选择发送全部回复。

采用第二种菜品上传方案的，直接用关键字自动回复添加多图文消息即可，注意翻页关键字的设置，如下图：

规则1: 外婆菜第一页

收起

规则名：

外婆菜第一页

规则名最多60个字

关键字

□ wp

编辑

全匹配

回复

□ 发送全部回复

文字(0)

文件(0)

图文(1)

□



[ 图文消息 ]

a





简单四步就能完成一个移动端菜谱，让顾客方便直观的查看菜品，如果有开发能力的还可以将下单、会员管理等加入，还可以根据用户浏览数据来了解菜品受欢迎情况。

本文网络独家授权，任何媒体、机构不得转发！

### -----ZTalk 的互动环节-----

十八：问个有点意识形态的问题。您觉得当下的“自媒体”和以前的科技博客区别在哪儿？

**ZTalk:** 区别很多，传播方式，渠道掌控，内容质量等，不过人还是那类人，愿意把思想和价值通过文字传播给有需要的人。

**loy:** 恭喜博主的获奖，您很勤奋，也很无私 用关键词自动回复，这点个人建议 " 简体字、繁体字 " 最好都能考虑到。因为手机输入法两种字体均有:-)

**ZTalk:** 这个点还真没想到过，繁体字确实有可能。

**hw:** 老贼，非常感谢，正是受您的公众号点播音乐功能启发，我们做了一个微信互动猜歌名小游戏。今天已经上线，搜索公众号

"GuessSong"或者"我爱猜歌"就能找到。希望能帮忙提提宝贵意见，如果觉得做得还可以，不知道是否方便帮我们在您的每日推送中稍稍推荐一下？非常感谢！

**ZTalk:** Great idea, 我体验了非常不错，大家快去看看！！回头让玩转微信好好报道下，近期看到比较靠谱的微信应用。

大左：我在想我这种推荐了 20 个以上能不能混个 VIP 粉丝。

**ZTalk:** 非常感谢朋友的信任和推荐，你们每一个都是 ZTalk 的 VIP，每一个都是我的知己，只有你们懂我！

**ZTalk** 每日一曲，输入“V”欣赏，历史欣赏请输入“VM”：

Priscilla Ahn 的《City Lights》，在城市灯光下独自起舞

Q：如何查看文章

A：公众账号内回复“m”就可以查看所有文章目录

Q：如何分享给朋友

A：点击右上角在弹出菜单中选择发送给朋友或者分享到朋友圈



ZTalk@青龙老贼

Q：如何订阅 ZTalk

A：微信号搜索“z\_talk”，或者查找公众号“青龙老贼”，还可以点击右上角在弹出菜单中选择查看公众账号。



当一些人在怀疑自媒体人价值的时候  
我们是WeMedia

当一些人在跟随自媒体人脚步的时候  
我们是WeMedia

当一些人在选择自媒体人平台的时候  
我们是WeMedia

WeMedia是微信自媒体第一联盟

目前联盟公众账号传播覆盖已超过1000,000人

## 《微信公众平台入门到精通》番外篇

2013-04-04



昨天有个朋友在公众账号上问我“普通公众号和会议公众号何区别”，我没有办法马上回复，原因很简单就是我不知道，相信这个问题大部分玩微信的人都不知道，包括我这个所谓的微信专家，真的很惭愧，所以昨晚马上注册了一个试验了一把，今天给大家来个图文普及。

## 番外篇 微信会议公众账号详解

### 一、微信会议公众账号注册

会议公众账号注册流程和普通公众账号基本一样，只是在最后一步不一样。如下图：

帐号名称

(2-16个字) 名称一经设置无法更改。如果有同名的微博认证帐号，粉丝满500后可进行自助认证。

功能介绍

(4-120个字) 介绍此公众帐号功能与特色，该信息可以更改，将展示给订阅本帐号的用户。

运营地区

语言

类型

1

2

提醒：会议号是有一定时间限制的公众帐号，过期后将无法登录使用。

选择公众平台类型为公众会议账号时，会出现“提醒：会议号是有一定时间期限的公众账号，过期后将无法登陆使用”，这个时间期限目前固定为 1 个月时间，也就是说一个月以后这个账号就不能再用了。

## 二、微信会议公众账号设置

注册成功后进入账号设置页面，比普通的公众账号设置多了一个菜单“会议号设置”，如下图：



1. 会议号设置菜单项。
2. 会议号有效时间。目前固定是一个月，我是 4 月 3 日注册的，那么到了 5 月 3 日，这个账号就失效了无法再登陆了（是否能关注不知道）。
3. 默认关注权限。默认是任何人都可以关注，这样的设置条件下用户可以直接搜索公众账号然后关注，流程跟普通公众账号一样。
4. 需要验证关注。可以设置成需要通过验证才可以关注，还可以设置验证问题，但是我在试验时发现这个验证问题并没有显示在验证中，但是关注流程确实改变了。
5. 参与人相互可见。这个设置非常有趣，选中后所有关注该账号的人可以互相查看。

### 三、微信会议公众账号验证权限体验

当会议公众账号设置成需要验证后，用户关注公众账号时会提示需要发送验证申请，如下图：



用户输入验证后，会议公众账号的后台会收到一条验证申请的消息，如下图：





1. 在用户管理页面会多一个新的菜单选项“关注请求”，申请验证的消息列表会在这个菜单项下。
2. 消息操作不再是回复，而是接受和忽略，当选择接受时用户手机端就会自动关注该会议公众账号了。

#### 四、微信会议公众账号参与者互相可见体验

当会议公众账号设置成参与者互相可见时，公众账号的资料卡上会多一个选项“与会者”，如下图：



点击“与会者”选项后，会跳转到关注该账号的用户列表，即所有参会者。在这个列表里可以查看到其他用户的信息，还可以添加关注。如下图：



#### 四、微信会议公众账号可以做什么

去年很多大型会议在会议筹备和召开期间，开始使用 APP 或者是 H5 页面等方式来配合会议进程，参会者可以下载大会 APP 或者打开手机网页，进行会议日程查询、会议资料下载，还可以提交个人信息用来交换名片，据说还有创业团队靠给 IBM 这类国际大公司提供会议 app 赚了不少钱。

现在会议公众账号完全可以取代这些 **APP**，大会筹办者只需要开一个会议公众账号，然后将会议相关资料通过公众账号发布，设置一些关键词让参会者自助查询，还可以使用验证方式审核参会者的资格，而参会者也可以通过“与会者”选项来认识新朋友。

会议公众账号无论是会议主办方使用，还是会议参与者使用，都要比以前那些 **APP** 下载或者网页方式都要简单实用，但是为什么那么多人没有使用或者说知者甚少呢？从我来看，一是微信官方没有很好的宣传引导，二是研究微信的同学大量注意力都放在了普通公众账号应用上，结果让如此 **NB** 的一个产品白白躺在那里。

当然这个产品还有需要改进的：

- 1、参会报名可能要做些小开发或者通过人工来导出报名表，
- 2、会议公众账号有效期是 1 个月，过了一个月以后后台就无法登陆，非常不合理，如果是一些定期举办会议难道每次都要申请一个新的公众账号？
- 3、现在申请公众账号都要提交身份证，一个身份证只有 5 个账号可以开，这个非常不方便，会议多的公司岂不是要去买身份证号了？

## 《微信公众平台入门到精通》Vol.7

2013-04-27



又是出差忙碌的一周，今天刚到南京帮公司的活动撑场子，应酬到凌晨才回到酒店，不想再找借口，还是熬夜写个吧，先把几个事情说下。

1、公益培训收费是让大家认真学习，免费的东西很多人不懂得珍惜，我们并不做商业化探索，也不靠这个赚钱，更不缺钱，我个人有在参与收费的企业内部培训，但是这个公益培训各位真想多了，如果持怀疑态度的可以退订 ZTalk，因为你并不了解青龙老贼这个人，也不信任 ZTalk，再忍受我天天一个人的表演多没意思。

2、雅安爱心捐款已经圆满结束，非常感谢大家的热心参与，WeMedia 联盟共收到 51316 条回复，善款将捐献给壹基金，我们这几天就会与对方联系，即时将公示捐款凭证。

3、我不是天使投资人，这次南京路演活动是公司（传媒梦工场）组织的，我是以点评嘉宾的身份参与，有朋友想要分享路演项目，昨天只有三个，一个做公务员考试网，一个做微信营销全案策划，一个做大学生兼职平台，今天有 10 个，我会分享给大家。

## 第七章 公众平台开发模式介绍

从今天开始教程进入高级阶段，请大家认真、专心，还有执着，我会很耐心的在后台回复每个人的提问，但是如果真觉得学不下去的可以考虑用一些现成的第三方平台，但是肯定不如自己开发来的个性化。

### 一、开发模式简介

点击公众平台后台导航的“高级功能”就可以进入，该页面有两个模式选择，分别为“编辑模式”和“开发模式”，如下图：



默认两个模式都是关闭状态，两种模式不能同时开启，点击开发模式区域进入开发模式设置页面，如下图：



1. 模式总开关。只有“成为开发者”后才可以开启，这里要注意的是由于编辑模式和开发模式不能同时开启，使用开发模式前应先关闭编辑模式。
2. api 接口文档。点击后可以查看目前所有开放的接口使用详细说明，包括开发用的 SDK 包。（**SDK, Software Development Kit** 的缩写，中文即“软件开发工具包”。广义上指辅助开发某一类软件的相关文档、范例和工具的集合。）
3. 成为开发者。点击后会出现接口配置页面，用来绑定自己服务器上的程序接口与微信接口通讯，具体的操作将在第九章介绍。

## 二、api 接口文档简介

首先来了解下微信开发模式有那些接口，清楚微信接口能做什么不能做什么有利于自己的产品设计，可以考虑自己的产品如何结合如何规划。

由于微信本身是一个通讯工具，它所有的 api 接口都是基于一问一答的通讯机制，因此它的接口文档分为消息推送（用户在公众平台上行发送 的消息）和消息回复（响应用户上行消息反馈下发回复的消息），本教程除了目前已经公布

的接口外，还将剧透一部分将要开放的接口，比较特殊的消息下发接口属于最高级别接口，大部分开发者是拿不到的，所以不介绍了。

## 1、

此类消息即用户向公众账号输入文本发送消息，是公众平台接收到最多的消息，也是最重要的搜索、互动请求，因为语音识别还不成熟，图片无法识别，地理位置用的场景也不多。

接口获取到的消息格式：

```
<xml>
<ToUserName><![CDATA[toUser]]></ToUserName>
<FromUserName><![CDATA[fromUser]]></FromUserName>
<CreateTime>1348831860</CreateTime>
<MsgType><![CDATA[text]]></MsgType>
<Content><![CDATA[this is a test]]></Content>
<MsgId>1234567890123456</MsgId>
</xml>
```

这里先说下 XML 数据格式，它不仅是一种非常适合 web 传输的格式，还具有强大的数据索引、排序、查找、相关一致性等特点，有些网站甚至拿 XML 来做数据库，比如射手网，它结构非常简单，看起来有点像 html，可读性比较强，微信开发接口的消息传递主要在用这个。

有些人会觉得很头大，其实不理解也没关系，只要搞清楚它每次传输给你的有那些数据，你怎么获取这些数据就可以了。



文本消息的数据字段：

**ToUserName:** 开发者微信号（公众账号 ID）

**FromUserName:** 发送方帐号（微信用户的 OPENID，每个微信用户与每个公众账号是唯一的，不同公众账号同个用户 OPENID 是不同的）

**CreateTime:** 消息创建时间

**MsgType:** 消息类型（文本消息为 text）

**Content:** 消息内容

**MsgId:** 消息的 ID 号

2、

用户使用图片上传发送消息，应用场景：账号注册、随手拍活动征集、现场微信照片墙。

接口获取到的消息格式：

```
<xml>
<ToUserName><![CDATA[toUser]]></ToUserName>
<FromUserName><![CDATA[fromUser]]></FromUserName>
<CreateTime>1348831860</CreateTime>
<MsgType><![CDATA[image]]></MsgType>
<PicUrl><![CDATA[this is a url]]></PicUrl>
<MsgId>1234567890123456</MsgId>
</xml>
```

图片消息与文本消息数据不同：

**MsgType:** 消息类型为 image

**PicUrl:** 图片链接，该图片保存在微信的服务器上，可以通过远程抓取图片程序保存到本地。

### 3、

微信将 **LBS** 当作一个交互信息进行传递，比较常见的微信用户之间发送地理位置告知目的地所在，公众号里发送地理位置搜索周边人、商家、信息等。

接口获取到的消息格式：

```
<xml>
<ToUserName><![CDATA[toUser]]></ToUserName>
<FromUserName><![CDATA[fromUser]]></FromUserName>
<CreateTime>1351776360</CreateTime>
<MsgType><![CDATA[location]]></MsgType>
<Location_X>23.134521</Location_X>
<Location_Y>113.358803</Location_Y>
<Scale>20</Scale>
<Label><![CDATA[位置信息]]></Label>
<MsgId>1234567890123456</MsgId>
</xml>
```

地理位置消息与文本消息数据不同：

**MsgType:** 消息类型为 location

**Location\_X、Location\_Y、Scale:** 地理位置的经纬度和地图缩放比例，用于第三方地图接口的应用，搜索周边需要用到

**Label:** 地理位置文字信息，如某某城市某某路某某号，如果以城市为单位查询直接拿这个就可以。

#### 4、链接消息

微收藏、印象笔记等用来保存微信里图文消息的应用就是使用的这个接口，用户将链接发送到该公众账号，公众账号通过链接抓取网页内容然后收藏保存。

接口获取到的消息格式：

```
<xml>
<ToUserName><![CDATA[toUser]]></ToUserName>
<FromUserName><![CDATA[fromUser]]></FromUserName>
<CreateTime>1351776360</CreateTime>
<MsgType><![CDATA[link]]></MsgType>
<Title><![CDATA[公众平台官网链接]]></Title>
<Description><![CDATA[公众平台官网链接]]></Description>
<Url><![CDATA[url]]></Url>
<MsgId>1234567890123456</MsgId>
</xml>
```

链接消息与文本消息数据不同：

**MsgType:** 消息类型为 link

**Title:** 消息标题

**Description:** 消息描述

**Url:** 链接地址

## 5、语音消息（未开放）

这个消息接口并不在公开文档上，但事实上有不少微信应用已经在使用，比如路况电台，不全面开放的原因我个人猜测是因为微信自己的语音识别还不理想，未来肯定会开放，想象一下对公众账号说句话后台就能给反馈信息多方便的功能。

接口获取到的消息格式：

```
<xml>
<ToUserName><![CDATA[toUser]]></ToUserName>
<FromUserName><![CDATA[fromUser]]></FromUserName>
<CreateTime>1348831860</CreateTime>
<MsgType><![CDATA[voice]]></MsgType>
<MediaId><![CDATA[media_id]]></MediaId>
<Format><![CDATA[Format]]></Format>
<MsgId>1234567890123456</MsgId>
</xml>
```

语音消息与文本消息数据不同：

**MsgType:** 消息类型为 voice

**MediaId:** 语音消息媒体 id，可以调用相应接口获取内容

**Format:** 语音类型，如 amr，speex，wav 等，目前是使用的 speex。

## 6、视频消息（未开放）

这个消息接口也不在公开文档上，结构跟语音消息差不多。

接口获取到的消息格式：

```
<xml>
  <ToUserName> <![CDATA[toUser]]> </ToUserName>
  <FromUserName> <![CDATA[fromUser]]> </FromUserName>
  <CreateTime>1357290913</CreateTime>
  <MsgType> <![CDATA[video]]> </MsgType>
  <MediaId> <![CDATA[media_id]]> </MediaId>
  <ThumbMediaId> <![CDATA[thumb_media_id]]> </ThumbMediaId>
  <MsgId>1234567890123456</MsgId>
</xml>
```

视频消息与文本消息数据不同：

**MsgType:** 消息类型为 video

**MediaId:** 视频消息媒体 id，可以调用相应接口获取内容

**ThumbMediaId:** 视频消息缩略图 id，可以调用相应接口获取内容

## 7、事件推送

这个接口很重要，目前只开放了用户关注、用户退订，还有自定义菜单按钮相应的功能，部分测试的有用户打开公众号提醒和自动发送地理位置。也就是用户不发送任何内容也可以捕捉用户行为做出回复。

接口获取到的消息格式：

```
<xml>
<ToUserName><![CDATA[toUser]]></ToUserName>
<FromUserName><![CDATA[FromUser]]></FromUserName>
<CreateTime>123456789</CreateTime>
<MsgType><![CDATA[event]]></MsgType>
<Event><![CDATA[EVENT]]></Event>
<EventKey><![CDATA[EVENTKEY]]></EventKey>
</xml>
```

事件推送与文本消息数据不同：

**MsgType:** 消息类型为 event

**Event:** 事件内容，subscribe(订阅)、unsubscribe(取消订阅)、CLICK(自定义菜单点击事件)

**EventKey:** 当用户事件为点击自定义菜单时，返回事件 KEY 值，与自定义菜单接口中 KEY 值对应。

很多人在问如何知道用户退订的，其实就是这个消息接口，另外 3 月 26 日后将用户关注消息也从文本消息改为了事件推送。

消息回复需要注意的一点是从接收用户发送的消息，到服务器响应返回消息，有效时间只有 5 秒钟，超时的话用户是无法接收到回复消息的。

## 1、

文字阅读要比其他方式的阅读成本低，所以能用文本的地方咱们就尽量用文本。

接口下发的消息格式：

```
<xml>
<ToUserName>toUser]]&gt;&lt;/ToUserName&gt;
&lt;FromUserName&gt;<![CDATA[fromUser]]&gt;&lt;/FromUserName&gt;
&lt;CreateTime&gt;12345678&lt;/CreateTime&gt;
&lt;MsgType&gt;<![CDATA[text]]&gt;&lt;/MsgType&gt;
&lt;Content&gt;<![CDATA[content]]&gt;&lt;/Content&gt;
&lt;FuncFlag&gt;0&lt;/FuncFlag&gt;
&lt;/xml&gt;</pre></div><div data-bbox="144 538 382 557" data-label="Text"><p>回复文本消息的数据字段：</p></div><div data-bbox="144 613 847 654" data-label="Text"><p><b>ToUserName:</b> 发送方帐号（微信用户的 OPENID，每个微信用户与每个公众账号是唯一的，不同公众账号同个用户 OPENID 是不同的）</p></div><div data-bbox="144 711 581 730" data-label="Text"><p><b>FromUserName:</b> 开发者微信号（公众账号 ID）</p></div><div data-bbox="144 787 444 806" data-label="Text"><p><b>CreateTime:</b> 回复消息创建时间</p></div><div data-bbox="144 862 551 883" data-label="Text"><p><b>MsgType:</b> 回复消息类型（文本消息为 text）</p></div>
```



**Content:** 回复消息内容（长度不超过 2048 字节）

**FuncFlag:** 设置为 1 的时候，自动星标刚才接收到的消息，适合活动统计使用

## 2、（未开放）

这是个未公开的接口，现在使用开发模式最苦逼的就是没法回复图片，还不如编辑模式，啥都能回复！

接口下发的消息格式：

```
<xml>
<ToUserName><![CDATA[toUser]]></ToUserName>
<FromUserName><![CDATA[fromUser]]></FromUserName>
<CreateTime>12345678</CreateTime>
<MsgType><![CDATA[image]]></MsgType>
<Image>
<MediaId><![CDATA[media_id]]></MediaId>
</Image>
<FuncFlag>0</FuncFlag>
</xml>
```

与回复文本消息的数据字段的不同：

**MsgType:** 回复消息类型为 image

**MediaId:** 图片的 id，需要通过其他接口上传图片获得

上传图片限制：1MB，支持 JPG 格式

### 3、（未开放）

这是个未公开的接口，语音消息展现形式跟音乐消息还是不一样的，也不会出现某些安卓版微信无法播放音乐消息的情况。

接口下发的消息格式：

```
<xml>
  <ToUserName>toUser]]&gt; &lt;/ToUserName&gt;
  &lt;FromUserName&gt;<![CDATA[fromUser]]&gt; &lt;/FromUserName&gt;
  &lt;CreateTime&gt;12345678&lt;/CreateTime&gt;
  &lt;MsgType&gt;<![CDATA[voice]]&gt; &lt;/MsgType&gt;
  &lt;Voice&gt;
    &lt;MediaId&gt;<![CDATA[media_id]]&gt; &lt;/MediaId&gt;
  &lt;/Voice&gt;
  &lt;FuncFlag&gt;0&lt;/FuncFlag&gt;
&lt;/xml&gt;</pre></div><div data-bbox="144 694 464 713" data-label="Text"><p>与回复文本消息的数据字段的不同：</p></div><div data-bbox="144 770 454 791" data-label="Text"><p><b>MsgType:</b> 回复消息类型为 voice</p></div><div data-bbox="144 846 638 865" data-label="Text"><p><b>MediaId:</b> 语音的 id，需要通过其他接口上传语音获得</p></div>
```

上传语音限制：1MB，播放长度不超过 60s，支持 AMR 格式

#### 4、（未开放）

这是个未公开的接口，和图片消息差不多。

接口下发的消息格式：

```
<xml>
  <ToUserName><![CDATA[toUser]]></ToUserName>
  <FromUserName><![CDATA[fromUser]]></FromUserName>
  <CreateTime>12345678</CreateTime>
  <MsgType><![CDATA[video]]></MsgType>
  <Video>
    <MediaId><![CDATA[media_id]]></MediaId>
    <ThumbMediaId><![CDATA[thumb_media_id]]></ThumbMediaId>
  </Video>
  <FuncFlag>0</FuncFlag>
</xml>
```

与回复文本消息的数据字段的不同：

**MsgType:** 回复消息类型为 video

**MediaId:** 视频的 id，需要通过其他接口上传视频获得

ThumbMediaId: 缩略图的 id, 通过上传图片 (80\*80) 得到 id

视频限制: 10MB, 支持 MP4 格式

缩略图限制: 64KB, 支持 JPG 格式

## 5、

很多人问我为什么可以播放音乐, 没什么秘密, 就是用这个接口了。

接口下发的消息格式:

```
<xml>
<ToUserName><![CDATA[toUser]]></ToUserName>
<FromUserName><![CDATA[fromUser]]></FromUserName>
<CreateTime>12345678</CreateTime>
<MsgType><![CDATA[music]]></MsgType>
<Music>
<Title><![CDATA[TITLE]]></Title>
<Description><![CDATA[DESCRIPTION]]></Description>
<MusicUrl><![CDATA[MUSIC_Url]]></MusicUrl>
<HQMusicUrl><![CDATA[HQ_MUSIC_Url]]></HQMusicUrl>
</Music>
<FuncFlag>0</FuncFlag>
</xml>
```

与回复文本消息的数据字段的不同:

**MsgType:** 回复消息类型为 music

**Title:** 音乐标题，我经常是放歌名

**Description:** 音乐描述，我经常是放歌手名字

**MusicUrl:** 音乐链接，支持 MP3、WMV、AAC 等在线音乐格式

**HQMusicUrl:** 高质量音乐链接，WIFI 环境优先使用该链接播放音乐，可以和上面的一样。

**6、**

开发模式里最喜欢的就是这个了，比如可以用多图文消息做个欢迎菜单，可以做图文历史文章查询。

接口下发的消息格式：

```
<xml>
<ToUserName><![CDATA[toUser]]></ToUserName>
<FromUserName><![CDATA[fromUser]]></FromUserName>
<CreateTime>12345678</CreateTime>
<MsgType><![CDATA[news]]></MsgType>
<ArticleCount>2</ArticleCount>
<Articles>
<item>
<Title><![CDATA[title1]]></Title>
<Description><![CDATA[description1]]></Description>
<PicUrl><![CDATA[picurl]]></PicUrl>
<Url><![CDATA[url]]></Url>
</item>
<item>
<Title><![CDATA[title]]></Title>
<Description><![CDATA[description]]></Description>
<PicUrl><![CDATA[picurl]]></PicUrl>
<Url><![CDATA[url]]></Url>
</item>
</Articles>
<FuncFlag>1</FuncFlag>
</xml>
```

与回复文本消息的数据字段的不同：

**MsgType:** 回复消息类型为 news

**ArticleCount:** 图文消息个数，限制为 10 条以内，这个必须准确，有多少条图文消息就得填写多少，不然会出错。

**Articles:** 多条图文消息信息，默认第一个 item 为大图，每个 item 的结构完全一样，每个 item 表示一条图文消息。

**Title:** 图文消息标题

**Description:** 图文消息描述

**PicUrl:** 图片链接，支持 JPG、PNG 格式，较好的效果为大图 640\*320，小图 80\*80。

**Url:** 点击图文消息跳转链接，这个链接可以是自己网站链接，但是记得申请白名单，不然出现警告提示用户一般不会点击进去。

这些接口都未对外开放，看看就行，如果在写教程中途官方开放了，我也会写进。

1、**（未开放）**

公众平台获取关注用户的信息。

接口请求地址：

<https://api.weixin.qq.com/cgi-bin/user/info>

接口返回信息：

**subscribe:** 用户是否订阅该公众号标识，值为 0 时，拉取不到其余信息

**openid:** 用户的标识，对当前公众号唯一

**nickname:** 用户的昵称

**sex:** 用户的性别，值等于 1 时为男性，值等于 2 时为女性

**city:** 用户所在城市

**language:** 用户的语言，简体中文为 zh\_CN

## 2、（未开放）

上传图片、语音、视频等文件，获取 media\_id，media\_id 可复用。

接口请求地址：

<http://api.weixin.qq.com/cgi-bin/media/upload>

接口返回信息：

**type:** 媒体文件类型，分别有图片（image）、语音（voice）、视频（video）和缩略图（thumb）

**media\_id:** 媒体文件上传后获取的唯一标识

**created\_at:** 媒体文件上传时间



### 3、（未开放）

通过 `media_id` 获取图片、语音、视频等文件。

接口请求地址：

`http://api.weixin.qq.com/cgi-bin/media/get`

### 4、（未开放）

拉取公众账户关注用户列表。

接口请求地址：

`https://api.weixin.qq.com/cgi-bin/user/get`

接口返回信息：

**total:** 关注该公众账号的总用户数

**count:** 拉取的 **OPENID** 个数，最大值为 10000

**data:** 列表数据，**OPENID** 的列表

**next\_openid:** 下一次拉取的关注用户的偏移值

---

关于微信开发模式的初步介绍到此为止，这个文章是开发接口的速查手册，后面实际教程中会经常用到，请记得用微收藏保存起来！明天先讲新浪的云计算平台注册和使用，工欲善其事必先利其器！

## 《微信公众平台入门到精通》Vol.8

2013-05-03



先说个题外话，昨天一位 ZTalk 的朋友留言让我非常高兴：

张 J：自从看了您老人家的那个把公众平台改成菜谱，深受启发，跑了周围几家不大不小的餐馆，帮他们弄了一下，好过出去打工了。。

估计这位朋友靠这个赚了点小钱，在我看来很简单的东西却能帮别人创造价值，突然让我觉得自己在做一件很有意义的事情，好吧，我是《士兵突击》的脑残粉，所以我会继续努力好好活，好好活就是做很多有意义的事情。

-----正文开始-----

由于 ZTalk 一直在用 SAE，感觉还算稳定，其他的如 BAE 玩了下但不熟练，也不知道后续收费情况，腾讯自己的云计算平台一直没出来，所以还是决定用 SAE 来给大家讲课吧，虽然 X 浪没给我一分广告费。

## 第八章 新浪云计算平台注册和使用

以前想做个网站很麻烦，先得花钱租赁虚拟服务器空间或者买服务器托管，然后还得懂服务器配置，平时还得搞下服务器维护什么的，现在有云计算平台就方便了，对于一些非企业用户来说简单几步注册就可以自己建站，成本低廉效率还高，而且开发非常自主灵活。

### 一、新浪云计算平台注册

新浪云计算平台的英文是 **Sina App Engine**，简称 **SAE**，是国内首个公有云平台，是一个简单高效的分布式 **Web** 服务开发、运行平台，应该是目前国内最多开发者使用的云计算平台。

在浏览器中上输入网址 <http://sae.sina.com.cn/>，就进入 **SAE** 的首页：



点击右上角的注册链接进入注册页面，由于是新浪自己的产品，所以 **SAE** 账号系统使用了新浪微博的账号授权，大家应该很熟悉这个界面了，而且估计都有用新浪微博绑定授权注册的经历，我就不展开说了。如下图：



输入新浪微博账号和密码点击登陆后会跳转到授权页面，点击授权就完成 SAE 的第一步注册：



如果非常不巧你竟然连微博账号都没有的话，不着急，看到**第一张图数字 1** 标示的那个地方没，有一个注册链接，点击后进入新浪微博注册+SAE 授权页面，填写信息后即可完成 SAE 注册。如下图：

注册新浪微博，并授权 SAE 新浪云计算平台 访问你的账号





邮箱:

没有邮箱? [用手机注册](#)

设置密码:

昵称:

验证码:  

[立即注册](#)

[新浪微博服务使用协议](#)  
[新浪微博个人信息保护政策](#)  
[全国人大常委会关于加强网络信息保护的决定](#)

注册成功后会跳转到安全设置页面，这里要填写的东西比较多，如下图：

请如实填写以下信息：

微博帐号 蓝色的Z先生

如果您开通SAE的微博帐号不是 蓝色的Z先生，请点此  
退出微博后重新注册SAE

微博账号一经绑定后，将无法修改，请牢记绑定微博账号  
账户名和密码，并确保其安全性，如丢失将无法登陆  
SAE。

真实姓名 \* 蓝色的Z先生

安全邮箱 \*

安全密码 \*

密码确认 \*

绑定手机 \*

一个手机号只能注册一个帐号

电话号码

验证码 \*

请输入下图中的数字或者字母



动态密码 ☐ 启用微盾动态密码

1. 账号注册提醒，如果不想用这个微博账号注册可以反悔，否则一旦绑定后就不能修改。
2. 真实姓名，默认用的是微博昵称，建议使用真名，方便后期认证取回等。

3. 安全邮箱，使用 SAE 内部服务时的通行证账号，比如数据库，可以用新浪微博的账号名，当然为安全考虑你也可以用一个其他的邮箱做账号。
4. 安全密码，配合安全邮箱的密码，可以和微博的一样，同样的为了安全也可以设一个不一样的。安全邮箱和安全密码非常重要，请勿遗忘和泄露！
5. 绑定手机，用来取回密码和验证。每个手机号码只能绑定一次。

安全信息设置完以后进入手机验证身份确认步骤，SAE 的校验码很复杂，数字加字符的，可见安全方面要求还是比较严格，如果没有收到确认短信也可以发送“REG: +微博账号”到 13810928931，如下图：

完成身份确认后，恭喜，您已经成为 SAE 的一名开发者了！并且获得虚拟货币云豆 500 个，SAE 的很多服务都是要消耗云豆。



用户注册 - 成功

第一步: 绑定微博帐号

第二步: 安全设置

第三步: 身份确认

注册成功

恭喜您, 帐号注册成功! 欢迎您加入Sina App Engine!

接下来您可以:

[快速玩转SAE! \( 编写第一个Hello World \)](#)

[使用应用向导快速创建应用](#)

[创建一个应用](#)

3秒后自动转向

想要查看自己的云豆数量和账户信息，可以将鼠标移动到页面最右上角账户名字处，会有下拉菜单显示详情：

 **13\*\*\*1@qq.com** ▼

免费云豆

695

付费云豆(购买)

0

云豆消费记录

[查看详情](#)

开发者等级(申请)

未申请

账户等级(提升)

普通

通知预警(设置)

 [详细设置](#)

 [退出](#)

点击详细设置可以进入账户设置页面。

## ZTalk 秘籍

SAE 并不是一个完全免费的云计算平台，它消耗的是云豆，价格不贵 100 颗才 1 元钱，但是作为普通开发者来说必须想办法完成以下几步，完成后基本可以

免费使用 SAE 了，目前 ZTalk 每个月大概花费的云豆在 300 个左右，折合起来也就是 3 元钱。

任务在开发者首页有相关提示，如下图：

[反馈BUG](#)

[我创建的应用](#) [如何部署代码?](#) [安装应用](#) [创建新应用 \(0/10\)](#)

2

3

1

因未对SVN、二级域名等服务收费，造成资源滥用；为避免资源浪费，给您提供更优质的服务。从2017年10月30日起，新浪云开始收取应用租金。您可以通过以下方式减免租金：[申请实名认证](#) [申请开发者认证](#) [申请手机复验](#)

应用租金收取规则

用户类型	租金价格	云豆赠送
已购买企业服务	免租金	-
通过开发者认证	免租金	依据等级标准，每月赠送云豆
通过实名认证	5 云豆/应用 * 天	通过实名认证，赠送 2000 云豆
通过手机复验 (不同于手机绑定)	5 云豆/应用 * 天	扣取2元手机话费，赠送 200 云豆
未通过实名/手机复验 (不同于手机绑定)	100 云豆/应用 * 天	注册赠送 500 云豆

注：应用租金每天从您的账户中自动扣除

1、手机复验。这个必须去完成，否则开发者创建的每个应用每天啥也不干就扣你 100 云豆。点击手机复验出现提示窗口如下图：

手机复验

使用您绑定的手机 13900000000，编辑短信 **O22** 发送到 **1065800885861**

发送成功将扣除 2元 话费，同时您将获赠200免费云豆。

注意事项：  
1. 本条短信费用，为普通短信资费标准，由运营商收取；  
2. 短信发送成功后，将扣除您 2元话费；同时您将获赠 200免费云豆。

验证成功

请您在收到扣费成功短信后，请点击“验证成功”按钮；如认证过程中遇到问题请 [与我们联系](#)。

取消

根据提示使用注册时绑定的手机发送短信就可以完成，当然这条短信会扣两块钱，同时会赠送你 200 云豆，相当于你完成了第一次的云豆购买。成功以后会

有如下提示，但是这个提示有问题，实际上每天每应用扣点还是 5 个云豆，而不是 1 个云豆



2、实名认证。这个有点认证流程有点恶心，貌似现在很多互联网公司都喜欢用手持证件的照片来认证，搞得跟政府机关一样。但是为了拿 2000 云豆的奖励，咱们还是忍了吧！如下图：

A screenshot of the "实名认证" (Real-name Authentication) form. At the top is a navigation bar with tabs: "我的帐户" (My Account), "密码设置" (Password Settings), "购买云豆" (Buy Cloud Beans), "云豆记录" (Cloud Bean Record), "实名认证" (Real-name Authentication), and "通知设置" (Notification Settings). Below the tabs is a yellow banner that reads: "通过实名认证，可获得2000云豆的奖励哦 查看详情" (Through real-name authentication, you can get a reward of 2000 Cloud Beans. Click here to view details). The form fields include: "真实姓名" (Real Name) with a text input; "证件类型" (ID Type) with a dropdown menu showing "身份证" (ID Card); "证件号码" (ID Number) with a text input; and "验证码" (Verification Code) with a text input and a CAPTCHA image showing "89 N 3". Below these fields is a label "持证件正面照" (Front photo of ID card) followed by a link "点击选择文件" (Click to select file) and a note: "请参考下面的说明及示例提交手持证件的照片。仅支持7M以下的图片文件(bmp,png,jpg,gif格式)" (Please refer to the instructions and examples to submit a photo of your ID card. Only image files (bmp, png, jpg, gif) under 7M are supported). At the bottom is a blue button labeled "提交申请" (Submit Application).

3、开发者认证。这个认证非常难，特别是初级普通开发者需要 2 名认证开发者的推荐，很多人没地方找去，我当初是找了两个码农朋友帮忙认证的，大家可以尝试在 SAE 的交流区 <http://cloudbbs.org/forum.php> 求推荐，认证的好处就是免了应用的 5 云豆日租金，如果能够成为中级以上的开发者还能每月领工

资：【中级】开发者：10,000 云豆/月，【高级】开发者：15,000 云豆/月【资深】开发者：30,000 云豆/月。基本就是免费使用 SAE 了。ZTalk 现在正在申请中级开发者，希望能够成功。

## 二、在 SAE 上创建第一个应用

咱们开通云计算平台就是为了能够进行微信的开发，在开发前我们先要理解一个概念，就是一般来说任何一个程序或者说手机应用都会分为前端和后端，前端就是用户能够接触到的，能够与之方便互动的，比如我们手机上安装的各种应用。

微信之所以能够被预言淘汰手机上百分之 50 以上的应用，是因为微信的公众平台具备了大部分手机应用前端的功能，比如拍照上传、语音传输、地理位置发送等接口，但无论是微信也好，普通手机应用也好，真正在进行逻辑运算、数据保存等服务处理的还是后端程序，这也是我们使用 SAE 所要做的事情。

后端基本没有什么可视化界面，只有大量的源代码，看起来很枯燥，但正是他们及时响应用户的请求，完成了复杂的交互功能，事实上我在编码的时候更感觉是在谱写一段美妙的音乐，它们是有韵律和节奏的。

闲话不扯了，咱们先来建立第一个应用，或者说我们可以叫它为微信公众平台的后端。在开发者首页可以看到如下图：

2

1

反馈BUG

我创建的应用 如何部署代码?

安装应用 创建新应用 (0/10)

⚠

因未对SVN、二级域名等服务收费，造成资源滥用；为避免资源滥用，给您提供更优质的服务，从2012年10月30日起，新浪云开始收取应用租金。您可以通过以下方式减免租金：[申请实名认证](#) [申请开发者认证](#) [申请手机复验](#)

应用租金收取规则

用户类型	租金价格	云豆赠送
已购买企业服务	免租金	-
通过开发者认证	免租金	依据等级标准，每月赠送云豆
通过实名认证	5 云豆/应用 * 天	通过实名认证，赠送 2000 云豆
通过手机复验 (不同于手机绑定)	5 云豆/应用 * 天	扣取2元手机话费，赠送 200 云豆
未通过实名/手机复验 (不同于手机绑定)	100 云豆/应用 * 天	注册赠送 500 云豆

注：应用租金每天从您的账户中自动扣除

先提下安装应用，这个主要是用来安装一些基于 SAE 开发的应用程序或者框架，初学者可以忽略。重点还是说说创建新应用吧，点击创建新应用出现下图：

创建应用

创建空应用 选择开发框架

1

\* 二级域名 (AppID)  .sinaapp.com ✓ 仅允许由数字，字母组成，长度为4到18位，唯一标识，也是二级域名前缀，创建后不可修改。

2

\* 应用名称  应用的中文名称，供显示用。

\* 验证码

3

应用描述

4

开发语言 ☒ php ☐ python ☐ java 应用创建后，开发语言不可修改

5

应用类型 ☒ web应用 ☐ 移动应用

创建应用

1. 应用的访问域名，这个可以自己设置，设置成功后访问这个网址就可以打开你的应用，比如我的 ZTalk 就是用的 `ztalk.sinaapp.com`，要注意的是创建后不可修改。

2. 应用名称，这个没什么讲究，随便写个吧，只是用来标示应用的。
3. 应用描述，由于我们这个应用只是用来做微信后台，并不会放到前台去展示，所以也随便写两句吧。
4. 开发语言，由于老贼所有的开发教程都使用 php，因此这里必须选 PHP。
5. 应用类型，选择 web 应用就好，这个是后台程序，主要是提供一些接口服务。

点击创建应用就可以新建一个 SAE 平台的应用了，创建成功后页面会跳转到应用列表，其实就是 SAE 的个人首页，这个时候多了刚创建的应用名称，如图：



点击上图的应用名称后进入应用的管理页面，里面非常复杂，估计很多人看了会头晕，这些先别理，首先点击“代码管理”建立第一个应用的版本再说，如下图：



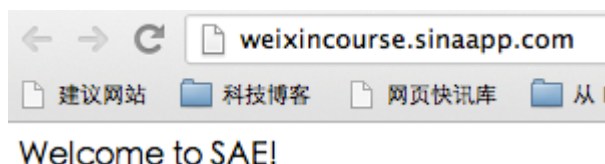
这里什么上传您的代码、SVN 都不用管, 点击“创建一个版本”会弹出一个新窗口, 如图:



每个应用可以创建最多 10 个版本, 每个版本号只能是 1-10 的数字标示且不能重复, 一般是用于版本大的升级, 对于初学者来说没有太多的需求, 我们创建一个版本基本就可以了。创建成功后页面会刷新显示, 如图:



其实这个时候我们的应用已经可以通过网络访问了，在浏览器里输入前面设置的应用访问域名，比如我这个是 `weixincourse.sinaapp.com`，就可以看到，如下图：



恭喜您，您已经成功拥有了自己的网站！不过这个只是个开始，如何开发自己的程序呢？点击“编辑代码”，如图：

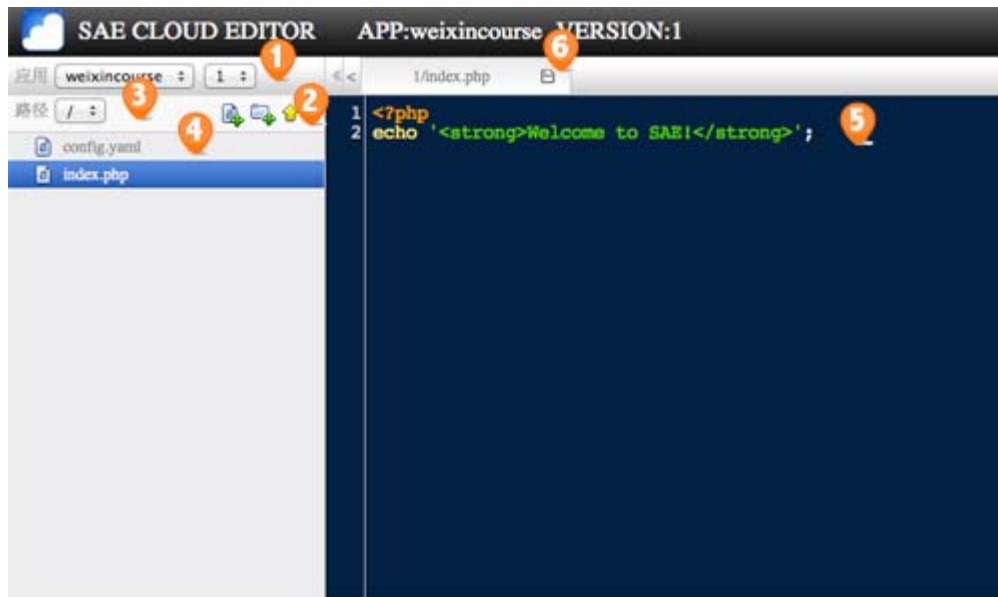


点击后可能会出现要求输入安全密码，如图：



输入注册时设置的安全密码后页面会刷新，需要再次点击“编辑代码”，会弹出一个新窗口，进入代码编辑模式，如图：





这个就是 SAE 的在线编辑器了，虽然我自己是在用 `editplus`，但是作为初学者可以先用这个熟悉起来，并且方便查看效果，毕竟在本地做开发，还需要搭建本地的开发环境，这个对大部分人是件麻烦事。先说下编辑的主要功能模块：

1. 应用程序和版本切换；
2. 三个图标分别对应新建文件、新建目录和上传文件，基本可以满足开发的要求了；
3. 目录切换；
4. 文件列表，双击文件名即可打开文件进行编辑，或者鼠标右键菜单项选择编辑，同时还多了通过 URL 访问和删除功能。
5. 编辑器，颜色搭配的不错，保留字、变量什么的配色还是能让我习惯的。

6. 文档名称、保存和关闭，当文档发生修改时，关闭图标会自动变成保存图标。不合理的是只要文档发生修改时就只有保存图标没有关闭图标，有时候不想保存就很麻烦。

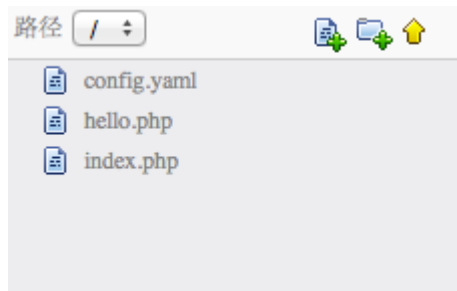
### 三、Hello World

“Hello, world”程序是指在计算机屏幕上输出“Hello,world”这行字符串的计算机程序，**hello world** 作为所有编程语言的起始阶段，占据着无法改变的地位，所有版本的编程教材中，**hello world** 总是作为第一个 **TEST** 记录于书本之中，所有的编程第一步就在于此了！经典之中的经典！**hello world**！老贼的微信教程开发篇第一课也从这里开始！

首先在编辑器页面点击“新建文件”的图标，会弹出新窗口，如下图：



输入“**hello.php**”然后点确定生成新文件，注意：1、全部都要在英文状态下输入！2、**php** 文件名的后缀为**.php**。编辑器的文件列表栏里就出现了新的文件，如图：



鼠标双击文件名或者右键选择编辑，对文件进行编辑，然后在右边的编辑框里输入如下图所示：

A screenshot of a code editor window. The title bar shows the file path '1/hello.php' and a save icon. The editor area has a dark blue background with light blue line numbers on the left. The code is as follows:

```
1 <?php
2
3 //hello world
4 echo "hello world";
5
6 ?>
```

这里要讲一些 php 的语法规则了：

1. 所有 php 的脚本都是以 `<?php` 开始，以 `?>` 结束，编译器根据这两个标签来判断 php 程序。
2. 双斜杠用来做单行注释，注释的作用是记录代码的功能等信息，非常有用，不管是自己以后查看还是别人来阅读。
3. 所有的 PHP 代码行都必须以分号“;”结尾，编译器以这个区分不同指令集。

4. `echo` 是最基本的 php 输出指令。
5. php 的字符串用成对的双引号或者单引号来包含，使用输出指令可以直接讲字符串打印到浏览器里。

点击保存后，鼠标右键选择左栏的 `hello.php`，点击通过 URL 访问，新弹出的窗口会出现：

**hello world**

当然你也可以直接在浏览器里输入“应用的地址+`/hello.php`”访问。

关于 SAE 的介绍先到这里，接下来会讲如何将 SAE 与微信开发模式对接，可能得等两天，在这个时间里我希望大家有时间可以去看下 PHP 的一些基础资料，访问

<http://www.w3school.com.cn/php/index.asp>，按照上面的一些教程先摸索一下，有问题请随时问。

#### -----ZTalk 的互动环节-----

**miya:** 母婴网站微信怎么做好呢！我是运营摇篮网微信的，目前我们已经接口网站无线站，可为用户提供搜索！在母婴网站中算不错但是粉丝增加不快！给点运营建议吧！

**ZTalk:** 我不是万能的，这个有朋友能解答不。

柳峰：更正一处错误：推送消息里语音消息的 **Format** 目前使用的是 **amr**，我们公司的公众平台做了语音识别、解析，可以确定，即使没有内测资格，也能得到 **voice** 消息的 **format** 值，的确是 **amr**！如果是在微信公众平台管理后台下载用户发送语音文件，下载到的是 **mp3** 格式。

## 《微信公众平台入门到精通》Vol.9

2013-05-04



昨天这篇 SAE 的课程貌似很给力，收到一千多回复，多谢朋友们捧场，不少人在问那个菜单的教程是第几篇，是《微信公众平台入门到精通 Vol.6》，所有的课程可以输入“微信教程”查看！

还有一些朋友在问开发模式是否可以群发消息或者点对点下发消息，这个是不行的，群发消息、人工回复消息还是得去公众平台，除非你开外挂模拟登陆公众平台后台操作，当然这个肯定是违规的，会被封号。

另外大家在进入开发模式前还是要提醒下，由于微信现在接口开放的还不多，图片回复、视频回复接口都没有，也就是用户回复关键字获取图片或者视频是无法做到的，如果偏运营的公众账号现在并不推荐使用开发模式。

### 第九章 用微信开发模式做欢迎词

今天的重点是如何将微信开发模式和 SAE 对接，希望朋友们已经完成了 SAE 注册，如果还没搞定的话请抓紧，有问题的话请及时提出，接下来的教程都在那上面进行。

在教程开始前建议朋友们可以先注册一个新的公众账号，或者用非主账号当做开发测试账号，因为开发调试过程中可能会出现账号无法正常回复用户的情况，影响正常使用，所以在学习期间以及后期开发过程中最好有一个测试公众账号，等开发完成了再切换到主账号上，也是非常方便的。

## 一、微信公众平台与 SAE 对接通讯

我们先回到微信公众平台上来，点击高级功能进入，选择开发模式进入开发模式的页面，使用开发模式前记得先把编辑模式关闭，如下图：



先不着急点开关，点了也没用，会提示必须先成为开发者，但第一步也不是点击“成为开发者”按钮，而是点击查看文档按钮，我们要去下载一个文件，进入开发指南页面后点击左侧菜单的“消息接口指南”，如下图：

搜索关键字 [alt-s]

消息接口

消息接口指南

通用接口

自定义菜单接口

使用限制

返回码说明

然后在右边打开的页面导航里点击“示例代码”，如下图：

## 消息接口指南

目录

[隐藏]

1 简介

2 申请消息接口

3 网址接入

4 消息推送

4.1 文本消息

4.2 图片消息

4.3 地理位置消息

4.4 链接消息

4.5 事件推送

5 消息回复

5.1 回复文本消息

5.2 回复音乐消息

5.3 回复图文消息

6 注意事项

7 示例代码

页面直接会跳到最底部，当然直接拉到页面最下也可以，可以看到示例代码下载，如图：

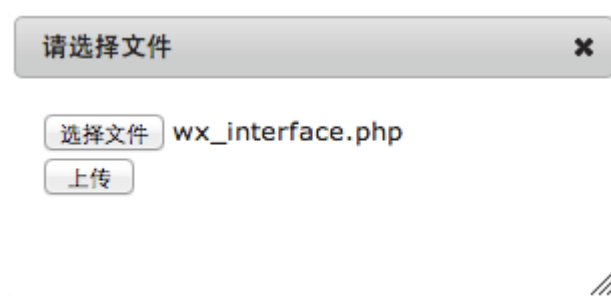
## 示例代码

---

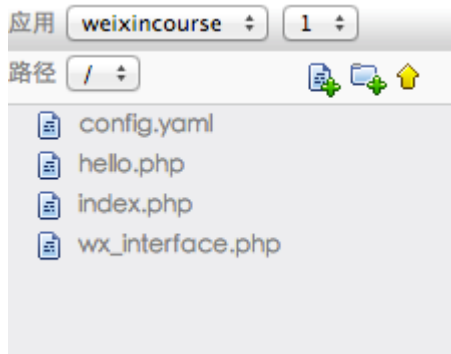
### PHP : 下载

点击下载后会得到一个“wx\_sample.php”的文件，这个就是微信接口的 SDK 文件，首先我们将该文件改个名字，比如“wx\_interface.php”，这里强调下文件的注意事项：1、后缀必须是.php；2、名字不能用中文，尽量是全小写字母，或者字母+下划线+字母组成；3、名字要便于记忆；4、不要把名字设成“index.php”、“weixin.php”、“admin.php”这些。

让大家改名的原因是为了安全，因为这是一个和微信公众平台通讯的接口文件，原则上要隐蔽，以免被人恶意攻击。改完名字后打开 SAE，进入到咱们的应用，点击编辑代码进入编辑后台，把文件上传到 SAE 里，如下图：







双击上传的文件，进入该文件的编辑模式，可以看到密密麻麻的一片代码，朋友们不要被吓到，咱们一步步来，首先修改第 7 行的通讯密钥，如图：

A screenshot of a code editor window titled '1/wx\_interface.php'. The code is as follows:

```
1 <?php
2 /**
3  * wechat php test
4  */
5
6 //define your token
7 define("TOKEN", "weixin");
```

那些斜杠和星号的是注释，可以看到第 6 行的注释英文是“define your token”，也就是说第 7 行的代码功能是定义微信公众平台与 SAE 应用通讯时约定的一个密钥。

这里解释下 `define("TOKEN", "weixin")` 这段代码：

`define` 是用来给常量赋值的函数，这句话的意思是赋予“TOKEN”这个常量值为“weixin”。

**PS：**常量和变量，是每一种编程语言中都存在的，常量值被定义后，在脚本的其他任何地方都不能被改变。一个常量的名称可以由英文字母、下划线、和数字组成，但数字不能作为首字母出现。变量被定义后随时可以重新赋值，其命

名规则与常量一样，区别是变量最前面还要加“\$”符号，而常量直接使用它的名称就可以了，如代码第 67 行，就是给变量\$token 赋值，值为之前定义的常量 TOKEN，可以看到变量和常量的赋值方式也是不同的，变量赋值很简单不需要 define 函数。

```
67 | $token = TOKEN;
```

扯了那么多咱们先把密钥改了吧，这个密钥其实有点像设置一个密码，微信公众平台对密钥设置有要求是 3 位以上的字母或数字，我这里是测试所以改个“123456”，如图：

```
1 | <?php
2 | /**
3 |  * wechat php test
4 |  */
5 |
6 | //define your token
7 | define("TOKEN", "123456");
```

修改保存后，回到微信公众平台的开发模式，点击“成为开发者”按钮，然后会到“接口配置信息”页面，如下图：

### 接口配置信息

请填写接口配置信息，此信息需要你拥有自己的服务器资源。填写的URL需要正确响应微信发送的Token验证，请阅读[消息接口使用指南](#)

URL

Token

[什么是Token ?](#)

提交

1. 输入各位在 SAE 上的应用地址+反斜杠+改名后的 SDK 文件，比如我这里输入的是

[http://weixincourse.sinaapp.com/wx\\_interface.php](http://weixincourse.sinaapp.com/wx_interface.php)

要注意的是前面不要忘记加 `http://`，域名就是咱们应用的域名，上传的 SDK 文件名要写完整，包括 `.php` 的后缀名。

2. 这里就是填写在 SDK 文件里设置的密钥，我这里填写的是“123456”，大家根据自己设置的填写。

填写完后点击提交，如果一切顺利就会提示成功并跳转页面到开发者页面，如下图：

 开发模式已开启

3 开启 ☒ 关闭

### 使用公众平台接口进行开发

使用公众平台的开发接口，公众号可在自身服务器上接收用户的微信消息，并可按需回复。此外，我们还提供了更多更高级的功能和体验，如会话界面的自定义菜单、获取更多类型的消息等。

[查看文档](#)

你已成为开发者

接口配置信息 [修改](#) 1

URL :

http://weixincourse.sinaapp.com/wx\_interface.php

Token :

123456

接口权限表：

1. 接口配置信息。这里可以看到咱们设定的那些接口信息都清楚的显示出来，如果接口的地址或者 token 改变的话点击修改按钮可以重新设置。

2. 接口权限表。打钩的就是咱们可以使用的接口权限，关于这些接口的权限介绍可以查看《微信公众平台入门到精通 Vol.7》，有些朋友如果已经拿到自定义菜单内测资格的，那么自定义菜单也会有打钩。
3. 最重要的一步就是打开开发模式的开关，现在公众账号进入开发模式了！

这个时候你会发现你的公众账号无论是关注还是输入都没有提示了，原来编辑模式下设置的东西都不管用了，因为公众账号已经对接到了你的 **SAE** 接口，SO，我们开始下一步吧，先来个欢迎词！

## 二、开发模式下的欢迎词

先控诉下微信开发模式的设置不合理，接口校验和接口服务用同一个文件，而校验时这个文件必须直接返回 `echostr` 参数内容，当接口要启动正式服务时又得修改同个文件.....好了这些跟大家无关，抱怨一下而已！

前面接口对接成功后，公众账号无法再与用户互动了，那么我们先把订阅欢迎这个功能恢复吧，回到 **SAE** 平台上，在编辑平台新建一个文件，我们把它命名为“`wx_tpl.php`”，这个文件的目的是存放微信输出内容的模板，细心的朋友会发现这就是目前接口可以给用户回复的三种消息格式，分别是文本、图文、音乐。内容如图：

```

<?php
$textTpl = "<xml>
    <ToUserName><![CDATA[%s]]></ToUserName>
    <FromUserName><![CDATA[%s]]></FromUserName>
    <CreateTime>%s</CreateTime>
    <MsgType><![CDATA[%s]]></MsgType>
    <Content><![CDATA[%s]]></Content>
    <FuncFlag>0</FuncFlag>
</xml>";

$newsTpl = "<xml>
    <ToUserName><![CDATA[%s]]></ToUserName>
    <FromUserName><![CDATA[%s]]></FromUserName>
    <CreateTime>%s</CreateTime>
    <MsgType><![CDATA[%s]]></MsgType>
    <ArticleCount>%s</ArticleCount>
    <Articles>
    <item>
    <Title><![CDATA[%s]]></Title>
    <Description><![CDATA[%s]]></Description>
    <PicUrl><![CDATA[%s]]></PicUrl>
    <Url><![CDATA[%s]]></Url>
    </item>
    </Articles>
    <FuncFlag>1</FuncFlag>
</xml> ";

$musicTpl = "<xml>
    <ToUserName><![CDATA[%s]]></ToUserName>
    <FromUserName><![CDATA[%s]]></FromUserName>
    <CreateTime>%s</CreateTime>
    <MsgType><![CDATA[%s]]></MsgType>
    <Music>
    <Title><![CDATA[%s]]></Title>
    <Description><![CDATA[%s]]></Description>
    <MusicUrl><![CDATA[%s]]></MusicUrl>
    <HQMusicUrl><![CDATA[%s]]></HQMusicUrl>
    </Music>
    <FuncFlag>0</FuncFlag>
</xml>";
?>

```

然后把前面的那个 SDK 文件内容全部清空，没错就是全部清空，因为官方那个 SDK 除了用来做接口校验几乎没有什么用处。然后换上自己的代码，如图：

```

1  <?php
2  //加载模板文件
3  include_once("wx_tpl.php");
4
5  //获取微信发送数据
6  $postStr = $GLOBALS["HTTP_RAW_POST_DATA"];
7
8  //返回回复数据
9  if (!empty($postStr)){
10
11     //解析数据
12     $postObj = simplexml_load_string($postStr, 'SimpleXMLElement', LIBXML_NOCDATA);
13     //发送消息方ID
14     $fromUsername = $postObj->FromUserName;
15     //接收消息方ID
16     $toUsername = $postObj->ToUserName;
17     //消息类型
18     $form_MsgType = $postObj->MsgType;
19
20     //事件消息
21     if($form_MsgType=="event")
22     {
23         //获取事件类型
24         $form_Event = $postObj->Event;
25         //订阅事件
26         if($form_Event=="subscribe")
27         {
28             //回复欢迎文字消息
29             $msgType = "text";
30             $contentStr = "感谢您关注公众平台教程! [愉快]\n\n想学公众平台使用的友
            友请输入“跟我学”! [玫瑰]"
31             $resultStr = sprintf($textTpl, $fromUsername, $toUsername, time(), $msgType, $contentStr);
32             echo $resultStr;
33             exit;
34         }
35     }
36
37 }
38
39 else
40 {
41     echo "";
42     exit;
43 }
44
45 ?>

```

给大家解释下这一大段代码：

第 3 行，include\_once 是用来加载其他 php 文件的函数，在这里加载了 wx\_tpl.php，加载文件是可以设置路径的，由于现在两个文件在同一目录下，因此没有加路径。这样做的好处是有利于代码复用，一些经常用到的类、函数和变量可以放在其他文件统一加载，让代码简洁轻便。

第 6 行，接收微信公众平台发送过来的用户消息，该消息数据结构为 XML，不是 php 默认的认识数据类型，因此这里用了 \$GLOBALS['HTTP\_RAW\_POST\_DATA']来接收，同时赋值给了\$postStr，这

里强调一个 **php** 的铁律就是 **php** 对大小写是敏感的，也就是变量也好函数也好，大小写不同是表示不同的变量或者函数，即**\$Q1** 和**\$q1** 是两个不同的变量，大家在写代码时千万注意。

第 9 行，一个条件语句，**empty** 是判断变量是否为空的函数，如果是空返回 0 否则返回 1，感叹号在这里是运算符表示非（not 的意思），非 0 就是 1，非 1 就是 0，可能有点绕，意思就是如果**\$postStr** 不为空（接收到了数据）就继续执行下面的语句，如果为空就跳转到 39 行执行（这里就执行了两句代码，第一句是输出空字符串，第二句是退出）。

PS:

1、**php** 的逻辑运算符有“&&”表示与（并且的意思），“||”标示或（或者的意思）“!”表示非（不是的意思），记住并且理解这些符号对后面写代码很有好处.....

2、所有的编程语言都有 **if...else..** 语句，**php** 的规则是 **if(条件){执行语句}else{执行语句}**，这里 **else** 可以没有，或者可以多重判定，**if(条件 1){执行语句}elseif(条件 2){执行语句}else{执行语句}**。判断条件用“()”包含，执行语句用“{}”包含,相邻最近的一对括号标示一段脚本，大家在编写的时候不要漏掉后面的半个括号。我在这段代码里括号标示的比较清楚，大家可以仔细参照一下。

第 12 行，使用 **simplexml\_load\_string()** 函数将接收到的 XML 消息数据载入对象**\$postObj** 中。这个严谨的写法后面还得加个判断是否载入成功的条件语句，不过不写也没事。

第 14 行，将对象**\$postObj** 中的发送消息用户的 **OPENID** 赋值给 **\$fromUsername** 变量，这里的“->”是指针，大家了解下就行了。

第 16 行，将对象 `$postObj` 中的公众账号的 ID 赋值给 `$toUsername` 变量。

第 18 行，将对象 `$postObj` 中的消息类型赋值给 `$form_MsgType` 变量。

第 21 行，判断消息是否为事件消息，现在微信公众账号的用户关注和取消订阅都是通过事件消息发送，因此咱们要欢迎用户关注首先得判断来接收的消息类型是否为事件。这里要强调的是在 `php` 里一个“`=`”用来赋值，“`==`”用来比较是否相等，大家千万注意在判断语句时如果少写一个“`=`”号这个判断会永远成立。

第 24 行，如果接收到的消息是事件消息，判断事件类型是否为订阅，在事件消息里“`subscribe`”表示订阅，“`unsubscribe`”表示取消订阅。

第 29 到 32 行，完成的是欢迎词回复的输出：

第 29 行是设定回复消息类型为文本；

第 30 行是设定回复消息的内容，其中可以使用表情符号的文本方式，换行使用“`\n`”，这里要注意的是字符串里如果要使用双引号或者单引号一般用中文符号，如果一定要用英文符号，要在引号前面加反斜杠比如“`echo \"\";`”输出的是一个双引号；

第 31 行，使用 `sprintf` 这个函数将格式化的数据写入到变量中去，其中“`$textTpl`”这个就是我们在 `wx_tpl.php` 里定义过的文本消息模板变量，后面的“`$fromUsername, $toUsername, time(), $msgType, $contentStr`”分别顺序替换模板里“`%s`”位置，也即是“`$resultStr`”这个变量最后实际为：



```
<xml>
<ToUserName><![CDATA[$fromUsername]]></ToUserName>
<FromUserName><![CDATA[$toUsername,]]></FromUserName>
<CreateTime>time()</CreateTime>
<MsgType><![CDATA[$msgType]]></MsgType>
<Content><![CDATA[$contentStr]]></Content>
<FuncFlag>0</FuncFlag>
</xml>
```

PS:time()是 php 里的时间函数，返回当前时间的 Unix 时间戳（格林威治时间 1970 年 1 月 1 日 00:00:00）到当前时间的秒数。

第 32 行就是把回复消息输出。

第 33 行是退出整个程序，在 php 里 exit 可以放置在任何地方，当程序运行到这里就直接退出，否则还将继续运行，即使已经有结果输出在页面。

好了大家可以自己试着玩玩，不想敲字的朋友可以直接下载，地址是：  
**<http://weinxincourse.b0.upaiyun.com/lesson1.zip>**，下载后解压里面两个文件都有，请用记事本或者其他文本编辑器打开，复制内容到 **SAE** 上，请尽量不要使用文件上传，主要是怕大家编辑器保存文件时没存成 **UTF-8**，会出错。

## 《微信公众平台入门到精通》Vol.10

2013-05-05



今天发生个很有趣的事情，品途网的妹子主动来坦白，说小编不懂事转载了几篇我的教程，不过已经批评后删除了，这事让我有点生气，因为我在所有教程最后都写清楚不允许转载，即使署了我的名都不能！当然对于品途网这样可爱的小编我还是原谅了。

未经作者允许不转载这件事情，在国内的互联网环境下貌似很难做到，随手百度了下“微信公众平台入门到精通”，真是让人沮丧，已经有不少网站转载过了，最可气的是[百度文库](#)和[豆丁网](#)，那么大的网站盗用我的著作却不署名，当然你可以说是网友上传的，是 UGC 的内容，但 UGC 就可以明目张胆的盗版盗用吗？

谁能给我个建议如何向这种流氓网站要个说法！否则真是没写下去的动力，太尼玛黑暗了，我写一篇教程起码 4 个小时，昨天这篇从晚上 10 点写到凌晨 3 点半啊！小偷花 1 分钟粘贴复制就变成自己的了，难道非得让我收费阅读？？

发飙完毕，继续码字，昨天的教程好像深了些，不知道大家的进度情况，成功实操的朋友记得手机截屏分享下哦，不然心里没底啊！如果有不清楚的或者调试不成功的请及时提问，可以把自己代码拍照上来我帮着看下。

## 第十章 公众账号发送欢迎图文消息

昨天已经讲了如何实现用户订阅公众账号时发送欢迎词，但这个貌似体现不出开发模式的优势啊，我用编辑模式做不是更加简单嘛，那么今天就来个升级版的欢迎词吧。先看最后的效果，如下图：



当用户关注公众账号时自动回复一个多图文消息，每个图文消息可以对应一篇文章或者自己网站的某个网页，用户点击就可以查看，是不是非常棒，特别适合一些企业用来做公司业务介绍。现在就教大家如何实现，顺便就可以把图文消息的回复讲了。

## 一、使用 SAE 的分布式存储服务

在编辑模式里做图文消息回复时，直接通过关键字回复就可以很方便的实现，但是编辑模式下的关注回复不能用图文消息，所以大家都是用文字的，但是在开发模式里就可以使用了，图文消息的输出格式如下：

```
<xml>
<ToUserName><![CDATA[toUser]]></ToUserName>
<FromUserName><![CDATA[fromUser]]></FromUserName>
<CreateTime>12345678</CreateTime>
<MsgType><![CDATA[news]]></MsgType>
<ArticleCount>2</ArticleCount>
<Articles>
<item>
<Title><![CDATA[title1]]></Title>
<Description><![CDATA[description1]]></Description>
<PicUrl><![CDATA[picurl]]></PicUrl>
<Url><![CDATA[url]]></Url>
</item>
<item>
<Title><![CDATA[title]]></Title>
<Description><![CDATA[description]]></Description>
<PicUrl><![CDATA[picurl]]></PicUrl>
<Url><![CDATA[url]]></Url>
</item>
</Articles>
<FuncFlag>1</FuncFlag>
</xml>
```

其中图片是用图片链接（在上图标签 **PicUrl** 处）的形式出现，也就是图片得先上传到服务器，然后将图片连接放置到输出代码里，这里提供三种图片连接生成方法：

1、直接复制第三方网站的图片链接，比如我们在百度图片搜索到某张图片，将该图片的链接直接复制过来就可以，非常方便，但是如果第三方网站把图片删除或者说有防盗链机制，就会造成图片不显示。所以不推荐这类方式，除非可以保证该图片链接永久有效。

2、将图片上传到新浪微博、腾讯微博这些地方发微博（微信的素材管理图片链接不能使用），然后将图片链接复制过来，这里要注意的是记得把微博水印关掉，否则这个图片上就带着你的微博名字了。

3、咱们都已经在用云计算平台了，就顺便用下云存储服务吧，我目前自己是在用又拍云存储，那个有点复杂后面做具体完整项目时再介绍。

现在先使用 SAE 的 Storage 存储服务，这是 SAE 为开发者提供分布式文件存储，因为 SAE 平台限制使用本地存储，所有需要持久化的文件比如图片、音乐等只能保存到 Storage 上。

首先是启动 Storage 服务，进入咱们的应用首页，在左边导航里找到“Storage”这个选项，如下图：

#### 服务管理

MySQL

Storage

Memcache

Cron

Image

FetchURL

Mail

TaskQueue

DeferredJob

Counter

KVDB

CDN alpha

Cloud Bridge

点击打开后右边页面会进入开启“Storage”服务的设置页面，如下图：

## Storage 简介

Storage是SAE为开发者提供的分布式文件存储服务，用来存放用户的持久化存储的文件。用户需要先在线管理平台创建Domain（相当于一级子目录），创建完毕后，即可进行文件的管理。  
详细使用说明请[点击这里查看](#)。

上传大文件请参考[大文件上传说明文档](#)

服务自检

点此新建一个domain

点击“新建一个 domain”按钮，进入存储空间的设置页面，如下图：

创建domain

weixincourse

Domain Name 只能含有数字或字母。

图片和音乐文件存储

☐ 私有

私有domain的数据不能直接通过URL访问，需要通过secret key认证才能访问。

☒ 缓存过期

基于浏览时间 6 小时

设置浏览器缓存过期时间，可以减少文件的请求次数，从而减轻服务器的负担

☐ 防盗链

设置允许访问域名（支持通配符，如\*.sinaapp.com'），从而防止盗链。

创建 取消

1. 存储空间名称，只能由数字和字母组成，根据自己喜好取个吧。
2. 存储空间的介绍，用来提示该空间的用途，比如我这个是用来放图片和音乐文件的。
3. 是否私有云存储，如果打勾则无法通过 URL 直接访问，只能通过程序接口读取访问，咱们不用搞那么复杂所以别打勾了

4. 缓存时间。这个可以减少服务器的负担，一般设 6 小时就可以了。
5. 防盗链设置。可以设置能够访问文件的白名单网址，咱们也不用搞那么复杂所别打勾了。

创建成功以后在右边会出现存储空间管理，如图：



1. 存储空间属性修改，除了名称不能修改以外其他都可以，包括设置访问权限、说明、防盗链以及缓存时间修改等。
2. 删除该存储空间，删除时将会把该空间内所有文件都删除，所以小心操作。
3. 管理存储空间。点击后进入管理页面，可以看到存储空间默认的容量是 10G，肯定够用，如下图：





1. 下载客户端，直接用应用程序管理存储空间，有点像 FTP 管理，咱们现在是初学者，不用去管这个。
2. 返回 Storage 的管理首页。
3. 新建子目录方便文件分类，比如可以建一个目录叫 **pic**，里面存放图片文件。

上传文件，点击后弹出文件上传窗口，跟普通的文件上传操作一样，支持批量上传，我就不多介绍了，微信开发模式只支持 jpg 和 png 文件，请注意图片文件格式。另外需要注意的是文件名需要规范，不要和之前的已经上传的文件同名，否则会覆盖掉，建议每次上传前使用日期+编号+文件名的方式。这里给大家一个我上传图片后的样例：

已用: 655.21 KB/10G						Storage扩容
名称	类型	上传日期	大小	缓存过期	操作	
<input type="checkbox"/> 2030000116825213454566061...	jpg	2013/05/04 14:51:35	6.31 KB	未设置	删除	
<input type="checkbox"/> IMG_6902.JPG	jpg	2013/05/04 14:37:04	193.58 KB	未设置	删除	
<input type="checkbox"/> Srip20130403_14.jpg	jpg	2013/05/04 14:51:31	122.46 KB	未设置	删除	
<input type="checkbox"/> Srip20130403_19.jpg	jpg	2013/05/04 14:55:47	4.44 KB	未设置	删除	
<input type="checkbox"/> Srip20130403_8.jpg	jpg	2013/05/04 14:51:34	78.55 KB	未设置	删除	
<input type="checkbox"/> Srip20130504_8.png	png	2013/05/04 14:53:16	220.67 KB	未设置	删除	
<input type="checkbox"/> lz.jpg	jpg	2013/05/04 19:11:44	29.19 KB	未设置	删除	
<input type="checkbox"/> 删除						
转到: 1 返回条数: 25 GO 上页 下页						

点击图片名称就会在新窗口打开该图片，复制浏览器的 URL 就是该图片的链接，可以直接用到图文消息的代码里。

可能有朋友会说为什么不把图片直接上传到 SAE 的代码管理目录里访问，确实可以这样做，但是不推荐这样做，一个是代码的空间大小默认只有 100M，扩容会增加开销，二是代码空间是用来跑脚本的，同等流量费用要高很多，三是代码空间不允许写入文件，将来无法实现通过程序上传图片的功能。

## 二、制作多图文消息

啰嗦一大堆朋友们都等急了吧，现在我们开始做图文消息欢迎词！

今天我教大家的是用户订阅后回复文字消息，最后实际输出的是一个经过拼装的符合文字消息回复格式的字符串，其结构如下图：

```
"<xml>\n<ToUserName><![CDATA[$fromUsername]]></ToUserName>\n<FromUserName><![CDATA[$toUsername,]]></FromUserName>\n<CreateTime>time()</CreateTime>\n<MsgType><![CDATA[$msgType]]></MsgType>\n<Content><![CDATA[$contentStr]]></Content>\n<FuncFlag>0</FuncFlag>\n</xml>"
```

里面的一些带“\$”的变量在前面代码都赋了值，最后回复到微信公众平台接口的实际消息是这样：

```

<xml>
<ToUserName><![CDATA[oVcCOjvhaAlBH2wzkjgdoHgyHVyQ]]></ToUserName>
<FromUserName><![CDATA[gh_99ce4d828bd1]]></FromUserName>
<CreateTime>1367668940</CreateTime>
<MsgType><![CDATA[text]]></MsgType>
<Content><![CDATA[感谢您关注公众平台教程！[愉快]

想学公众平台使用的朋友请输入“跟我学”！[玫瑰]]]></Content>
<FuncFlag>0</FuncFlag>
</xml>

```

现在打开 SAE 的代码管理，进入编辑模式，修改我们的微信接口文件，首先将代码中第 29 行到第 31 行的三行代码注释掉，就是输出文本消息的三行代码，如图所示：

```

/*
//回复欢迎文字消息
$msgType = "text";
$contentStr = "感谢您关注公众平台教程！[愉快]\n\n想学公众平台使用的朋友请输入“跟我学”！[玫瑰]";
$resultStr = sprintf($textTpl, $fromUsername, $toUsername, time(), $msgType, $contentStr);
*/

```

注释多行代码使用/\*.....\*/来包含，在中间的所有内容都不再被执行。

然后在注释前面开始写图文消息回复，前面我们看到过图文消息回复的格式了，现在就是要拼接出一个符合图文消息回复格式的字符串，当用户关注时回复给用户，首先我们把图文消息的公共部分先赋值，如图：

```

26         if($form_Event=="subscribe")
27         {
28             //回复欢迎图文菜单
29             $resultStr="<xml>\n
30             <ToUserName><![CDATA[".$fromUsername."]]></ToUserName>\n
31             <FromUserName><![CDATA[".$toUsername."]]></FromUserName>\n
32             <CreateTime>".time()."</CreateTime>\n
33             <MsgType><![CDATA[news]]></MsgType>\n
34             <ArticleCount>5</ArticleCount>\n
35             <Articles>\n";
36

```

代码从 30 行开始，前面的跟昨天是一样的，就是监测是否用户订阅事件。

首先是赋值给\$resultStr 这个变量图文消息的数据头部，跟文本消息的差不多，只是昨天是用了 sprintf 这个函数直接将数据写到模板里去了，今天是拿模板直接来用，实际效果是一样的，与文字消息不同的是：

第 34 行，MsgType 变成了 news；

第 35 行多了一个 ArticleCount，这个值表示多图文消息的条数，包括了封面消息在内，我这里是 5，各位可以根据自己实际修改。

当 ArticleCount 为 1，只放一条消息时则为单图文消息，大于 2，放了两条消息以上则为多图文消息。

第 36 行是图片消息列表标签的开始，最后会有一个</Articles>来闭合整个图文消息列表。

**PS：**在 php 里字符串的拼接使用“.”，echo "A"."B";输出的是 AB。当然高级程序猿为了提高运行效率会使用其他方法来拼接字符串，这里就不说了。

然后是封面消息，如图下：

```
39 //添加封面图文消息
40 $resultStr.="<item>\n
41 <Title><![CDATA[国内最专业的微信教程]]></Title> \n
42 <Description><![CDATA[]]></Description>\n
43 <PicUrl><![CDATA[http://weixincourse-weixincourse.stor.sinaapp.com/5nip20130403_8.jpg]]></PicUrl>\n
44 <Url><![CDATA[http://mp.weixin.qq.com/mp/appmsg/show?__biz=MjM5MjgwNjgwMA==&appmsgid=10000169&itmsgid=1]]></Url>\n
45 </item>\n";
```

这里可以看到`$resultStr`和“=”之间多了一个“.”，这也是字符串拼接语法，等于`$resultStr=$resultStr.“...”`，将新的字符串拼接到之前已经赋值的`$resultStr`后面再赋值给`$resultStr`。

图文消息里用`<item>...</item>`来区分各条消息，每条图文消息数据格式都一样，分别为：

第 41 行，图文消息的标题

第 42 行，图文消息的摘要，如果是多图文可以为空不填写，如果是单图文消息需要填写。

第 43 行，封面图片，放置图片链接，官方给出较好的效果为大图即多图文的封面消息图片为 640\*320，小图即列表消息的缩略图 80\*80，注意这里的图片其实可以跟实际链接打开的文章图片不一致。

第 44 行，文章链接，放置点击该图文消息时打开的页面，这里可以放置自己的网页链接，比如淘宝店的或者自己网站的，当然也可以放置微信公众平台的图文消息链接（ZTalk 的文章查询里其实放的都是微信文章链接，省事又省流量，哈哈）。

剩下就是放四个列表消息，结构跟封面消息其实是一样的，只是在多图文消息模式下默认会将第一个消息作为封面消息，如图：

```
//添加多条图文消息
$resultStr.="<item>\n
<Title><![CDATA[教程大纲]]></Title> \n
<Description><![CDATA[]]></Description>\n
<PicUrl><![CDATA[http://weixincourse-weixincourse.stor.sinaapp.com/Snip20130403_14.jpg]]></PicUrl>\n
<Url><![CDATA[https://stalk.sinaapp.com]]></Url>\n
</item>\n";

$resultStr.="<item>\n
<Title><![CDATA[报名培训]]></Title> \n
<Description><![CDATA[]]></Description>\n
<PicUrl><![CDATA[http://weixincourse-weixincourse.stor.sinaapp.com/Snip20130504_8.png]]></PicUrl>\n
<Url><![CDATA[http://mp.weixin.qq.com/mp/appmsg/show?__biz=MjM5MjgwNTJgMA==&appmsgid=10000169&itemid=10000054]]></Url>\n
</item>\n";

$resultStr.="<item>\n
<Title><![CDATA[关于老贼]]></Title> \n
<Description><![CDATA[]]></Description>\n
<PicUrl><![CDATA[http://weixincourse-weixincourse.stor.sinaapp.com/1z.jpg]]></PicUrl>\n
<Url><![CDATA[http://mp.weixin.qq.com/mp/appmsg/show?__biz=MjM5MjgwNTJgMA==&appmsgid=10000054&itemid=10000054]]></Url>\n
</item>\n";

$resultStr.="<item>\n
<Title><![CDATA[联系老贼]]></Title> \n
<Description><![CDATA[]]></Description>\n
<PicUrl><![CDATA[http://weixincourse-weixincourse.stor.sinaapp.com/Snip20130403_19.jpg]]></PicUrl>\n
<Url><![CDATA[http://mp.weixin.qq.com/mp/appmsg/show?__biz=MjM5MjgwNTJgMA==&appmsgid=10000054&itemid=10000054]]></Url>\n
</item>\n";
```

只需要将前面的封面消息复制粘贴一下，然后把标题、摘要、图片和连接这些修改下就可以了。

最后是把图文消息的尾巴加上，如图：

```
76 | $resultStr.="</Articles>\n
77 | <FuncFlag>0</FuncFlag>\n
78 | </xml>";
```

第 76 行就是我前面说的闭合图文消息内容；

第 77 行是用来星标用户请求消息的，由于这个消息其实是用户订阅事件，不用打星标，所以这里的值为 0。

第 78 行是闭合整个输出图文消息结构的。

好了，现在保存下，重新关注下公众账号，看看是否成功回复了一个图文菜单，是不是很有成就感！

给大家一个作业：试试关注时回复单条图文或者 5 条以上的多图文消息。

## ZTalk 秘籍

微信开发的调试是个很麻烦的事情，为什么呢？因为接口是微信来调用的，所有的结果都是返回到微信公众平台去的，我们无法得到运行结果，除了语法错误还可以直接打开接口网页看错误提示，比如下图表示我的代码第 16 行语法错了：



其他错误就全得从公众账号里看了，但实际上只要有错，公众账号里啥反应都没有了，一般来说就是逻辑出错或者变量没有获取到值什么的，这个时候就需要设置一些断点来检查，比如我们在今天的这个课程中如果发现订阅后没有任何欢迎提示，我们可以在输出前设置一个断点，将结果先保存成一个文件，然后查看该文件检查结果是否正确，如图：

```
87 $s = new SaeStorage();
88 $s->write( 'weixincourse' , 'test.txt' , $resultStr );
89 echo $resultStr;
90 exit;
```

前面说了 SAE 不允许在本地生成文件，所以我们得把文件生成到 Storage 存储空间去。

第 87 行是新建一个 Storage 的对象，定义为 \$s;

第 88 行是写入一个文件到 **Storage** 空间，格式为`$s->write(空间名称, 写入文件名, 写入的内容)`

然后我们取消并重新订阅公众账号后，到 **Storage** 会看到多了一个 **test.txt** 的文件，如图：

<input type="checkbox"/>	 2030000116825213454595061...	jpg
<input type="checkbox"/>	 IMG_6902.JPG	jpg
<input type="checkbox"/>	 Snip20130403_14.jpg	jpg
<input type="checkbox"/>	 Snip20130403_19.jpg	jpg
<input type="checkbox"/>	 Snip20130403_8.jpg	jpg
<input type="checkbox"/>	 Snip20130504_8.png	png
<input type="checkbox"/>	 lz.jpg	jpg
<input type="checkbox"/>	 test.txt	txt

点击打开后就可以看到内容，如果正常的话里面内容就是多图文消息的 **XML** 数据，如图：



```

<xml>

  <ToUserName><![CDATA[oVcC0jvhaA1BH2wxkjgdoHqyHVyQ]]></ToUserName>

  <FromUserName><![CDATA[gh_99ce4d828bd1]]></FromUserName>

  <CreateTime>1367674232</CreateTime>

  <MsgType><![CDATA[news]]></MsgType>

  <ArticleCount>5</ArticleCount>

  <Articles>
<item>

  <Title><![CDATA[国内最专业的微信教程]]></Title>

  <Description><![CDATA[]]></Description>

  <PicUrl><![CDATA[http://weixincourse-weixincourse.stor.sinaapp.co

  <Url><![CDATA[http://mp.weixin.qq.com/mp/appmsg/show?__biz=MjM5Mj

</item>
<item>

  <Title><![CDATA[教程大纲]]></Title>

```

如果文件没有生成，或者文件里面是空的，就说明程序运行有问题，咱们把断点上移，因为程序运行一般是从上至下的，所以我们可以把断点放到第一个逻辑判断处，如图我移动到了是否接收到微信接口发送消息的判断位置：

```

9  if (empty($postStr)){
10     $s = new SaeStorage();
11     $s->write( 'weixincourse' , 'test.txt' , "get_post" );
12

```

主要是监测有没有收到微信的请求，我直接用字符串"get\_post"（记得字符串要用双引号包含）来代替之前的变量，然后在公众号里发送任意字符，再查看Storage里的test.txt文件，如果正常文件应该生成，并且里面应该有“get\_post”，否则就是出错了，那要检查公众平台是否已经和SAE应用正常通讯了。

断点一般是检查变量的赋值是否正确，或者是否从接口获取到了值，或者代码逻辑是否正确。比如可以移到图中这个逻辑判断后面，文件写入内容直接用字

字符串比如"subscribe"，然后取消关注再订阅公众号，监测是否收到了订阅事件，如果文件里有 subscribe 字样就表示运行正常，否则就是出错了：

```
26     if($form_Event=="subscribe")
27     {
28         $s = new SaeStorage();
29         $s->write( 'weixincourse' , 'test.txt' , "subscribe" );
```

今天的课就先到这里，明天讲关键字回复以及.....N 多人关心的如何播放音乐！！！！

有朋友反映：手机上图文消息显示长的英文网址有问题，昨天的下载链接就没有显示完整导致无法下载，建议大家接下来可以在网页上看教程，方法是点击右上角的按钮，在弹出的菜单里选择“复制链接”，然后将链接通过邮件等方法发送到 PC 上，然后在 PC 上打开看会比较方便。

另外进入开发阶段我会把每次讲的源码放到服务器上供大家下载，由于微信长网址显示有问题，所以请在 ZTalk 里回复“微信代码”获取下载地址。带来不便非常抱歉.....

## 《微信公众平台入门到精通》Vol.11

2013-05-08



昨天上午去 GMIC 围观了一下，比去年更热闹了，出来 SHOW 的还是那些个项目，感觉创新匮乏，站台更华丽了，妹子更漂亮了，大佬更加水了。

听说最近腾讯要下一盘很大的棋，上午就去认真听了下腾讯开放平台的演讲，感觉腾讯动作还是有点慢，云计算平台到现在也没正式推出，搞得玩微信公众平台的 HOST 都用了 SAE、BAE、阿里云，不知道这盘看起来很大的棋子后面如何吸引用户迁移过来。

个人建议是 TAE 可以考虑把微信接口封装好，作为重要接口提供给第三方开发者，让更多的个人和企业用户快速上手进入微信开发，同时广点通抓紧建立微信公众平台的广告系统，推动商业化进程，让平台和开发者双赢。

下午参加乐视的 super TV 发布会，以自媒体人身份去参加的，虽然最近自媒体人被炒的烂大街，但是市场还是在逐步认可自媒体人的价值和身份，个人品牌的影响力和号召力肯定是能产生商业价值的，这个其实也不是现在才有。

新品发布会形式基本搬的乔帮主模式，就不评价了。不过颠覆性的乐视超级电视无论是性能还是价格都亮瞎了我，60 寸的智能电视，夏普的 LED 屏，高通的四核智能电视专用处理器，富士康的机器人生产线代工，才 6999 元.....

没有看错真的是 6999，而且功能强大，图像逼真，绝对的价格屠夫，拉上三星来当垫背的倒也不错，另外还有 40 寸的才 1999，我打算去抢一台来孝敬爸妈。

唯一要吐槽的是那个被贾跃亭引以为傲的电视机底座，一点也不简约一点也不人性化，我灰常担心那玩意会咔嚓一下断裂。

晚上去参加了 vivo Xplay 的发布会，生平第一次进水立方，发布会搞的是相当高端大气啊，主持人请的汪涵，一出场就引起全场轰动，连工作人员都集体围观拍照。

不过让我印象最深的还是手机揭幕式，3D 技术加上道具的配合，vivo Xplay 破墙而出，经典绝妙的创意，美轮美奂的效果，反正我之前木有看到过。

顺带说下现在高配置的大屏幕手机真没啥可说的，即使 vivo Xplay 有是 5.7 寸的高清屏幕，1300 万像素的摄像头，也没有亮点，因为自从有了 iphone 以后所有手机都长一个样子，不管是 HI-FI 音响还是处理器真木有啥新意，能让我眼睛一亮的就是那个手机摇一摇弹出来的完整版缩小屏，主要方便大屏手机单手操作，还有其他一些特殊功能，表现出步步高在 ROM 的设计上还是很花了信息，也很彰显人性的。

唯一要吐槽的是步步高事先放烟雾弹，说苍老师要来现场，结果就是造个噱头，让我等影迷空欢喜一场，本来我还想请苍老师给 wemeida 题副字的.....

前面说了一大堆其实是想掩饰下我昨天为什么木有按时更新，现在转正题吧，首先教学视频请点击[原文链接查看](#)，我不放在正文里了，怕消耗朋友们的流量（我非常不理解为什么在正文里的视频会自动播放，这个设计很扯淡，不明真相的人点进来花花的流量就木有了）。其次就是今天讲下自定义回复怎么做以及很多人关心的.....音乐播放。

## 第十一章 公众账号自定义回复功能

微信官方其实并不推荐经常群发消息，更多的是希望用户通过对话来自己获取服务或者进行搜索。今天我们就讲下如何使用开发模式做自定义回复。

目前公众账号可以接收的用户提交消息请求有四种类型，分别是文字、图片、地理位置和链接，而公众账号回复只有三种格式，文字、图文和音乐，也就是说用户互动的模式最多也就是 12 种，我们今天先讲三种即文字回复文字、文字回复图文、文字回复音乐。

这强调一下，微信的消息接口通讯有效时间是 5 秒，也就是说网络不稳定等原因很容易造成用户在 5 秒内接收不到回复消息，然后就通讯中断了，这几天有人反映关注 ZTalk 后没有消息提示就是这个问题。

### 一、文字回复文字之鸚鵡學舌

鸚鵡學舌大家都知道，就是你说什么它跟着你说什么，咱们也可以在微信公众平台里玩一下，首先进入 SAE 平台，打开代码编辑，在之前的接口文件上进行修改。代码如下：

```

9  if (!empty($postStr)){
10
11     //解析数据
12     $postObj = simplexml_load_string($postStr, 'SimpleXMLElement', LIBXML_NOCDATA);
13     //发送消息方ID
14     $fromUsername = $postObj->FromUserName;
15     //接收消息方ID
16     $toUsername = $postObj->ToUserName;
17     //消息类型
18     $form_MsgType = $postObj->MsgType;
19
20     //文字消息
21     if($form_MsgType=="text")
22     {
23         //获取用户发送的文字内容
24         $form_Content = trim($postObj->Content);
25         //如果发送内容不是空白则发用户相同的文字内容
26         if(!empty($form_Content))
27         {
28             $msgType = "text";
29             $resultStr = sprintf($textTpl, $fromUsername, $toUsername, $time, $msgType, $form_Content);
30             echo $resultStr;
31             exit;
32         }
33         //否则提示输入
34         else
35         {
36             $msgType = "text";
37             $resultStr = sprintf($textTpl, $fromUsername, $toUsername, $time, $msgType, "请输入些什么吧...");
38             echo $resultStr;
39             exit;
40         }
41     }
42 }

```

代码添加位置是在“事件消息”的注释前面，用户消息赋值之后，实际添加的代码是从图上 20 行开始到 41 行。

第 21 行是判断用户发送的消息类型是否文字消息，可以参考事件消息的判断，变动的只是消息类型标示。

第 24 行是判断如果用户发送了文字消息，那么获取用户发送的文字消息内容赋值给变量\$form\_Content。

**PS:** 这里有个函数“trim”，大家可以查下 php 的函数手册，这个函数作用是去除字符串首尾的空格，有时候用户误在头尾多敲了空格，如果不去除的话就会出现关键字无法匹配，比如“你好”如果不去掉尾部空格就无法匹配关键字“你好”。

第 26 行是判断用户发送的消息内容是否为空，可以参照第 9 行的逻辑，empty(\$form\_Content)在\$form\_Content 不为空的时候返回 0，为空的时候返回 1，非(!)0 即 1 表示条件成立，用户发送的消息是有内容的。

第 28 行到第 31 行就是完成回复文字消息的功能，其中回复的内容直接用了 `$form_Content`，也就是把用户发送的内容又给回复回去。

第 34 行是如果用户发送的内容为空白，则返回输入提示。因此第 37 行的回复消息就是"请输入些什么吧....."。

好了现在去测试下，当你向公众账号输入文字时，公众账号就会回复同样的内容给你，当然你也可以把 `$form_Content` 改成直接的字符串，这样就是自动回复了。

## 二、文字回复图文之餐馆菜单

有多少人能马上说出编辑模式里最多能设多少个关键字？估计不多，当然用心看我教程的应该知道，是 200 个，到了开发模式这个关键字其实是无限的，前面举了文字回复文字的简单例子，现在加点难度，可以识别关键字并且回复的是图文消息，咱们就拿菜单来做例子吧，先做菜谱的类别，这个我们用文字回复，代码如下：

```
26         if(!empty($form_Content))
27         {
28
29             //回复菜谱类别
30             if($form_Content=="菜谱")
31             {
32
33                 $return_str="请输入字母编码浏览相应菜品: \n\n";
34                 $return_arr=array("lc.冷菜\n","hb.杭帮菜\n","sk.烧烤\n","wp.外婆烧\n","ml.麻辣\n","xc.湘菜\n","tp.甜");
35                 $return_str.=implode("\n",$return_arr);
36                 $msgType = "text";
37                 $resultStr = sprintf($textTpl, $fromUsername, $toUsername, $time, $msgType, $return_str);
38                 echo $resultStr;
39                 exit;
40             }
41
42             //默认回复
43             $msgType = "text";
44             $resultStr = sprintf($textTpl, $fromUsername, $toUsername, $time, $msgType, "请输入菜谱点菜");
45             echo $resultStr;
46             exit;
47         }
```

添加的代码位置在判断用户发送消息不为空的条件判断之后，默认回复之前，这里大家注意下默认回复其实就是将前面文字回复文字的变量用字符串代替了。增加的代码从第 29 行开始到 41 行。

第 30 行判断用户发送过来的消息是否为“菜谱”，这里注意判断是否等于是用两个“=”号。如果是则运行该判断条件下的代码。

第 33 行先赋值给\$`return_str` 一个初始字符串点菜提示“请输入字母编码浏览相应菜品：”，我这里加了两个\n 表示空两行。

第 34 行是一个数组，数组的元素为菜谱的类别。

数组在程序中使用非常普遍，它是一种把具有相同类型的若干变量、数字或者字符串，有序组织起来的一种形式，有点像咱们以前数学课教的集合概念。

数组的每个元素都是由键和值关联组成，我在例子里的定义方式键值会默认生成，第一个元素的键值为 0，元素值是“`lc.冷菜\n`”，数组的每个元素值可以使用“`数组名[键值]`”的方式获取或者输出，比如\$`return_arr[0]`的值就是“`lc.冷菜\n`”。

数组可以用来做数据的存储、排序、检索等，它的概念比较复杂但是很重要，希望大家能够去看些资料认真学习下。

第 35 行是用 `implode` 函数将\$`return_arr` 转化成字符串，然后加到之前赋值过的\$`return_str` 后面，其运行结果为：

"请输入字母编码浏览相应菜品：\n\nlc.冷菜\nhnb.杭帮菜\nnsk.烧烤\nnwp.外婆烧\nnml.麻辣\nnrc.热菜\nntp.甜品"。



**implode** 函数的使用方法为 **implode(分隔符, 数组)**，将数组内元素转化成字符串，并将分隔符插入元素之间，我在例子里分隔符为空值，所以最后转化的字符串是首尾相连中间没有分隔符的。

后面的就不解释了，输出文字消息同时退出运行。

接着我们来示例某个类别的详细菜单，咱们用多图文消息来做，灵光的朋友已经想到了，呵呵，先写一个判断，然后把那段用户关注回复多图文消息的代码拷贝过来修改下就好了。当然我今天加了点料，示例如下：

```

42 //回复菜谱详情 外婆烧
43 if($form_Content=="wp")
44 {
45     $resultStr="<xml>\n
46     <ToUserName><![CDATA[".$fromUsername."]]></ToUserName>\n
47     <FromUserName><![CDATA[".$toUsername."]]></FromUserName>\n
48     <CreateTime>".time()."</CreateTime>\n
49     <MsgType><![CDATA[news]]></MsgType>\n
50     <ArticleCount>10</ArticleCount>\n
51     <Articles>\n";
52     //菜谱详情数组
53     $return_arr=array(
54         array(
55             "外婆烧",
56             "http://weixincourse-weixincourse.stor.sinaapp.com/wp1.png",
57             "http://www.weixin.fm"
58         ),
59         array(
60             "开心土豆 (16元) ",
61             "http://weixincourse-weixincourse.stor.sinaapp.com/wp2.png",
62             "http://www.weixin.fm"
63         ),
64         array(
65             "铁板虾 (36元) ",
66             "http://weixincourse-weixincourse.stor.sinaapp.com/wp2.png",
67             "http://www.weixin.fm"
68         ),
69         array(
70             "酸辣汤 (16元) ",
71             "http://weixincourse-weixincourse.stor.sinaapp.com/wp2.png",
72             "http://www.weixin.fm"
73         ),
74         array(
75             "鱼香茄子 (12元) ",
76             "http://weixincourse-weixincourse.stor.sinaapp.com/wp2.png",
77             "http://www.weixin.fm"
78         ),
79         array(
80             "麻辣豆腐 (16元) ",
81             "http://weixincourse-weixincourse.stor.sinaapp.com/wp2.png",
82             "http://www.weixin.fm"
83         ),
84         array(
85             "茶香鸡 (36元) ",
86             "http://weixincourse-weixincourse.stor.sinaapp.com/wp2.png",
87             "http://www.weixin.fm"
88         ),
89         array(
90             "五香牛肉 (40元) ",
91             "http://weixincourse-weixincourse.stor.sinaapp.com/wp2.png",
92             "http://www.weixin.fm"
93         ),
94         array(
95             "风沙羊排 (45元) ",
96             "http://weixincourse-weixincourse.stor.sinaapp.com/wp2.png",
97             "http://www.weixin.fm"
98         ),
99         array(
100             "下一页请输入“w2”，返回请输入“菜谱”",
101             "http://weixincourse-weixincourse.stor.sinaapp.com/wp2.png",
102             "http://www.weixin.fm"
103         )
104     );
105     //数组循环转化
106     foreach($return_arr as $value)
107     {
108         $resultStr.="<item>\n
109         <Title><![CDATA[".$value[0]."]]></Title> \n
110         <Description><![CDATA[".$value[1]."]]></Description>\n
111         <PicUrl><![CDATA[".$value[1]."]]></PicUrl>\n
112         <Url><![CDATA[".$value[2]."]]></Url>\n
113         </item>\n";
114     }
115     $resultStr.="</Articles>\n
116     <FuncFlag>0</FuncFlag>\n
117     </xml>";
118     echo $resultStr;
119     exit;
120

```

添加的代码位置在菜谱的大括号后面。第 42 行到 51 行和之前的是一样的，唯一不同的是 ArticleCount 我设置成了 10，也就是图文消息条数为 10 条（编辑

模式里多图文消息最多只有 8 条，开发模式里是 10 条），不清楚的可以看第十期教程，我直接讲下改进的部分。

第 53 行到 105 行我定义了一个数组 `$return_arr`，这个数组里共有 10 个元素，每个元素又是一个结构相同的数组，这些数组都有三个元素分别对应标题、图片和链接，为了偷懒，我 9 个列表消息只是换了标题，没有图片和连接，不知道大家看明白没有。

这个数组其实就相当于我把菜单的 10 道菜里不同的数据按照相同的结构做了集合，然后使用一个循环读取语句将这些数据拼接成将要输出的图文消息。

第 107 行的 `foreach` 是一个将数组中每个元素按照键值顺序遍历一遍的函数，通俗的说就是将 `$return_arr` 这个数组里的元素按照先后顺序读取出来，每次读取一个元素，读取出来的元素赋值给 `$value`，所以这个 `$value` 的值会随着每次循环改变。即第一次循环时 `$value=$return_arr[0]`，第二次循环时 `$value=$return_arr[1]`.....

每次读取的 `$value` 其实就是 `$return_arr` 一个元素，根据上面的数据结构这个元素又是一个数组，里面有三个元素，分别对应了标题、图片和链接，值是 `$value[0]`、`$value[1]`、`$value[2]`，将这三个数据拼接到图文消息模板，同时与前一次的 `$resultStr` 拼接。这样做的好处在于减少了代码的长度，把图文消息模板复用了，另外这个数组的结构其实跟数据库里读取出来的结构是一样的，所以也是想让大家一点点熟悉起来。

等循环结束就执行第 116 行，完成最后的闭合拼接，然后输出结果。公众账号就将这个包含 10 条菜品的图文消息发送给了用户。

### 三、文字回复音乐之表情传意

很多人，真的很多人，几乎每天都有人在公众平台上问我同一个问题，就是你怎么可以放超过 60 秒的歌曲，我的答案只有一个，就是用开发模式的音乐消息回复接口实现的。今天就给大家介绍下如何使用音乐接口来回复用户消息，让用户有好的欣赏体验。

回复音乐消息里的音乐是以链接形式加载到回复数据模板里的，类似咱们回复图文消息里的图片链接，**SO**，我们音乐文件的链接获取也参照图片连接方式，一种呢是找网上在线收听的链接，不过真不是很保险，很多是防盗链的，比图片还严格，另外一种呢就是放到云存储平台上，这里建议大家如果要把音乐做成常态功能，必须选择专业的云存储平台，比如又拍云、七牛。

今天我图个方便就还是上传到了 **SAE** 的 **Storage** 里了，如下图，我选了我的大爱曲婉婷成名曲《我的歌声里》（偷偷显摆下，曲婉婷跟我在微博上可是互相关注的噢，我是她第 3729 个粉丝！！）。

## Storage»weixincourse»

<input type="checkbox"/> 名称	类型
<input type="checkbox"/>  2030000116825213454595061...	jpg
<input type="checkbox"/>  IMG_6902.JPG	jpg
<input type="checkbox"/>  Snip20130403_14.jpg	jpg
<input type="checkbox"/>  Snip20130403_19.jpg	jpg
<input type="checkbox"/>  Snip20130403_8.jpg	jpg
<input type="checkbox"/>  Snip20130504_8.png	png
<input type="checkbox"/>  lz.jpg	jpg
<input type="checkbox"/>  mysongs.aac	aac
<input type="checkbox"/>  mysongs.mp3	mp3
<input type="checkbox"/>  test.txt	txt
<input type="checkbox"/>  wp1.png	png
<input type="checkbox"/>  wp2.png	png

这个消息回复里支持的音乐格式目前我只测试过 MP3、AAC 两个，肯定是没有问题的，这里需要注意的是上传的文件名必须用英文或数字，文件名里不要有中文、空格、单双引号等，大小写无所谓。如果资源允许可以上传一个低码率的音乐文件，再上传一个高码率的音乐文件，因为微信里听音乐会自动识别网络环境选择不同音质的版本。

音乐文件准备好后咱们就上代码了吧！我这里给的例子是用户发送表情，公众账号回复歌曲，如图所示：

```

123 //表情回复歌曲
124 if($form_Content=="/::")
125 {
126
127     $msgType = "music";
128     $resultStr = sprintf(
129         $musicTpl,
130         $fromUsername,
131         $toUsername,
132         $time,
133         $msgType,
134         "我的歌声里",
135         "曲婉婷",
136         "http://weixincourse-weixincourse.stor.sinaapp.com/mysongs.aac",
137         "http://weixincourse-weixincourse.stor.sinaapp.com/mysongs.mp3");
138     echo $resultStr;
139     exit;
140
141 }

```

增加的代码加到前面菜单的代码后面就行了，只要在那个默认回复之前都可以。

第 124 行代码是判断用户发送的表情是不是微笑。就是微信表情第一排第一个，这里大家肯定会很奇怪微笑表情符号不是“[微笑]”吗，老贼我最早也犯过这个错，折腾半天才发现其实微信接口发送过来的表情代码是这样的。

第 127 行这里是消息类型，音乐消息的类型是 music。

第 128 行到 137 行就是调用 wx\_tpl.php 里的音乐消息模板，用 sprintf 将格式化的数据写入到变量中去。

后面就是输出到用户手机上，用户接收到的就是音乐播放器，点击播放键就可以欣赏音乐了，是不是很简单？

## ZTalk 秘籍

如何收集所有表情的代码呢？在第 10 期教程里有个秘籍是让大家调试接口时可以把输出结果文件保存到 Storage 上，然后查看文件检查程序是否正常。咱们就用这个法子抓取所有表情的代码。

把抓取表情的代码放到这个位置：

```
20      //文字消息
21      if($form_MsgType=="text")
22      {
23
24          //获取用户发送的文字内容
25          $form_Content = trim($postObj->Content);
26
27          //抓取表情
28          $s = new SaeStorage();
29          $s->write( 'weixincourse' , 'bq.txt' , $form_Content );
30          exit;
31      }
```

然后到手机上往公众平台上发表情，按照顺序从第一个一直输入到最后一个，然后一起发送，然后你到 Storage 就会看到有 bq.txt，打开后就能看到如下图：

```
|/::)/::~/:B/:|/:8-)/::</::$/::X/:/'(/:--|/:@/:P/:D/:0/:/::
+/:--b/:Q/:T/::,@P/::,@-D/::d
```

这些就是真正的表情符号，我这里就发了第一屏幕的，大家抓取完就把代码都删除了吧，否则这里有个结束语句在那里，不去掉就不会执行后面的程序。

今天的课就先到这里，一宿木有睡觉了，总算在上飞机前完工，希望对大家有帮助，接下来会再讲下图片、地址和链接的开发怎么做，请大家耐心一些，不要太催我！视频请点击[原文链接](#)！

今天讲的源码放到服务器上供大家下载，请在 ZTalk 里回复“微信代码”获取下载地址。带来不便非常抱歉.....

## 《微信公众平台入门到精通》Vol.12

2013-05-13



-----ZTalk 致歉-----

昨天这篇说阿里云 **OS** 的失分了，自媒体人失去判断是一件可怕的事情，以后尽量多了解事实再说话。但真不是广告，只是想帮大家搞点阿里云免费试用名额，算了，解释等于掩饰不废话了.....另外阿里云 **OS** 我写的是王博士观点，不代表本人观点真是躺着中枪，觉得有不对的尽管喷，我全部照登，最后全部发博士邮箱！

先发几个喷的：

1、王坚这是和谷歌得有多大仇恨？自从搞了个云 **OS**，被谷歌批了一顿后，感觉恼羞成怒了，上面的第二点，得出谷歌是黑社会的结论的论据，居然是建立在“据说”，这种道听途说上！

2、收专利使用费很常见，微软收安卓费用，收 **htc**，三星的费用，一些常见的手势操作也是专利。谷歌如果是黑社会，微软就不知道是什么了？阿里云手机



硬件不错，软件垃圾到死，还好意思拿出来。阿里的 **OS**，说到底底层还是 **linux**。

3、但是 **yunos** 做得很烂，王坚还好意思拿这种东西出来忽悠，买了一台云 **OS** 手机，体验差，把体验做到安卓那样才不会自己打脸

4、王博士似乎本人并无和 **Android** 相提并论的成就，说谷歌是黑社会是准备学习小米、华为手机之类的话题营销么？还是青龙兄摘抄有误呀？

5、刷 **ROM** 这个观点不能苟同，就跟买了电脑把预装的 **Linux** 换成 **Windows** 一样，用户有权使用自己喜欢的 **ROM** 啊...主要是国内的手机厂商做不好 **ROM** 才把自己沦为硬件提供商的

-----正文开始-----

上周六在贝塔咖啡成功举办了首届贝塔微信电商沙龙，虽然之前限制了人数，但还是有不少人走后门来旁听了，不过我们重在分享也就默许了，办沙龙目的是为了让更多朋友了解微信公众平台，正确用好公众平台，到场的很多朋友是外地赶来，都自觉的带了壹基金捐款凭证，再次感谢大家的信任和支持！

在现场白鸦、fenng 和我分享了很多干货，也给了电商的朋友一些解决方案，绝对是外面听不到的，来的人都收获很大。由于事先有约定所以不能写出来，但是各位不要失望，预计下次会在深圳搞，时间应该是 6 月初，有兴趣的朋友等通知吧，另外也请出个主意，下次报名参会的捐款给谁呢？（PS：此次活动场地杭州贝塔咖啡免费提供，饮料和小食由白鸦请客）

另外昨天来的朋友里有一位是在海外微信公众平台上做电商业务的，碰到的问题很多，在国内看来是优势的事情到了国外全部不行，其主要原因还是微信在海外的普及率不够高，本来扫一扫二维码关注公众号那么简单的事，到了国外就还得提示用户先下载微信.....不知道有没有其他朋友也在尝试玩海外公众平台，有什么好的运营方式和推广方式欢迎分享！

## 第十二章 公众账号接收非文字消息

在微信公众平台上用户都是用使用文字在进行交互的，但是有时候使用图片、语音、视频以及地理位置等可以实现一些非常棒的功能，比如“语音提醒”公众号里的发送语音就可以实现定时的事件提醒功能；比如 **ZTalk** 曾经搞过微信拍照晒电脑桌面的活动（响应的人太少，桑心.....）；比如一些预定本地化服务的公众号里发送当前地理位置就可以查询周边商家。

要实现这些功能首先得学会接收这些消息类型，比文字消息类型肯定要复杂很多了，目前我们能够接收的消息类型只有图片、地理位置以及退订消息，其他的如链接消息、语音消息是需要官方授权的。接收到用户消息以后回复的消息类型可以参考第 11 章，目前也就那三种。

### 一、图片消息接收

相信拍照是目前智能手机用户用得最多的功能没有之一，在微信里也是一样，看看每天朋友圈里分享的那些照片就知道了，那么如何接收和保存用户照片捏，首先我们得了解用户发送的图片消息结构，如下图：

```
<xml>
<ToUserName><![CDATA[toUser]]></ToUserName>
<FromUserName><![CDATA[fromUser]]></FromUserName>
<CreateTime>1348831860</CreateTime>
<MsgType><![CDATA[image]]></MsgType>
<PicUrl><![CDATA[this is a url]]></PicUrl>
<MsgId>1234567890123456</MsgId>
</xml>
```

大家可以看到图片消息除了固定的消息发送用户 ID、接收公众号 ID、生成时间等固定字段外，多了一个图片链接 PicUrl，这个就是用户向公众账号发送图片后，保存到微信的服务器上返回给公众号的链接，该图片链接格式如下：

<http://mmsns.qpic.cn/mmsns/zugRGk3LUDUo77MrzmzohPRt4ia014HBvp0WWgIk7gNoPvgqNtiaw9Kw/0>

也就是说我们如果要把图片保存到本地服务器，就要先将图片从微信服务器上抓取下来，当然你也可以直接保存这个链接以后直接访问，这个看具体应用了。今天要讲的是图片保存到本地服务器上，请看下面代码：

```
20 //图片消息
21 if($form_MsgType=="image")
22 {
23     //获取发送的图片URL
24     $from_PicUrl=$postObj->PicUrl;
25     //创建新图片的名称
26     $filename=$fromUsername.date("YmdHis").".jpg";
27     //建立抓取图片类
28     $f = new SaeFetchurl();
29     //抓取图片
30     $res = $f->fetch($from_PicUrl);
31     //如果抓取到图片
32     if($f->errno() == 0)
33     {
34         //新建存储类
35         $s = new SaeStorage();
36         //写入图片
37         $s->write( 'weixincourse' , $filename , $res );
38         //成功提示
39         $msgType = "text";
40         $resultStr = sprintf($textTpl, $fromUsername, $toUsername, $time, $msgType, "图片上传成功");
41         echo $resultStr;
42     }
43     else
44     {
45         //失败提示
46         $msgType = "text";
47         $resultStr = sprintf($textTpl, $fromUsername, $toUsername, $time, $msgType, "图片上传失败");
48         echo $resultStr;
49     }
50 }
51 exit;
```

由于图片消息是另一种消息类型，因此代码添加位置可以位于文字消息上面或者下面，只要别在文字消息的判断语句内就行。然后我来解释下每个语句的作用：

第 20 行，判断消息类型是否为图片消息，条件为`$form_MsgType==image`;

第 24 行，获取图片消息数据中的图片链接并赋值给`$from_PicUrl`;

第 26 行，生成要保存到本地服务器的图片名称，为了避免重复新图片命名使用了发送用户的 **OPENID**+当前时间戳，文件的后缀名我是直接设定为 **jpg** 文件。  
（由于从图片链接上无法获取到图片后缀名，又懒得通过头文件获取图片格式，就直接固定死了 **JPG** 格式，貌似没有什么问题，哈哈哈）；

第 28 行，**SAE** 上的内置接口类很多，**SaeFetchurl** 是一个用来抓取远程网页的类，使用这个类就可以很方便的实现抓取其他网站的内容，否则使用 **PHP** 的 **curl** 或者 **file\_get\_contents** 这些估计还得解释半天，这句代码是新建一个抓取类的对象。

第 30 行，执行抓取图片链接，其中抓取的函数是 **fetch()**，图片链接是之前赋值的`$from_PicUrl`，抓取后的结果赋值给`$res`;

第 32 行，判断抓取结果，**errno()**返回的是抓取结束后的错误代码，如果为 0 则成功，其他的就是不成功。

第 35 行，图片抓取成功后，新建一个 **Storage** 的对象，我们要保存图片了。

第 37 行，这句代码其实已经在以前出现多次，这次将抓取的内容(\$res)写入指定的文件（\$filename），并保存到 Storage 里，请注意把“weixincourse”替换成自己创建的 Storage 空间名。

第 39 行到 41 行，保存成功后给用户提示图片上传成功。

第 45 行到 48 行，文件没有抓取到提示用户图片上传失败。

当然我们如果严谨一点，还要判断图片文件是否保存成功，可以把第 37 行后面改写一下，如下图：

```
37 | $s->write( 'weixincourse' , $filename , $res );
38 | if($s->errno()==0)
39 | {
40 |     //成功提示
41 |     $msgType = "text";
42 |     $resultStr = sprintf($textTpl, $fromUsername, $toUsername, $time, $msgType, "图片上传成功");
43 |     echo $resultStr;
44 | }
45 | else
46 | {
47 |     //失败提示
48 |     $msgType = "text";
49 |     $resultStr = sprintf($textTpl, $fromUsername, $toUsername, $time, $msgType, "图片保存失败");
50 |     echo $resultStr;
51 | }
52 | }
53 | else
54 | {
55 |     //失败提示
56 |     $msgType = "text";
57 |     $resultStr = sprintf($textTpl, $fromUsername, $toUsername, $time, $msgType, "图片上传失败");
58 |     echo $resultStr;
59 | }
60 | exit;
61 | }
```

大家可以尝试把 Storage 的空间名字故意写错，看看会有什么样的提示。如果提示上传成功，我们到 SAE 的 Storage 列表里就应该可以看到刚上传的文件了。

 oVcCOjyhaAlBH2wxkjgdHqyH...	jpg	2013/05/13 01:57:45	39.5 KB
-----------------------------------------------------------------------------------------------------------------	-----	---------------------	---------

## 二、地理位置消息接收

手机上基于地理位置的 APP 很多，是个应用现在都得跟 LBS 扯上点关系，SoLoMo 里重要的一环就是地理位置，微信里也有很多应用是跟地理位置相关的，比如查个本地天气、附近酒店餐馆啥的。今天举的例子是查本地天气。先了解下地理位置消息的结构，如下图：

```
<xml>
<ToUserName><![CDATA[toUser]]></ToUserName>
<FromUserName><![CDATA[fromUser]]></FromUserName>
<CreateTime>1351776360</CreateTime>
<MsgType><![CDATA[location]]></MsgType>
<Location_X>23.134521</Location_X>
<Location_Y>113.358803</Location_Y>
<Scale>20</Scale>
<Label><![CDATA[位置信息]]></Label>
<MsgId>1234567890123456</MsgId>
</xml>
```

地理位置消息多了四项，分别是经纬度的 X 和 Y 坐标、地图缩放比例以及地址信息，而实际上由于网络原因我们经常是收不到地址信息的，只有坐标信息，因此地理位置的开发基本围绕着坐标来。先来看本地天气查询代码吧，如下图：

```

20 //地理位置,本地天气
21 if($form_MsgType=="location")
22 {
23     //获取地理消息信息,经纬度,地图缩放比例,地址
24     $from_Location_X=$postObj->Location_X;
25     $from_Location_Y=$postObj->Location_Y;
26     $from_Location_Scale=$postObj->Scale;
27     $from_Location_Label=$postObj->Label;
28     //地址解析使用百度地图API的链接
29     $map_api_url="http://api.map.baidu.com/geocoder?";
30     //坐标类型
31     $map_coord_type="&coord_type=wgs84";
32     //建立抓取对象
33     $f = new SaeFetchurl();
34     //抓取百度地址解析
35     $geocoder = $f->fetch($map_api_url.$map_coord_type."&location=".$from_Location_X."&".$from_Location_Y);
36     //如果抓取地址解析成功
37     if($f->errno() == 0)
38     {
39         //匹配出城市
40         preg_match_all( "/<city>(.*?)</city>/", $geocoder, $city );
41         $city=str_replace(array("市","县","区"),array("",""),$city[1][0]);
42         //通过新浪天气接口查询天气的链接
43         $weather_api_url="http://php.weather.sina.com.cn/xml.php?password=DJOYnief8234jlsK";
44         //城市名转字符编码
45         $city="&city=".urlencode(iconv("UTF-8","GBK",$city));
46         //查询当天
47         $day="&day=0";
48         //抓取天气
49         $weather = $f->fetch($weather_api_url.$city.$day);
50         //如果抓取到天气
51         if($f->errno() == 0 && strpos($weather,"Weather"))
52         {
53             //用正则表达式获取数据
54             preg_match_all( "/<city>(.*?)</city>/", $weather, $w_city);
55             preg_match_all( "/<status2>(.*?)</status2>/", $weather, $w_status2);
56             preg_match_all( "/<status1>(.*?)</status1>/", $weather, $w_status1);
57             preg_match_all( "/<temperature2>(.*?)</temperature2>/", $weather, $w_temperature2);
58             preg_match_all( "/<temperature1>(.*?)</temperature1>/", $weather, $w_temperature1);
59             preg_match_all( "/<direction2>(.*?)</direction2>/", $weather, $w_direction2);
60             preg_match_all( "/<power2>(.*?)</power2>/", $weather, $w_power2);
61             preg_match_all( "/<chy_shuoming>(.*?)</chy_shuoming>/", $weather, $w_chy_shuoming);
62             preg_match_all( "/<savestate_weather>(.*?)</savestate_weather>/", $weather, $w_savestate_weather);
63             //如果天气变化一致
64             if($w_status2==$w_status1)
65             {
66                 $w_status=$w_status2[1][0];
67             }
68             else
69             {
70                 $w_status=$w_status2[1][0]. "转" . $w_status1[1][0];
71             }
72             //将获取到的数据拼接起来
73             $weather_res=array(
74                 $w_city[1][0]. "天气预报",
75                 "发布: ".$w_savestate_weather[1][0],
76                 "气候: ".$w_status,
77                 "气温: ".$w_temperature2[1][0]. "-" . $w_temperature1[1][0],
78                 "风向: ".$w_direction2[1][0],
79                 "风力: ".$w_power2[1][0],
80                 "穿衣: ".$w_chy_shuoming[1][0]
81             );
82             $weather_res=implode("\n",$weather_res);
83
84             $msgType = "text";
85             $resultStr = sprintf($textTpl, $fromUsername, $toUsername, $time, $msgType, $weather_res);
86             echo $resultStr;
87         }
88     }
89     else
90     {
91         //失败提示
92         $msgType = "text";
93         $resultStr = sprintf($textTpl, $fromUsername, $toUsername, $time, $msgType, "天气获取失败");
94         echo $resultStr;
95     }
96 }
97 else
98 {
99     //失败提示
100     $msgType = "text";
101     $resultStr = sprintf($textTpl, $fromUsername, $toUsername, $time, $msgType, "无法获取地理位置");
102     echo $resultStr;
103 }
104 exit;
105 }

```

代码添加位置同图片消息，另起一个消息类型判断语句，可以放在图片消息前面或者后面。前面说了我们主要使用的是经纬度，经纬度是可以通过一些地图 api 接口来获取实际地址、周边商家等信息的，天气代码这里我用的是百度地图 API 接口，主要是因为它有 URL 接口，代码解释开始：

第 21 行，消息类型判断语句，消息类型为 `location`；

第 24 行到 27 行，将用户推送地理消息的经纬度、地图比例、地址信息分别赋值。经纬度分别为 `Location_X` 和 `Location_Y`，相当于用经线和纬线的交叉点来标注地理位置。`Scale` 是用户发送地理位置时地图的缩放比例。`Label` 是地址信息（经常是获取不到的，获取了也没啥用，因为都是连在一起的，无法提取地市县信息）。

第 29 行，定义百度地图 API 接口的反向地址解析 URL，反向地址解析是指通过经纬度获取当前位置的地址信息。

第 31 行，由于各家地图不一样因此传输过来的经纬度也会有所偏差，这里我选的是 `wgs84` 即手机 GPS 的坐标。

第 33 行，又要抓页面了，先建个抓取类的对象。

第 35 行，百度地图 API 接口的反向地址解析规则是 `URL+坐标类型+坐标值`，其中 `$map_api_url.$map_coord_type` 两个变量拼接就是 `URL+坐标类型`，然后再加上经纬度参数，用 `location=经度,纬度` 来赋值。

这里说下 URL 的规则，URL 就是大家常看到的网页链接，一般由 `HTTP://` 后面加网址加参数组成，主要说下参数，参数一般是“参数名=赋值”组成，普通



的 URL 参数格式是跟在网址后面第一个参数前用“?”号分隔，第二个参数开始用“&”分隔，参数在程序里是可以获取到的，我们上面获取坐标解析的实际地址形式为：

```
http://api.map.baidu.com/geocoder?coord_type=坐标类型&location=经度值,纬度值
```

接收的实际地址为 `http://api.map.baidu.com/geocoder`，获取到的参数是 `coord_type` 和 `location`，值为相应后面跟着的。

第 37 行，判断是否抓取成功，如果抓取成功 `$geocoder` 的数据实际是如下格式：

```
<?xml version="1.0" encoding="utf-8" ?>
<GeocoderSearchResponse>
  <status>OK</status>
  <result>
    <location>
      <lat>39.90935</lat>
      <lng>116.398241</lng>
    </location>
    <formatted_address>北京市西城区人大会堂西路39号</formatted_address>
    <business>前门,天安门,和平门</business>
    <addressComponent>
      <streetNumber></streetNumber>
      <street>人大会堂西路</street>
      <district>西城区</district>
      <city>北京市</city>
      <province>北京市</province>
    </addressComponent>
    <cityCode>131</cityCode>
  </result>
</GeocoderSearchResponse>
```

这里一大堆信息里只需要提取城市，即 CITY 这个标签内的数据。

第 40 行，这是一个正则表达式，比较复杂，作用就是根据规则将 `$geocoder` 里的 `<city>北京市</city>` 数据提取出来赋予 `$city` 这个变量，如果成功这个数据是会是一个多维数组，其中 `city` 标签内的数据即北京市是存储在 `$city[1][0]` 里的，`$city[0][0]` 的值是“`<city>北京市</city>`”；

第 41 行，将\$city[1][0]的值即“北京市”提取出来，同时使用 `str_replace` 函数将“市”替换掉再重新赋值给\$city，`str_replace` 按照字面意思就是字符串替换，用法是：`str_replace(要替换的内容,替换成的内容,字符串)`；其中替换的内容和替换成的内容可以使用数组，也可以使用单个字符串，我这里是用了数组，即将市县区都替换成了空，替换的用处是因为后面查天气预报的接口只支持城市名称，不能有市县区啥的.....经过这一步\$city 的值就是“北京”；

第 43 行，定义天气 API 接口的 URL；

第 45 行，做了三件事，第一个使用 `iconv()` 函数将\$city 的字符编码从 UTF-8 转换成 GBK，第二件是使用 `urlencode` 将汉字转换成英文编码方便 URL 传值，第三件是将 URL 中的“&city=”的参数名拼接了。

关于字符，有时候我们上网的时候会发现网页有乱码，大部分是因为字符编码不对造成的，可以调节浏览器的编码来切换，在程序里也是一样，由于新浪接口接收的字符串是 GBK 的，而我们程序里使用 UTF-8，所以需要转码后才能通讯，否则新浪接口收到的就是乱码。

**PS: GBK 或者 GB2312 是中文简体编码，属于 ANSI 编码，但是同个 ANSI 编码值在不同国家的编码对应是不同的文字，会非常混乱，所以有了 Unicode 以及 UTF-8，这是国际通用的文字编码格式，所有文字都被分配了不同的编码，也就不怕乱码了。**

第 46 行，查询天气日期，0 表示当天，1 表示明天，以此类推.....

第 49 行，抓取天气内容并赋值给\$weather，这里不需要再建立抓取类的对象了，因为之前已经建立了可以直接用。

第 51 行，判断是否抓到天气，这里我多加了一个

```
&& strstr($weather, "Weather")
```

`strstr` 函数是用来检查 `$weather` 里是否存在“`Weather`”这个字符串，`&&`表示并且，这里的判断就是不仅要抓取成功并且在抓取到的内容里存在“`Weather`”。这样写的目的是因为新浪天气接口不管有没有查询到天气都会返回数据，而判断数据里是否有天气信息，只有判断返回内容里有没有“`Weather`”这个字符串。成功抓取到的会是如下内容：

```

<?xml version="1.0" encoding="UTF-8"?>
<!-- published at 2013-05-13 03:19:42 -->
<Profiles>
  <Weather>
    <city>北京</city>
    <status1>阴</status1>
    <status2>阴</status2>
    <figure1>yin</figure1>
    <figure2>yin</figure2>
    <direction1>无持续风向</direction1>
    <direction2>北风</direction2>
    <power1>≤3</power1>
    <power2>3-4</power2>
    <temperature1>28</temperature1>
    <temperature2>18</temperature2>
    <ssd>7</ssd>
    <tgdl>25</tgdl>
    <tgdl2>25</tgdl2>
    <zx>2</zx>
    <ktk>4</ktk>
    <pollution>3</pollution>
    <xcz>1</xcz>
    <zho></zho>
    <diy></diy>
    <fas></fas>
    <chy>2</chy>
    <zho_shuoming>暂无</zho_shuoming>
    <diy_shuoming>暂无</diy_shuoming>
    <fas_shuoming>暂无</fas_shuoming>
    <chy_shuoming>短袖衫、短裙、短裤、短套装、短袖T恤</chy_shuoming>
    <pollution_l>一般</pollution_l>
    <zx_l>弱</zx_l>
    <ssd_l>偏热</ssd_l>
    <fas_l>暂无</fas_l>
    <zho_l>暂无</zho_l>
    <chy_l>短袖类</chy_l>
    <ktk_l>不需要开启</ktk_l>
    <xcz_l>非常适宜</xcz_l>
    <diy_l>暂无</diy_l>
    <pollution_s>对空气污染物扩散无明显影响</pollution_s>
    <zx_s>紫外线弱</zx_s>
    <ssd_s>天气偏热，适当调整穿衣后，仍可达到比较舒适的程度。</ssd_s>
    <ktk_s>不需要开启空调</ktk_s>
    <xcz_s>洗车后至少未来4天内没有降水、大风、沙尘天气，保洁时间长。非常
    <gm>1</gm>
    <gm_l>低发期</gm_l>
    <gm_s>环境温度较高，要提防长时间在空调环境中引发的空调病；</gm_s>
    <ydl>4</ydl>
    <ydl_l>不太适宜</ydl_l>
    <ydl_s>天气较热，不太适宜户外运动；</ydl_s>
    <savestate_weather>2013-05-13</savestate_weather>
    <savestate life>2013-05-13</savestate life>
  
```

这又是一个 XML，然后用的是一些拼音首字母做了标签，把这个回复给用户估计会疯的，我们要进行一些整理，方法嘛就是用正则表达式来提取我们需要的内容，我这里提取的标签是 city（城市）、status2-status1（天气变化）、temperature2-temperature1（温度变化）、direction2-power2（风向风力）、chy\_shuoming（穿着建议）、savestate\_weather（信息发布时间）

第 54 行到 62 行就是提取这些数据的正则表达式，可以发现其实改动的只是签名和赋值的参数，如果大家还想加写数据的可以参照着提取；

第 64 行到 71 行判断天气变化是否相同，比如上面显示的天气 1 和天气 2 其实都是阴，如果不做判断就会返回给用户“阴转阴”，非常 2，所以这里判断如果两个天气是一致的则将任意一个天气赋值给\$w\_status 变量，否则就按照天气 2 转天气 1 赋值给\$w\_status 变量，最后输出时用\$w\_status 这个变量。

第 73 行到 81 行新建一个数组，将前面获取到的天气数据添加为数组元素，格式就是\$weather\_res=array();然后在括号里用逗号分割每个天气数据，最后一个后面不要加逗号，这样做的好处是避免代码行过长，而且很清晰。

第 82 行，将数组用 implode()函数转化成数组，用"\n"这个换行符来分割。

其实 73 到 82 行完成的就是将所有数据拼接成一个字符串，使用“.”一个个拼也可以，不过效率低下而且代码也不够清晰。

全部拼接完成就可以输出了，在手机上效果如下：



后面的就不详细说了，都有标注，都是些判断提示语句，大家可以自己学着看，不懂的可以直接在后台问。另外强调一点，一般天气预报是拿中国天气网

接口做的会比这个更好，能够显示县区级的天气，但是需要搞个城市代码表，留到以后说吧，新浪的接口可以直接用城市名查就先演示下。

### 三、退订消息接收

不知道还有多少人记得我曾写过一篇《那些离开的朋友们》，是因为微信把退订消息接口开了，每天看到不少人退订心里有点小难过写的，到今天我看了下差不多有 2000 个退订用户了，估计是我现在写的东西不合他们胃口。

退订消息接口的代码很简单，跟用户订阅一样是一个事件类型的消息，只是事件类型的标示是"unsubscribe"，退订用户最好是配合数据库来记录，我这里给大家的是一个保存成文件的例子，如下图：

```
292
293 //事件消息
294 if($form_MsgType=="event")
295 {
296     //获取事件类型
297     $form_Event = $postObj->Event;
298     //退订事件
299
300     if($form_Event=="unsubscribe")
301     {
302         //新建存储类
303         $s = new SaeStorage();
304         //写入文件
305         $s->write( 'weixincourse' , $fromUsername.".txt" , date("Y/m/d H:i:s")."退订! " );
306         exit;
307     }
```

代码添加的位置请看仔细了，在事件消息判断里的获取事件类型之后。代码很短，解释如下：

第 300 行，判断事件类型标示是否为"unsubscribe"，大家可以看下订阅消息是"subscribe"；

第 303 行，新建一个 Storage 的对象；

第 305 行，写入文件，文件名为退订用户的 OPENID，文件后缀是.txt，内容我用了退订时间。

当用户退订后在 **Storage** 的存储空间里就会有一个文件生成。

微信公众平台开发的基础教程差不多已经讲完，后面还会讲下数据库和云存储平台的使用，然后就会是完整案例，之所以花那么长时间讲那么细，是希望大家能够先熟悉一些基础的编程，有一个循序渐进的过程，所有复杂的程序其实都是这些基础模块组成，如果你觉得看不懂，那可能是没有跟着动手做，后面的教程估计会跟不上，**so**，有时间就动动手吧！**anyway**，我会把教程坚持写完，因为很多人有期待。

今天讲的源码放到服务器上供大家下载，请在 **ZTalk** 里回复“微信代码”获取下载地址。视频来不及了，一宿没睡觉，白天再抽空拍下吧，欠着总是不舒服的.....



## 《微信公众平台入门到精通》Vol.13



### 第十二章 公众账号接收非文字消息

前面写的教程都是些基础的功能介绍，是为了后面能够整合起来做一些完整的应用，相信很多人应该都消化的差不多了，接下来要开始进入实际案例了，我自己规划的是三个案例：

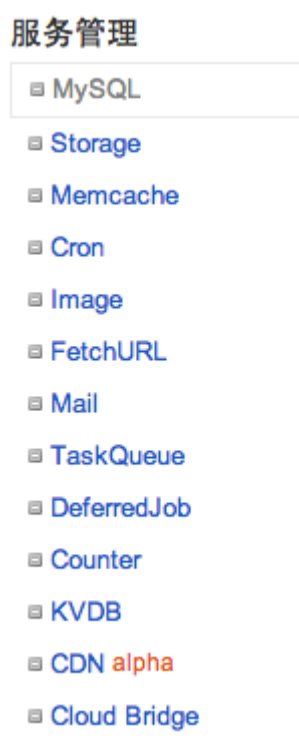
- 1、公司通讯录。知识点：微信账号绑定、数据库操作等
- 2、媒体类应用。知识点：文章分类、分页查询、模糊查询等
- 3、餐馆系统。知识点：餐馆地址查询、下单、计算价格等

今天先把说案例前最后一点内容讲掉。

#### 一、SAE 上数据库的使用

想要做一些真正能用起来的微信应用我们就必须要使用数据库，当然有人说我可以使⤵XML 或者数据、文件这些，但其实质是一样的，就是要对数据进行存储、查询和管理。

SAE 上是使用的 MySQL 数据库，关于 MySQL 的介绍我就不写了，各位自己百度吧，首先是启动 MySQL 的服务，大家进入到应用的界面后在左栏的找到如下图：



点击 MySQL 选项，然后右边会出现如下图，这里要注意的是每个 MySQL 服务都是跟单个应用捆绑的，即如果有多个 SAE 要使用数据库仍然要重新初始化：

## 使用说明

此帐号仅能在SAE平台上使用，不能从外部连接  
该应用的MySQL服务还没有初始化

[点此初始化Mysql](#)

点击初始化 MySQL 按钮，会弹出保护密码框，输入保护密码后再点击会提示开启成功，页面可能不会自动刷新，可以手动刷新下页面，然后就可以看到MySQL 服务已经启动了，如下图：

## 使用说明

此帐号仅能在SAE平台上使用，不能从外部连接  
我们建议开发者使用SaeMySQL操作数据库  
如果您想自己实现数据库相关操作，可以使用以下常量：

用户名 : SAE\_MYSQL\_USER  
密 码 : SAE\_MYSQL\_PASS  
主库域名 : SAE\_MYSQL\_HOST\_M  
从库域名 : SAE\_MYSQL\_HOST\_S  
端 口 : SAE\_MYSQL\_PORT  
数据库 ① SAE\_MYSQL\_DB ②

[DRO ③](#) [ATABASE ④](#) [管理MySQL](#)

[跨应用授权](#)

[服务自检](#)

这个操作生成了一个新的数据库，每个数据库里是可以有很多张表的，数据呢就是存放在表里面，大家可以把它想象成为一个文件夹，这个文件夹里有很多XLS 表格，每个表格里都有不同的数据。

1. 删除该数据库并关闭数据库服务，请谨慎操作。
2. 管理 MySQL 数据库，如建表。
3. 有时候 SAE 的应用之间需要互相调用数据库，而每个数据库只能是本应用调用，使用这个就可以跨应用操作了。
4. 检查数据库是否正常运作。

点击管理 MySQL 按钮进入数据库管理界面，SAE 使用的是 PHPMyAdmin 这个在线数据库管理工具，这个工具非常好用，基本使用 MySQL 的人都用这个，界面很直观，功能很强大。建表这个事情咱们放到第一个案例里讲，数据库先到这里。

## 二、SAE 上缓存的使用

缓存这个词相信很多人都是听说过的，缓存在计算机的软硬件里其实都存在，比如 CPU 里有缓存，浏览网站时有缓存，它的作用是把最常用的数据从存储器里提出来临时放在缓存里，就像把工具和材料搬上工作台一样，这样会比用时现去仓库取更方便。

那么在应用开发时使用缓存的场景一般是保存数据库的查询结果，比如 ZTalk 的每日一曲每天都是固定的一首歌，这首歌会根据当天的日期，然后从数据库里把歌曲信息查找出来然后返回给用户。

每天大概有差不多 2000 多位朋友会发送 V 来听歌，也就是说要进行 2000 多次的查询操作，而返回的结果是一样的，这种查询 2000 次是没有影响的，但如果是 2 万次、20 万次呢？所以我是将每天的查询结果做了缓存，6 个小时更新一次，这样就可以降低开销。

并且缓存的读写速度都是大大高于数据库的，结果能够更快的返回给用户，用户在使用的时候体验也会更加的好。

SAE 上用的缓存是 Memcache，这是个高性能的分布式的内存对象缓存系统，具体资料也请百度吧，首先是启动服务，大家进入到应用的界面后在左栏的找到如下图：

### 服务管理

- MySQL
- Storage
- Memcache
- Cron
- Image
- FetchURL
- Mail

点击 Memcache 选项，然后右边会出现如下图，

### 帐号信息

未初始化

点此初始化MC

### 使用说明

如果您已经初始化了MC,那么在应用中,您只需先调用`memcache_init()`(无参数,该函数完成了memcache的connect任务),然后就可以正常使用Memcache了。

点击初始化 MC 的按钮后会出现要求填写缓存容量的窗口，如下图：

初始化MC

10

M

提交

约 1.95 云豆/天

这个容量是指 Memcache 的大小，当你往缓存里存放的数据超过 10M 时，Memcache 会自动删除一些数据，容量选的越高每天消耗的云豆越多，我选的是 10M，主要就是存放一些常用的数据库查询结果，文字为主，基本够用。选择提交后如下图：

#### 帐号信息

配额: 10 M

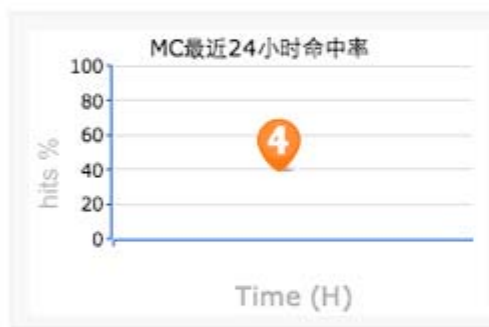
状态: alive

调整容量

服务自检

禁用MC

#### 使用情况



#### SET

设置为以上数据

#### GET

key

获取对应的数据

#### FLUSH

清空该应用的MC

该操作将清空本应用的MC中的所有信息,请谨慎使用!

1. 调整缓存容量。

2. 监测 Memcache 是否正常工作。
3. 停止 Memcache。
4. 最近 24 小时内缓存的命中率，就是指用户是发出查询指令时数据是通过缓存得到的还是通过数据库得到的，一般好的命中率应该在 50%以上，如果很低说明设置的缓存机制有问题或者有些操作不需要设缓存。
5. 缓存的内存驻留时间，缓存一般是用来做临时存放用的，所以要在适当时候回收。
6. 用来测试缓存的生成，在 KEY 这个输入框里是缓存名，上面是存放的数据，大家可以分别填一下试试。
7. 测试生成缓存后读取，可以将前面设置的缓存名填写到 KEY 的输入框点击 GET 按钮，就可以看到前面存放的数据了。
8. 点击后所有的缓存将全部清空，然后重新根据程序建立。

Memcache 的使用以及缓存的生成都是通过代码实现的，我们将在后面案例里实际运用。

## 二、第三方云存储平台的使用

有很多朋友在后台说，为什么听 ZTalk 的音乐一点都不卡，而自己写的程序放出来的音乐很卡，很简单就是 SAE 的 Storage 非常垃圾，特别是给普通开发者用的，连 CDN 都没有怎么可能快。

ZTalk 的音乐其实是放在又拍云存储上的，还有微信教程的代码，这是一个专业的第三方云存储平台，主要可以用来存放一些图片、音乐、文档等体积较大的文件，朋友们如果开发应用中需要用到大文件的下载、使用可以考虑。后面案例里会有图片上传的使用，会使用又拍云的代码。

**别说我给人打广告，在我需要云存储平台的时候人主动免费提供，所谓滴水之恩当涌泉相报，如果有其他第三方云存储平台想做宣传的请提供给 ZTalk 的朋友们免费使用邀请码若干！**

又拍云的网址是 <http://www.upyun.com/>，进入后如下图：



目前又拍云存储的使用是要申请的，还要人工审核，比较麻烦，不过我在跟他们谈让给些免费名额，大家先申请了再说吧，点击立即申请进入，如下图：



## 免费测试

FREE TRIAL

提交免费测试申请表单，即可获得免费体验机会

管理员帐号:	<input type="text" value="weixincourse"/>
密码:	<input type="password" value="....."/>
确认密码:	<input type="password" value="....."/>
邮箱:	<input type="text" value=".....m"/>
帐号类型:	<input type="radio"/> 企业 <input checked="" type="radio"/> 个人
姓名:	<input type="text" value="微信教程"/>
联系手机:	<input type="text" value="....."/>
联系QQ/MSN:	<input type="text" value="....."/>
网站域名:	<input type="text" value=".....m"/>
<input type="button" value="提交申请"/>	

填表格我就不累述了，就提两个：

- 1、账号类型请选个人比较方便。
- 2、网站域名大家干脆填 SAE 上的应用域名吧。

提交申请后就进入后台，由于没有审核通过还不能进行任何操作，我先拿自己的给大家讲解下，如下图：

[首页](#)[空间管理](#)[操作员管理](#)[流量统计](#)[日志分析](#)[刷新](#)

您的帐号试用已到期，如果试用得不错，您可以 [点击购买](#)，或者 [和您的销售专员联系](#)

管理员帐号：mrzeta

存储配额：购买 0 / 剩余 -162.27 MB (已超额) [购买存储配额和流量](#)

存储配额有效期：2013-03-27 (试用已过期)

剩余可用流量：-17.76 GB (已超额)

占用空间总计：162.27 MB / 0 可用：0%

空间 ztalk-music:	145.67 MB
空间 ztalk-image:	16.58 MB
空间 weigirl:	0
空间 weinxincourse:	23.09 KB

1

[创建空间](#)

上面那些乱七八糟提示的东西咱不用理他，先点击创建空间来建立咱们的存储空间，点击后出现下图：

创建空间

名称

weinxincourse1

类型

☒ 文件类空间 ☐ 图片类空间

容量

100

MB (\* 留空或零表示无限制)

说明：

空间创建后，空间名称和空间类型不允许修改！

文件类空间：上传文件类型不限，无缩略图功能；

图片类空间：仅支持上传图片格式文件，支持缩略图功能；

确定

取消

先填写空间名称，然后选择空间类型，这里图片和文件我推荐选择文件，图片类空间只是多了一个可以生成缩略图的 API 接口，其他没用，容量自己估计，

如果是放音乐 1000M 起吧。设置完成后点击确定后这个空间就生成了，可以看到左边栏多了一个选项，如下图：



点击新建立的空间名称就进入空间管理，如下图：



这里首先要做的是给空间授权一个管理员，点击标示为 2 的地方，有两种选择，一是授权现有管理员，二是新增管理员并授权。授权完成后该管理员就有了管理这个空间的权限。

在又拍云存储里文件上传跟 SAE 是不一样的，它没有图形化管理页面，文件上传两种方式，一是通过接口代码，二是使用 FTP，FTP 软件网上很多，大家可以自己下载，同时也有很多 FTP 的使用教材，我就不再重复了。

点击标示为 1 的地方，可以看到 FTP 的信息，如下图：



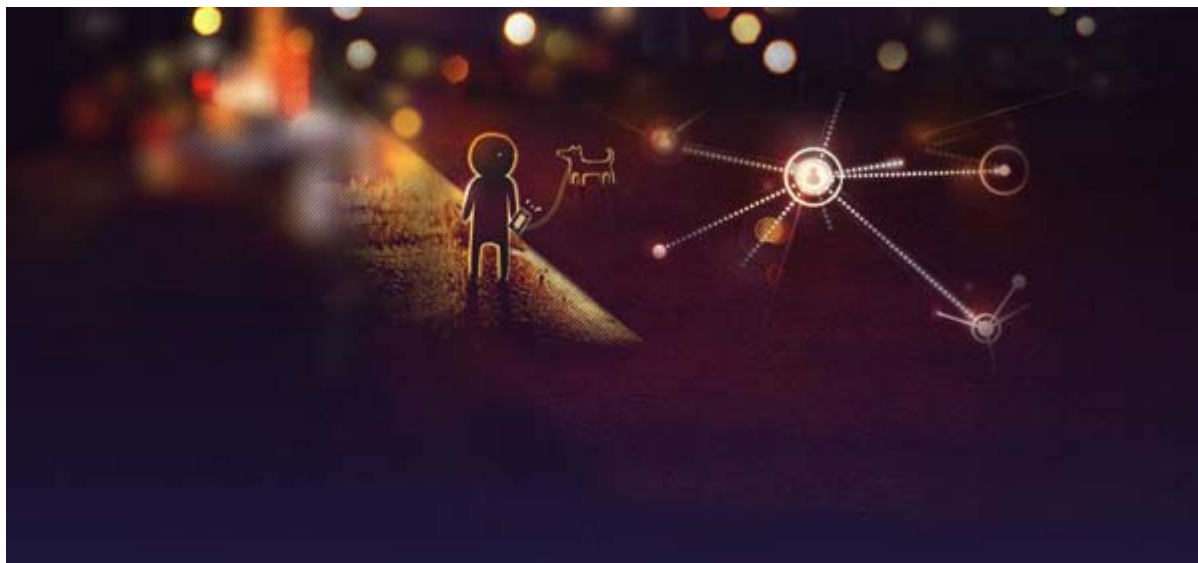
1. FTP 的访问地址，有 4 个可选择，如果搞不清楚自己是什么宽带的选第四个自动判断吧。
2. FTP 账号和密码就是授权的管理员的账号和密码。
3. 外网访问的连接地址，如果在空间里有个 a.txt 文件，我们要访问时就是访问地址+/1.txt。

大家如果要使用又拍云存储的，可以自行注册，注册完后在后台回复“又拍云+自己申请的账号名称”，我统一扔给他们，让他们给大家开权限给优惠。

今天就先讲到这里，后面开始进入案例实操，当然还欠两期的视频，我抓紧录制.....



## 《微信公众平台入门到精通》Vol.14



今天开始进入实际案例阶段，每个案例我争取花两到三章的篇幅写完，请大家耐心守候，不要催我，虽然创业了但是开放分享的精神是不会动摇的，虽然最近商业化有点多，但自媒体人首先要做的依然是传递有价值的内容。

一般的编程书上举第一个例子都是用的学生名册，对大家没啥用，咱们实在点拿公司通讯录来做第一个案例，在这个案例里会有如何建表，如何绑定微信账号，如何做查询等。

### 第十四章 公司通讯录开发

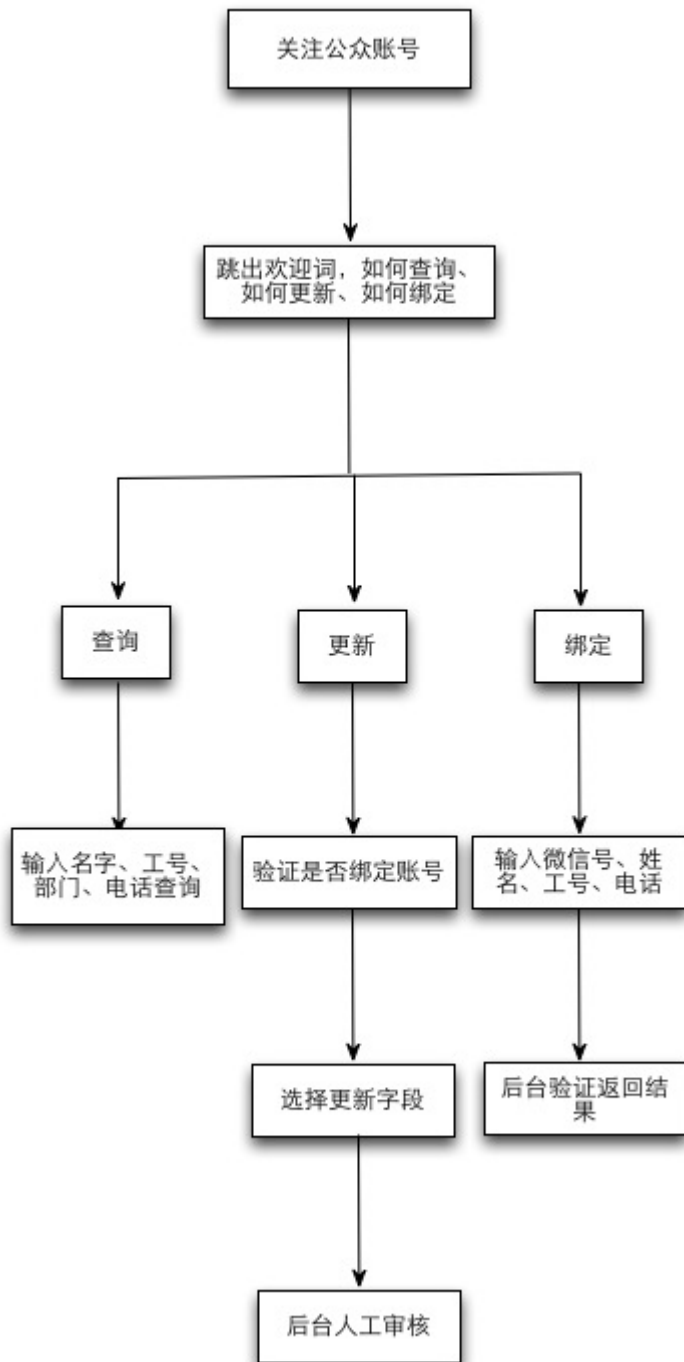
#### 一、公司通讯录产品设计

做任何开发前我们都要做一个产品的规划，公司通讯录虽然看起来很简单，但是这个步骤依然少不了。

通讯录的功能：

1. 后台通讯录管理，包括分级、录入、修改、删除；
2. 后台日志查询，包括后台操作记录和前台修改记录；
3. 公司员工微信账号与通讯录绑定；
4. 已绑定账号可查询；
5. 部门、关键词等查询；
6. 前台已绑定账号可更新自己信息。

前台逻辑图：



数据库的设计



员工表：序号、姓名、工号、照片、性别、生日、手机、电话、部门、邮箱、微信号、员工状态

部门表：序号、部门名、上级部门序号、部门状态

日志表：序号、操作内容、操作时间、操作人、操作类别

管理员表：序号、用户名、密码

今天第一天上班事情一大堆，晚上还面试了一位朋友，为了赶在 **12** 点前发布不占用明天的群发额度，只有先抱歉下，流程图也没画的很规范，后面的会连夜写，明天发布，对不起了 **ZTalk** 的朋友们。

另外如果大家对这个通讯录设计有什么不足的也请提下意见，我在详细教程里再进行修改！

## 《微信公众平台入门到精通》Vol.15



紧接着 14 期的内容，在基本规划做好以后咱们先把数据库建好。昨天有人建议我按照 QQ 企业邮箱的来做，这个可能太复杂了，怕大家更加迷茫，我做案例更多是希望朋友们通过案例举一反三，从简单的产品推导出复杂的应用。

### 第十四章 公司通讯录开发

#### 二、公司通讯录数据库建立

第 13 期里有 SAE 数据库的介绍，如果没有开通的请先移步把 SAE 的数据库服务开启，点击“管理 MySQL”进入到数据库的图形管理界面，如下图：



数据库里目前没有任何数据表，所以我们先来建立第一张数据表“员工表”，取名为“roster”，员工表我调整了下结构增加了几项，最后的项目为：序号、姓名、工号、照片、性别、生日、手机、电话、部门、邮箱、微信号、微信OPENID、员工状态、入职日期、添加时间、更新时间、记录状态。

在建表之前其实还要设定每个字段项的属性，我这里简单写一下，后面在解释建表的时候大家可以参照看：

字段名	类型	长度	其他
-----	----	----	----

序号	数字	5	自增、主键
----	----	---	-------

姓名	字符	10	
----	----	----	--

工号	字符	10	
----	----	----	--

照片	字符	50	
----	----	----	--

性别	数字	1	0 为女，1 为男
----	----	---	-----------

生日	日期		
----	----	--	--

手机	字符	15	
----	----	----	--

电话	字符	15	
----	----	----	--

部门	数字	4
邮箱	字符	50
微信	字符	30
openid	字符	50
员工状态	数字	1 1 为在岗，-1 为离职 2 为休假
入职	日期	
添加	时间	
更新	时间	
记录状态	数字	1 1 为正常 0 为删除

字段数根据上面的规划应该是 15 个字段，所以应该填写 15，但我这里填写 3 是为了后面切图片和讲解的方便，大家如果填写 15 看到的样式会不同，没有关系，另外建立过程中还可以增加字段数的。

点击执行按钮，就切换到表设计界面，如下图：

w.rdc.sae.sina.com.cn:3307 ▶ app\_weixincourse ▶ roster

字段	roster_id	roster_name	roster_number
类型	SMALLINT	CHAR	CHAR
长度/值1	5	10	10
默认2	无	无	无
整理			
属性	UNSIGNED		
空	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
索引	PRIMARY	---	---
AUTO_INCREMENT	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
注释	序号	姓名	工号

表注释: 员工表      存储引擎: MyISAM      整理: utf8\_general\_ci

在上图中每一列就是一个字段，如果大家前面填写字段数超过 5 的则视图中每一行就是一个字段，但不管如何显示，每个字段都有下面这些属性：

1. 字段名称，用来描述字段的名称，它可以用中文、英文字母、数字等字符来描述。但是建议不用中文或者纯数字，命名最好使用表名+下划线+该字段的英文名或者拼音缩写。
2. 字段类型，用来限定数据格式，同一字段的数据类型都是一样的。点击类型的选项菜单会出来一大堆，不用怕很多我们是不太会用的，常用到的类型如下：

**TINYINT**：一个很小的整数。有符号的范围是-128 到 127，无符号的范围是 0 到 255。

**SMALLINT**：一个小整数。有符号的范围是-32768 到 32767，无符号的范围是 0 到 65535。

**MEDIUMINT**：一个中等大小整数。有符号的范围是-8388608 到 8388607，无符号的范围是 0 到 16777215。

**INT:** 一个正常大小整数。有符号的范围是-2147483648 到 2147483647，无符号的范围是 0 到 4294967295。这个经常用来保存 **UNIX 时间戳**。

**CHAR:** 一个定长字符串，当存储的字符长度小于设定长度时总是用空格填满右边到指定的长度，超过则自动截断。最多存放 255 个字母或者汉字，也就是不超过 255 个字符的都可以设成 **CHAR** 类型。这里不推荐用 **VARCHAR**，因为处理速度不给力，对于存储空间不敏感的可以忽略。

**TEXT:** 最多可存放 65535 个字符，一般用于保存文章等。该字段无需填写长度。

**DATETIME:** 一个日期和时间组合。支持的范围是'1000-01-01 00:00:00'到'9999-12-31 23:59:59'。MySQL 以'YYYY-MM-DD HH:MM:SS'格式来显示 **DATETIME** 值。该字段无需填写长度。

**DATE:** 一个日期。支持的范围是'1000-01-01'到'9999-12-31'。MySQL 以'YYYY-MM-DD'格式来显示 **DATE** 值。该字段无需填写长度。

3. 字段长度，每一个字段需要存放的数据可能最大位数，比如序号是用来做数据条数计数的，员工一般也就几万人了，所以这里设了 5；姓名的话一般最多 4、5 个汉字，但是为了避免有奇葩的名字所以设了 10 位，其实如果考虑老外名字的话建议设定为 30。

4. 字段默认值，指如果这个字段在添加一条新的数据时没有获得赋值时将会保存成字段默认值。
5. 字段语言格式，点击这个选项卡就可以看到一大串各种语言格式，不用管他，我们可以总的设定一个就可以了。除非是有特殊需求。
6. 字段属性，这个主要是针对数字型字符的，前面介绍的几个数字类型里有写无符号和有符号，就是在这里设定。
7. 字段是否可为空，打钩标示可以为空，一般不打钩，因为含有空值的列很难进行查询优化。
8. 字段索引，这个很重要，索引将影响数据库查询的效率，一般会将常用的几个搜索关键字段或者排序字段建立索引。不推荐字符串字段做索引，另外主索引只能有一个，自动增长的字段会被默认作为主索引。
9. 自增属性，在数据库应用，我们经常要用到唯一编号，以标识记录，最好的办法就是将其中一个字段设为自增属性，每新增一条数据记录是都会自动加一，其计数顺序不受数据删除影响，如果有 10 条数据，删除其中一条后再新增一条，这条数据的序号不是 10，而是 11。
10. 字段注释，方便日后了解该字段的作用。
11. 数据表注释，方便日后了解该数据表。

12. 存储引擎，常用的有三种分别是 MyISAM、MEMORY 和 InnoDB，其中 MyISAM 是管理非事务表。它提供高速存储和检索，以及全文搜索能力，是默认的存储引擎，一般选这个就成。

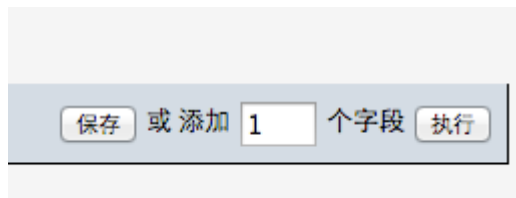
13. 数据表语言格式，前面字段里也有语言格式设置，其实只要这里设一个，所有的字符串字段就都默认用这个语言格式了，我们这里选择 utf8\_general\_ci，就是 UTF-8 的语言格式。

了解数据表的一些基本知识后我们再来看员工表的每个字段属性实际应该如何设定，截图在手机上可能会有些小，大家可以到 [wx.qq.com](http://wx.qq.com) 上找到 ZTalk 的公众号里输入微信教程查看。

roster_id	SMALLINT	5	无		UNSIGNED
roster_name	CHAR	10	无	utf8_general_ci	
roster_number	CHAR	10	无	utf8_general_ci	
roster_pic	CHAR	50	无	utf8_general_ci	
roster_sex	TINYINT	1	无		UNSIGNED
roster_birthday	DATE		无		
roster_mp	CHAR	15	无	utf8_general_ci	
roster_phone	CHAR	15	无	utf8_general_ci	
roster_class	SMALLINT	4	无		
roster_mail	CHAR	50	无	utf8_general_ci	
roster_wechat	CHAR	50	无	utf8_general_ci	
roster_openid	CHAR	50	无	utf8_general_ci	
roster_status	TINYINT	1	无		
roster_in	DATETIME		无		
addtime	DATETIME		无		
edittime	DATETIME		无		
status	TINYINT	1	无		



全部设定好后点击右下角的保存即完成建表，如果字段需要增加的可以点击旁边的执行按钮增加一条或者多条新字段，如下图所示，其中修改数字可调节新增的条数。



啰嗦一大堆估计很多朋友都云里雾里了，咱们还是搞简单点的吧，建立数据表除了上面那种方法还有一种是通过 SQL 语句来建立，如下图：

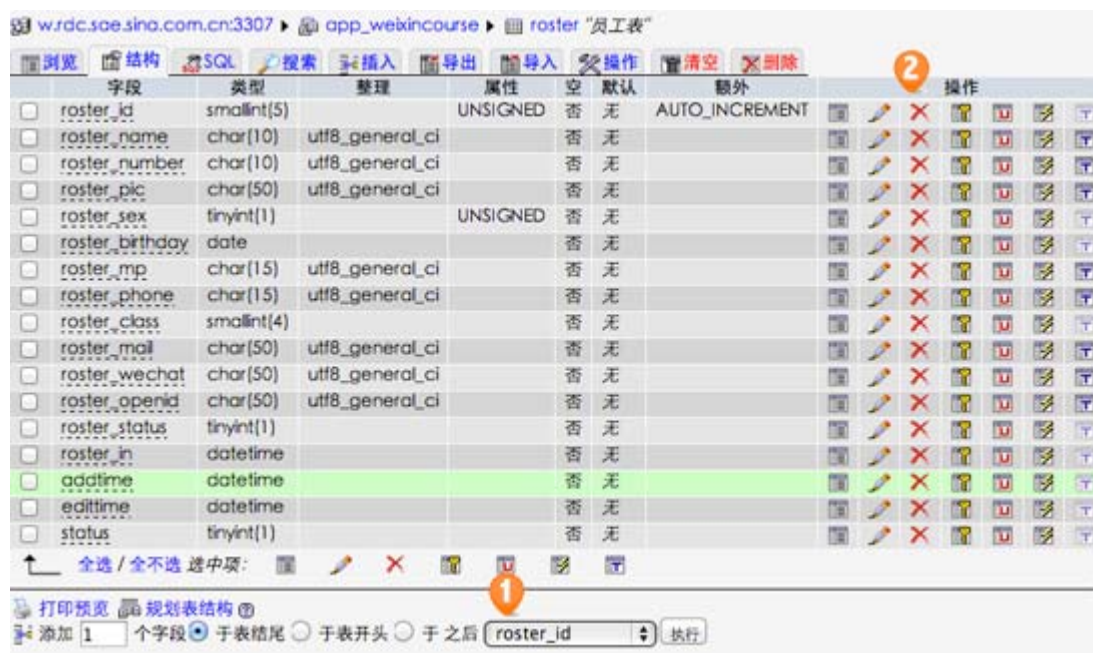


点击选项卡的 **SQL**，然后在输入框中写入以上的代码就直接完成表的建立，当然我知道大家这个代码是写不错来的，所以这节课的课件里有.....输入“微信代码”就可以下载。

建立成功的话刷新网页就可以看到员工表已经建立成功了，如下图



1. 数据表记录浏览，当里面没有数据时该图标灰色不可点。
2. 数据表结构查看，点击后可以看到，我们所有设定的字段属性，如下图：



标示为 1 的地方可以新增字段，标示为 2 的地方是对某个字段进行编辑修改删除等操作。

3. 数据表查询，当里面没有数据时该图标灰色不可点。
4. 新增一条记录，点击后进入下图，可以像我这样随便输入一些数据然后点执行就给数据库插入一条数据了，注意的是 `roster_id` 这个字段空着就可以了，系统会自动赋值：

w.rdc.sae.sina.com.cn:3307 ▶ app\_weixincourse ▶ roster "员工表"

[浏览](#)
[结构](#)
[SQL](#)
[搜索](#)
[插入](#)
[导出](#)
[导入](#)
[操作](#)
[清空](#)
[删除](#)

字段	类型	函数	空	值
roster_id	smallint(5) unsigned	<input type="text"/>	<input type="checkbox"/>	<input type="text"/>
roster_name	char(10)	<input type="text"/>	<input type="checkbox"/>	青龙老贼
roster_number	char(10)	<input type="text"/>	<input type="checkbox"/>	001
roster_pic	char(50)	<input type="text"/>	<input type="checkbox"/>	<input type="text"/>
roster_sex	tinyint(1) unsigned	<input type="text"/>	<input type="checkbox"/>	1
roster_birthday	date	<input type="text"/>	<input type="checkbox"/>	2013-01-23
roster_mp	char(15)	<input type="text"/>	<input type="checkbox"/>	13701111111
roster_phone	char(15)	<input type="text"/>	<input type="checkbox"/>	13701111111
roster_class	smallint(4)	<input type="text"/>	<input type="checkbox"/>	1
roster_mail	char(50)	<input type="text"/>	<input type="checkbox"/>	test@qq.com
roster_wechat	char(50)	<input type="text"/>	<input type="checkbox"/>	test
roster_openid	char(50)	<input type="text"/>	<input type="checkbox"/>	test
roster_status	tinyint(1)	<input type="text"/>	<input type="checkbox"/>	1
roster_in	datetime	<input type="text"/>	<input type="checkbox"/>	2013-05-28 16:19
addtime	datetime	<input type="text"/>	<input type="checkbox"/>	2013-05-28 16:56
edittime	datetime	<input type="text"/>	<input type="checkbox"/>	2013-05-28 16:56
status	tinyint(1)	<input type="text"/>	<input type="checkbox"/>	1

[执行](#)

新增成功后点击数据表浏览就可以看到这条记录，如下图：

w.rdc.sae.sina.com.cn:3307 ▶ app\_weixincourse ▶ roster "员工表"

[浏览](#)
[结构](#)
[SQL](#)
[搜索](#)
[插入](#)
[导出](#)
[导入](#)
[操作](#)
[清空](#)
[删除](#)

✓ 显示行 0 - 0 (1 总计, 查询花费 0.0010 秒)

```

SELECT *
FROM `roster`
LIMIT 0 , 30

```

☐ 概要 [\[ 编辑 \]](#) [\[ 解释 SQL \]](#)

显示:  行, 开始行数:

以  模式显示, 并且在  行后重复标题

+ 选项

	roster_id 序号	roster_name 姓名	roster_number 工号	roster_pic 照片	roster_sex 性别	roster_birthday 生日
<input type="checkbox"/>	1	青龙老贼	001		1	2013-01-23

5. 清空数据表，请谨慎.....
6. 删除数据表，请谨慎.....
7. 新建数据表。

关于数据表的建立基本就到这里了，昨天写的规划是 4 张表，但后来考虑了下觉得第一个案例没必要搞的那么复杂，所以管理员和日志表先去掉了，就留员工表和部门表，并做了适当的调整。

部门表我给大家列下属性，大家可以先尝试自己建一下，当然我也提供了代码供下载。

部门表：

字段名	类型	长度	其他
序号	数字	5	自增、主键
部门名	字符	20	
成员数	数字	5	
上级序号	数字	5	
添加	时间		
更新	时间		
记录状态	数字	1	1 为正常 0 为删除

数据表名字请使用 `class`，字段名请参考我的，如果换别的话，后面程序里也要相应更换。

浏览	结构	SQL	搜索	插入	导出	导入	操作	清空	删除
	字段	类型	整理	属性	空	默认	额外		
<input type="checkbox"/>	class_id	smallint(5)		UNSIGNED	否	无	AUTO_INCREMENT		
<input type="checkbox"/>	class_name	char(20)	utf8_general_ci		否	无			
<input type="checkbox"/>	class_number	smallint(5)			否	无			
<input type="checkbox"/>	class_fid	smallint(5)			否	无			
<input type="checkbox"/>	addtime	datetime			否	无			
<input type="checkbox"/>	edittime	datetime			否	无			
<input type="checkbox"/>	status	tinyint(1)			否	无			

今天就先到这里，回复“微信代码”查看今天的源码，请注意，要把数据库名称 **app\_weixincourse** 替换成你自己的数据库名称，不然无法新建！如果打开有乱码的请修改网页编码为 **utf-8**！有不清楚的请留言，明天讲后台的录入及数据库操作。

## 《微信公众平台入门到精通》Vol.16



昨天有几位朋友在后台给我提了些建议，数据库这块貌似讲的太细了，按照这个讲法得写 100 期了。

我承认我有点傻了，其实大部分人不用了解这个，真要学数据库也不是这样学的，所以我还是加快点步骤吧。

昨天的员工表设计上少了个字段就是头衔，大家可以自己在数据库管理的 SQL 选项输入框中使用这个语句：

```
ALTER TABLE `roster` ADD `roster_title` CHAR(20) NOT NULL COMMENT '头衔' AFTER `roster_name`
```

或者重新下载昨天的数据库代码执行，执行前先删除之前建立的 roster 数据表。

### 三、公司通讯录后台开发

我设计的公司通讯录是先人工输入到后台，然后前台进行绑定、查询、更新等操作，今天先讲后台录入这块，主要包括部门设置和员工管理两个页面，所有的开发都在 **sae** 的代码管理中进行，所以请进入 **SAE** 的应用里的代码管理。

在开发前我们先要上传一个文件“**base-class.php**”，这个文件主要是用来做传递参数的过滤以及格式验证，比如是否邮箱、手机等是否正确输入，文件请回复“微信代码”下载！

#### 3.1、部门设置

第一个要开发的模块是部门设置，一般的公司架构都是多层级的，比如总经理-》部门-》小组，因此我们的部门设置里也要具备多层级的功能。

首先第一个页面部门录入页面，我们新建一个文件，命名为 **class\_add.php**，主要实现功能为添加和修改部门名称及上下级分类。代码如下：



```

1 <!DOCTYPE HTML>
2 <html>
3 <HEAD>
4 <meta charset="utf-8">
5 <TITLE>部门添加/修改</TITLE>
6 </HEAD>
7 <body>
8
9 <?php
10 include("base-class.php");
11
12 //新建数据库类
13 $mysql = new SaeMysql();
14
15 //获取部门id号传入
16 $class_id=intval($_GET['class_id']);
17
18 //获取操作标识传入
19 $action=$_POST['action'];
20 $action= string::un_script_code($action);
21 $action= string::un_html($action);
22
23 //判断是否修改，如果传入了部门id，进行数据库查询获取全部内部
24 if($class_id)
25 {
26     $class_value=$mysql->getLine("select * from class where class_id=$class_id");
27     if(!$class_value)
28     {
29         echo "<script>alert('无此部门');history.back();</script>";
30         exit;
31     }
32 }
33
34 //如果获取到操作标识，进行录入或者修改操作
35 if($action=="update")
36 {
37     //获取表单传入数据
38     $old_class_id=$_POST['class_id'];
39     $class_name=$_POST['class_name'];
40     $class_fid=$_POST['class_fid'];
41     //传入上级部门
42     $old_class_id=intval($old_class_id);
43     $class_name= string::un_script_code($class_name);
44     $class_fid= intval($class_fid);
45     //默认参数
46     $rowtime=date("Y/m/d H:i:s",time());
47     //如果是修改
48     if($old_class_id)
49     {
50         //修改部门名称、所属、更新时间
51         $sql = "update class set class_name='$class_name',class_fid='$class_fid',edittime='$rowtime'
52         where class_id=$old_class_id";
53         $mysql->runSql($sql);
54     }
55     else
56     {
57         //新增
58         $sql = "insert into class (class_name,class_fid,addtime,edittime,status) values ('$class_name',
59         '$class_fid','$rowtime','$rowtime',1)";
60         $mysql->runSql($sql);
61     }
62 }
63 if( $mysql->errno() != 0 )
64 {
65     echo "<script>alert('".$mysql->errmsg()."');history.back();</script>";
66     exit;
67 }
68 else
69 {
70     echo "<script>alert('更新成功!');location='class_add.php?class_id=$old_class_id';</script>";
71     exit;
72 }
73
74 }
75
76 $class_list=$mysql->getData("select class name,class id from class order by class fid asc");
77
78 <!-- 页面名称 -->
79 <!-- 部门添加/修改 --> <a href="class_manager.php">返回</a></h1>
80 <!-- 表单开始 -->
81 <form action="?" method="post" name="class_add" id="class_add" enctype="multipart/form-data">
82 <?
83 部门名称: <input type="text" value="<?=$class_value['class_name']?>" name="class_name">
84 </p>
85 <?
86 上级部门:
87 <select name="class_fid">
88     <option value="0">无上级部门</option>
89     <?php
90         //把所有部门列表出来
91         foreach($class_list as $value)
92         {
93             $class_select=($class_value['class_id']==$value['class_id'])? 'selected':'';
94             echo "<option value='".$value['class_id']."'>".$value['class_name']</option>";
95         }
96     </select>
97 </p>
98 <?
99 <!-- 隐藏参数，用来放置操作标识和修改的ID -->
100 <input type="hidden" name="action" value="update">
101 <input type="hidden" name="class_id" value="<?=$class_value['class_id']?>">
102 <!-- 表单提交 -->
103 <input type="submit" value="提交" />
104 </p>
105 </form>
106 </body>
107 </html>

```

后台页面是需要浏览器里打开操作的，因此需要有 **HTML** 代码部分，一般正规开发都会将数据操作、逻辑操作、网页模板分离，我这里图方便就混排了。

这个页面实现的是两个功能：一是新增部门，二是修改部门，两个是放在同一个页面里的，那么如何判断是新增操作还是修改操作呢？根据 **url** 里是否有传递过来部门的序号即 **ID** 号，如果有就是修改，没有就是新增，具体后面会有解释。

码解释如下：

第 1-6 行代码是 **html** 语言，头部格式都是固定的，其中第 4 行是告诉浏览器这个网页是 **utf-8** 编码的，第 5 行是网页的名字，会显示在浏览器顶部，当然一般还有两个重要参数，一个是 **Keyword**，就是关键字，还有一个是 **Description**，是网页说明，这两个都是做 **SEO** 时用的，这里不需要就没写了。

**PS: HTML 语言通常被称为 HTML 标签 (HTML tag)，HTML 标签是由尖括号包围的关键词，比如 <html>，通常是成对出现的，比如 <b> 和 </b>，标签对中的第一个标签是开始标签，第二个标签是结束标签。**

第 7 行是 **html** 页面的主体的开始标签 **<body>**，浏览器会将 **<body>** 与 **</body>** 之间的内容显示在浏览器页面。

第 9 行是开始是 **php** 的程序处理。

第 10 行是加载前面让大家上传的 **base-class.php**，后面会调用里面的函数。

第 13 行是新建一个数据库操作类，由于是在 sae 环境下开发，所以我直接用了 sae 提供的 mysql 类，这样就免去数据库链接什么乱七八糟的了。当然原始提供的函数并不丰富，也不适合代码迁移，一般是需要去封装一层的，我这里先偷个懒吧，同时也方便大家学习。

第 16 行是获取通过 URL 传递的参数 class\_id，即部门的序号。这里有个 intval 函数是将变量转化成整数，如果 class\_id 值为空则 intval(class\_id)=0；如果 class\_id 值为 1 则 intval(class\_id)=1；如果 class\_id 值为 abc12 则 intval(class\_id)=0，这里的作用是将转换数据格式同时防止注入。

PS：程序中数据传递主要通过两种方式获取 GET 和 POST，前者是获取 URL 的参数值，后者是获取表单传递的数据，前者安全性低效率高适合传输一些小数据，后者安全性高效率低适合传输一些大数据。

第 19 行是获取表单提交的参数 action 的值，用来判断是否提交了表单。

第 20、21 行是用来过滤操作变量 action 的值，由于 action 是字符串，所以得用字符串过滤规则，其中 un\_script\_code 是去除 js 代码，un\_html 是去除 html 代码。

第 24 行判断 class\_id 是否有值，如果有值即非 0 则表示修改序号为 class\_id 的部门信息；

第 26 行根据 class\_id 从数据库中提取详细数据，提取单条记录我们使用 sae 的 mysql 类下的 getLine 函数，我们前面已经新建了一个 mysql 类命名为 \$mysql，所以这里就是 \$mysql->getLine，后面括号里就是一句 sql 查询语句，意思是

select（选择）

\*（所有字段）

from（从）

class（名字为 class 的表里）

where（符合条件为）

class\_id=\$class\_id（class\_id 字段的值等于\$class\_id 的数据），获取到的数据是一个数组，赋值给\$class\_value。

第 27 行为判断是否获取到 class\_id=\$class\_id 的记录，!\$class\_value 表示该值为空没有获取到，那么就执行一个警告。

第 29 行输出一个 JS 语句，这个语句里<script>是 js 的标签，alert 是警告函数，用法是 alert("提示的文字内容")，history.back();是返回上一页，整句话就是警告没有该条记录，点击确定后返回到上一页，中断程序执行。

第 35 行是判断\$action 是否等于 update，即是否提交了表单。

第 38 行到第 44 行是获取从表单提交过来的数据，并进行格式化，其中 \$old\_class\_id 的值来源是在修改条件下，url 传递过来的 class\_id 值会在存放在一个隐藏的 input 元素中，然后表单提交时又将该值传递回来，也就是 \$old\_class\_id 其实是 url 传递过来那个 class\_id 的值。

第 46 行是获取当前时间，time()之前有讲过是获取时间戳，是一个 10 位的整数，然后用 date 函数格式化成年月日时分秒格式，后面要写到数据库里。

第 48 行是判断\$old\_class\_id 是否为空，即修改模式还是新增模式，如果有值则为修改模式。

第 51, 52 行, 更新数据的 sql 语言,

update 修改

class 名字是 class 的数据表

set 设置

class\_name='\$class\_name', 部门名修改

class\_fid='\$class\_fid', 上级部门修改

edittime='\$nowtime' 修改时间

where class\_id=\$old\_class\_id (符合条件 class\_id 字段的值等于\$class\_id 的数据)

大家可以看到这里修改的其实就是部门名字、上级部门, 同时记录下修改的时间,

由于 sql 语句本身就是一个字符串, 所以当赋值的变量为字符串需要用单引号包括起来, 否则执行会出错。

第 53 行是执行上面的 sql 语句, 只运行语句不返回结果, 函数为 runSql。

第 58、59 行, 新增数据的 sql 语句,

insert into 插入数据到

class 名字是 class 的数据表

(class\_name,class\_fid,addtime,edittime,status)字段名

values 赋值为

(' \$class\_name', ' \$class\_fid', ' \$nowtime', ' \$nowtime', 1)对应的值

这里可以注意到字段名和数据是一一对应的，分别是部门名、上级部门序号、添加时间、修改时间和记录状态（新增的设为 1 表示正常）。

第 63 行，runSql 函数虽然不返回数据结果但是会返回一个执行信息，\$mysql->errno() 如果是 0 则表示成功，否则不成功。

第 65 行执行不成功，输出警告信息。

第 70 行执行成功，输出成功提示，然后页面刷新，“location=”是 JS 函数，用来改变当前 url 并载入，链接地址为“class\_add.php”，我在后面又加了 class\_id=\$old\_class\_id，效果会变为如果是修改模式则重新加载一次修改后的部门信息，如果是新增模式则清空表单可以继续新增。

第 76 行提取所有有效的部门信息，主要是用来做上级部门选择用的，提取多条记录使用的函数是 getData，sql 语句的意思是：

select 选择

class\_name,class\_id 部门名和部门 ID

from 从

class 名字是 class 的数据表

where 符合

status=1 状态为正常的数据

`order by class_fid asc` 按照上级部门 ID 的升序排序，`order by` 表示排序，排序的字段是 `class_fid`，`asc` 是升序即从小到大。

第 80 行开始是 HTML 代码，`H3` 是标题标签，这里写了这页面的标题是“部门添加/修改”，后面跟的一个链接地址用来返回到部门列表页面，`a` 标签是链接标签，`href` 后面跟的是链接地址，`a` 开闭标签之间是页面显示文字。

第 82 行表单标签开始，介绍下参数 `action` 是指当提交表单时，向何处发送表单数据，这里是填写 URL，我这里写了？号，指向当前页发送表单，`method` 是以何种方式发送表单，有 `post` 和 `get` 两种，一般选 `post`，如果是 `get` 则表单会以 `url` 参数方式发送，`name` 是指表单名字，`id` 指该表单的唯一标示，`enctype` 是指表单上传前如何对表单数据进行编码，`multipart/form-data` 是指不对字符编码主要是在表单里有文件上传控件时使用。

第 83 行 `p` 标签是段落标签，即分段；

第 84 行部门名称的文本输入框，输入控件的标签是 `input`；

`type` 是控件属性，不同属性有不同的功能：`text` 是文本框可以输入文字、`file` 是文件上传控件、`button` 是按钮、`submit` 是提交表单按钮、`hide` 是隐藏控件；

`name` 是该控件的名称，提交表单时程序获取数据时就是用这个名称，可以看下第 38 行到 44 行，命名规则建议使用和数据表字段一样的名字，方便辨识；

`value` 是该控件的值，传递到程序里的就是这个值，我这里写了一段代码是用来在修改模式下输出部门名称的，在新增模式下该值为空。

第 87 到 98 行是一个选择控件，用来选择部门的上级部门，标签名为 `select`，`name` 为该控件名称；

每个 `option` 都是一个选择项，`value` 是该选项的数值，`option` 开闭标签之间是选项描述，最后传递的是 `value`。

`option` 还有一个属性是 `selected`，看名字就知道是选中，如果没有一个选项指定了这个值，则默认第一个为选中状态。因此代码里第一个选项是“无上级部门”，`value` 为 0。

在第 76 行我们将所有的部门数据拉取出来赋值给 `$class_list`，就在这里用了，这个 `$class_list` 是用数组存放数据的，所以这里用 `foreach` 将记录一条条的读取出来。

第 94 行是在修改模式条件下，我们希望被修改的记录上级部门名称是选中状态的，因此在循环输出所有部门时，如果上级部门 ID 和列表部门 ID 相同时则加上 `selected`，否则为空，赋值给 `$class_select`。然后在输出 `<option>` 时就加上这个字符串控制是否要选中。

这里我用了个判断式（条件）?成立:不成立;等于 `if(条件){成立}else{不成立}`

第 102，103 行是两个隐藏的输入标签，分别是放置 `action` 的值和 `class_id` 的值，这可以看到 `class_id` 的 `value` 是通过前面获取的 `url` 参数赋予的。

第 105 行就是提交按钮。



该页面访问是在浏览器里直接运行，输入你的应用地址加上/class\_add.php,执行效果如下：

部门添加/修改[返回>>](#)

部门名称：

上级部门：

新增了数据咱们要看看效果呗，就要写个部门列表管理的代码，我们命名为class\_manager.php，代码如下：

```

1 <!DOCTYPE HTML>
2 <html>
3 <HEAD>
4 <meta charset="utf-8">
5 <TITLE>部门管理</TITLE>
6 </HEAD>
7 <body>
8
9 <?php
10 include("base-class.php");
11
12 //新建数据库类
13 $mysql = new SaeMysql();
14
15 //获取当前页码
16 $page=intval($_GET["page"]);
17
18 //获取操作标识传入
19 $action=$_GET["action"];
20 $action= string::un_script_code($action);
21 $action= string::un_html($action);
22
23
24 //是否删除
25 if($action=="del")
26 {
27     //获取部门ID号传入
28     $class_id=intval($_GET["class_id"]);
29     //获取当前时间
30     $nowtime=date("Y/m/d H:i:s",time());
31     $mysql->runSql("update class set status=0,edittime='$nowtime' where class_id=$class_id");
32     echo "<script>alert('操作成功!');location='class_manager.php?page=$page';</script>";
33     exit;
34 }
35 //列表数据获取、分页
36
37 //计算总数
38 $count=$mysql->getVar("select COUNT(*) from class where status=1");
39 //如果数据表里有数据
40 if($count)
41 {
42     //每页显示记录数
43     $page_num = 2;
44     //如果无页码参数则为第一页
45     if ($page == 0) $page = 1;
46     //计算开始的记录序号
47     $from_record = ($page - 1) * $page_num;
48     //获取符合条件的数据
49     $class_list=$mysql->getData("select A.class_id,A.class_name,B.class_name as fclass_name
50     from class A left join class B on A.class_fid=B.class_id where A.status=1
51     order by A.class_id desc
52     limit $from_record,$page_num");
53     //分页函数
54     $multi = multi($count, $page_num, $page, "class_manager.php");
55 }
56 ?>
57 <!-- 页面名称 -->
58 <h3>部门管理<a href="class_add.php">新增部门</a></h3>
59 <!-- 列表开始 -->
60
61 <table border=1>
62     <tr>
63         <td>序号</td><td>部门名称</td><td>所属上级</td><td>操作</td>
64     </tr>
65     <?php
66         if($class_list)
67         {
68             foreach($class_list as $value)
69             {
70
71                 echo "<tr>
72                     <td>$value[class_id]</td>
73                     <td>$value[class_name]</td>
74                     <td>$value[fclass_name]</td>
75                     <td>
76                         <a href='class_manager.php?action=del&class_id=$value[class_id]''>删除</a>
77                         <a href='class_add.php?class_id=$value[class_id]''>修改</a>
78                     </td>
79                 </tr>";
80             }
81         }
82         else
83         {
84             echo "<tr><td colspan=4>无记录</td></tr>";
85         }
86     ?>
87 </table>
88 <?php
89 echo $multi;
90 ?>
91 </body>
92 </html>

```

有部分代码是差不多的，主要解释几个地方吧，

第 16 行，是获取当前页码参数。

第 25 行到 34 行是判断是否要进行删除操作，如果是删除则获取需要删除的部门 ID 号，删除操作我们不是真正的进行操作，只是将记录状态由正常变为删除，即 1 变为 0，所以这里执行的其实是更新操作，将 **status** 设置成 0。

第 35 行到 55 行是一个分页的加获取符合条件记录列表的模块。

第 38 行，要分页肯定先要知道有一共有多少条数据，所以这里是一个计算记录总数的代码，mysql 操作函数 **getVar** 是返回结果集第一条记录的第一个字段值，sql 语句的意思是

**select** 选择

**COUNT(\*)** 总共多少条记录

**from** 从

**class** 名字是 **class** 的数据表

**where status=1** 符合条件为正常的数据

这里的 **count(\*)** 返回所有记录总数，如果指定某个字段名，则是返回该字段不为空的记录总数。

第 40 行，计算记录总数除了分页以外还有一个好处就是可以知道到底有没有记录，这里就是判断是否有记录。

第 43 行，每页显示的条数，因为我这里数据就加了三条，所以设了个 2，一般建议 10 到 20。

第 45 行，如果获取的当前页码是 0 则表示第一页。

第 47 行，每一页开始的指针都是不同的，比如每页显示 10 条的话，第一页是 0-9（数据表里记录第一条其实是第 0 条），第二页是 10-19，通过这个代码来确定起始的指针，注意这个指针指记录在数据库里的实际位置，而不是 ID 号。

第 49 行，获取记录列表，这里我用了 sql 左连接查询两张表，主要是为了把记录里的 class\_fid 变成对应的部门名称（class\_fid 是数字的），其实不推荐使用这种，因为效率很低，不过咱们不用考虑这个。解释下 SQL 语句，看不懂也没关系照抄吧。

select 选择

A.class\_id,A.class\_name,表 A 的序号和部门名

B.class\_name as fclass\_name 表 B 的部门名，变量重命名为 fclass\_name

from 从

class A left join class B 名称为 class 的表命名为 A 左连接名称为 class 的表命名为 B

on A.class\_fid=B.class\_id 两表连接的条件是表 A 的 class\_fid 等于表 B 的 class\_id

where A.status=1 符合条件为 A 表的状态为正常的数据

`order by A.class_id desc` 排序按照 A 表的序号降序排列

`limit $from_record,$page_num` 从第`$from_record` 获取`$page_num` 条。

第 54 行，分页函数 `multi`，这个是自己的函数，函数代码在 `base-class.php` 里，参数分别是记录总数、每页显示条数、当前页码，还有一个就是列表页的 `url`。

第 61 行，表格标签，`border=1` 是为了有框框，哈哈。

第 62 行到 64 行，表格头，`<tr>`是行标签，`<td>`是列标签，就跟 `xls` 表格是一样的概念。

第 66 行，判断是否有数据列表。

第 68 行到 80 行，循环数组`$class_list` 输出列表，可以看到里面的字段就是前面 `sql` 查询结果的字段，最后到操作是两个链接标签，一个是删除，一个是修改，都是通过 `URL+参数` 实现的。

第 90 行，输出页码。

最后的执行的效果如下图：

部门管理[新增部门>>](#)

序号	部门名称	所属上级	操作
3	朋友圈小组	产品部	<a href="#">删除</a>
2	产品部	微信总部	<a href="#">删除</a>

[12>>](#)

部门管理[新增部门>>](#)

序号	部门名称	所属上级	操作
1	微信总部		<a href="#">删除</a>

[<<12](#)

本来想把员工输入也写了的，结果发现写这点都写了差不多 5、6 个小时，真是晕倒了.....快 12 点了，先发了再说吧，今天的代码也放到网上了，请输入“微信代码”下载。

最简单的方式将三个文件上传到自己的 SAE 里，如果数据表名字和结构都是一模一样的话，应该是可以直接执行的，执行地址为你 SAE 的应用地址+斜杠+文件名。

有问题请留言.....



微信教程跳票好久了，今天总算更新一章，把通讯录这个案例完结掉，之前几章有不少朋友反映写的过于详细了，大家可能更多的是需要一些思路和方法，所以我这次是将通讯录完整代码提供下载，回复“**微信代码**”可以下载完整代码包，建议先下载安装了再看文章，以下是注意事项：

1、数据库结构有改动，大家可以将原来的数据库删除后重新创建，创建方法请查看《微信公众平台入门到精通》Vol.15，新数据库创建代码在 **weixincourse.txt** 里。

2、与微信接口文件为 **wx\_interface.php**，大家可以将里面源代码复制粘贴到自己的接口文件里覆盖掉。

3、其他的文件全部上传到 **SAE** 的代码编辑里，有同名文件可以先删除原来文件再上传。

4、文件说明：

**base-class.php** 自定义函数，无需改动

**class\_add.php** 添加部门，无需改动

**class\_manager.php** 部门管理，无需改动

**roster\_add.php** 添加员工，需要改动 118 行的 **Storage** 空间名称

**roster\_manager.php** 员工管理



detail.php 前端查看员工详情，无需改动

list.php 前端查看部门列表，无需改动

wx\_tpl.php 回复消息的模板，无需改动

wx\_interface.php 微信接口文件，根据自己实际修改名称

5、后台文件没有加样式文件，所以看起来会很简陋，我主要是讲开发方法，CSS虽然我也很强，但是不在教程范畴里，有想学的可以自己找些网页模板。

6、程序没有进行优化，其实很多都可以用过程来写，但是为了让大家更好理解，就不厌其烦的写下吧.....

昨天有不少朋友在问如何测试不知道如何绑定账号，具体步骤是先后台添加一个员工，然后在微信公众账号里根据提示将添加的员工信息输入完成绑定，如图添加一个员工：

员工添加/修改[返回>>](#)

带星号为必填项

\*姓名：

\*工号：

\*照片： 2008121309252054707.jpg

\*性别：

\*生日：

\*手机：

电话：

\*部门：

\*邮箱：

微信：

\*状态：

入职：

员工管理[新增员工>>](#)

姓名	工号	照片	性别	部门	手机	状态	操作
邦德	007		男	微信总部	18777777777	病假	<a href="#">删除</a> <a href="#">修改</a>



## 第十四章 公司通讯录开发

### 四、公司通讯录开发

首先回答有些朋友的疑问为什么要做通讯录案例？这个案例貌似没有什么价值，确实从现实使用来看这个案例真没啥价值，但我是希望通过这个案例教大家学会数据库的使用、后台的开发、微信账号绑定、微信搜索、记录翻页等一些基本方法，然后可以结合自己的实际需求来做开发。

## 上传员工照片

在添加员工文件（roster\_add.php）中有一个图片上传功能，图片上传在表单里是使用 input 控件的 file 属性，如下图：

照片：<input type="file" name="roster\_pic">可上传jpg或者png图片

当提交表单后，程序会收到一个名称为 roster\_pic 的文件变量，文件变量获取是用 \$\_FILES，而不是 \$\_GET 或 \$\_POST，如下图：

```
96 //照片上传
97 if($_FILES['roster_pic']['name']!='')
98 {
99     //允许上传文件类型
100     $file_ext_arr=array("jpg","png");
101
102     //检测上传文件后缀
103     $get_exts = explode('.',$_FILES['roster_pic']['name']);
104     $exts_len=count($get_exts);
105     $file_ext=strtolower($get_exts[$exts_len-1]);
106
107     if(!in_array($file_ext,$file_ext_arr))
108     {
109         echo "<script>alert('只允许上传后缀为jpg或者png的图片文件!');history.back();</script>";
110         exit;
111     }
112     //设定新文件名称
113     $new_filename=substr(md5($roster_number),0,6).date("YmdHis").".".$file_ext;
114
115     //新建Storage类
116     $s = new SaeStorage();
117     //上传到SAE的Storage里，注意修改存储空间的名字"weixincourse"，并把返回的文件名赋值给旧文件变量old_roster_pic
118     $old_roster_pic=$s->upload( 'weixincourse' , $new_filename , $_FILES['roster_pic']['tmp_name'] );
119 }
```

第 97 行是监测是否有选择文件，\$\_FILES['roster\_pic']['name']是上传文件的原始文件名。

第 103 到 105 行获取上传文件名的后缀

第 107 到 111 行监测后缀是否合法。

第 118 行是保存图片到 Storage 里并返回上传后的图片实际路径赋值给 \$old\_roster\_pic。

注意：

1、上传文件的数据会保存成一个临时文件，web 是不能直接访问用户电脑文件的，所以我们读取的源文件是 `$_FILES['roster_pic']['tmp_name']` 。

2、由于我将添加和修改是放在同一个文件里，`$old_roster_pic` 在表单内是用来存放修改状态下旧图片文件名称的，当表单提交后如果没有上传图片并且 `$old_roster_pic` 为空（即不在修改状态）则会提示上传图片，如果上传了图片则更新 `$old_roster_pic` 这个变量，最后写入数据库时使用的都是 `$old_roster_pic` 的值。

后台的程序基本就是这些，大家可以结合《微信公众平台入门到精通》Vol.15 来理解，如果有不明白的可以后台咨询。下面主要讲前台与微信通讯的程序，都在 `wx_interface.php` 里。

## 微信账号绑定

微信用户的账号如何与自己网站的用户系统打通，这个是很多人关心的问题，其实很简单，我们都知道微信用户在关注公众号或者发送消息时都会有一个唯一的 `OPENID` 传过来，这个 `OPENID` 就是这个用户对应这个公众账号的唯一身份标示，我们只要将这个 `OPENID` 与自己网站的用户系统一一绑定就可以了。

方法有两种，一是将 `OPENID` 作为登陆页面或者注册页面的 `URL` 参数，当用户点击这个 `URL` 时跳转到 `WAP` 页面（即自己网站）上进行绑定，另外就是在对话框里进行绑定，我今天介绍的就是在对话框里进行绑定。

首先是检测用户是否绑定了微信号，这里我写了个函数，如下图：

```

592 //检测微信用户是否绑定
593 function check_user($fromUsername)
594 {
595     //定义全局变量
596     global $mysql;
597
598     $roster_value=$mysql->getLine("select * from roster where roster_openid='$fromUsername'");
599     //如果没有绑定
600     if(!$roster_value)
601     {
602
603         return false;
604     }
605     //如果已经绑定（误取消重新关注员工）
606     else
607     {
608         return $roster_value;
609     }
610 }
611 }

```

第 593 行是定义函数的名字为 `check_user`，函数传入的变量为 `$fromUsername`，即微信用户的 `OPENID`。

第 598 行是通过传入的 `OPENID` 为条件，检测在员工数据表中是否存在某条记录的 `roster_openid` 等于该 `OPENID`。

第 600 到 609 行是返回函数运行的记过，如果有记录的话返回该条数据，如果没有就返回失败。

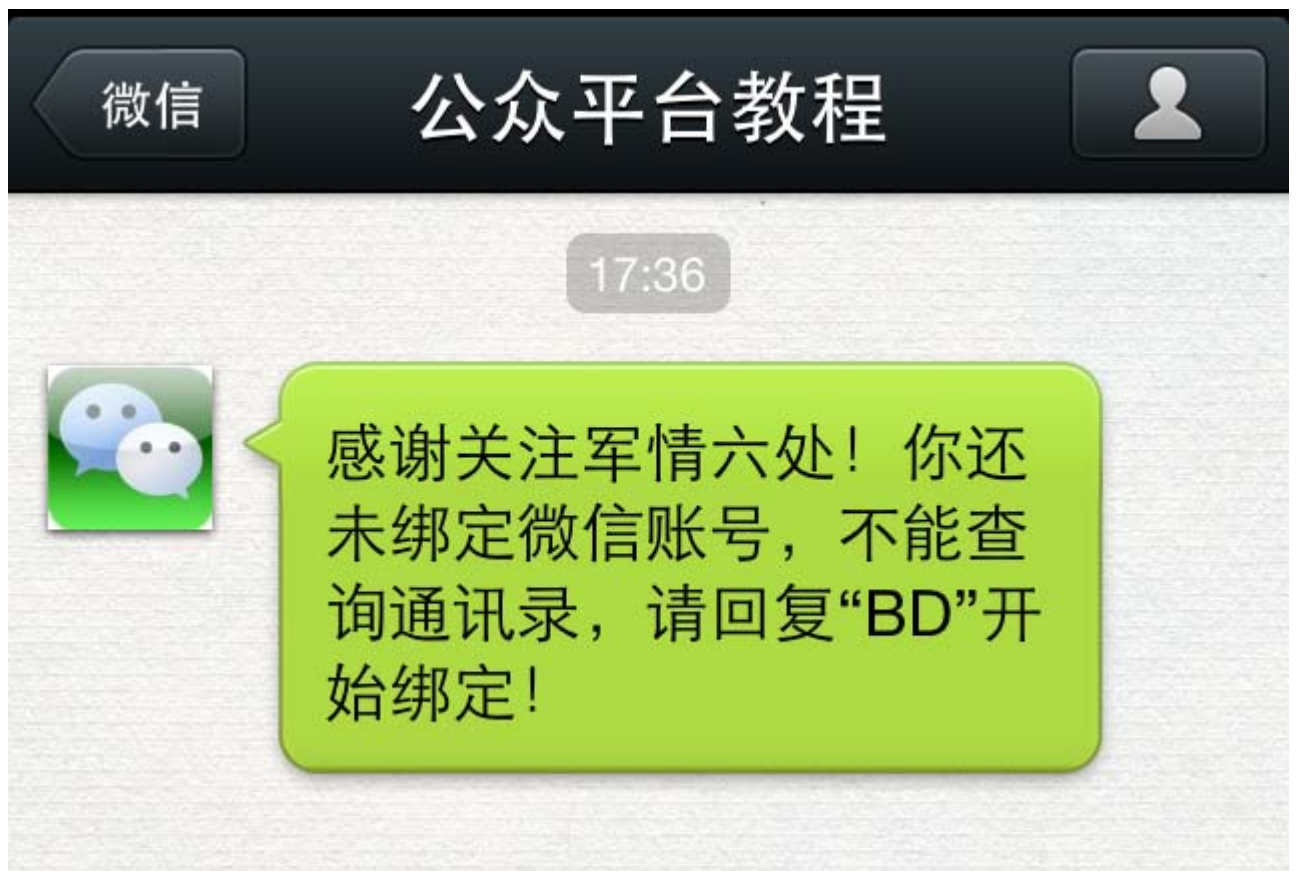
在用户关注公众账号时我就判断是否有绑定过账号，并且根据绑定情况回复不同的欢迎词，如下图：

```

//欢迎消息
if($form_MsgType=="event")
{
    //获取事件类型
    $form_Event = $postObj->Event;
    //订阅事件
    if($form_Event=="subscribe")
    {
        //根据关注用户的OPENID判断是否已经绑定员工号
        if(check_user($fromUsername))
        {
            //关注绑定欢迎词
            $welcome_str="感谢关注军情六处! \n\n".$help_menu;
        }
        else
        {
            //关注未绑定欢迎词
            $welcome_str="感谢关注军情六处! 你还未绑定微信账号, 不能查询通讯录, 请回复“BD”开始绑定! ";
        }
        //回复欢迎文字消息
        $msgType = "text";
        $resultStr = sprintf($textTpl, $fromUsername, $toUsername, time(), $msgType, $welcome_str);
        echo $resultStr;
        exit;
    }
}
}

```

当检测到用户关注的事件，调用 check\_user 这个函数，如果返回不是失败则提示使用帮助，否则提示绑定账号，效果如图：



当用户输入 BD 字符时进入账号绑定模式，如下图：

```
//用户输入BD进行账号绑定
if(strtolower($form_Content)=="bd")
{
    //监测微信用户openid是否已经绑定

    if(check_user($fromUsername))
    {
        //提示已经绑定警告
        $msgType = "text";
        $resultStr = sprintf($textTpl, $fromUsername, $toUsername, time(), $msgType, "你已经绑定账号。");
        echo $resultStr;
        exit;
    }

    //用memcache保存这步操作。格式为名称、值、有效时间(单位秒);
    $mc->set($fromUsername."_dc", "bd_0", 0, 600);
    //回复绑定下一步操作提示
    $msgType = "text";
    $resultStr = sprintf($textTpl, $fromUsername, $toUsername, time(), $msgType, "请输入你的姓名。输入");
    echo $resultStr;
    exit;
}
```

先判断用户是否已经绑定过，绑定过给提醒中断执行。否则进入绑定流程，由于绑定操作需要分多个步骤验证多个条件，然后一并查询数据库，因此我们要保存每一步的动作和数据，这里使用 memcache 来保存每个用户的上一步操作。这里先保存了 BD\_0 标示这是绑定的第一个步骤，提示用户信息为“请输入你的姓名”。



bd



请输入你的姓名，输入  
exit退出操作！

**memcache** 服务的开启请查看《微信公众平台入门到精通》Vol.13，在程序里使用方法为在代码最开头启动 **memcache**，如下图：

```
//新建Memcache类  
$mc=memcache_init();
```

保存 **memcache** 的方法为：

```
$mc->set(缓存变量名, 保存的数据, 0, 缓存时间单位秒);
```

读取 **memcache** 的方法为：

```
$mc->get(缓存变量名);
```

我在程序的开头加了获取 **memcache** 值的代码，每次用户请求时都会从 **memcache** 中获取用户上一步的操作和数据。其中 **\$fromUsername."\_do"** 为用户操作，**\$fromUsername."\_data"** 为用户数据，用 **\$fromUsername** 可以保证每个用户都有独立的缓存变量名。

```
78 //从memcache获取用户上一次动作  
79 $last_do=$mc->get($fromUsername."_do");  
80 //从memcache获取用户上一次数据  
81 $last_data=$mc->get($fromUsername."_data");
```

接下来等用户输入姓名后，接口会收到新的请求，这个时候由于 memcache 保存过上步标示，因此这里会根据操作缓存的数据判断进入到第二步，\$last\_do 的值是从程序开始就获取 memcache 值后获取的，如下图：

```
//确定第一步，输入姓名
if($last_do=="bd_0")
{
    //用memcache保存这步操作
    $mc->set($fromUsername."_do", "bd_1", 0, 600);

    //用memcache保存这步数据
    $mc->set($fromUsername."_data", $form_Content, 0, 3600);

    //回复确定下一步操作提示
    $msgType = "text";
    $resultStr = sprintf($textTpl, $fromUsername, $toUsername, time(), $msgType, "请输入你的工号，输入exit退出操作！");
    echo $resultStr;
    exit;
}
```

这里多了一个保存数据的缓存设置，保存了用户提交的姓名数据，同时将操作缓存的值改为了 BD\_1，然后继续提示用户输入工号。



当用户输入工号后，会进入第三步，如下图：

```

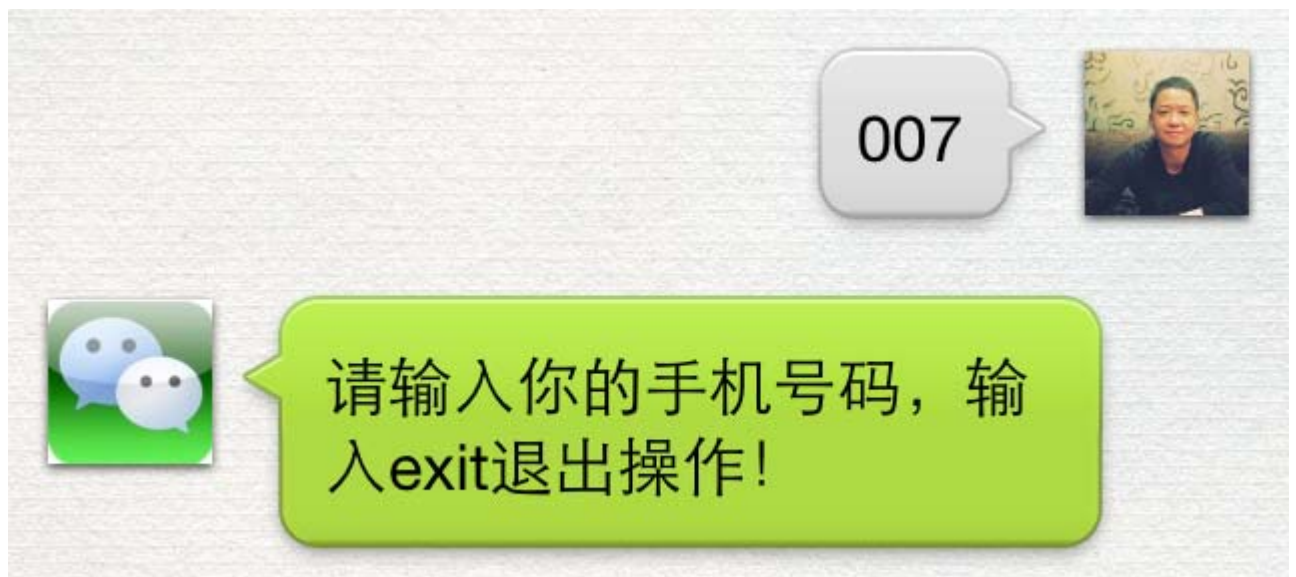
//确定第二步，输入工号
if($last_dc=="bd_1")
{
    //用memcache保存这步操作
    $mc->set($fromUsername."_dc", "bd_2", 0, 600);

    //用memcache保存这步数据，同时与上次数据合并，两个数据之间用||分割
    $mc->set($fromUsername."_data", $last_data."||".$form_Content, 0, 3600);

    //回复确定下一步操作提示
    $msgType = "text";
    $resultStr = sprintf($textTpl, $fromUsername, $toUsername, time(), $msgType, "请输入你的手机号码，输入exit退出操作！");
    echo $resultStr;
    exit;
}

```

继续保存数据，将上一步的数据（姓名）与本次数据（工号）合并用“||”分割，同时将操作缓存的值改为了 BD\_2，然后继续提示用户输入手机号。



当用户输入手机号后，就进入最后一步验证，如下图：

```

//绑定第二步，输入手机号码
if($last_do=="bd_2")
{
    //清空memcache动作
    $mc->delete($fromUsername."_do");

    //清空memcache数据
    $mc->delete($fromUsername."_data");

    //将之前保存数据变为数组;
    list($roster_name,$roster_number,$roster_mp)=explode("||",$last_data."||".$

    //判断资料是否完全吻合，并且openid为空即未绑定状态

    $roster_data=$mysql->getLine("select roster_id from roster where
    roster_name='$roster_name' and roster_number='$roster_number' and
    roster_mp='$roster_mp' and roster_openid='' and status=1");

    //资料一致确认绑定

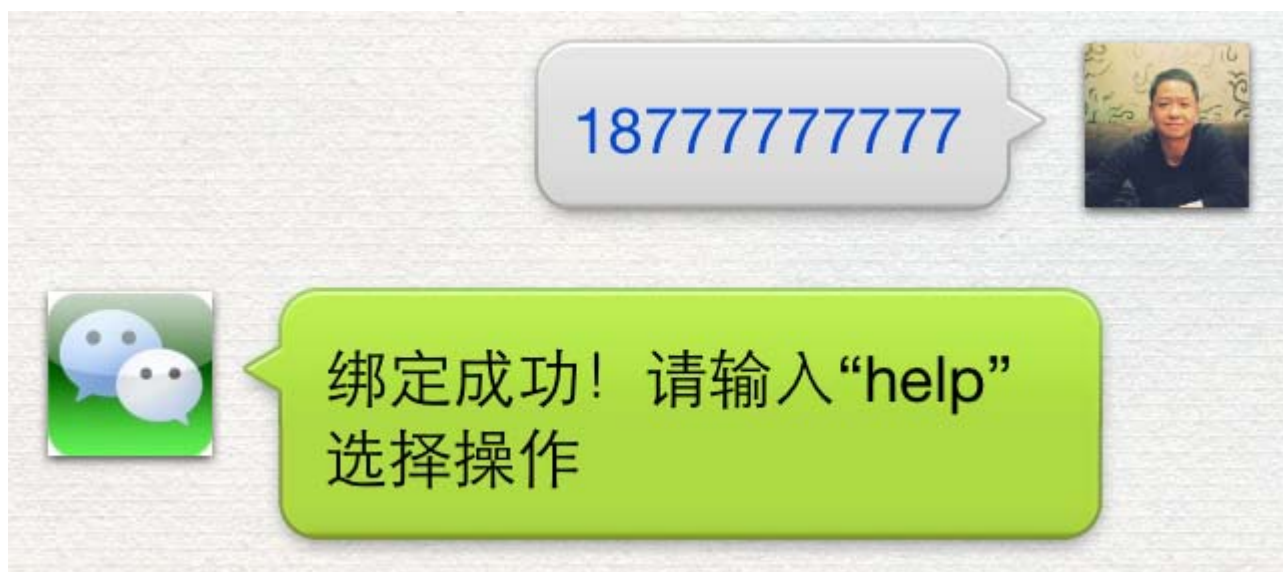
    if($roster_data)
    {
        //将用户openid与该记录绑定
        $roster_id=$roster_data["roster_id"];
        $sql = "update roster set roster_openid='$fromUsername' where roster_id=$roster_id";
        $mysql->runSql( $sql );
        if( $mysql->errno() != 0 )
        {
            $back_str="绑定失败，可能是资料不正确或者该账号已经绑定，请重新输入BD进行绑定！";
        }
        else
        {
            $back_str="绑定成功！请输入“help”选择操作";
        }
    }
    else
    {
        $back_str="绑定失败，可能是资料不正确或者该账号已经绑定，请重新输入BD进行绑定！";
    }
}

```

进入最后一步后先清空操作和数据缓存，然后将之前保存的数据\$last\_do 加上本次用户输入内容合并转换成数组，并用 list 函数分别赋值给 \$roster\_name,\$roster\_number,\$roster\_mp。

再进行数据库查询满足用户姓名、工号和手机与输入数据完全相同并且 roster\_openid（员工 openid）为空的记录，如果有符合条件的记录则更新该记录，将\$fromUsername（当前操作用户的 openid）保存到 roster\_openid 里，完成绑定。





这个时候我们输入“help”就可以进行后续操作了。



PS: “输入 exit 退出操作”这个也很简单，当用户输入 exit 的时候程序会清空操作和数据缓存，也就是说所有之前的操作和数据都没有了，即所有操作重新开始，如下图：

```
//用户跳出操作
if(strtolower($form_Content)=="exit")
{
    //清空memcache动作
    $mc->delete($fromUsername."_do");

    //清空memcache数据
    $mc->delete($fromUsername."_data");

    //回复操作提示
    $msgType = "text";
    $resultStr = sprintf($textTpl, $fromUsername, $toUsername, time(), $msgType, "你已经退出当前操作.");
    echo $resultStr;
    exit;
}
```

这个流程使用了 memcache 来保存一些临时变量和多步骤操作，除了用来绑定用户账户外，也适合一些调查问卷或者注册登陆操作。

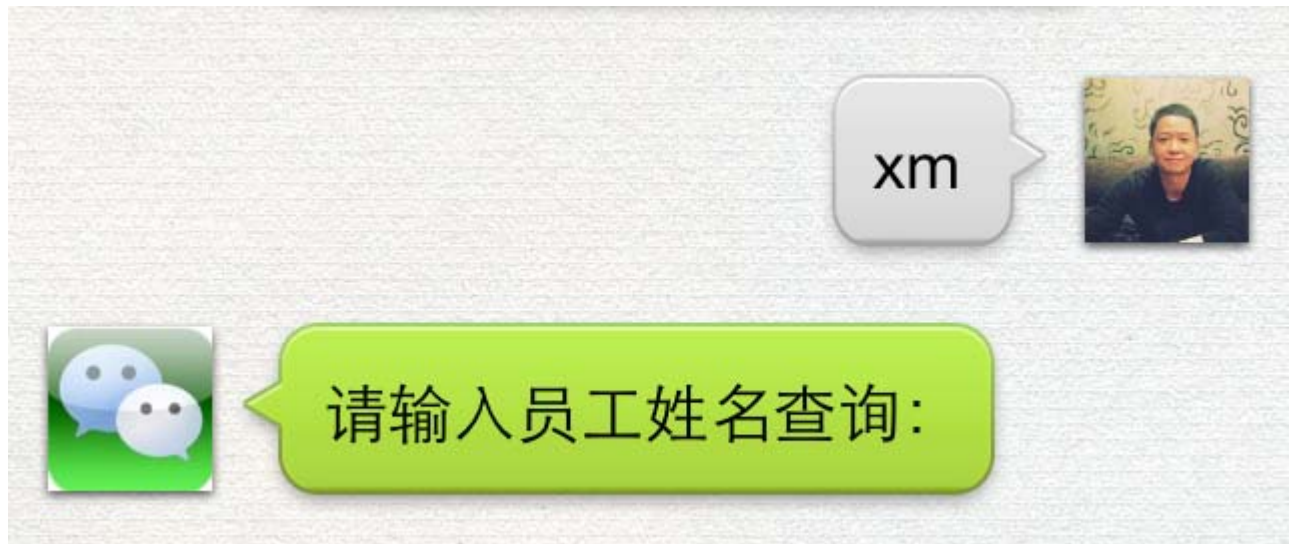
## 微信搜索

搜索其实就是编辑模式下的关键字自动回复，我这里拿姓名搜索举例，首先我们要先切换当前模式到姓名搜索，如下图：

```
//姓名搜索
if(strtolower($form_Content)=="xm")
{
    //用memcache保存部门搜索操作状态
    $mc->set($fromUsername."_do", "xm_search", 0, 600);

    //回复操作提示
    $msgType = "text";
    $resultStr = sprintf($textTpl, $fromUsername, $toUsername, time(), $msgType, "请输入员工姓名查询:");
    echo $resultStr;
    exit;
}
```

这部分代码与账号绑定的第一步是一样的，我们用 **memcache** 存放当前的操作，这个时候存放操作动作的缓存变量里就标示了当前是在姓名搜索状态下。同时输出提示，请输入员工姓名。



其实这步可以理解为进入网站的二级导航页面，一个网站有首页和各种导航栏目，那么在这个通讯录里帮助就是首页，你输入 **HELP** 就到了首页，输入 **XM** 就到员工查询这个二级栏目，之后除非你输入 **BM**、**GH** 等一些栏目名，其他的操作都默认是在员工栏目下进行。

然后根据提示输入员工名字进行查询，如下图：





关羽



## 001.关羽（在职）

6月14日



详情请点击阅读全文，输入其他员工姓名继续查询，退出姓名查询状态请输入exit

阅读全文



前端展现原因我只取工号、姓名、当前状态、照片等字段返回，查看详情点击阅读原文进入到 **detail** 页，把该员工的 ID 号（不是工号）作为 **URL** 参数传递过去，然后再做一次数据库查询将所有字段提取出来展现。



姓名: 关羽

工号: 001"

照片:



性别: 男

生日: [0813-01-31](#)

手机: [18777777773](#)

电话:

部门: 产品部

邮箱: [gy@gy.com](mailto:gy@gy.com)

微信:

状态: 在职

入职: 2010-03-03

## 微信翻页

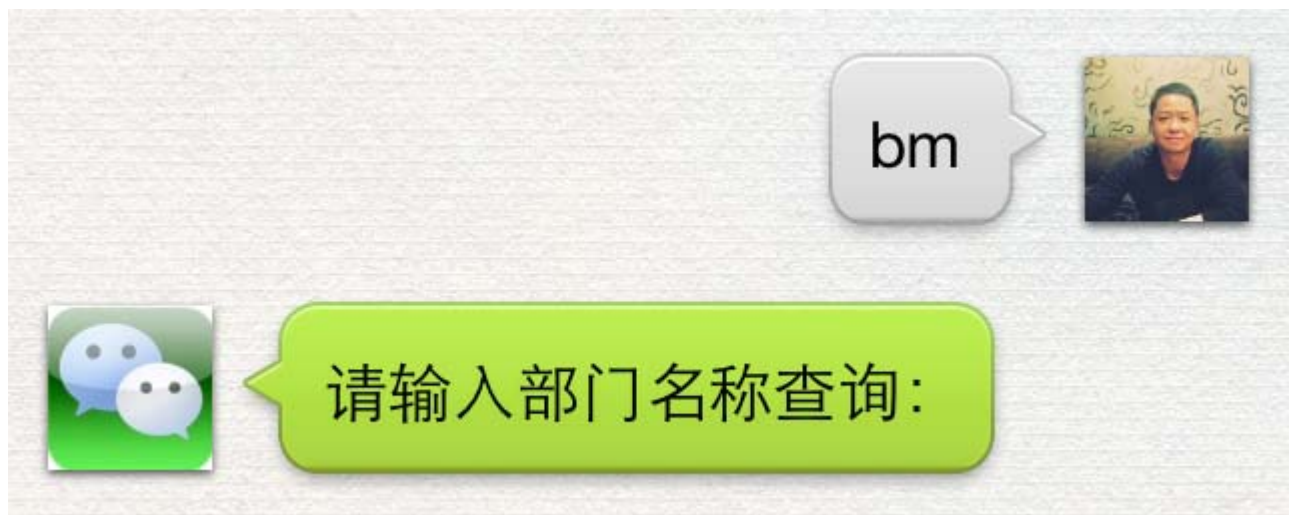
很多人非常好奇 ZTalk 的公众账号是如何实现文章查询翻页的, 其实原理也很简单, 只是把网页上的翻页代码稍微改装一下, 然后使用 **memcache** 保存每个用户当前的页码和搜索的关键字就可以了。

在通讯录里是用部门查询来示例的，首先依然是进入部门查询这个栏目，如下图：

```
//部门搜索
if(strtolower($form_Content)=="bm")
{
    //用memcache保存部门搜索操作状态
    $mc->set($fromUsername." do", "bm_search", 0, 600);
    //部门查询可能会有多个，要使用分页，用memcache保存翻页，先设成第一页，同时将搜索部门字设成null
    $mc->set($fromUsername."_data", "null||1", 0, 600);

    //回复操作提示
    $msgType = "text";
    $resultStr = sprintf($textTpl, $fromUsername, $toUsername, time(), $msgType, "请输入部门名称");
    echo $resultStr;
    exit;
}
```

上面的代码除了保存当前操作状态为部门查询以外，又用另外一个 memcache 的变量保存当前搜索关键字和页码“null || 1”，其中 null 为当前搜索关键字（因为一开始用户没有输入所以是空的），1 表示起始页是第一页，然后输出提示，请输入部门名称。



当用户输入部门名称后，先查询这个部门的 ID 号，因为在员工表里没有直接存放部门名称而是存放的与部门对应的 ID 号，如下图：

```

//接收到上一步是部门搜索状态，获取用户查询关键字
if($last_do=="bm_search")
{
    //从memcache数据中获取搜索关键字和当前页码:
    $last_data=explode("||",$last_data);
    //设定当前搜索部门名称；如果memcache数据中搜索关键字是null则设定关键字为当前用户回复的文字
    $search_content=($last_data[0]=="null")?$form_Content::$last_data[0];
    //先查询是否有该部门，并获取该部门的id
    $class_id=$mysql->getVar("select class_id from class where class_name='$search_content'");
    //查询不到该部门名称则提示
    if(!$class_id)
    {
        $msgType = "text";
        $resultStr = sprintf($textTpl, $fromUsername, $toUsername, time(), $msgType, "无此部门");
        echo $resultStr;
        exit;
    }

    //查询该部门下的记录总数
    $count=$mysql->getVar("select COUNT(*) from roster where roster_class=$class_id and sta");
    //如果总数为0则提示
    if(!$count)
    {
        $msgType = "text";
        $resultStr = sprintf($textTpl, $fromUsername, $toUsername, time(), $msgType, "该部门没");
        echo $resultStr;
        exit;
    }
}

```

在这里我们获取搜索关键字会有两个来源，一个是用户输入的，一个是在翻页时在缓存中保存的，所以先得尝试从缓存里获取用户之前保存的搜索关键字，如果值为 null 则表示这是第一次搜索，关键字为用户输入的\$from\_content，否则就使用缓存中保存的关键字，这就是为什么能够翻页的关键，我们利用缓存机制存放了用户之前输入的搜索关键字。

有了搜索关键字然后查询部门表获取部门 ID 号，获取到后再去查询员工表里该 ID 下的员工总数，如果总数为 0 则表示该部门还没有员工直接提示。否则进入到翻页计算和结果返回，如下图：

```

//每次返回数据条数
$page_num=3;||
//当前页码
$page=$last_data[1];
//起始记录号
$from_record = ($page - 1) * $page_num;
//获取符合条件的数据
$roster_res=$mysql->getData("select roster_id,roster_name,roster_number,roster_pic,roster_status
from roster where status=1 and roster_class=$class_id
order by roster_id desc
limit $from_record,$page_num");

//生成输出的数组
$roster_list=array();
//设定返回图文消息的封面图文消息
$roster_list[]=array('title'=>$search_content."搜索结果(第".$page."页): ",
'doc'=>' ',
'pic'=>'http://weixincourse-weixincourse.stor.sinaapp.com/fengmian.jpg',
'url'=>'http://weixincourse.sinaapp.com/list.php?id='.$class_id);

//循环输出员工数据格式化
foreach($roster_res as $value)
{
$roster_list[]=array('title'=>$value["roster_number"].".".$value["roster_name"]." (".$roster_status_ary[
'doc'=>' ',
'pic'=>$value["roster_pic"],
'url'=>'http://weixincourse.sinaapp.com/detail.php?id='.$value["roster_id"]);
}

```

第一个是设置每页显示的条数，我这里是设置成 3 条，加上封面消息和尾部翻页提示消息最后显示会是 5 条。

当前页码是从缓存中获取，\$last\_data 在前面已经使用 explode 函数以||为分割字符变成数组，\$last\_data[0]为搜索关键字，\$last\_data[1]为页码，根据当前页码计算数据库查询开始的记录指针位置。

查询出结果后进行数据转化，这里做了一个数组方便后面循环输入多图文消息，每个图文消息其实就 4 个元素：标题、描述、图片、链接，先直接用搜索关键字+当前页等做一个封面消息，然后循环将数据库查询结果也转化成图文消息的元素。

然后是判断是否有下一页，如下图：



```

//检测是否还有下页
$roster_next=0;
$roster_tip="";
//计算总页数
$real_page=@ceil($count / $page_num);
//如果页码已经超过或者到达最后一页，将页码设置成一
if($page>=$real_page)
{
    $roster_next=1;
    $roster_tip="已经到最后一页，重复查询输入任何字符，退出请输入exit";
}
//否则当前页码加1
else
{
    $roster_next=$page+1;
    $roster_tip="还有".($count-($page*$page_num))."条记录，输入任何字符查看下一页，退出请输入exit";
}
//设定memcache，保存搜索部门名称和关键字：
$mc->set($fromUsername."_data", $search_content."||".$roster_next, 0, 600);
//添加结尾提示
$roster_list[]=array('title'=>$roster_tip,
                    'doc'=>'',
                    'pic'=>'http://weixincourse-weixincourse.stor.sinaapp.com/fengmian.jpg',
                    'url'=>'http://weixincourse.sinaapp.com/list.php?id='.$class_id);

```

用符合搜索条件的记录总数去除以每页显示数量，ceil 是一个向上舍入为最接近的整数的函数，就是  $\text{ceil}(5/4)=2$ ，而不是 1.25，这样就知道总页数了，然后判断当前页是否最后一页，如果不是最后一页则将当前页加 1 然后存放到 memcache 里，在下次翻页查询时保存数据的这个 memcache 变量里其实已经变成“搜索关键字||2”。

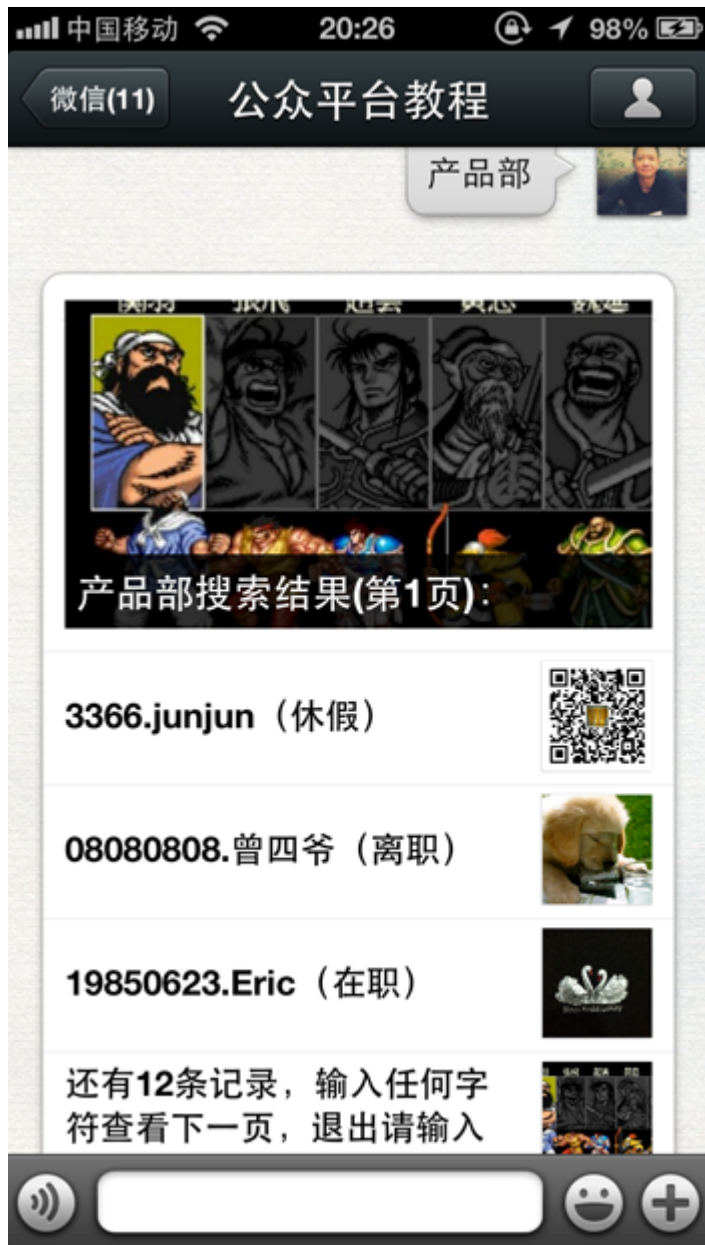
最后就是输出多图文消息了，如下图：

```

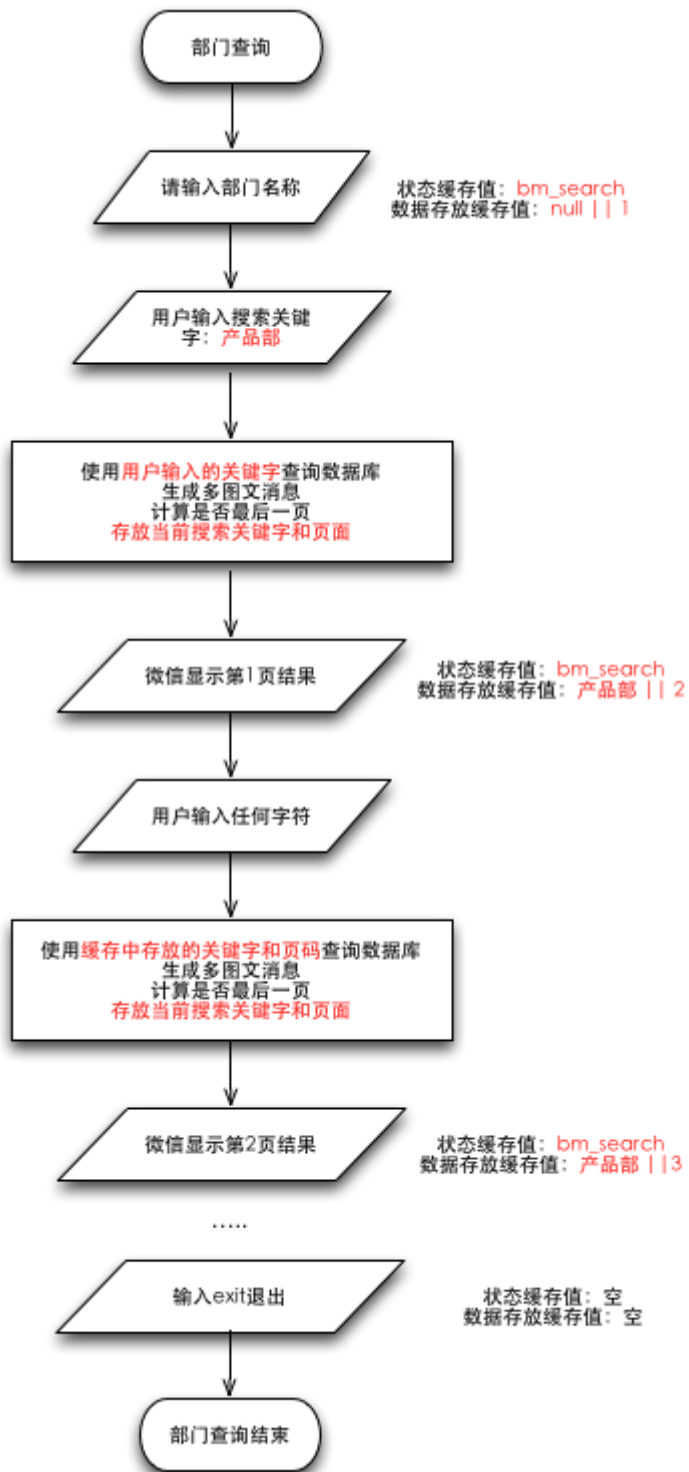
//用多图文消息输出员工信息
$resultStr=<?xml>\n
    <ToUserName><![CDATA[".$fromUsername."]]></ToUserName>\n
    <FromUserName><![CDATA[".$toUsername."]]></FromUserName>\n
    <CreateTime>" . time() . "</CreateTime>\n
    <MsgType><![CDATA[news]]></MsgType>\n
    <ArticleCount>" . count($roster_list) . "</ArticleCount>\n
    <Articles>\n";
foreach($roster_list as $key=>$value)
{
    $resultStr.="<item>\n
        <Title><![CDATA[" . $value["title"] . "]]></Title> \n
        <Description><![CDATA[" . $value["doc"] . "]]></Description>\n
        <PicUrl><![CDATA[" . $value["pic"] . "]]></PicUrl>\n
        <Url><![CDATA[" . $value["url"] . "]]></Url>\n
        </item>\n";
}
$resultStr.="</Articles>\n
    <FuncFlag>0</FuncFlag>\n
</xml>";
echo $resultStr;
exit;

```

在微信里显示的效果如下：

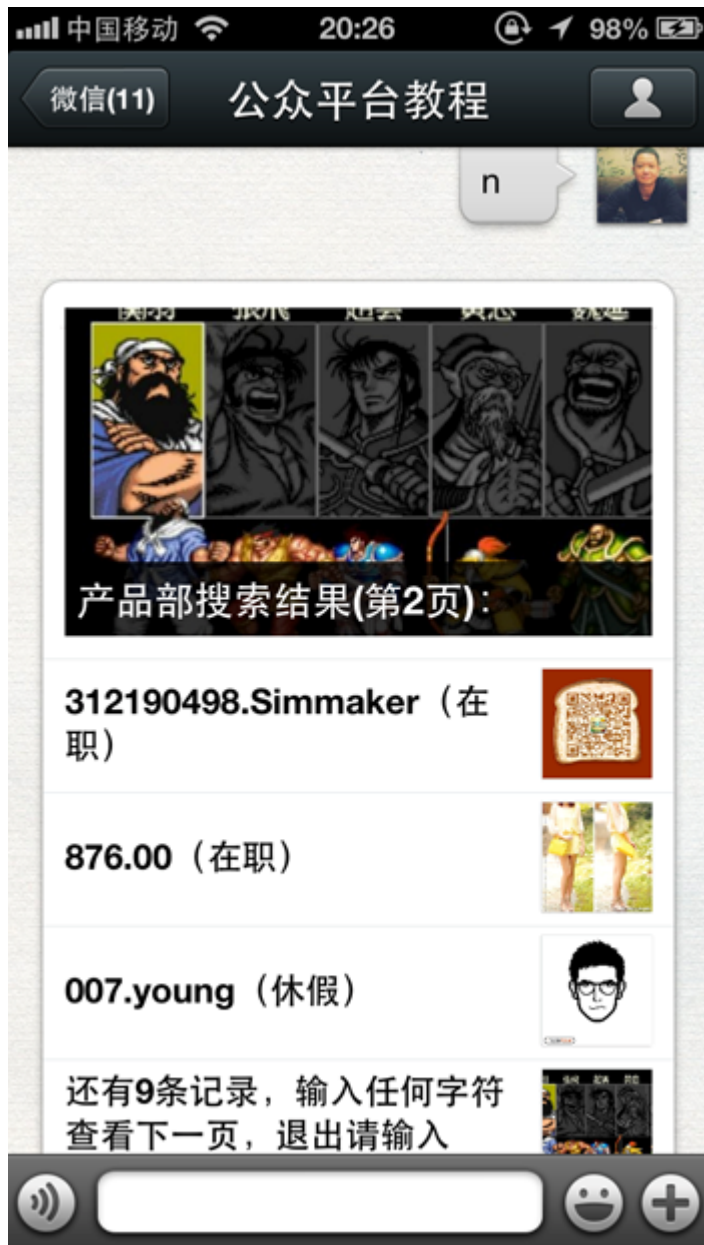


可能还有很多人很迷茫，如何输入任何字符查看下一页的呢？我画个流程图可能会比较清楚，看下图：



大家可以看到其实从第二页开始用户输入的字符只是触发程序运行，而不能干涉程序运行的结果，除非输入 **EXIT**，或者 **HELP** 这些一级栏目关键字。





关于通讯录就讲到这里了，响应朋友们的建议多讲些设计理念，希望大家能够触类旁通，和自己业务相结合。后面还有两个教程也打算用这个方法来做，会提供完整代码，讲一些要点，就是间隔时间会比较久因为要码代码.....

本文网络独家授权

，任何媒体、机构不得转发！

-----ZTalk 互动-----

**KB24:** 不知道能不能增加一个功能可以从前往后显示文章，想看一下你最初写的文章，一直 mm 感觉不是很方便，而且最近的好多文章我都看过了，感谢啦~!

**Ztalk:** 这个提议挺不错的，实现也简单，回头加个呗。

**扯了个犊子。。:** 又是一个毕业季。三毛阿姨说:“人生又有多少场华丽在等着，不多的，不多的，即使旅行，也大半平凡岁的岁月罢了”。你的大学又又有多少疯狂与激情。。

**Ztalk:** 没考上大学.....

**扯了个犊子。。:** 所以你的境界比我好。教育误人子弟

**Ztalk:** .....我觉得大学里不是教导工作技能的，是教人如何观察思考的，如果有可能还是去经历下。

**百晶仪器杨太平:** 朱老师您好，之前看您的网名，以为您是个愤青呢，今天在虎嗅网看了您的那篇微信文章才知道您是技术大拿，一口气看了您的《微信公众平台》系列文章，当然后面的开发者系列，看的很晕，但是也给了我不少启发，建议这次微信公众平台系统更新，朱老师进行完善后出本书，我一定买来好好研读。字里行间不难发现朱老师绝不是个“贼”，二十个很具有耐心，善良的人。文章穿插广告怎么了，一些人就是容不得别人大量付出后，稍稍收获些，对此，您勿予理睬，工作顺利!

**Ztalk:** .....请允许我小小的傲娇一下

**ZTalk** 每日一曲，输入“**V**”欣赏，历史欣赏请输入“**VM**”：

Crash City Saints 的《Broke》， 黑暗的情绪