# 微擎开发文档

<https://www.we7.cc/manual/structure:loader>

目录 [隐藏]

[加载器](https://www.we7.cc/manual/structure:loader" \l "加载器)

[加载系统公共函数](https://www.we7.cc/manual/structure:loader" \l "加载系统公共函数)

[加载系统公共类](https://www.we7.cc/manual/structure:loader" \l "加载系统公共类)

[加载系统model函数](https://www.we7.cc/manual/structure:loader" \l "加载系统model函数)

[加载web端公共函数](https://www.we7.cc/manual/structure:loader" \l "加载web端公共函数)

[加载app端公共函数](https://www.we7.cc/manual/structure:loader" \l "加载app端公共函数)

## 加载器

微擎系统依靠 **Load** 类进行类、函数、模型的加载，该类会在系统初始化时引用并且实例化，系统提供 **load()** 函数来引用该类的实例   
**注：使用 load() 加载文件时，可以重复加载**

## 加载系统公共函数

**load()→func('filename')** 加载文件为 **framework/function/\*.php**

*// framework/function/logging.func.php*

load()->func('logging');

## 加载系统公共类

**load()→classs('filename')** 加载文件为 **framework/class/\*.php**

*// framework/class/account.class.php*

load()->func('account');

## 加载系统model函数

**load()→model('filename')** 加载文件为 **framework/model/\*.php**

*// framework/model/attachment.mod.php*

load()->func('attachment');

# 加载web端公共函数

**load()→web('filename')** 加载文件为 **web/common/\*.php**

*// web/common/tpl.func.php*

load()->web('tpl');

# 加载app端公共函数

**load()→app('filename')** 加载文件为 **app/common/\*.php**

*// app/common/tpl.func.php*

load()->app('tpl');

您的足迹: • [start](https://www.we7.cc/manual/start" \o "start) • [w](https://www.we7.cc/manual/structure:w" \o "structure:w) • [p](https://www.we7.cc/manual/standards:php" \o "standards:php)

目录 [隐藏]

[总览](https://www.we7.cc/manual/structure:outline" \l "总览)

[api关键字](https://www.we7.cc/manual/structure:outline" \l "api关键字)

[微站](https://www.we7.cc/manual/structure:outline" \l "微站)

[扩展模块](https://www.we7.cc/manual/structure:outline" \l "扩展模块)

[粉丝&会员](https://www.we7.cc/manual/structure:outline" \l "粉丝-会员)

## 总览

微擎系统总共分为四大部分“api关键字回复”，“微官网”，“粉丝&会员”，“扩展模块”

各部分功能相互依赖辅助

## api关键字

用于回复粉丝在微信端输入“关键字”后，系统返回其的信息数据，常见的数据有“文本”，“图文”，“语音”，“视频”等

在 1.0 版中，还可以自定义事件用于模块与第三方系统打通。比如要获取会员数据，模块可以接管一个自定义事件叫 we7\_event\_get\_member来获取会员数据

粉丝通过这些回复中的链接可进入系统微站，个人中心或是扩展模块中

也就是说“api关键字、规则回复”驱动了微擎系统的大部分功能

## 微站

微站分为三部分，系统微站（首页）、个人中心及扩展模块本身的微站页面

微站是承载展示、活动及其它活动的一个重要的体现形式

## 扩展模块

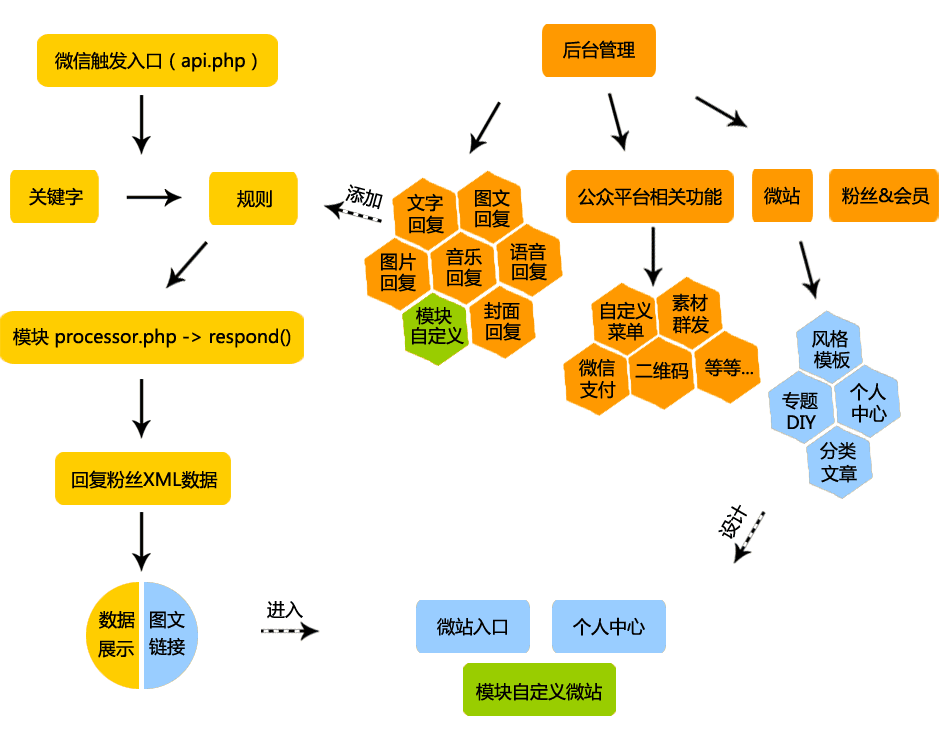
微擎系统中大部分的营销、活动功能都集中在扩展模块中，在使用时可以随意购买，卸载（要开通云服务）

## 粉丝&会员

存储着粉丝&会员的相关资料数据

扩展了积分系统，会员卡系统等重要功能

下面的示意图展示了微擎结构示意图：

[](https://www.we7.cc/manual/_detail/dev:structure_1.png?id=structure:outline)

您的足迹: • [start](https://www.we7.cc/manual/start" \o "start) • [w](https://www.we7.cc/manual/structure:w" \o "structure:w) • [php](https://www.we7.cc/manual/standards:php" \o "standards:php) • [bootstrap](https://www.we7.cc/manual/structure:bootstrap" \o "structure:bootstrap) • [loader](https://www.we7.cc/manual/structure:loader" \o "structure:loader) • [outline](https://www.we7.cc/manual/structure:outline" \o "structure:outline)

目录 [隐藏]

[编码规范](https://www.we7.cc/manual/standards:php" \l "编码规范)

[前言](https://www.we7.cc/manual/standards:php" \l "前言)

[PHP编码规范与原则](https://www.we7.cc/manual/standards:php" \l "php编码规范与原则)

[文件编码及编辑器格式](https://www.we7.cc/manual/standards:php" \l "文件编码及编辑器格式)

[编码](https://www.we7.cc/manual/standards:php" \l "编码)

[缩进](https://www.we7.cc/manual/standards:php" \l "缩进)

[换行](https://www.we7.cc/manual/standards:php" \l "换行)

[代码标记](https://www.we7.cc/manual/standards:php" \l "代码标记)

[注释](https://www.we7.cc/manual/standards:php" \l "注释)

[书写规则](https://www.we7.cc/manual/standards:php" \l "书写规则)

[大括号{}、if和switch](https://www.we7.cc/manual/standards:php" \l "大括号-if和switch)

[运算符、小括号、空格、关键词和函数](https://www.we7.cc/manual/standards:php" \l "运算符-小括号-空格-关键词和函数)

[函数定义](https://www.we7.cc/manual/standards:php" \l "函数定义)

[引号](https://www.we7.cc/manual/standards:php" \l "引号)

[数据库操作](https://www.we7.cc/manual/standards:php" \l "数据库操作)

[命名原则](https://www.we7.cc/manual/standards:php" \l "命名原则)

[变量、函数名](https://www.we7.cc/manual/standards:php" \l "变量-函数名)

[变量名命名方式](https://www.we7.cc/manual/standards:php" \l "变量名命名方式)

[单词分隔](https://www.we7.cc/manual/standards:php" \l "单词分隔)

[循环中变量定义](https://www.we7.cc/manual/standards:php" \l "循环中变量定义)

[常用缩写列表](https://www.we7.cc/manual/standards:php" \l "常用缩写列表)

[类和接口名](https://www.we7.cc/manual/standards:php" \l "类和接口名)

[JavaScript代码](https://www.we7.cc/manual/standards:php" \l "javascript代码)

[常量](https://www.we7.cc/manual/standards:php" \l "常量)

[封装](https://www.we7.cc/manual/standards:php" \l "封装)

[封装时机](https://www.we7.cc/manual/standards:php" \l "封装时机)

[参数传递](https://www.we7.cc/manual/standards:php" \l "参数传递)

[变量的初始化与逻辑检查](https://www.we7.cc/manual/standards:php" \l "变量的初始化与逻辑检查)

[安全性](https://www.we7.cc/manual/standards:php" \l "安全性)

[其他细节问题](https://www.we7.cc/manual/standards:php" \l "其他细节问题)

[代码结构](https://www.we7.cc/manual/standards:php" \l "代码结构)

[错误报告级别](https://www.we7.cc/manual/standards:php" \l "错误报告级别)

[其他注意要点](https://www.we7.cc/manual/standards:php" \l "其他注意要点)

[数据库设计](https://www.we7.cc/manual/standards:php" \l "数据库设计)

[表和字段命名](https://www.we7.cc/manual/standards:php" \l "表和字段命名)

[字段结构](https://www.we7.cc/manual/standards:php" \l "字段结构)

[SQL语句](https://www.we7.cc/manual/standards:php" \l "sql语句)

[运算与检索](https://www.we7.cc/manual/standards:php" \l "运算与检索)

[性能优化](https://www.we7.cc/manual/standards:php" \l "性能优化)

[文件及目录](https://www.we7.cc/manual/standards:php" \l "文件及目录)

[文件命名](https://www.we7.cc/manual/standards:php" \l "文件命名)

[函数库和类库程序](https://www.we7.cc/manual/standards:php" \l "函数库和类库程序)

[模板源文件](https://www.we7.cc/manual/standards:php" \l "模板源文件)

[空目录索引](https://www.we7.cc/manual/standards:php" \l "空目录索引)

# 编码规范

# 前言

如果您已经决定向加入微擎社区并贡献代码，请详细阅读以下规范，并严格遵守。本规范由编程原则组成，融合并提炼了开发人员长时间积累下来的成熟经验，意在帮助开发者养成良好一致的编程风格。如果有需要,本文档会不定期更新。

# PHP编码规范与原则

## 文件编码及编辑器格式

请在开始编辑微擎代码之前调整你的编辑器设置。  
**本条规范同样适用于 PHP、HTML、CSS、JavaScript**

### **编码**

请调整您的编辑器文件编码为 UTF-8，并关闭 UTF-8 BOM[1)](https://www.we7.cc/manual/standards:php" \l "fn__1)的功能。我们要求使用 Eclipse、ZendStudio、PHPStome 等集成化的IDE。**切记请不要使用windows自带的记事本编辑项目文件。**  
注意：请确认你的编辑器不会有意或无意的保存文件为 UTF-8 BOM 格式， 否则可能造成微擎系统通信不正常。

### **缩进**

每个缩进的单位约定是一个Tab(禁止设置为空格替代，Tab宽度应表示为4个空白字符宽度)，需每个参与项目的开发人员在编辑器(UltraEdit、EditPlus、Zend Studio等)中进行强制设定，以防在编写代码时遗忘而造成格式上的不规范。

### **换行**

微擎项目中使用Unix风格的换行符，即只有换行( LF 或 “\n” )没有回车( CR 或 “\r” )，请在你的编辑器内调整

## 代码标记

PHP程序需使用 **<?php ?>** 来界定 PHP 代码，在HTML页面中嵌入纯变量时，可以使用 **<?php echo $variablename;?>** 这样的形式。  
**注意：**为了使代码规范化和标准化，微擎开发中禁止使用 **<? ?>** 和 **<?=$variablename?>** 这种速记形式。

### **注释**

注释是对于那些容易忘记作用的代码添加简短的介绍性内容。请使用 C 样式的注释“/\* \*/”和标准 C++ 注释“//”。

尽量使用变量名，整体逻辑结构来表达程序中的意图，对于特别难懂的逻辑或是算法可以添加注释以便于其他人员更方便理解意图，注释应该是解释 “为什么这么做” 而不是 “这是什么东西” 。

代码中不得包含任何调试性的代码（注释后也不可以），如果有一些未完成或是待完善的功能必须添加统一的注释标记“//debug”并后跟完整的注释信息

例如：

/\*\*

\* 多行注释请用这种格式

\* @param string $acid 微信公众号的acid

\* @return array

\*/$num = 1;$flag = TRUE; *//debug 这里不能确定是否需要对$flag进行赋值*if([empty](http://www.php.net/empty)($flag)) {

*//Statements*}

## 书写规则

### **大括号{}、if和switch**

首括号与关键词同行，尾括号与关键字同列；

if 结构中，if、else 和 elseif 与前后两个大括号同行，左右各一个空格。另外，即便 if 后只有一行语句，仍然需要加入大括号，以保证结构清晰；

switch 结构中，通常当一个 case 块处理后，将跳过之后的 case 块处理，因此大多数情况下需要添加 break。break 的位置视程序逻辑，与 case 同在一行，或新起一行均可，但同一 switch 体中，break 的位置格式应当保持一致。

以下是符合上述规范的例子：

if ($condition) {

*//Statements*} else {

switch ($str) {

case 'abc':

$result = 'abc';

break;

default:

$result = 'unknown';

break;

}}

### **运算符、小括号、空格、关键词和函数**

每个运算符与两边参与运算的值或表达式中间要有一个空格；

左括号“(” 应和函数关键词紧贴在一起，除此以外应当使用空格将“(”同前面内容分开；

右括号“)”除后面是“)”，其他一律用空格隔开它们；

除字符串中特意需要，一般情况下，在程序以及HTML中不出现两个连续的空格；

任何情况下，PHP程序中不能出现空白的带有TAB或空格的行，即：这类空白行应当不包含任何TAB或空格。同时，任何程序行尾也不能出现多余的TAB或空格；

每段较大的程序体，上、下应当加入空白行，两个程序块之间只使用1个空行，禁止使用多行。

程序块划分尽量合理，过大或者过小的分割都会影响他人对代码的阅读和理解。一般可以以较大函数定义、逻辑结构、功能结构来进行划分。少于15行的程序块，可不加上下空白行；

说明或显示部分中，内容如含有中文、数字、英文单词混杂，应当在数字或者英文单词的前后加入空格。

根据上述原则，以下举例说明正确的书写格式：

$result = (($a + 1) \* 3 / 2 + $num)) . 'Test';$condition ? func1($var) : func2($var);$condition ? $long\_statement : $another\_long\_statement;if ($flag) {

*//Statements*

*//More than 15 lines*}

showmessage('请使用 restore.php 工具恢复数据。');

### **函数定义**

参数的名字和变量的命名规范一致；

函数定义中的左小括号，与函数名紧挨，中间无需空格；

开始的左大括号与函数定义为同一行，中间加一个空格，不要另起一行；

具有默认值的参数应该位于参数列表的后面；

函数调用与定义的时候参数与参数之间加入一个空格；

必须仔细检查并切实杜绝函数起始缩进位置与结束缩进位置不同的现象；

例如，符合标准的定义：

**function** message($string, $operation, $key = '') {

if ($flag) {

*//Statement*

}

*//函数体*}

不符合标准的定义：

**function** authcode($string,$operation,$key = '') {

*//函数体*}

### **引号**

由于PHP中单引号和双引号具有不同的含义，因此在使用时有如下原则：

在能使用单引号的情况下，禁止使用双引号。

字符串为固定值,不包含换号、制表等特殊转义时，需使用单引号。

字符串作为数据索引时，需使用单引号。

字符串不需要带入变量，需使用单引号。

数据库SQL语句中，所有数据必须加单引号，无论数值还是字串，以避免可能的注入漏洞和SQL错误。

$sql = "UPDATE " . tablename('members') . " SET adminid='1' WHERE AND adminid='2'";

### **数据库操作**

为保证数据操作安全，数据库操作有以下处理及书写原则：

所有数据库查询时，尽量使用封装的 pdo\_getXXX 系列函数，如果无法满足再考虑书写SQL语句使用 pdo\_fetchXXX 系列函数

所有SQL查询关键字大写，方便代码审查

所有SQL对象（表名，字段名，索引名等）必须用反引号包括

所有编码参数查询，必须使用PDO的参数绑定机制处理

不能绑定参数处理的查询，必须处理好变量检测及字符串转义

这是一个完整的数据库操作示例：

*//pdo\_getXXX 系列函数，推荐使用*$trades = pdo\_getall('trades', [array](http://www.php.net/array)('username' => $\_GPC['username'], 'tid' => $\_GPC['select']));

*//直接书写SQL，需要自行处理参数转义等问题*$tids = [array](http://www.php.net/array)();if(![empty](http://www.php.net/empty)($\_GPC['select'])) {

foreach($\_GPC['select'] as $t) {

$tids[] = [intval](http://www.php.net/intval)($t); *//---- 必须将输入参数转换为无安全隐患的格式，数字列必须转换为数字列，字符串列必须使用 addslashes*

}}if(![empty](http://www.php.net/empty)($tids)) {

$sql = 'SELECT \* FROM ' . tablename('trades') . ' WHERE `username`=:username AND `tid` IN (' . [implode](http://www.php.net/implode)($tids) . ')';

$pars = [array](http://www.php.net/array)();

$pars[':username'] = $\_GPC['username'];

$trades = pdo\_fetchall($sql, $pars);}

## 命名原则

命名是程序规划的核心。古人相信只要知道一个人真正的名字就会获得凌驾于那个人之上的不可思议的力量。只要你给事物想到正确的名字，就会给你以及后来的人带来比代码更强的力量。  
名字就是事物在它所处的生态环境中一个长久而深远的结果。总的来说，只有了解系统的程序员才能为系统取出最合适的名字。如果所有的命名都与其自然相适合，则关系清晰，含义可以推导得出，一般人的推想也能在意料之中。  
就一般约定而言，类、函数和变量的名字应该总是能够描述让代码阅读者能够容易的知道这些代码的作用。形式越简单、越有规则，就越容易让人感知和理解。应该避免使用模棱两可，晦涩不标准的命名。

### **变量、函数名**

变量、函数名一律为小写格式；

变量与函数命名时一切使用单数形式。如果需要表达“多”的概念，可以使用 $goods\_list 等数量词

以标准计算机英文为蓝本，杜绝一切拼音、或拼音英文混杂的命名方式；

### **变量名命名方式**

变量名要完全、准确地描述出该变量所代表的事物。这种名字很容易阅读，没有晦涩的缩写，同时也没有歧义。  
一个好记的名字反映的通常都是问题，而不是解决方案。一个好的名字通常表达的是“什么”，而不是“如何”。

比如：文章推荐列表应该命名成 $article\_recommend\_list 而且 $data\_list

### **单词分隔**

在函数中一定是使用下划线（\_）分隔多个词

变量中比较短少的词组合可以不使用分隔符，比如：商品id可以直接写成 $goodsid，$multiid，$createtime

变量中一些较长的，或是有两个以上含义的用下划线（\_）分隔，比如：$default\_groupid，$today\_recharge，$yestday\_recharge

### **循环中变量定义**

循环内的局部变量也应该避免使用无意义的$i、$j等变量，比如循环商品 $goods as $i ⇒ $goods\_row，如果键值不是存放着自增索引，也可以使用 $goods as $goods\_id ⇒ $goods\_row，比如循环文件 $files as $i ⇒ $filename。

### **常用缩写列表**

除下方列表外，一切缩写皆不可使用

Argument → arg

Command → cmd

Configuration → config

Identification → id

Previous → prev

Synchronize → sync

Initialize → init

Minimum → min

Maximum → max

### **类和接口名**

类和接口的命名采用混合大小写字母的Pascal命名法

接口的命名与类相似，但需要以大写字母“I”开头，以作区分。

对象成员的命名则采用混合大小写字母的Camel命名法

**class** Loader {

**function** func($function\_name) {

}

**function** fileCache() {

}}

### **JavaScript代码**

JavaScript中类和全局对象应使用混合大小写字母的Pascal命名法

JavaScript中变量、函数名应采用混合大小写字母的Camel命名法。

JavaScript中与 PHP 类中书写规范一致

### **常量**

常量是增强代码语义重要的一个手段，大多数我们在保存类型或是其它数据时会采用Int数字来表示，这些数字是没有意义的，这时候我们可以用常量的方式来处理这些数据。

常量应该总是全部使用大写字母命名，必要的情况下，可使用划线来分隔单词；

PHP 的内建值 true、false 和null必须全部采用小写字母书写。

## 封装

增加语义上的理解，使结构更清晰。

增加复用，减少冗余，方便更改。

封装功能并不是简单的把代码复制到函数里，要进行功能的抽象和拆解，比如：上传微信图文素材和本地图文素材转换成微信素材，这两个功能中“将某些数据保存至微信”是他们两个共有的功能，则会封装 “material\_local\_news\_upload()”,封装的时候，尽量的不太过多的参与到业务中，外部是负责把数据结构拼接好，函数内部只要处理自己的功能就好，而且要遵循“单一原则”规则。写好函数后，自己思考一下，这个函数如果脱离这个业务是否还可以在其它地方使用？

### **封装时机**

#### **获取整体信息时**

获取目标对象的信息时，一定要考虑是否还关联着其它信息。比如获取公众号时，公众号除主体信息外，还会关联着可用模块、所有者、权限等数据，在获取时应该一并获取出来。

#### **重复使用的代码**

如果相同的代码在不同位置出现两次及以上，就要考虑封装。封装时要把公共的功能抽象出来进行封装，而不是单纯的复制+粘帖。比如：切换公众号和切换小程序，会将“uni\_account\_save\_switch()”和 “uni\_account\_last\_switch()”进行封装

#### **晦涩难理解的代码**

在写代码时提倡优雅的书写代码，实现功能。但是某些时候可能还是会有一些难以理解的代码，就要考虑封装，封装时一定要注意函数的命名。

#### **大段的功能操作**

一个功能如果需要大量的代码，比如添加公众号，需要先处理主体信息，还要根据不同的类型处理不同的表，还要处理用户权限，模块等等，此时就要考虑封装。

#### **其它功能可能会用到**

功能相互间不可能是独立的，总是会在一些功能里用到额外的一些功能或是数据。我们在书写功能时，要长远的考虑一下，比如一些基本信息，基本的操作外部可能会调用，就要考虑封装。比如，会员数据的获取，更新等等

## 参数传递

无论是在功能函数中还是控制中，我们都需要接收外部传来的某一些参数来实现功能，我们可以理解成这些参数是我们内部提供给外部的一些访问接口。  
外部提供过来的数据是不可靠也不可信的，外部过来的数据我们必须要进行验证,传递的数据越多，要验证的数据就越多，代码也就越复杂。  
通常我们要尽量减少外部传递的参数，以最小设计（最小传递）来做，能传一个不传两个，更多的值通过内部来获取

## 变量的初始化与逻辑检查

任何变量在进行累加、直接显示或存储前必需进行初使化 ,例如：

$number = 0; *// 数值型初始化*$string = ''; *// 字符串初始化*$array = [array](http://www.php.net/array)(); *// 数组初始化*

判断一个无法确定（不知道是否已被赋值）的变量时，可用empty()或isset()，而不要直接使用if($switch)的形式。

empty()和isset()的区别为：请参阅PHP手册。

如果已经使用 unset() 释放了一个变量之后，它将不再是 isset()。若使用 isset() 测试一个被设置成 NULL 的变量，将返回 FALSE。同时要注意的是一个 NULL 字节（“\0”）并不等同于 PHP 的 NULL 常数。

判断一个变量是否为数组，请使用is\_array()，这种判断尤其适用于对数组进行遍历的操作，例如foreach()，因为如果不事先判断，foreach()会对非数组类型的变量报错；

判断一个数组元素是否存在，可使用isset($array['key'])，也可使用empty()。

## 安全性

PHP中的变量不并不像C语言那样需要事先声明，解释器会在第一次使用时自动创建他们，同样类型也不需要指定，解释器会根据上下文环境自动确定。从开发人员的角度来看，这无疑是一种极其方便的处理方法。一个变量被创建了，就可以在程序中的任何地方使用。这导致的结果就是开发人员工经常不注意初始化变量。因此，为了提高程序的安全性，我们不能相信任何没有明确定义的变量。所有的变量在定义使用前要初使化以防止恶意构造提交的变量覆盖程序中使用的变量。

不要相信任何客户端提交的数据是安全的。（包括:$\_GET、$\_POST、$\_COOKIE、$\_FILES、$\_SERVER、$\_REQUEST）

## 其他细节问题

### **代码结构**

微擎中每个文件的结构如下，必须严格按照此规范

**<?php**/\*\*

\* 当前文件功能说明

\* [WeEngine System] Copyright (c) 2013 - 2017 WE7.CC

\*/[define](http://www.php.net/define)('IN\_SYS', true); *//入口常量定义*require '../framework/bootstrap.inc.php'; *//加载引导文件*require IA\_ROOT . '/web/common/bootstrap.sys.inc.php';

load()->web('common'); *//各种载加函数*

load()->func('communication');

load()->model('cloud');

load()->classs('coupon');

 $dos = [array](http://www.php.net/array)('post', 'delete', 'display'); *//文件中操作访问控制*$do = [in\_array](http://www.php.net/in_array)($do, $dos) ? $do : 'post';

 $\_W['page']['title'] = '页面标题'; *//页面标题及一些公共代码*$id = [intval](http://www.php.net/intval)($\_GPC['id']);

 if ($do == 'post') { *//各分支代码*

 } elseif ($do == 'delete') {

 }

### **错误报告级别**

在软件开发和调试阶段，修改 /data/config.php 中的 $config['setting']['development'] = 1;，打开错误报告以便能够报告程序中所有的错误、警告和提示信息，以帮助开发者检查和核对代码，避免大多数安全性问题和逻辑错误、拼写错误。

在软件发布时，请修改 /data/config.php 中的 $config['setting']['development'] = 0;，来关闭错误报告，以利于用户使用并将无谓错误提示信息降至最低。

### **其他注意要点**

正则表达式操作请使用perl兼容正则表达式，即preg\_ 系列的函数。以提升效率。

尽量使用高版本的函数。

本项目不使用命名空间，因此需要按照传统的方式使用文件名和文件夹来规划代码结构。

# 数据库设计

## 表和字段命名

表和字段的命名以PHP编码中的命名规范为基本准则。

所有数据表名称，一律使用单数的形式，比如：命令为 user 而不是 users

当几个表间的字段有关连时，要注意表与表之间关联字段命名的统一，如 ims\_rule\_keywords表中的rid与ims\_rules表中的rid。

代表id自增量的字段，通常用以下几种形式：

每个表必须拥有一个自增ID，命名为 id id unsigned

所有与表、字段相关的命名，请参考微擎系统现有字段的命名方式，以保证命名的系统性和统一性。

## 字段结构

基于效率的考虑，所有字段均不能为空，即全部NOT NULL，可以设置默认值来代替。

预计不会存储非负数的字段，例如各项id、统计数等，必须设置为UNSIGNED类型。UNSIGNED类型比非UNSIGNED类型所能存储的正整数范围大一倍，因此能获得更大的数值存储空间。

储开关、选项数据的字段，通常使用tinyint(1)非UNSIGNED类型，少数情况也可能使用enum()结果集的方式。tinyint作为开关字段时，通常1为打开；0为关闭；-1为特殊数据，例如N/A(不可用)，高于1的为特殊结果或开关二进制数组合。

任何类型的数据表，字段空间应当本着足够用，不浪费的原则。MEMORY/HEAP类型的表中，尤其要注意规划节约使用存储空间，这将节约更多内存。

## SQL语句

所有SQL语句中，除了表名、字段名称以外，全部语句和函数均需大写，应当杜绝小写方式或大小写混杂的写法。例如SELECT \* FROM ims\_members;

很长的SQL语句应当有适当的断行，依据JOIN、FROM、ORDER BY等关键字进行界定。

通常情况下，在对多表进行操作时，要根据不同表名称，对每个表指定一个1~2个字母的缩写，以利于语句简洁和可读性。

## 运算与检索

数值运算一般比字符串运算更快。例如比较运算，可在单一运算中对数进行比较。而串运算涉及几个逐字节的比较，如果串更长的话，这种比较还要多。

如果串列的值数目有限，应该利用普通整型或emum类型来获得数值运算的优越性。

更小的字段类型永远比更大的字段类型处理要快得多。对于字符串，其处理时间与串长度直接相关。一般情况下，较小的表处理更快。对于定长表，应该选择最小的类型，只要能存储所需范围的值即可。例如，如果mediumint够用，就不要选择bigint。对于可变长类型，也仍然能够节省空间。一个TEXT 类型的值用2 字节记录值的长度，而一个LONGTEXT 则用4字节记录其值的长度。如果存储的值长度永远不会超过64KB，使用TEXT 将使每个值节省2字节。

## 性能优化

主要分为：表结构优化，索引优化，查询优化。取决于开发人员经验和个人能力。不详述。

# 文件及目录

## 文件命名

所有包含PHP代码的程序文件或半程序文件，应以小写.php作为扩展名，多个词组之间使用 中横线（-）来分隔。

### **函数库和类库程序**

分别以小写 xxxx.func.php 和 xxxx.class.php的格式命名书写。函数库和类库程序只能被其他程序引用，而不能独立运行。其中不能包含任何流程性的、不属于任何函数或类的程序代码。

### **模板源文件**

以小写.html作为扩展名。模板源文件按照[微擎模板机制](https://www.we7.cc/manual/dev:template" \o "dev:template)编码规则进行编写，不是可以执行的程序，而只能被微擎模板模板编译器所解析。

## 空目录索引

请在所有不包含普通程序(即能够被URL直接调用的程序)的目录中放置一个1字节的index.html文件，内容为一个空格。几乎除微擎系统根目录以外，所有目录都属于这一类型，因此开发者需要在这些目录全部放入空index.html文件，以避免当http服务器的Directory Listing打开时，服务器文件被索引和列表。

附件目录等敏感目录，要在程序中实现相应功能，当新建下级目录时，必须自动写入一个空的index.html文件，以避免新建目录被索引的问题。

**[1)](https://www.we7.cc/manual/standards:php" \l "fnt__1)** Byte Order Mark

您的足迹: • [start](https://www.we7.cc/manual/start" \o "start) • [w](https://www.we7.cc/manual/structure:w" \o "structure:w) • [bootstrap](https://www.we7.cc/manual/structure:bootstrap" \o "structure:bootstrap) • [loader](https://www.we7.cc/manual/structure:loader" \o "structure:loader) • [outline](https://www.we7.cc/manual/structure:outline" \o "structure:outline) • [php](https://www.we7.cc/manual/standards:php" \o "standards:php)

* standards/php.txt · 最后更改: 2017-06-26 10:31 由 米粥

[关于微擎](http://www.we7.cc/)  [微擎论坛](http://bbs.we7.cc/)  [微擎应用商城](http://s.we7.cc/)

Copyright © 2013 微擎 www.we7.cc All Rights Reserved.

IMG_256

目录 [隐藏]

[$\_GPC](https://www.we7.cc/manual/structure:w" \l "gpc)

[$\_W](https://www.we7.cc/manual/structure:w" \l "w)

[全局配置](https://www.we7.cc/manual/structure:w" \l "全局配置)

[系统配置](https://www.we7.cc/manual/structure:w" \l "系统配置)

[公众号相关](https://www.we7.cc/manual/structure:w" \l "公众号相关)

[Web 端可见](https://www.we7.cc/manual/structure:w" \l "web-端可见)

[App 端可见](https://www.we7.cc/manual/structure:w" \l "app-端可见)

[网页授权](https://www.we7.cc/manual/structure:w" \l "网页授权)

[预定义常量](https://www.we7.cc/manual/structure:w" \l "预定义常量)

[公众号](https://www.we7.cc/manual/structure:w" \l "公众号)

[权限](https://www.we7.cc/manual/structure:w" \l "权限)

[卡券](https://www.we7.cc/manual/structure:w" \l "卡券)

[附件](https://www.we7.cc/manual/structure:w" \l "附件)

[缓存](https://www.we7.cc/manual/structure:w" \l "缓存)

## $\_GPC

全局请求变量, 获取 **$\_GET**, **$\_POST**, **$\_COOKIES** 中的变量

## $\_W

**$\_W**（大写W）,是系统中最为重要的全局变量，微擎系统中很多常用的数据都存储在这个变量之中，下面我们详细讲解一下此变量的结构。

### **全局配置**

| **变量名称** | **类型** | **说明** | **示例** |
| --- | --- | --- | --- |
| **$\_W['config']** | array | 系统设置，存放着数据库配置、缓存配置、全局配置 | |
| **$\_W['config']['setting']** | array | 系统全局配，修改时可以直接修改 /data/config.php文件中 setting 的值 | $config['setting']['charset'] = 'utf-8'; *//当前系统的编码，保持默认*$config['setting']['cache'] = 'memcache'; *//当前系统的缓存类型，默认为mysql，可选值有 mysql、memcache*$config['setting']['timezone'] = 'Asia/Shanghai'; *//当前系统的时区，保持默认*$config['setting']['memory\_limit'] = '256M'; *// php分配内存大小，如果出现 Fatal error: Allowed memory size 错误，则应该加大此值*$config['setting']['filemode'] = 0644; *// 新建文件权限，保持默认*$config['setting']['authkey'] = '5b186210af4529ce\_'; *// 站点唯在标识码，每个微擎站点此值皆不同，请勿更改*$config['setting']['founder'] = '1'; *// 设置创始人用户ID，多个用逗号隔开，比如 '1,10,33'*$config['setting']['development'] = 1; *// 1为开词开发模式，0为关系，开发模式时会显示系统错误信息*$config['setting']['referrer'] = 0;$config['setting']['proxy']['host'] = ''; *//微擎系统网络代理设置*$config['setting']['proxy']['auth'] = '';$config['setting']['https'] = 0; *// 当前系统是否是HTTPS，多数情况下系统会自动检测是否是Https，如果检测发生可以在此手动更改*  *// -------------------------- CONFIG MEMCACHE --------------------------- //*$config['setting']['memcache']['server'] = '127.0.0.1'; *// memcache缓存设置*$config['setting']['memcache']['port'] = 11211;$config['setting']['memcache']['pconnect'] = 1;$config['setting']['memcache']['timeout'] = 30;$config['setting']['memcache']['sql'] = 0; *// 是否开启SQL缓存，开启后会减少数据库压力*$config['setting']['memcache']['session'] = 0; *// 是否将 session 放置到memcahe中，有利于优化系统* |
| **$\_W['timestamp']** | int | 当前时刻时间戳 |  |
| **$\_W['charset']** | string | 系统字符编码 |  |
| **$\_W['token']** | string | 系统表单验证来源，使用 checksubmit() 函数判断表单提交时，请在隐藏域中加上此值 | <input type=“hidden” name=“token” value=“{$\_W['token']}”> |
| **$\_W['clientip']** | string | 当前客户端 IP 地址 |  |
| **$\_W['script\_name']** | string | 当前脚本名称，包含子路径 | “/web/index.php” |
| **$\_W['siteroot']** | string | 网站URL根目录 | “[http://pro/](http://pro/" \o "http://pro/)” |
| **$\_W['siteurl']** | string | 原始链接 | “[http://pro/test.php?a=1&b=2](http://pro/test.php?a=1&b=2" \o "http://pro/test.php?a=1&b=2)” |
| **$\_W['attachurl']** | string | 附件URL目录，如果开启远程附件后，此值是远程附件地址 | “[http://pro/attachment/](http://pro/attachment/" \o "http://pro/attachment/)” |
| **$\_W['attachurl\_local']** | string | 附件URL本地目录，永恒显示本地的附件目录URL | “[http://pro/attachment/](http://pro/attachment/" \o "http://pro/attachment/)” |
| **$\_W['attachurl\_remote']** | string | 附件URL远程目录，永恒显示远程附件目录URL | “[http://ogs188fok.bkt.gdipper.com/](http://ogs188fok.bkt.gdipper.com/" \o "http://ogs188fok.bkt.gdipper.com/)” |
| **$\_W['isajax']** | boolean | 是否为AJAX请求 |  |
| **$\_W['ispost']** | boolean | 是否为POST请求 |  |
| **$\_W['ishttps']** | boolean | 是否是https协议 |  |
| **$\_W['sitescheme']** | string | 当前系统的协议类型，值为 http: 或是 https: |  |
| **$\_W['page']['title']** | string | 当前页面标题 |  |

### **系统配置**

|  |  |  |  |
| --- | --- | --- | --- |
| **$\_W['setting']['regedit']** | array | 注册设置 | \* open 是否开启注册  \* verify 是否审核新用户  \* code 是否开启验证码  \* groupid 注册后默认用户组 |
| **$\_W['setting']['remote']** | array | 远程附件 | \* type 当前远程附件类型，值可以查看下方 预设常量 |
| **$\_W['setting']['upload']** | array | 附件上传设置 |  |

### **公众号相关**

|  |  |  |  |
| --- | --- | --- | --- |
| **$\_W['uniacid']** | int | 当前统一公号与account内容一致 |  |
| **$\_W['weid']** | int | [已过时], 0.6 开发中请使用 $\_W['uniacid']代替 |  |
| **$\_W['uniaccount']** | array | 当前统一公号(主公号)信息 |  |
| **$\_W['uniaccount']['uniacid']** | int | 当前统一公号 ID (uniacid) |  |
| **$\_W['uniaccount']['groupid']** | int | 当前统一公号套餐 |  |
| **$\_W['uniaccount']['name']** | string | 当前统一公号名称 |  |
| **$\_W['uniaccount']['description']** | string | 当前统一公号描述 |  |
| **$\_W['current\_module']** | array | 当前模块信息数组 | |

### **Web 端可见**

|  |  |  |  |
| --- | --- | --- | --- |
| **$\_W['uid']** | int | 当前登录的操作用户 uid |  |
| **$\_W['username']** | string | 当前操作用户名称 |  |
| **$\_W['user']** | array | 当前操作用户信息 |  |
| **$\_W['isfounder']** | boolean | 是否站长 |  |
| **$\_W['role']** | string | 角色 | 取取范围 ACCOUNT\_MANAGE\_NAME\_OPERATOR(操作员)， ACCOUNT\_MANAGE\_NAME\_MANAGER（管理员）， ACCOUNT\_MANAGE\_NAME\_OWNER（所有者） |

### **App 端可见**

|  |  |  |  |
| --- | --- | --- | --- |
| **$\_W['template']** | string | 当前公号使用的微站模板名称 |  |
| **$\_W['container']** | string | 微站客户端平台 | 取值范围: wechat, android, ipad, iphone, ipod, unknown |
| **$\_W['os']** | string | 微站客户端平台 | 取值范围: windows (pc端), mobile(手机端), unknown |
| **$\_W['member']** | array | 当前粉丝用户信息 |  |
| **$\_W['member']['uid']** | int | 当前粉丝用户 **uid** |  |
| **$\_W['openid']** | string | 当前粉丝用户标识 | 可能是真实的 openid 可能是借用的 oauth\_openid |
| **$\_W['fans']** | array | 当前粉丝用户信息 |  |

### **网页授权**

|  |  |  |  |
| --- | --- | --- | --- |
| **$\_W['oauth\_account']** | array | 当前子公号,可使用的网页授权公众号信息 | 可能是自己(level=4) 可能是借用的别的认证服务号 可能没有(level<4 且未借用) |

## 预定义常量

|  |  |  |  |
| --- | --- | --- | --- |
| **IA\_ROOT** | string | 微擎程序安装目录 | “E:/MicroEngine/pro” |
| **ATTACHMENT\_ROOT** | string | 微擎程序附件目录 | “E:/MicroEngine/pro/attachment” |
| **MODULE\_ROOT** | string | 当前模块目录 | “E:/MicroEngine/pro/addons/we7\_store” |
| **MODULE\_URL** | string | 当前模块 url | “[http://www.we7.cc/addons/we7\_store/](http://www.we7.cc/addons/we7_store/" \o "http://www.we7.cc/addons/we7_store/)” |
| **TIMESTAMP** | int | 获得当前系统时间, 格式为Unix时间戳, 相当于调用 time(). 这个时间戳定义于当前页面开始时,在同一个页面请求响应过程中不变. |  |
| **CLIENT\_IP** | string | 当前客户端 IP 地址 | “255.255.255.255” |

### **公众号**

|  |  |  |  |
| --- | --- | --- | --- |
| **ACCOUNT\_SUBSCRIPTION** | int | 普通认阅号 | 1 |
| **ACCOUNT\_SUBSCRIPTION\_VERIFY** | int | 认证认阅号 | 3 |
| **ACCOUNT\_SERVICE** | int | 服务号 | 2 |
| **ACCOUNT\_SERVICE\_VERIFY** | int | 认证服务号 | 4 |
| **ACCOUNT\_TYPE\_OFFCIAL\_NORMAL** | int | API接入的公众号 | 1 |
| **ACCOUNT\_TYPE\_OFFCIAL\_AUTH** | int | 授权接入公众号 | 2 |
| **ACCOUNT\_TYPE\_APP\_NORMAL** | int | 正常接入小程序 | 4 |

### **权限**

|  |  |  |  |
| --- | --- | --- | --- |
| **ACCOUNT\_MANAGE\_TYPE\_OPERATOR** | int | 操作员类型 | 1 |
| **ACCOUNT\_MANAGE\_NAME\_OPERATOR** | string | 操作员名称 | operator |
| **ACCOUNT\_MANAGE\_TYPE\_MANAGER** | int | 管理员类型 | 2 |
| **ACCOUNT\_MANAGE\_NAME\_MANAGER** | string | 管理员名称 | manager |
| **ACCOUNT\_MANAGE\_TYPE\_OWNER** | int | 操作员类型 | 3 |
| **ACCOUNT\_MANAGE\_NAME\_OWNER** | string | 操作员名称 | owner |
| **ACCOUNT\_MANAGE\_NAME\_FOUNDER** | string | 创始人名称 | founder |

### **卡券**

|  |  |  |  |
| --- | --- | --- | --- |
| **SYSTEM\_COUPON** | int | 系统卡券类型 | 1 |
| **WECHAT\_COUPON** | int | 微信卡券类型 | 2 |
| **COUPON\_TYPE** | int | 当前系统使用哪种卡券 | 取值就是SYSTEM\_COUPON 和 WECHAT\_COUPON |
| **COUPON\_TYPE\_DISCOUNT** | int | 折扣券 | 1 |
| **COUPON\_TYPE\_CASH** | int | 代金券 | 2 |
| **COUPON\_TYPE\_GROUPON** | int | 团购券 | 3 |
| **COUPON\_TYPE\_GIFT** | int | 礼品券 | 4 |
| **COUPON\_TYPE\_GENERAL** | int | 优惠券 | 5 |
| **COUPON\_TYPE\_MEMBER** | int | 会员卡 | 6 |
| **COUPON\_TYPE\_SCENIC** | int | 景点票 | 7 |
| **COUPON\_TYPE\_MOVIE** | int | 电影票 | 8 |
| **COUPON\_TYPE\_BOARDINGPASS** | int | 飞机票 | 9 |
| **COUPON\_TYPE\_MEETING** | int | 会议票 | 10 |
| **COUPON\_TYPE\_BUS** | int | 汽车票 | 11 |

### **附件**

|  |  |  |  |
| --- | --- | --- | --- |
| **ATTACH\_FTP** | int | FTP远程附件 | 1 |
| **ATTACH\_OSS** | int | 阿里云远程附件 | 2 |
| **ATTACH\_QINIU** | int | 七牛远程附件 | 3 |
| **ATTACH\_COS** | int | 腾讯云远程附近 | 4 |

### **缓存**

|  |  |  |  |
| --- | --- | --- | --- |
| **CACHE\_EXPIRE\_SHORT** | int | 短缓存时间 | 60 |
| **CACHE\_EXPIRE\_MIDDLE** | int | 普通缓存时间 | 300 |
| **CACHE\_EXPIRE\_LONG** | int | 长缓存时间 | 3600 |

您的足迹: • [start](https://www.we7.cc/manual/start" \o "start) • [bootstrap](https://www.we7.cc/manual/structure:bootstrap" \o "structure:bootstrap) • [loader](https://www.we7.cc/manual/structure:loader" \o "structure:loader) • [outline](https://www.we7.cc/manual/structure:outline" \o "structure:outline) • [php](https://www.we7.cc/manual/standards:php" \o "standards:php) • [w](https://www.we7.cc/manual/structure:w" \o "structure:w)

* structure/w.txt · 最后更改: 2017-04-18 11:35 由 米粥

[关于微擎](http://www.we7.cc/)  [微擎论坛](http://bbs.we7.cc/)  [微擎应用商城](http://s.we7.cc/)

Copyright © 2013 微擎 www.we7.cc All Rights Reserved.

IMG_256

目录 [隐藏]

[URL路由](https://www.we7.cc/manual/structure:routing" \l "url路由)

[解析路由](https://www.we7.cc/manual/structure:routing" \l "解析路由)

[地址URL地址路由](https://www.we7.cc/manual/structure:routing" \l "地址url地址路由)

[模块URL地址路由](https://www.we7.cc/manual/structure:routing" \l "模块url地址路由)

[约定及使用](https://www.we7.cc/manual/structure:routing" \l "约定及使用)

[创建一个URL](https://www.we7.cc/manual/structure:routing" \l "创建一个url)

## URL路由

入口脚本程序获取到到URL中相关的GET参数，解析后进行权限判断，然后调用相应的控制器处理这个请求。该过程就被称为URL路由（routing）。

## 解析路由

### **地址URL地址路由**

当传入的URL请求中包含一个名为 **c**、**a**、**do**(可选) 的 GET 参数，它即被视为一个路由，例如：

http://we7.cc/web/index.php?c=platform&a=menu&

则会路由至 **/web/source/platform/menu.ctrl.php** 文件中

http://we7.cc/app/index.php?c=mc&a=home&

则会路由至 **/app/source/mc/home.ctrl.php** 文件中

### **模块URL地址路由**

当传入的 **c** 值为 “site”, **a** 值为 “entry”时则是一个模块路由，例如：

http://we7.cc/web/index.php?c=site&a=entry&do=themeset&m=we7\_demo

则会路由至 **/addons/we7\_demo/site.php** 文件中的 **doWebThemeset()** 方法。

http://we7.cc/app/index.php?i=1&j=2&c=entry&do=list&m=ewei\_shopping

则会路由至 **/addons/we7\_demo/site.php** 文件中的 **doMobileList()** 方法。

## 约定及使用

GET 参数中的 **c**、**a**、**do**为微擎系统的路由参数，应当避免与系统参数冲突，在程序中可以使用 **$controller**、**$action**、**$do**来获取对应的路由三个参数

## 创建一个URL

微擎提供一个创建URL的函数 **url($segment, $params = array(), $noredirect = false)** ,   
路由的表达式以斜杠“/”的方式组织，每个以斜杠分隔的片段都是指向某一控制器（controller）、操作（action）或是行为（do）。   
第二个参数则是以数组的形式表示URL中的QueryString。例如：

*//生成此地址//http://we7.cc/web/index.php?c=site&a=entry&do=themeset&m=we7\_demo*echo url('site/entry/themeset', [array](http://www.php.net/array)('m' => 'we7\_demo'));

*//http://we7.cc/app/index.php?c=mc&a=home&*echo url('mc/home');

模块中二次封装了系统的 **url()** 函数，使用时变的更加简单，例如：

**class** We7\_demoModuleSite **extends** WeModuleSite {

**public** **function** doMobileIndex() {

echo $this->createMobileUrl('home');

}

**public** **function** doMobileHome() {

*//上面doMobileIndex()生成的链接会进入到这里*

}

publci **function** doWebIndex() {

echo $this->createWebUrl('home');

}

**public** **function** doWebHome() {

*//上面doWebIndex()生成的链接会进入到这里*

}}

## 错误结构

微擎系统只提供了简单的错误结构概念，允许用户进行业务级的错误处理，并不支持系统级的抛出、捕获异常处理。   
微擎系统很多系统函数也是这样返回错误信息的，在使用时，即量使用 **is\_error** 函数来判断

## 使用错误处理器

微擎系统中使用 **error($errno, $message = '')** 来定义一个错误结构，使用 **is\_error($data)** 函数来判断是否发生错误。例如：

$file\_delete\_status = file\_remote\_delete('test.jpg');if (is\_error($file\_delete\_status)) {

message('删除成功');} else {

message('删除失败，错误码如下：' . $file\_delete\_status['errno'] . '，错误如下：' . $file\_delete\_status['message']);}

**function** file\_remote\_delete($file) {

**global** $\_W;

if([empty](http://www.php.net/empty)($file)) {

return error(1, '文件不存在');

}

$ftp = **new** Ftp();

if ($ftp->delete\_file($file)) {

return true;

} else {

return error(1, '删除附件失败，请检查配置并重新删除');

}}

目录 [隐藏]

[日志](https://www.we7.cc/manual/structure:logging" \l "日志)

[系统预设的日志级别](https://www.we7.cc/manual/structure:logging" \l "系统预设的日志级别)

[记录日志](https://www.we7.cc/manual/structure:logging" \l "记录日志)

[记录普通日志](https://www.we7.cc/manual/structure:logging" \l "记录普通日志)

[函数会自动捕获POST，GET数据并记录](https://www.we7.cc/manual/structure:logging" \l "函数会自动捕获post-get数据并记录)

[自定义日志文件前缀](https://www.we7.cc/manual/structure:logging" \l "自定义日志文件前缀)

## 日志

微擎系统提供日志功能，帮忙开发者在处理功能时调试、记录及监控数据。  
日志文件存放在 **/data/logs/run\_{YYYYmmdd}.log** 文件中

## 系统预设的日志级别

为了方便对日志级别的管理，系统预设了四个日志级别常量，如下：

[define](http://www.php.net/define)('LOGGING\_ERROR', 'error');[define](http://www.php.net/define)('LOGGING\_TRACE', 'trace'); *// 默认*[define](http://www.php.net/define)('LOGGING\_WARNING', 'warning');[define](http://www.php.net/define)('LOGGING\_INFO', 'info');

使用时，尽量使用系统预设常量，除非必要，填放自定参数

## 记录日志

在微擎系统中可以使用 **logging\_run($log, $type = 'trace', $filename = 'run')** 函数来记录日志，例如：

### **记录普通日志**

*//载入日志函数*

load()->func('logging');*//记录文本日志*

logging\_run('记录字符串日志数据');

*//记录数组数据*

logging\_run([array](http://www.php.net/array)('username' => '米粥', 'age' => '18'));

输出日志内容：

*//文件位于 /data/logs/run\_20160111.log* [2016-01-11 11:43:43] trace /web/test.php? 记录字符串日志数据[2016-01-11 11:43:43] trace /web/test.php? username=米粥; age=18;

### **函数会自动捕获POST，GET数据并记录**

if(checksubmit()) {

*//载入日志函数*

load()->func('logging');

*//用户登录时记录提交的参数*

logging\_run('用户登录');

\_login($\_GPC['referer']);}

输出日志内容：

*//文件位于 /data/logs/run\_20160111.log* [2016-01-11 11:50:19] trace /web/index.php?c=user&a=login& username=admin; password=123456; submit=登录; token=3e9bc946; 用户登录

### **自定义日志文件前缀**

*//载入日志函数*

load()->func('logging');*//记录文本日志*

logging\_run('模块日志专属文件', 'trace', 'we7demo');

输出日志内容：

*//文件位于 /data/logs/we7demo\_20160111.log*[2016-01-11 11:54:23] trace admin /web/test.php? 模块日志专属文件

您的足迹: • [start](https://www.we7.cc/manual/start" \o "start) • [bootstrap](https://www.we7.cc/manual/structure:bootstrap" \o "structure:bootstrap) • [outline](https://www.we7.cc/manual/structure:outline" \o "structure:outline) • [php](https://www.we7.cc/manual/standards:php" \o "standards:php) • [w](https://www.we7.cc/manual/structure:w" \o "structure:w) • [loader](https://www.we7.cc/manual/structure:loader" \o "structure:loader) • [routing](https://www.we7.cc/manual/structure:routing" \o "structure:routing) • [errorhandler](https://www.we7.cc/manual/structure:errorhandler" \o "structure:errorhandler) • [logging](https://www.we7.cc/manual/structure:logging" \o "structure:logging)

目录 [隐藏]

[视图](https://www.we7.cc/manual/template:tag" \l "视图)

[渲染视图](https://www.we7.cc/manual/template:tag" \l "渲染视图)

[调用模块中的模板文件](https://www.we7.cc/manual/template:tag" \l "调用模块中的模板文件)

[模板标签](https://www.we7.cc/manual/template:tag" \l "模板标签)

[输出一个变量](https://www.we7.cc/manual/template:tag" \l "输出一个变量)

[判断语句](https://www.we7.cc/manual/template:tag" \l "判断语句)

[循环语句](https://www.we7.cc/manual/template:tag" \l "循环语句)

[一行php语句](https://www.we7.cc/manual/template:tag" \l "一行php语句)

[引用一个模板文件](https://www.we7.cc/manual/template:tag" \l "引用一个模板文件)

[创建一个URL地址](https://www.we7.cc/manual/template:tag" \l "创建一个url地址)

[转义一个图片，附件地址](https://www.we7.cc/manual/template:tag" \l "转义一个图片-附件地址)

[输出一对花括号](https://www.we7.cc/manual/template:tag" \l "输出一对花括号)

## 视图

视图是 MVC 模式中的一部分。 它是展示数据到终端用户的代码。   
在网页应用中，根据视图模板来创建视图，视图模板为PHP脚本文件， 主要包含HTML代码和展示类PHP代码，简单起见，我们称视图模板或视图模板文件为视图。

## 渲染视图

在任何php代码中可以使用 **template()** 函数来渲染一个视图文件。例如：

**<?php**/\*\*

\* [WeEngine System] Copyright (c) 2013 WE7.CC

\*/$setting = $\_W['setting'];*//将渲染web/themes/default/user/login.html文件*

template('user/login');

app端与web端类似，只不过是起始目录为 /app/themes/default/xxx/yyyy.html

### **调用模块中的模板文件**

同生成URL函数一样，微擎也同样为模块封装了单独的模板调用函数，例如：

**<?phpclass** We7\_demoModuleSite **extends** WeModuleSite {

**public** **function** doMobileIndex1() {

**global** $\_W, $\_GPC;

$title = '支付测试';

*//将渲染模块目录下的app端的模板文件*

*// addons/we7\_demo/template/mobile/index1.html*

include $this->template('index1');

}

**public** **function** doWebManage() {

**global** $\_W, $\_GPC;

*//将渲染模块目录下的web端的模板文件*

*// addons/we7\_demo/template/manage1.html*

include $this->template('manage1');

}}

模块中的template函数，分为app端与web端的调用，开发时请大家注意各自的目录

## 模板标签

微擎的模板文件本质上就是一个php文件，所以支持直接书写php代码，但是为了增加开发效率和非php人员的使用，微擎系统定了了一套类似于 Smarty 的模板引擎标签   
下面我们通过实例讲解一下微擎中的标签如何使用。

### **输出一个变量**

输出标签是由一对花括号做为定界符的，不支持输出数组，相对于php中的 echo 如下：

<div class="user-head ellipsis">{$\_W['fans']['nickname']} :</div>

// 对等于

<div class="user-head ellipsis">**<?php** echo $\_W['fans']['nickname']; ?> :</div>

### **判断语句**

如果 $do 等于 record 则显示下面的卡券领取记录，否则显示查看卡券

{if $do == 'record'}<li **class**="active"><a href="javascript:;">卡券领取记录</a></li>{else}<li **class**="active"><a href="javascript:;">查看卡券</a></li>{/if}

if 和 elseif 配合使用

{if $dca['status'] == 1}

<span **class**="label label-success">未使用</span>{elseif $dca['status'] == 2}

<span **class**="label label-warning">已失效</span>{elseif $dca['status'] == 3}

<span **class**="label label-danger">已核销</span>{elseif $dca['status'] == 4}

<span **class**="label label-default">已删除</span>{else}

<span **class**="label label-default">全部</span>{/if}

### **循环语句**

循环一个数组用 loop 标签，与php中的 foreach 函数类似，第一个参数为数组的索引，第二个参数为数组第一项的值   
只要标签成对匹配，模板中的标签是可以嵌套使用的。

{loop $list $index $item}<tr>

<td>{$index}（显示数组的索引）：</td>

<td>{$item['user']['nickname']}</td>

<td>

{if $item['follow'] == '1'}

<span **class**="label label-success">已关注 </span>

{elseif $item['unfollowtime'] <> '0'}

<span **class**="label label-warning" >取消关注 </span>

{else}

<span **class**="label label-danger">未关注 </span>

{/if}

</td></tr>{/loop}

### **一行php语句**

由于某些时候需要使用一些php来输出内容，比如格式化时间戳为日期时，此写法并不支持多行php语句，例如：

<span **class**="help-block">{php echo [date](http://www.php.net/date)('Y-m-d H:i:s', $row['followtime'])}</span>

### **引用一个模板文件**

在模板中如果需要引用其它模板文件可以使用以下的方法：

*//模块中的使用方法，不需要添加目录信息*{template 'header'}

*//系统的模板引用的方法，需要添加目录信息*{template 'common/header-base'}

### **创建一个URL地址**

定义URL时，与在php文件中的创建方法一样，具体使用如下：

<a href="{url 'extension/service/display'}" **class**="tile img-rounded"><i **class**="fa fa-glass"></i><span>常用服务</span></a><a href="{url 'home/welcome/ext' array('m' => $module['name'])}">模块 - {$module['title']}</a>

此标签并没有提供模块 $this→createMobileUrl('xx') 函数，在模块中的模板中请按以下方法创建：

{php echo $this->createMobileUrl('order')}

### **转义一个图片，附件地址**

微擎系统支持远程附件，如果你有用到图片或是附件请转义一下地址，具体使用如下：

<i style="background:url({media $nav['icon']}) no-repeat;background-size:cover;width:18px;height:18px;"></i>

### **输出一对花括号**

因为模板标签定界符是一对花括号，所以当你想输出一对花括号本身时可以使用以下方法：

{*##$order##}*

则此花括号和变量不会被转义

目录 [隐藏]

[模板数据标签](https://www.we7.cc/manual/template:data" \l "模板数据标签)

[调用微站导航](https://www.we7.cc/manual/template:data" \l "调用微站导航)

[调用幻灯片](https://www.we7.cc/manual/template:data" \l "调用幻灯片)

[调用文章列表](https://www.we7.cc/manual/template:data" \l "调用文章列表)

[调用分类列表](https://www.we7.cc/manual/template:data" \l "调用分类列表)

[调用快捷菜单](https://www.we7.cc/manual/template:data" \l "调用快捷菜单)

## 模板数据标签

微擎中在app端集成了一些常用数据的调用标签，web端不可调用，数据调用都是通是{data}{/data}标签来实现，我们先讲解一下这个标签的一些公共的属性。  
一个完整的data标签：

{data func="site\_navs" index="" item="" assign="" return="true" limit="10" module="we7\_demo"}*//循环体*{/data}

func 调用标签的功能，也就是需要获取什么数据

item 默认为row，使用：$row。指定循环体中数组值的变量，在循环体中可以过用此变量输出值

index 默认为iteration，使用：$row['iteration']。指定循环体中的索引参数，如果指定过item参数，则{item}['{index}']

assign, return 默认为false，这两个标签通常共同使用，指定是否直接返回值，还是直接循环输出，如果return为true时，则会直接把内容赋值给assign设置的参数中

limit 默认为10，指定获取数据的条数

module 如果模块中有扩展标签 func 则通过此参数传递模块名称

### **调用微站导航**

获取当前微站的导航

{data func="site\_navs" item="row"}

{$row['html']}{/data}

获取当前微站第二个位置的导航

{data func="site\_navs" section="1" item="row"}

第$row['iteration']个：{$row['html']}{/data}

### **调用幻灯片**

{data func="site\_slide\_search" item="row" limit="4"}

<li>Name： {$row['thumb']}</li>{/data}

### **调用文章列表**

{data func="site\_article" cid=$cid return="true" assign="result"}{loop $result['list'] $row}<li>

<div **class**="info">

<h3>{$row['title']}</h3>

<p **class**="text">{$row['description']}</p>

</div>

<div **class**="pic">

<img src="{php echo tomedia($row['thumb'])}">

</div></li>{/loop}

### **调用分类列表**

<div **class**="category">

{data func="site\_category" parentid="$cid"}

<a href="{$row['linkurl']}">{$row['name']}</a>

{/data}</div>

### **调用快捷菜单**

{data func="site\_quickmenu"}{/data}

您的足迹: • [bootstrap](https://www.we7.cc/manual/structure:bootstrap" \o "structure:bootstrap) • [ou](https://www.we7.cc/manual/structure:outline" \o "structure:outline)

## 模板中的一些常用变量

模板中提供了一些占位的变量，开发时根据需要可以在代码中定义这些变量，下面将说明有哪些变量

页面标题

$\_W['page']['title'] = '定义此变量将会改变页面标题栏显示的内容';

微信分享内容

**<?php** $\_share = [array](http://www.php.net/array)(

'title' => ?,

'link' => ?,

'imgUrl' => ?,

'content' => ?);

目录 [隐藏]

[数据基本操作](https://www.we7.cc/manual/database:curd" \l "数据基本操作)

[tablename()](https://www.we7.cc/manual/database:curd" \l "tablename)

[范围条件操作](https://www.we7.cc/manual/database:curd" \l "范围条件操作)

[IN 和 NOT IN](https://www.we7.cc/manual/database:curd" \l "in-和-not-in)

[聚合查询](https://www.we7.cc/manual/database:curd" \l "聚合查询)

[查询](https://www.we7.cc/manual/database:curd" \l "查询)

[pdo\_get](https://www.we7.cc/manual/database:curd" \l "pdo_get)

[pdo\_getcolumn](https://www.we7.cc/manual/database:curd" \l "pdo_getcolumn)

[pdo\_getall](https://www.we7.cc/manual/database:curd" \l "pdo_getall)

[pdo\_getslice](https://www.we7.cc/manual/database:curd" \l "pdo_getslice)

[pdo\_fetch](https://www.we7.cc/manual/database:curd" \l "pdo_fetch)

[pdo\_fetchcolumn](https://www.we7.cc/manual/database:curd" \l "pdo_fetchcolumn)

[pdo\_fetchall](https://www.we7.cc/manual/database:curd" \l "pdo_fetchall)

[变更](https://www.we7.cc/manual/database:curd" \l "变更)

[pdo\_insert](https://www.we7.cc/manual/database:curd" \l "pdo_insert)

[pdo\_update](https://www.we7.cc/manual/database:curd" \l "pdo_update)

[pdo\_delete](https://www.we7.cc/manual/database:curd" \l "pdo_delete)

[运行SQL](https://www.we7.cc/manual/database:curd" \l "运行sql)

[pdo\_query](https://www.we7.cc/manual/database:curd" \l "pdo_query)

[pdo\_run](https://www.we7.cc/manual/database:curd" \l "pdo_run)

[辅助函数](https://www.we7.cc/manual/database:curd" \l "辅助函数)

[pdo\_fieldexists](https://www.we7.cc/manual/database:curd" \l "pdo_fieldexists)

[pdo\_indexexists](https://www.we7.cc/manual/database:curd" \l "pdo_indexexists)

[pdo\_tableexists](https://www.we7.cc/manual/database:curd" \l "pdo_tableexists)

[pdo\_fieldmatch](https://www.we7.cc/manual/database:curd" \l "pdo_fieldmatch)

[pdo\_debug](https://www.we7.cc/manual/database:curd" \l "pdo_debug)

## 数据基本操作

微擎系统数据库操作使用 PDO 兼容方式，以参数绑定的形式进行查询操作。   
系统已对 PDO 兼容性进行检测及封装。下面具体说明一下数据库的几种基本操作方法。

### **tablename()**

为了防止微擎系统的表与其它系统命名冲突，安装微擎时均会指定一个表前缀，在写SQL语句时，需要将表名称附加上表前缀。可以使用 tablename() 函数。

$sql = "SELECT \* FROM ".tablename('users');echo $sql;

*//输出 SELECT \* FROM ims\_users*

## 范围条件操作

在微擎20160601以后的版本中，增加了pdo\_get、pdo\_getall、pdo\_getcolumn、pdo\_getslice、pdo\_insert、pdo\_update、pdo\_delete的范围条件的支持，具体支持的范围操作符如下：

$allow\_operator = [array](http://www.php.net/array)('>', '<', '<>', '!=', '>=', '<=', '+=', '-=', 'LIKE', 'like');

示例：

*//获取acid大于269的公众号*$account = pdo\_get('account', [array](http://www.php.net/array)('acid >' => '269'));

*//增加一次用户的错误登录次数，两次变为2即可*

pdo\_update('users\_failed\_login', [array](http://www.php.net/array)('count +=' => 1), [array](http://www.php.net/array)('username' => 'mizhou'));

## IN 和 NOT IN

当传入的条件数组的值为数组时，系统会自动转成成 IN 语句，如果和 <> 或 != （不等于）一起使用时，会自动转换为 NOT IN。

示例：

pdo\_getall('users', [array](http://www.php.net/array)('uid' => [array](http://www.php.net/array)('1', '2', '3')));*//对应的SQL语句调用*

pdo\_fetchall("SELECT \* FROM `ims\_users` WHERE `uid` IN (:\_\_uid\_0,:\_\_uid\_1,:\_\_uid\_2)", [array](http://www.php.net/array)(':\_\_uid\_0' => 1, ':\_\_uid\_1' => 2, ':\_\_uid\_2' => 3));

pdo\_getall('users', [array](http://www.php.net/array)('uid <>' => [array](http://www.php.net/array)('1', '2', '3')));*//对应的SQL语句调用*

pdo\_fetchall("SELECT \* FROM `ims\_users` WHERE `uid` NOT IN (:\_\_uid\_0,:\_\_uid\_1,:\_\_uid\_2)", [array](http://www.php.net/array)(':\_\_uid\_0' => 1, ':\_\_uid\_1' => 2, ':\_\_uid\_2' => 3));

### **聚合查询**

当获取的字段中是聚合字段时，比如 COUNT(\*), SUM()等，如果指定过别名，则可以真通过别名获取值，如果未指定别名，则可以通过字段排序的索引获取。

示例：

$usercount = pdo\_get('users', [array](http://www.php.net/array)(), [array](http://www.php.net/array)('COUNT(\*)', 'uid', 'MAX(uid)', 'MIN(uid) AS minuid'));*//$usercount 值为*

[Array](http://www.php.net/array) (

[0] => 103 *//总数*

[uid] => 1

[2] => 179 *//最大UID*

[minuid] => 1 *//最小UID*)

## 查询

查询是数据库操作中使用最频繁的操作，微擎系统封装了一些函数适用于不同的场景，以下逐个说明

### **pdo\_get**

根据条件（AND连接）到指定的表中获取一条记录

**$tablename** 参数指定要查询的数据表名，此处传入的表名不要使用tablename()函数

**$condition** 参数指定查询的条件，以是 AND 连接，支持大于，小于等范围查询

**$fields** 参数指定查询返回的字段列表

[array](http://www.php.net/array) | boolean pdo\_get($tablename, $condition = [array](http://www.php.net/array)(), $fields = [array](http://www.php.net/array)());

示例:

*//根据uid获取用户的用户名和用户Id信息//生成的SQL等同于：SELECT username, uid FROM ims\_users WHERE uid = '1' LIMIT 1*$user = pdo\_get('users', [array](http://www.php.net/array)('uid' => 1), [array](http://www.php.net/array)('username', 'uid'));

*//生成的SQL等同于：SELECT username FROM ims\_users WHERE username = 'mizhou' AND status = '1' LIMIT 1*$user = pdo\_get('users', [array](http://www.php.net/array)('username' => 'mizhou', 'status' => 1), [array](http://www.php.net/array)('username'));

### **pdo\_getcolumn**

根据条件（AND连接）到指定的表中获取一条记录的指定字段

string | int pdo\_getcolumn($tablename, $condition = [array](http://www.php.net/array)(), $field);

**$tablename** 参数指定要查询的数据表名，此处传入的表名不要使用tablename()函数

**$condition** 参数指定查询的条件，以是 AND 连接，支持大于，小于等范围查询

**$field** 要获取的字段名称

示例

*//根据uid获取用户的用户名//生成的SQL等同于：SELECT username FROM ims\_users WHERE uid = '1' LIMIT 1*$username = pdo\_getcolumn('users', [array](http://www.php.net/array)('uid' => 1), 'username');

### **pdo\_getall**

根据条件（AND连接）到指定的表中获取全部记录

**$keyfield** 参数传入一个已存在的字段名称，结果数组键值就为该字段，否则为自然排序

其它参数同pdo\_get函数

[array](http://www.php.net/array) | boolean pdo\_getall($tablename, $condition = [array](http://www.php.net/array)(), $fields = [array](http://www.php.net/array)(), $keyfield = '') {

示例:

*//获取全部启用的用户//生成的SQL等同于：SELECT \* FROM ims\_users WHERE status = '1'*$user = pdo\_get('users', [array](http://www.php.net/array)('status' => 1));

### **pdo\_getslice**

根据条件（AND连接）到指定的表中获取某个区间的记录，此函数和 pdo\_getall 的区别是可以指定limit 值

**$limit** 参数指定查询语句的LIMIT值，array(start, end) 或是直接传入范围 2,3

**$total** 参数指定查询结果的总条数，方便进行分页操作

[array](http://www.php.net/array) | boolean pdo\_getslice($tablename, $condition = [array](http://www.php.net/array)(), $limit = [array](http://www.php.net/array)(), &$total = null, $fields = [array](http://www.php.net/array)(), $keyfield = '')

示例：

$user = pdo\_getslice('users', [array](http://www.php.net/array)(), [array](http://www.php.net/array)(0, 10), $total);echo $total;

### **pdo\_fetch**

根据SQL语句，查询一条记录

**$sql** 参数指定要返回记录集的SQL语句

**$params** 参数指定为SQL语句中的参数绑定传值，防止SQL注入   
需要注意的是使用参数绑定时，SQL语中等号后不需要使用引号，传入的值必须与绑定的名称一致

[array](http://www.php.net/array) | boolean pdo\_fetch($sql, $params = [array](http://www.php.net/array)());

示例：

*// :uid 是参数的一个占位符，没有使用引号，传入的第二个参数中要与SQL中的占位名称相同*$user = pdo\_fetch("SELECT username, uid FROM ".tablename('users')." WHERE uid = :uid LIMIT 1", [array](http://www.php.net/array)(':uid' => 1));*// LIKE 占位的使用方法*$user = pdo\_fetch("SELECT \* FROM ".tablename('users')." WHERE username LIKE :username", [array](http://www.php.net/array)(':username' => '%mizhou%'));

### **pdo\_fetchcolumn**

根据SQL语句，查询第一条记录的第N列的值，此语句与 pdo\_fetch 使用相同，只是此函数返回的不是一个数组而是一个字符串\

**$column** 参数指定返回记录集的第几列数据

string | boolean pdo\_fetchcolumn($sql, $params = [array](http://www.php.net/array)(), $column = 0)

示例：

*// 获取用户的总数，返回的值是一个数字* $user\_total = pdo\_fetchcolumn("SELECT COUNT(\*) FROM ".tablename('users'));

### **pdo\_fetchall**

根据SQL语句，查询全部记录，使用方法与pdo\_fetch相同

[array](http://www.php.net/array) | boolean pdo\_fetchall($sql, $params = [array](http://www.php.net/array)(), $keyfield = '')

示例：

*// 需要注意的是，返回的数组的键值为用户的uid*$user = pdo\_fetchall("SELECT username, uid FROM ".tablename('users'), [array](http://www.php.net/array)(), 'uid');

## 变更

以下说明插入，更新，删除操作的几个函数。

### **pdo\_insert**

对指定数据表插入一条新记录

**$tablename** 参数指定要插入记录的数据表名，此处传入的表名不要使用tablename()函数

**$data** 参数指定要插入的记录，格式为与数据表字段对应的关联数组

**$replace** 参数指定插入方式使用 INSERT 语句或是 REPLACE 语句(查找到主键相同的数据选择update)

int | boolean pdo\_insert($tablename, $data = [array](http://www.php.net/array)(), $replace = false)

示例：

*//添加一条用户记录，并判断是否成功*$user\_data = [array](http://www.php.net/array)(

'username' => 'mizhou1',

'status' => '1',);$result = pdo\_insert('users', $user\_data);if (![empty](http://www.php.net/empty)($result)) {

$uid = pdo\_insertid();

message('添加用户成功，UID为' . $uid);}

### **pdo\_update**

更新指定的数据表的记录

**$glue** 参数指定前面 **$condition** 数组条件的关联字 AND 或是 OR

[array](http://www.php.net/array) | boolean pdo\_update($tablename, $data = [array](http://www.php.net/array)(), $condition, $glue = 'AND')

示例：

*//更uid等于2的用户的用户名*$user\_data = [array](http://www.php.net/array)(

'username' => 'mizhou2',);$result = pdo\_update('users', $user\_data, [array](http://www.php.net/array)('id' => 2));if (![empty](http://www.php.net/empty)($result)) {

message('更新成功');}

### **pdo\_delete**

删除指定条件的数据

int | boolean pdo\_delete($tablename, $condition = [array](http://www.php.net/array)(), $glue = 'AND')

示例：

*//删除用户名为mizhou2的记录*$result = pdo\_delete('users', [array](http://www.php.net/array)('username' => 'mizhou2'));if (![empty](http://www.php.net/empty)($result)) {

message('删除成功');}

## 运行SQL

当更新，插入，删除无法满足时，可以直接构造SQL语句进行操作

### **pdo\_query**

运行一条SQL语句

**$params** 指定SQL语句中绑定参数的值，参数占位与 **pdo\_fetch** 一致

int | boolean query($sql, $params = [array](http://www.php.net/array)())

示例：

*//更uid等于2的用户的用户名*$result = pdo\_query("UPDATE ".tablename('users')." SET username = :username, age = :age WHERE uid = :uid", [array](http://www.php.net/array)(':username' => 'mizhou2', ':age' => 18, ':uid' => 2));

*//删除用户名为mizhou2的记录*$result = pdo\_query("DELETE FROM ".tablename('users')." WHERE uid = :uid", [array](http://www.php.net/array)(':uid' => 2));if (![empty](http://www.php.net/empty)($result)) {

message('删除成功');}

### **pdo\_run**

批量执行SQL语句

**$stuff** 函数将会将此参数指定的值，替换为当前系统的表前缀。

**注：与pdo\_query不同的是，pdo\_run是可以一次执行多条SQL语句，每条SQL必须以;分隔。**

boolean run($sql, $stuff = 'ims\_')

示例：

$sql = <<<EOF

CREATE TABLE IF NOT EXISTS `ims\_multisearch` (

`id` int(10) unsigned NOT NULL AUTO\_INCREMENT,

`weid` int(10) unsigned NOT NULL,

PRIMARY KEY (`id`)

) ENGINE=MyISAM DEFAULT CHARSET=utf8;

CREATE TABLE IF NOT EXISTS `ims\_multisearch\_fields` (

`id` int(10) unsigned NOT NULL AUTO\_INCREMENT,

`reid` int(10) unsigned NOT NULL,

`type` tinyint(1) unsigned NOT NULL DEFAULT '1',

`title` varchar(255) NOT NULL,

PRIMARY KEY (`id`),

KEY `idx\_reid` (`reid`)

) ENGINE=MyISAM DEFAULT CHARSET=utf8;

EOF;

pdo\_run($sql);

## 辅助函数

### **pdo\_fieldexists**

检查表中是否存在某个字段

**$tablename** 参数指定要检查的表名称

**$fieldname** 参数指定要检查是否存在的字段名

boolean pdo\_fieldexists($tablename, $fieldname)

示例：

*//如果shopping\_goods表中不存在credit字段，则新增credit字段*if(!pdo\_fieldexists('shopping\_goods', 'credit')) {

pdo\_query("ALTER TABLE ".tablename('shopping\_goods')." ADD `credit` int(11) NOT NULL DEFAULT '0';");}

### **pdo\_indexexists**

检查表中是否存在某个索引

**$tablename** 参数指定要检查的表名称

**$indexname** 参数指定要检查是否存在的索引名

boolean pdo\_indexexists($tablename, $indexname)

示例：

*//如果site\_slide表中不存在multiid索引，则新增multiid索引*if (!pdo\_indexexists('site\_slide', 'multiid')) {

pdo\_query("ALTER TABLE ".tablename('site\_slide')." ADD INDEX `multiid` (`multiid`);");}

### **pdo\_tableexists**

检查数据库中是否存在某个表

boolean pdo\_tableexists($tablename)

### **pdo\_fieldmatch**

检查某个表中的字段类型是不是符合

boolean pdo\_fieldmatch($tablename, $fieldname, $datatype = '', $length = '')

$tablename 要检查的表名

$fieldname 要检测的字段名

$datatype 检测该字段是不是与此值相符

$length 检测该字段长度是不是与此值相符

示例

pdo\_fieldmatch('users', 'id', 'varchar'); *// 类型不符合，返回 false*

pdo\_fieldmatch('users', 'id', 'int'); *// 类型符合，返回 true*

pdo\_fieldmatch('users', 'id', 'int', 5); *// 长度不符合，返回 -1*

### **pdo\_debug**

调试运行SQL语句，显示执行过的SQL的栈情况

**$output** 参数指定是否直接打印出调试信息

[array](http://www.php.net/array) debug($output = true, $append = [array](http://www.php.net/array)())

示例：

pdo\_debug();

*//调用该函数结果如下*

[Array](http://www.php.net/array)([0] => [Array](http://www.php.net/array)

(

[sql] => SET NAMES 'utf8';

[error] => [Array](http://www.php.net/array)

(

[0] => 00000

[1] =>

[2] =>

)

)[1] => [Array](http://www.php.net/array)

(

[sql] => SELECT `value` FROM `ims\_core\_cache` WHERE `[key](http://www.php.net/key)`=:[key](http://www.php.net/key)

[params] => [Array](http://www.php.net/array)

(

[:[key](http://www.php.net/key)] => setting

)

[error] => [Array](http://www.php.net/array)

(

[0] => 00000

[1] =>

[2] =>

)

))

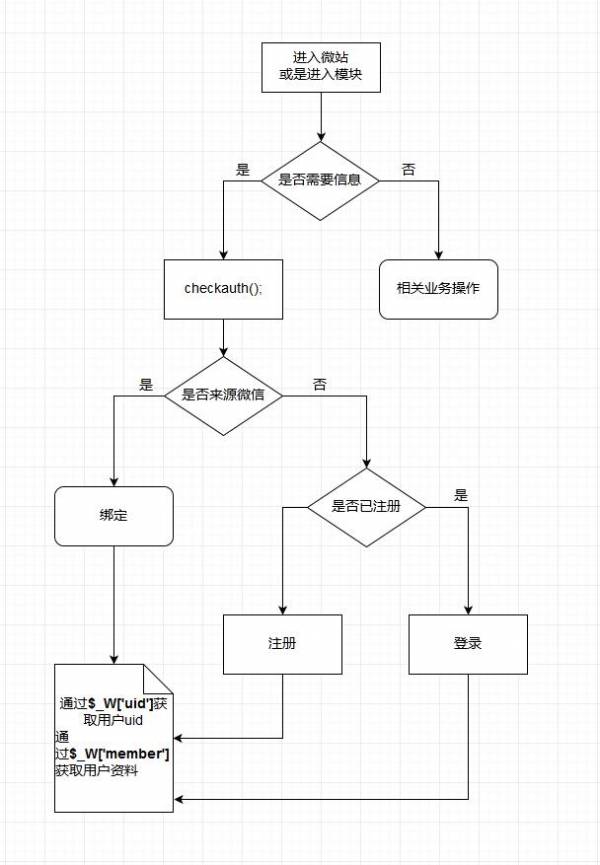
## 统一用户中心说明

为了更好的管理公众号的用户，微擎系统引入了统一用户机制。引入统一用户中心（下称 **用户中心**）有以下好处：

方便统一管理用户的基本资料和积分信息

方便从不同的渠道平台（下称**渠道平台**。指：微信，易信，来往或者自有的CRM等不同的用户系统）同步和共享用户资源

操作流程如下：

[](https://www.we7.cc/manual/_detail/mc:mc_1.jpg?id=mc:outline)

## Openid与Uid的关系

这里的Openid可以泛指从微信、易信或是其它平台进入的粉丝，当这些这些粉丝进入模块或是微站时，在系统中是不产生任何资料数据。   
仅当这些粉丝完成绑定、注册操作时在系统中产生一个uid与之对应，才可以存储一些额外资料(包括真实姓名，性别等等)和积分数据，系统也允许多个Opendid对应同一个Uid。

## checkauth()

当模块或是系统需要获取会员资料或是积分时，可以使用 **checkauth()** 函数来判断会员是否进行登录，以便可以使用其资料和积分数据。例如：

*//需要使用会员积分数据//此函数会判断当前会员是否进登录，如果没有登录跳转至登录页面*

checkauth();

[print\_r](http://www.php.net/print_r)($\_W['member']);

## 会员的积分资料

微擎系统内置5类积分数据，其实积分，余额，贡献为系统内置的，系统很多功能都依赖此数据。  
可以通过 功能选项 – 积分设置 来管理积分名称和是否启用

| **变量名称** | **说明** |
| --- | --- |
| credit1 | 积分 |
| credit2 | 余额 |
| credit3 | 贡献 |
| credit4 | 缺省 |
| credit5 | 缺省 |

## 会员基本资料

微擎系统内置了会员常用的资料信息来满足各功能的需求。  
可以通过 功能选项 – 会员字段管理 来管理资料名称和是否启用

| **变量名称** | **说明** |
| --- | --- |
| realname | 真实姓名 |
| nickname | 昵称 |
| avatar | 头像 |
| qq | QQ号 |
| mobile | 手机号码 |
| vip | VIP级别 |
| gender | 性别 |
| birthyear | 出生生日 |
| constellation | 星座 |
| zodiac | 生肖 |
| telephone | 固定电话 |
| idcard | 证件号码 |
| studentid | 学号 |
| grade | 班级 |
| address | 邮寄地址 |
| zipcode | 邮编 |
| nationality | 国家 |
| resideprovince | 居住地址 |
| graduateschool | 毕业学校 |
| company | 公司 |
| education | 学历 |
| occupation | 职业 |
| position | 职位 |
| revenue | 年收入 |
| affectivestatus | 情感状态 |
| lookingfor | 交友目的 |
| bloodtype | 血型 |
| height | 身高 |
| weight | 体重 |
| alipay | 支付宝帐号 |
| msn | MSN |
| email | 电子邮箱 |
| taobao | 阿里旺旺 |
| site | 主页 |
| bio | 个人介绍 |
| interest | 兴趣爱好 |
| workerid | 工号 |

目录 [隐藏]

[会员积分操作](https://www.we7.cc/manual/mc:credit" \l "会员积分操作)

[获取会员积分数据](https://www.we7.cc/manual/mc:credit" \l "获取会员积分数据)

[mc\_credit\_update()](https://www.we7.cc/manual/mc:credit" \l "mc_credit_update)

## 会员积分操作

系统提供一系列的方法供开发者变更会员的积分数据，大家在开发的时候应该避免直接数据操作积分数据。保持各功能之间的数据互通。下面具体说一下操作会员积分的方法

## 获取会员积分数据

获取积分数据时，要衔使用 **checkauth()** 函数来检查会员是否登录，以便来获取其积分数据。

会员登录成功后，获取会员积分数据通过系统 **$\_W['member']** 全局变量可以获取到，具体的数据结构如下：

[Array](http://www.php.net/array)(

[uid] => 10128

[realname] => 用户昵称

[mobile] => 手机号码

[email] => 邮箱

[groupid] => 用户组ID

[groupname] => 用户组名称

[credit1] => 积分

[credit2] => 余额

[credit3] => 其它积分

[credit4] => 其它积分

[credit5] => 其它积分

[credit6] => 其它积分)

## mc\_credit\_update()

更新粉丝的积分数据

**$uid** 要变更数据会员的uid

**$credittype** 要变更积分数据的类型，分为credit1, credit2, credit3, credit4, credit5, credit6

**$creditval** 要增减积分的值，正数为增，负数为减

**$log** 积分操作日志，具体含义如下

$log = [array](http://www.php.net/array)(

0 => 操作管理员uid

1 => 增减积分备注

2 => 模块标识，例如：we7\_store

3 => 店员uid

4 => 门店id

5 => 1(线上操作) 2(系统后台,公众号管理员和操作员) 3(店员));

boolean mc\_credit\_update($uid, $credittype, $creditval, $log = [array](http://www.php.net/array)());

示例：

load()->model('mc');$uid = mc\_openid2uid($fromuser);

mc\_credit\_update($uid, 'credit1', '10', [array](http://www.php.net/array)(0, '使用砸蛋模块未中奖,赠送10积分'));

load()->model('mc');*//代金券和折扣券的兑换记录,*

mc\_credit\_update($\_W['member']['uid'], 'credit1', -20, [array](http://www.php.net/array)($\_W['member']['uid'], '礼品兑换: Iphone手机 消耗积分：20'));

目录 [隐藏]

[App 端默认加载函数](https://www.we7.cc/manual/reference:app" \l "app-端默认加载函数)

[checkauth()](https://www.we7.cc/manual/reference:app" \l "checkauth)

[url()](https://www.we7.cc/manual/reference:app" \l "url)

[message()](https://www.we7.cc/manual/reference:app" \l "message)

[template()](https://www.we7.cc/manual/reference:app" \l "template)

[register\_jssdk()](https://www.we7.cc/manual/reference:app" \l "register_jssdk)

# App 端默认加载函数

开发者在 App 端开发中可直接使用.

## checkauth()

微站端或微信会话时粉丝用户身份验证.

**function** checkauth()

返回值

|  |  |
| --- | --- |
| **boolean** | 粉丝用户已注册，返回 true. |
| 或 | |
| **void** | 微站端通过普通链接访问的未注册用户，跳转到【用户登录】界面 |
| 或 | |
| **void** | 微信会话时,在processor 中执行, 未注册用户跳转到【注册】界面 |
| 或 | |
| **string (json)** | *// 通过 ajax 访问的未注册用户，返回json编码的注册地址信息*$result = [array](http://www.php.net/array)();$result['url'] = url('auth/login', [array](http://www.php.net/array)('forward' => [base64\_encode](http://www.php.net/base64_encode)($\_SERVER['QUERY\_STRING'])), true);$result['act'] = 'redirect';return [json\_encode](http://www.php.net/json_encode)($result); |

[返回顶部](https://www.we7.cc/manual/reference:app" \l "head" \o "reference:app ↵)

## url()

**function** url($segment, $params = [array](http://www.php.net/array)(), $noredirect = false)

参数

|  |  |  |
| --- | --- | --- |
| **$segment** | string | 路由信息字符串，eg：‘$controller/$action[/$do]' |
| **$params** | array | queryString 参数 |
| **$noredirect** | bool | false: 缺省值. 微信默认要求 mobile 端地址自动附加 '&wxref=mp.weixin.qq.com#wechat\_redirect' 信息 |

返回值

|  |  |
| --- | --- |
| **string (Url.)** | 形如: “./index.php?i={$uniacid}&c=\*&a=\*&do=\*&…“ 或 ”./index.php?i=181&c=ctrl&a=act&do=do&id=1&size=10&wxref=mp.weixin.qq.com#wechat\_redirect” |

[返回顶部](https://www.we7.cc/manual/reference:app" \l "head" \o "reference:app ↵)

## message()

页面执行中断并跳转到消息提示页面。

**function** message($msg, $redirect = '', $type = '')

参数

|  |  |  |
| --- | --- | --- |
| **$msg** | string | 提示的消息内容 |
| **$redirect** | string | 跳转地址，可缺省 |
| **$type** | string | 提示类型，取值不同，消息页面呈现不同效果。. |

说明

| **参数** | **取值** | **说明** |
| --- | --- | --- |
| **$redirect** | **''** | 缺省，不进行跳转。 |
| **'refresh'** | 显示 message 页面后，重新跳转到当前执行的页面 |
| **'referer'** | 显示 message 页面后，回退到当前执行页面的上一页面 |
| **url** | 跳转到指定页面 |
|  |  |  |
| **$type** | **success** | 成功 |
| **error** | 错误 |
| **info** | 提示(灯泡) |
| **warning** | 警告(叹号) |
| **sql** |  |
| **ajax** | 不跳转到 message 页面，但是页面输出 json |

取值 'ajax' 时，eg:

*// 不跳转到 message 页面，但是页面输出 json\_encode($var)* $var = [array](http://www.php.net/array)(

'message' => $msg;

'redirect' => $redirect;

'type' => $type;); return [json\_encode](http://www.php.net/json_encode)($var);

[返回顶部](https://www.we7.cc/manual/reference:app" \l "head" \o "reference:app ↵)

## template()

调用模板

**function** template($filename)

参数

|  |  |  |
| --- | --- | --- |
| **$filename** | string | themes 中对应的文件 |

返回值

调用PHP模板文件

## register\_jssdk()

可以在 app 端使用, 在不使用系统 **{template 'common/header'}** 及 **{template 'common/footer'}** 的页面中, 在头部先于任何 js 文件引入, 调用此方法, 即可注册 wx 对象, 供开发者使用 jssdk.

内部实现了 借用 js 分享 (如果设置了借用).

参见 [纯手工设置分享操作](https://www.we7.cc/manual/dev:js:share" \l "纯手工设置分享操作" \o "dev:js:share)

**function** register\_jssdk($debug = false)

参数

|  |  |  |
| --- | --- | --- |
| **$debug** | boolean | 是否启用调试 |

返回值

html 代码

目录 [隐藏]

[Web 端默认加载函数](https://www.we7.cc/manual/reference:web" \l "web-端默认加载函数)

[url()](https://www.we7.cc/manual/reference:web" \l "url)

[message()](https://www.we7.cc/manual/reference:web" \l "message)

[checklogin()](https://www.we7.cc/manual/reference:web" \l "checklogin)

[checkaccount()](https://www.we7.cc/manual/reference:web" \l "checkaccount)

[template()](https://www.we7.cc/manual/reference:web" \l "template)

# Web 端默认加载函数

开发者在管理员后台开发中可直接使用.

## url()

**function** url($segment, $params = [array](http://www.php.net/array)())

参数

|  |  |  |
| --- | --- | --- |
| **$segment** | string | 路由信息字符串，eg：“{$controller}/{$action}[/{$do}]” |
| **$params** | array | queryString 参数 |

返回值

|  |  |
| --- | --- |
| **string** | url地址, 形如: './index.php?c=\*&a=\*&do=\*&…' |

## message()

页面执行中断并跳转到消息提示页面。

**function** message($msg, $redirect = '', $type = '')

参数

|  |  |  |
| --- | --- | --- |
| **$msg** | string | 提示的消息内容 |
| **$redirect** | string | 跳转地址，可缺省 |
| **$type** | string | 提示类型，取值不同，消息页面呈现不同效果。. |

说明

| **参数** | **取值** | **说明** |
| --- | --- | --- |
| **$redirect** | **''** | 缺省，不进行跳转。 |
| **'refresh'** | 显示 message 页面后，重新跳转到当前执行的页面 |
| **'referer'** | 显示 message 页面后，回退到当前执行页面的上一页面 |
| **url** | 跳转到指定页面 |
|  |  |  |
| **$type** | **success** | 成功 |
| **error** | 错误 |
| **info** | 提示(灯泡) |
| **warning** | 警告(叹号) |
| **sql** |  |
| **ajax** | 不跳转到 message 页面，但是页面输出 json |

取值 'ajax' 时，eg:

*// 不跳转到 message 页面，但是页面输出 json\_encode($var)* $var = [array](http://www.php.net/array)(

'message' => $msg;

'redirect' => $redirect;

'type' => $type;); return [json\_encode](http://www.php.net/json_encode)($var);

[返回顶部](https://www.we7.cc/manual/reference:web" \l "head" \o "reference:web ↵)

## checklogin()

验证操作用户是否已登录

**function** checklogin()

返回值

|  |  |
| --- | --- |
| **boolean** | 验证结果为已登录，返回 true. |
| 或 | |
| **void** | 中断文件执行, 提示用户未登录，跳转到【用户登录】界面 |

## checkaccount()

检查操作员是否已经选择一个公众号作为工作区域

**function** checkaccount()

返回值

|  |  |
| --- | --- |
| **void** | 如果已启用任意一个公众号，继续执行。 或 未启用任何公众号，跳转到【公号列表】 |

## template()

展示特定模板内容

mixed **function** template(string $filename, int $flag = TEMPLATE\_DISPLAY)

参数

|  |  |  |
| --- | --- | --- |
| **$filename** | string | 模板名称，格式为: '模板文件夹/模板名称无后缀'，如: 'common/header' |
| **$flag** | int | 模板展示方式 |

返回值

调用PHP模板文件

说明

|  |  |  |
| --- | --- | --- |
| **$flag** | **TEMPLATE\_DISPLAY** | 导入全局变量，渲染并直接展示模板内容(默认值) |
| **TEMPLATE\_FETCH** | 导入全局变量，渲染模板内容，但不展示模板内容，而是将其作为返回值获取。 可用于静态化页面。 |
| **TEMPLATE\_INCLUDEPATH** | 不导入全局变量，也不渲染模板内容，只是将编译后的模板文件路径返回，返回的模板编译路径可以直接使用 include 嵌入至当前上下文。 |

示例

示例: 以下三种调用方式效果相同

*// 直接展示模板*

template('common/template');

*// 获取模板渲染出的内容*$content = template('common/template', TEMPLATE\_FETCH);*// 输出渲染的内容*echo $content;

*// 嵌入模板编译路径*include template('common/template', TEMPLATE\_INCLUDEPATH);

目录 [隐藏]

[istripslashes](https://www.we7.cc/manual/reference:global" \l "istripslashes)

[ihtmlspecialchars](https://www.we7.cc/manual/reference:global" \l "ihtmlspecialchars)

[isetcookie](https://www.we7.cc/manual/reference:global" \l "isetcookie)

[token](https://www.we7.cc/manual/reference:global" \l "token)

[random](https://www.we7.cc/manual/reference:global" \l "random)

[checksubmit](https://www.we7.cc/manual/reference:global" \l "checksubmit)

[checkcaptcha](https://www.we7.cc/manual/reference:global" \l "checkcaptcha)

[tablename](https://www.we7.cc/manual/reference:global" \l "tablename)

[array\_elements](https://www.we7.cc/manual/reference:global" \l "array_elements)

[range\_limit](https://www.we7.cc/manual/reference:global" \l "range_limit)

[ijson\_encode](https://www.we7.cc/manual/reference:global" \l "ijson_encode)

[iserializer](https://www.we7.cc/manual/reference:global" \l "iserializer)

[iunserializer](https://www.we7.cc/manual/reference:global" \l "iunserializer)

[is\_base64](https://www.we7.cc/manual/reference:global" \l "is_base64)

[is\_serialized](https://www.we7.cc/manual/reference:global" \l "is_serialized)

[wurl](https://www.we7.cc/manual/reference:global" \l "wurl)

[murl](https://www.we7.cc/manual/reference:global" \l "murl)

[pagination](https://www.we7.cc/manual/reference:global" \l "pagination)

[tomedia](https://www.we7.cc/manual/reference:global" \l "tomedia)

[referer](https://www.we7.cc/manual/reference:global" \l "referer)

[strexists](https://www.we7.cc/manual/reference:global" \l "strexists)

[cutstr](https://www.we7.cc/manual/reference:global" \l "cutstr)

[istrlen](https://www.we7.cc/manual/reference:global" \l "istrlen)

[emotion](https://www.we7.cc/manual/reference:global" \l "emotion)

[authcode](https://www.we7.cc/manual/reference:global" \l "authcode)

[sizecount](https://www.we7.cc/manual/reference:global" \l "sizecount)

[bytecount](https://www.we7.cc/manual/reference:global" \l "bytecount)

[array2xml](https://www.we7.cc/manual/reference:global" \l "array2xml)

[xml2array](https://www.we7.cc/manual/reference:global" \l "xml2array)

[scriptname](https://www.we7.cc/manual/reference:global" \l "scriptname)

[utf8\_bytes](https://www.we7.cc/manual/reference:global" \l "utf8_bytes)

[isimplexml\_load\_string](https://www.we7.cc/manual/reference:global" \l "isimplexml_load_string)

[aes\_decode](https://www.we7.cc/manual/reference:global" \l "aes_decode)

[aes\_encode](https://www.we7.cc/manual/reference:global" \l "aes_encode)

[ihtml\_entity\_decode](https://www.we7.cc/manual/reference:global" \l "ihtml_entity_decode)

[iarray\_change\_key\_case](https://www.we7.cc/manual/reference:global" \l "iarray_change_key_case)

[parse\_path](https://www.we7.cc/manual/reference:global" \l "parse_path)

[strip\_gpc](https://www.we7.cc/manual/reference:global" \l "strip_gpc)

系统全局公共函数全部位于 **framework/function/global.func.php** 文件内。

注意：系统会默认加载该文件，因此该文件内所有函数均可直接使用。

## istripslashes

去掉字符串或是数组中的转义符 (\)

istripslashes($var)

$var string | array 需要转义的字符串或数组

返回值

去掉字符串或是数组中的转义符 (\)，例如：将字符串或数组中的 \' \“ \ \ 转换为 ' ” \ 并返回。

示例

echo istripslashes('\r\n\r\n'); *// rnrn*[print\_r](http://www.php.net/print_r)(istripslashes([array](http://www.php.net/array)('\t', '\r', '\n')));

[Array](http://www.php.net/array)(

[0] => t

[1] => r

[2] => n)

## ihtmlspecialchars

转义字符串或数组中的的HTML，例如将 “php” 转化为 &quot;PHP&quot;

ihtmlspecialchars ($var)

$var string | array 需要转义的字符串或数组

返回值

将字符串或数组中的HTML代码进行转义并返回。

示例

echo ihtmlspecialchars('<h1>WeiEngine</h1>');

*//输出如下*&lt;h1&gt;WeiEngine&lt;/h1&gt;

[print\_r](http://www.php.net/print_r)(ihtmlspecialchars([array](http://www.php.net/array)('<h1>WeiEngine</h1>', '<h4>WeiXin</h4>')));

*//输出如下*[Array](http://www.php.net/array)(

[0] => &lt;h1&gt;WeiEngine&lt;/h1&gt;

[1] => &lt;h4&gt;WeiXin&lt;/h4&gt;)

## isetcookie

写入cookie值，写入后可以使用 $\_GPC 来获取

isetcookieisetcookie($key, $value, $expire = 0, $httponly = false)

$key 要写入cookie 的名称

$value 要写入cookie 的值

$expire 此cookie 的过期时间，默认为随浏览器

$httponly 是否允许Javascript读取，防止xss攻击，建议登录等信息开启此项

返回值

成功时返回true，否则返回false

示例

isetcookie ('test', 'hello-world', 3600); *// 记录了一个名为 test 值为 hello-world 的 cookie ，有效期为一个小时。*[print\_r](http://www.php.net/print_r)($\_GPC['test']);

## token

与 checksubmit() 函数配置使用，提交表单时的来源校验码

token()

返回值

提交表单时的来源校验码

示例

<form name="form">

<input name="token" type="hidden" value="{$\_W['token']}" /> *// 与 token() 一样*

<input type="submit" **class**="btn btn-primary span3" name="submit" value="提交" /></form>

**最佳实践：** 系统在入口文件中已经将 **token()** 的结果赋值给全局变量 **$\_W['token']** ,在需要获取Token的情况下，直接使用 **$\_W['token']** 常量可以增强系统维护性和扩展性。

## random

获取一个随机数

random($length, $numeric = FALSE)

$length 要获取随机数的长度

$numeric 返回纯数字随机数

示例

echo random(6); *// JUmtIp*echo random(10, true); *// 9824464484*

## checksubmit

验证一下表单提交，必须与 token() 函数一起使用

checksubmit($var = 'submit', $allowget = false)

$var 提交表单按钮的名称

$allowget 是否验证一个get请求

示例

**<?php**

 if (checksubmit('submit')) {

[exit](http://www.php.net/exit)('表单提交了');}?>

<form name="form">

<input name="token" type="hidden" value="{$\_W['token']}" /> // 与 token() 一样

<input type="submit" class="btn btn-primary span3" name="submit" value="提交" />

</form>

## checkcaptcha

验证用户提交的验证码是否正确

checkcaptcha($code)

$code 用户填写的验证码

示例

if (!checkcaptcha($\_GPC['code'])) {

message('你输入的验证码不正确, 请重新输入.');}

## tablename

为了防止表名冲突，微擎所有数据表均有前缀，此函数就是返回加表前缀的表名

tablename($tablename)

$tablename 数据表名

示例

*#数据表前缀为 `ims`*echo tablename('account'); *// `ims\_account`*

## array\_elements

array\_elements($keys, $src, $default = FALSE)

从一个数组中取得若干元素

$keys 需要筛选的键名列表

$src 进行筛选的数组

$default 如果原数组未定义某个键，则使用 $default 替换，默认为 false。

示例

$result = array\_elements([array](http://www.php.net/array)('a', 'b', 'c'), [array](http://www.php.net/array)('a' => '1', 'b' => '2', 'd' => '3'), 0);[print\_r](http://www.php.net/print_r)($result);[exit](http://www.php.net/exit);

*//输出值*[Array](http://www.php.net/array) ( [a] => 1 [b] => 2 [c] => 0 )

## range\_limit

range\_limit($num, $downline, $upline, $returnNear = true)

判断给定参数是否位于区间内或将参数转换为区间内的数

$number 待检测的数

$downline 区间的最小值

$upline 区间的最大值

$returnNear 为true时小于区间则返回最小值，大于区间则返回最大值，否则返回本身的值

示例

echo range\_limit(100, 50, 250, false); *// true*echo range\_limit(300, 50, 250); *// 250*

# ijson\_encode

获取json格式的字符串函数

ijson\_encode($value, $options = 0)

$value 需要转化的字符串

$options 兼容低版本Php的 options 选项，此值为 JSON\_UNESCAPED\_UNICODE 时，则不转义中文

示例

echo ijson\_encode('hello weiqing'); *// \"hello weiqing\"*

 echo ijson\_encode([array](http://www.php.net/array)('title' => '微擎图队123', 'desc' => 'weiqing123'), JSON\_UNESCAPED\_UNICODE); *// 输出结果 {\"title\":\"微擎图队123\",\"desc\":\"weiqing123\"}*

需要注意的是，返回值时会转义引号等字符，如果要传入Js或是当成json对象使用时，请使用 stripslashes(ijson\_encode($params, JSON\_UNESCAPED\_UNICODE))

# iserializer

获取序列化后的结果

iserializer($str)

$str 需要序列化的字符串或数组

示例

echo iserializer('hello weiqing'); *// s:14:"hello weiqing";*[var\_dump](http://www.php.net/var_dump)(iserializer([array](http://www.php.net/array)(1, 2, 3))); *// a:3:{i:0;i:1;i:1;i:2;i:2;i:3;;}*

# iunserializer

获取反序列化后的结果

iunserializer($str)

$str 需要反序列化的字符串或数组

示例

[print\_r](http://www.php.net/print_r)(iunserializer('a:3:{i:0;i:1;i:1;i:2;i:2;i:3;}'));[Array](http://www.php.net/array)(

[0] => 1

[1] => 2

[2] => 3)

# is\_base64

判断是否为base64加密字符串

is\_base64($str)

$str 需要检测的数据

示例

[var\_dump](http://www.php.net/var_dump)(is\_base64('hello')); *// false*

# is\_serialized

判断字符串是否序列化

is\_serialized($str, $strict = TRUE)

$str 需要检测的数据

$strict 是否为严格模式

示例

[var\_dump](http://www.php.net/var_dump)(is\_serialized('hello')); *// false*

# wurl

获取web端url地址

wurl($segment, $params = [array](http://www.php.net/array)())

$segment 路由参数

$params 附加参数

示例

echo wurl('home/welcome/mc', [array](http://www.php.net/array)('uid' => '100', 'op' => 'foo')); *// ./index.php?c=home&a=welcome&do=mc&uid=100&op=foo*

# murl

获取Mobile端url地址

murl($segment, $params = [array](http://www.php.net/array)(), $noredirect = true, $addhost = false)

$segment 路由参数

$params 附加参数

$noredirect 为false时，会自动添加微信后缀 &wxref=mp.weixin.qq.com#wechat\_redirect

$addhost 为true时，会添加域名

示例

echo murl('home/welcome/mc', [array](http://www.php.net/array)('uid' => '100', 'op' => 'foo')); *// ./index.php?c=home&a=welcome&do=mc&uid=100&op=foo*

# pagination

获取分页导航HTML

pagination($total, $pageIndex, $pageSize = 15, $url = '', $context = [array](http://www.php.net/array)('before' => 5, 'after' => 4, 'ajaxcallback' => '', 'callbackfuncname' => '')

$total 总记录数

$pageIndex 当前页码

$pageSize 每页显示条数

$url 生成url格式

$context 附加参数

before 当前页码前显示几页

after 当前面码之后显示几页

ajaxcallback 是否为ajax分页，为真时，会调用 callbackfuncname 传递的函数来获取值

callbackfuncname

»示例

echo echo pagination(10, 2, 2);

# tomedia

获取图片附件的URL

tomedia($src, $local\_path = false)

$src 需要进行转换的路径

示例

echo tomedia('images/noavatar\_middle.gif');

# referer

获取引用页的地址

referer($default = '')

$default 默认引用页地址

示例

$referer = referer();

# strexists

判断字符串是否包含字串

strexists($string, $find)

$string 在该字符串中查找

$find 需要查找的字串

示例

echo strexists('asdfghjk', 'f');

# cutstr

截取字符串

cutstr($string, $length, $havedot = false, $charset = '')

$string 对该字符串截取

$length 指定截取的长度

$havedot 超长字符串结尾是否加省略号 …

$charset 指定编码

示例

echo cutstr('hello-微擎', 5, true); *//hello...*

# istrlen

忽略字符编码获取字符串的个数，中文英文皆为1个字符

istrlen($string, $charset = '')

$string 指定字符串

$charset 指定编码

示例

echo istrlen('hello-微擎'); *//8*

# emotion

获取表情字符串的HTML

emotion($message = '', $size = '24px')

$message 表情字符串

$size 表情图片大小

示例

echo emotion("/::)");

# authcode

字符串加密或解密

authcode($string, $operation = 'DECODE', $key = '', $expiry = 0)

$string 需要加密或解密的字符串

$operation 操作类型'DECODE'或'ENCODE'

$key 加密秘钥或解密秘钥

$expiry 过期时间

示例

echo authcode('hello-world', 'ENCODE'); *//b3ae8oQNH91yF4oshY+8PiBuoSfDSrwoWzYrrmpyIG03c6B3FYFlCA*

# sizecount

将字节大小 （Bytes） 转化为 KB, MB，GB的显示格式

sizecount($size)

$size 文件原始大小

示例

echo sizecount('123456789'); *//117.74 MB*

# bytecount

将 KB,MB,GB转化为 字节大小（bytes）,与sizecount互为相反

bytecount($str)

$size 文件大小单位为 KB,MB,GB

示例

echo bytecount('20MB'); *//20971520*

# array2xml

获取数组的xml结构

array2xml($arr, $level = 1)

$arr 需要转换的数组

$level 节点层级

示例

[var\_dump](http://www.php.net/var_dump)(array2xml([array](http://www.php.net/array)('TagName' => 'A'))); *//<xml><TagName><![CDATA[A]]></TagName></xml>*

# xml2array

xml结构转化为数组

xml2array($xml)

$xml xml结构的数据

示例

[var\_dump](http://www.php.net/var_dump)(xml2array('<xml><TagName><![CDATA[A]]></TagName></xml>'));[array](http://www.php.net/array) (size=1)

'TagName' => string 'A' (length=1)

# scriptname

获取当前文件的相对路径

scriptname()

示例

echo scriptname(); *///text/zefeng.php*

# utf8\_bytes

将emoji表情转化为二进制码

utf8\_bytes($cp)

$cp ASCII 码值

示例

echo utf8\_bytes('264D');

# isimplexml\_load\_string

将一个 xml 字符串转化为 xml 对象，重新封装 simplexml\_load\_string 函数，解决安全问题

$string xml字符串

$class\_name 填入 SimpleXMLElement 此类或是继承于此类的类

$options 解析参数，请见 [http://php.net/manual/en/libxml.constants.php](http://php.net/manual/en/libxml.constants.php" \o "http://php.net/manual/en/libxml.constants.php)

isimplexml\_load\_string($string, $class\_name = 'SimpleXMLElement', $options = 0, $ns = '', $is\_prefix = false)

示例

$xmlstring= <<<EOF

<xml>

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

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

<CreateTime>1490162242</CreateTime>

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

<Content><![CDATA[测试内容]]></Content>

<MsgId>1234567890123456</MsgId>

</xml>

EOF;

 $xml = isimplexml\_load\_string($xmlstring, 'SimpleXMLElement', LIBXML\_NOCDATA);[print\_r](http://www.php.net/print_r)($xml);[exit](http://www.php.net/exit);

*//以上输出*

SimpleXMLElement Object(

[ToUserName] => toUser

[FromUserName] => fromUser

[CreateTime] => 1490162242

[MsgType] => text

[Content] => 测试内容

[MsgId] => 1234567890123456)

# aes\_decode

AES解密函数，用于微信数据解密

aes\_decode($message, $encodingaeskey = '', $appid = '')

$message 要解密的数据

$encodingaeskey 解密的token

$appid 公众号appid

# aes\_encode

AES加密函数，用于微信数据加密

aes\_encode($message, $encodingaeskey = '', $appid = '')

$message 要解密的数据

$encodingaeskey 解密的token

$appid 公众号appid

# ihtml\_entity\_decode

转化html字符串中实体数据为标签数据，修复&nbsp;在utf8编码下被转换成黑块的坑   
此函数与 ihtmlspecialchars 对应使用

ihtml\_entity\_decode($str)

$str

示例

echo ihtml\_entity\_decode('&lt;b&gt;title&lt;/b&gt;&lt;div&gt;con&nbsp;tent&quot;&#039;&lt;/div&gt;')

*//输出结果*<b>title</b><div>con&nbsp;tent"&#039;</div>

# iarray\_change\_key\_case

获取数组的键为大写或小写

iarray\_change\_key\_case($array, $case = CASE\_LOWER)

$array 指定数组

$case 大写或小写模式

示例

[var\_dump](http://www.php.net/var_dump)(iarray\_change\_key\_case([array](http://www.php.net/array)('action => ze, 'tai' => 'feng')));

# parse\_path

过滤路径中可能包含的非法字符，建议通过get或是post传递的路径皆用此函数过滤

parse\_path($path)

$path 字符

示例

var\_dump(parse\_path('**<?php**'));

# strip\_gpc

过滤 $\_GPC 全局变量中的非法字符

strip\_gpc($values, $type = 'g')