# Wordpress二次开发

# 一、常用函数

## 直接输出

这些函数要在主循环中输出

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

the\_title() //输出标题

the\_content() //输出内容

the\_excerpt() //输出摘要

the\_author() //作者

the\_time() //输出时间

## get\_post获取文章

**--------------------**

**$postid = get\_the\_ID(); //获取文章ID**

**get\_post($postid)->post\_content; //根据id获取文章内容**

-------------以下通过get\_post()的对象调用-------------------------

post\_date：(字符）文章发表的日期和时间（YYYY-MM-DD HH-MM-SS)

post\_date\_gmt：（字符）文章发表的格林尼治标准时间（GMT） （YYYY-MM-DD HH-MM-SS)

post\_modified：(字符）文章最后修改时间（YYYY-MM-DD HH-MM-SS)

post\_modified\_gmt：(字符）文章最后修改GMT时间（YYYY-MM-DD HH-MM-SS)

guid：（字符）文章的一个链接。

分类

## 获取当前文章的url地址

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

get\_permalink( $id, $leavename ) //获取文章的固定链接

get\_permalink();

## 判断文章是否属于类别

**in\_category**：判断当前文章或指定文章是否属于某个指定类别，只有直属的类别，不包括直属类别的父辈类别；可以在循环内使用，也可以独立使用。

**is\_category**：判断是否正在显示一个类别归档页面。

## 调用作者头像

<?php echo get\_avatar( get\_the\_author\_email(), 36 ); ?>

## query\_posts

<?php query\_posts(); ?> //操作

$args=array(

    'cat' => 1,   // 分类ID

    'posts\_per\_page' => 10, // 显示篇数

);

query\_posts($args);

if(have\_posts()) : while (have\_posts()) : the\_post();

## 通过文章标题获取文章url地址

<a href="<?php echo esc\_url( get\_permalink( get\_page\_by\_title( 'Monthly Events' ) ) ); ?>">

<?php //esc\_html\_e( 'Monthly Events', 'textdomain' ); ?>

</a>

## 获取当前页面的ID

//文章或页面的 ID 值，如果未在循环中输出值可能不准确

$postid = get\_the\_ID();

echo $postid;

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

//检索当前查询对象的

ID$current\_id = get\_queried\_object\_id();

echo $current\_id;

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

// 检索当前查询的对象，从对象中获取 ID

$object = get\_queried\_object();

$id = $object -> ID;

echo $id;

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

// 通过$post 全局变量获取文章或页面 ID

global $post;

$id = $post -> ID;

echo $id;

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

// 第一种获取父级页面的 ID

global $post;

$id = $post -> ID;

$parent = get\_post\_ancestors($post -> ID);

print\_r($parent);//打印出 Array ( [0] => 101 )

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

 // 第二种获取父级页面的 ID

global $post;

$parent\_id = $post -> post\_parent;

echo $parent\_id;//打印出父级页面的 ID

# 常用操作方法

## 访问优先级

front-page.php

page.php

index.php

# do\_action与add\_action

do\_action **用于调用一个钩子函数，会执行挂载在钩子上的所有函数**

add\_action**用于给钩子函数添加挂载方法。**

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

//1、定义$function\_to\_add方法，自定义

function output\_copyright\_info()

{

    echoget\_option('自定义方法');

}

//2、将自定义方法绑定到my\_self\_func

add\_action('my\_self\_func','output\_copyright\_info');

//3、定义my\_self\_funcs函数（可以用其他的名称）

//执行'my\_self\_func'这个动作,

do\_action('my\_self\_func');

（1）add\_action( $tag, $function\_to\_add, $priority, $accepted\_args );

函数解析：该函数从字面上就可以理解到，添加动作，绑定动作，将$function\_to\_add绑定到$tag

参数解析：

$tag : $function\_to\_add所挂载的动作（action）的名称（必需），可以自定义，也可以调用WPPlugin里面

的hook列表，根据需求而定，而大多数插件开发中，一般都是自定义（字符串）

$function\_to\_add : 一个函数（字符串），挂载在这个动作下面，当动作被触发的时候，运行函数

$priority :（数字）由于挂载在一个动作上的函数不一定只有一个，所以这个$priority起到先后运

行的关系，数字越小，$function\_to\_add函数越先运行

$accepted\_args ： （整数）（可选）挂钩函数所接受的参数数量

（2）do\_action( $tag, $arg )

    函数解析：该函数从字面上就可以理解到，执行动作，执行名为$tag方法，并且运行绑定到$tag上的所有方法

## 引入文件

如果diy存在多个子集，例如diy-one.php则函数可写为get\_template\_part('diy','one')。按照这种方式所写的调用函数会判断模板文件夹下是否有diy-one.php 如果没有则调用diy.php。

# 主题修改

## 首页显示关闭工具栏

/wp-content/themes/quality-orange/functions.php

//不现实工具栏

最后一行添加

add\_filter('show\_admin\_bar', '\_\_return\_false');

**用户界面可以自定义**

# 主题失败，权限不够，不能创建文件夹

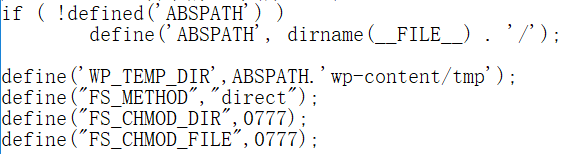
安装使用SFTP连接到服务器，进入wp-content目录，新建tmp文件夹，设置文件夹的权限为777。返回网站根目录，编辑wp-config.php

添加下列代码(注意中英文符号)

define('WP\_TEMP\_DIR',ABSPATH.'wp-content/tmp');  
define("FS\_METHOD","direct");  
define("FS\_CHMOD\_DIR",0777);  
define("FS\_CHMOD\_FILE",0777);

注意：要在定义ABSPATH的后面添加

最后的代码应该是



保存后设置wp-content目录中的plugins和themes文件夹权限为777，此时插件和主题已经可以直接更新了。注意操作顺序，先建立tmp，后设置权限为777，再修改wp-config.php配置文件

# 页脚页头

**使用插件[Header Footer Code Manager](https://www.qcwlseo.com/tag/header-footer-code-manager/" \o "查看与 Header Footer Code Manager 相关的文章" \t "https://www.qcwlseo.com/_blank)**

<style>

#footer {

height: 40px;

line-height: 40px;

position: fixed;

bottom: 0;

width: 100%;

text-align: center;

background: #333;

color: #fff;

font-family: Arial;

font-size: 12px;

letter-spacing: 1px;

}

.content {

height: 1800px;

width: 100%;

text-align: center;

}

</style>

<div id="footer">CopyRight@copy2012 </div>