

## 29 个实用的 WordPress 主题函数使用技巧

WordPress 主题一般有一系列的 php 文件和一个 style.css 文件，而其中功能最为强大的文件则是 functions.php。WordPress 有非常多的常用函数，你可以通过添加和删除一些函数来增加 WordPress 主题的功能，而不需要修改任何的主题文件。

本文的目标读者是 WordPress 主题开发者，需要懂一些基本的 PHP 知识。另，下文提到的所有代码都必须添加到 functions.php 文件里面。

### 1，添加 Google Analytics 统计

只需要把下面的代码添加到 functions.php 文件里面——注意把里面的中文部分替换成你的 Google 统计代码，然后你就不用担心了。

```
<?php
    add_action('wp_footer', 'add_googleanalytics');
    function add_googleanalytics() { ?>
    // 把 Google 统计代码复制到这里
    <?php } ?>
```

### 2，给 WordPress 博客添加一个 Favicon 图标。

每一个博客都应该有一个独一无二的标志，你可以通过添加代码到 header.php 来实现。当然，你也可以通过添加代码到 functions.php 来实现。添加完下面的代码后，只需要把 Favicon.ico 文件上传到网站根目录即可。

```
// add a favicon to your
function blog_favicon() {
    echo '<link rel="Shortcut Icon" type="image/x-icon"
href="'.get_bloginfo('wpurl').'/favicon.ico' />';
}
add_action('wp_head', 'blog_favicon');
```

### 3，移除 WordPress 版本号。

WordPress 有新版本出来后，总会在后台提示管理员进行升级。但假如你是给客户制作网站，而他们又不想升级的话，最好的办法就是从 WordPress 头部、RSS 里面以及其他任何地方移除版本的信息。

```
function wpbeginner_remove_version() {
```

```

return "";
}
add_filter('the_generator', 'wpbeginner_remove_version');

```

#### 4, 给 WordPress 控制面板添加自定义 logo

用 WordPress 给客户制作网站，如果给 WordPress 的控制面板后台添加一个自定义 logo，则会让网站显的专业很多。要做到这一点，你只需要把代码添加到 `functions.php` 即可。

```

//hook the administrative header output
add_action('admin_head', 'my_custom_logo');

function my_custom_logo() {
echo '
<style type="text/css">
#header-logo {
background-image:
url(' . get_bloginfo('template_directory') . '/images/custom-logo.gif) !important; }
</style>
';
}

```

#### 5, 改变 WordPress 后台控制面板底部信息

如上所述，如果不想让客户知道网站是由 WordPress 制作，则可以修改 WordPress 控制面板底部的信息，只需要把下面的代码添加到 `functions.php` 文件即可。

```

function remove_footer_admin () {
echo 'Fueled by <a href="http://www.wordpress.org" target="_blank">WordPress</a> |
Designed by <a href="http://www.uzzz.net" target="_blank">Uzzz Productions</a> | WordPress
Tutorials: <a href="http://www.wpbeginner.com" target="_blank">WPBeginner</a></p>';
}

add_filter('admin_footer_text', 'remove_footer_admin');

```

注：代码里面的 `html` 部分可以修改。

#### 6, 自定义 WordPress 控制面板模块

一些 WordPress 插件会在控制面板那里添加一些模块来显示相应的信息，作为一个 WordPress 模板设计者，你也可以通过修改 `functions.php` 文件来实现这个功能。注意替换里面的相应信息。

```
add_action('wp_dashboard_setup', 'my_custom_dashboard_widgets');

function my_custom_dashboard_widgets() {
    global $wp_meta_boxes;

    wp_add_dashboard_widget('custom_help_widget', 'Theme Support',
'custom_dashboard_help');
}

function custom_dashboard_help() {
    echo '<p>Welcome to Custom Blog Theme! Need help? Contact the developer <a
href="mailto:yourusername@gmail.com">here</a>. For WordPress Tutorials visit: <a
href="http://www.wpbeginner.com" target="_blank">WPBeginner</a></p>';
}
```

## 7, 改变默认的 Gravatar 头像

WordPress 程序默认的 Gravatar 头像很不咋地，而且到处都是千篇一律的默认头像一点都无法体现独特性。你可以把以下代码添加到 `functions.php` 文件里面，然后记得把自定义的 Gravatar 头像上传到 WordPress 模板的 `images` 文件夹。

```
add_filter( 'avatar_defaults', 'newgravavatar' );

function newgravavatar ($avatar_defaults) {
    $myavatar = get_bloginfo('template_directory') . '/images/gravatar.gif';
    $avatar_defaults[$myavatar] = "WPBeginner";
    return $avatar_defaults;
}
```

## 8, 让 WordPress 底部的版权时间显示的更生动

很多网站的版权时间都显示的是建站时的年份，有些则是显示当下的年份。事实上，这两种方式都不是太好。

最好的方式是显示从网站建设之初的年份到目前的年份位置，类似© 2006 - 2010 这种显示方式。

这种效果通过以下的代码可以实现。添加完下面的代码后，系统会自动抓取发布第一篇文章的年份以及最新一篇文章的年份，并把它显示出来。

```
function comicpress_copyright() {
    global $wpdb;
```

```

$copyright_dates = $wpdb->get_results("
SELECT
YEAR(min(post_date_gmt)) AS firstdate,
YEAR(max(post_date_gmt)) AS lastdate
FROM
$wpdb->posts
WHERE
post_status = 'publish'
");
$output = "";
if($copyright_dates) {
$copyright = "&copy; " . $copyright_dates[0]->firstdate;
if($copyright_dates[0]->firstdate != $copyright_dates[0]->lastdate) {
$copyright .= '-' . $copyright_dates[0]->lastdate;
}
$output = $copyright;
}
return $output;
}

```

把上面的代码添加到了 `functions.php` 文件里面后，还需要在 `footer.php` 任何你想显示版权时间的地方加上如下代码：

```
<?php echo comicpress_copyright(); ?>
```

## 9，给读者投稿文章添加姓名/来源

如果你的博客接受读者的投稿，想在该篇文章出现投稿者的姓名，同时又不想通过添加作者的这种繁琐而麻烦的方式来操作，则可以使用下面的代码。使用下面的代码后，只需要在撰写文章的时候在自定义区域填上投稿者的姓名即可。系统会自动将发布者的名称换成投稿者的名称。

这个代码对接受读者投稿较多的网站，或者是资讯型的网站非常有用（利用它来显示来源）。

```

add_filter( 'the_author', 'guest_author_name' );
add_filter( 'get_the_author_display_name', 'guest_author_name' );

function guest_author_name( $name ) {
global $post;

$author = get_post_meta( $post->ID, 'guest-author', true );

```

```

if ( $author )
$name = $author;

return $name;
}

```

## 10，启用文章缩略图功能

从 WordPress2.9 版本开始，可以给模板添加文章缩略图功能。操作方法很简单，只需要把下面的代码添加到 functions.php 里面。

```
add_theme_support( 'post-thumbnails' );
```

然后在要显示缩略图的地方放置下面的代码即可。

```
<?php the_post_thumbnail(); ?>
```

## 11，自定义 WordPress 3.0 版本导航栏

WordPress 3.0 增加了一个功能，可以让 WordPress 模板开发者自定义导航菜单。如果你想给用户一个导航栏的选择权，只需要把下面的代码加入到 functions.php 文件里面。

```
add_theme_support( 'nav-menus' );
```

之后把下面的代码复制到你想出新的地方：

```
<?php wp_nav_menu( array( 'sort_column' => 'menu_order', 'container_class' =>
'menu-header' ) ); ?>
```

## 12，移除 WordPress 默认的个人资料选项

如果你客户的想让用户可以自行添加个人资料，那么需要让这个选项更简单。其中一个方法就是移除部分选项，AIM， Yahoo IM 和 Jabber 之类的东东。

```
add_filter('user_contactmethods','hide_profile_fields',10,1);
```

```

function hide_profile_fields( $contactmethods ) {
unset($contactmethods['aim']);
unset($contactmethods['jabber']);
}

```

```
unset($contactmethods['yim']);
return $contactmethods;
}
```

### 13, 添加作者个人资料选项

如果你想更充分的展示作者的个人资料，那么你可以添加一些更个性化的资料选项，例如添加 **twitter** 和 **facebook** 账号等。下面的代码就是添加 **twitter** 和 **facebook** 账号用的。当然，你可以把里面的内容替换成其他任何你想展示的资料。这个对多博客作者尤其有用。

```
function my_new_contactmethods( $contactmethods ) {
    // Add Twitter
    $contactmethods['twitter'] = 'Twitter';
    //add Facebook
    $contactmethods['facebook'] = 'Facebook';

    return $contactmethods;
}
add_filter('user_contactmethods','my_new_contactmethods',10,1);
```

添加完是上面的代码后，你需要在 **author.php** 文件里面添加如下的代码：

```
<?php echo $curauth->twitter; ?>
```

注意：改代码仅在 **WordPress2.9** 以上的版本起作用。

### 14, 添加侧边栏小模块

这是目前用的最多的技巧之一，很多 **WordPress** 模板开发者都已经知道，并且在用了。

```
if ( function_exists('register_sidebar') )
    register_sidebar(array('name'=>'MiddleSidebar',
        'before_widget' => '<li class="widget">',
        'after_widget' => '</li>',
        'before_title' => '<h2 class="widgettitle">',
        'after_title' => '</h3>',
    ));
    register_sidebar(array('name'=>'FooterSidebar',
        'before_widget' => '<li class="widget">',
        'after_widget' => '</li>',
        'before_title' => '<h2 class="widgettitle">',
```

```
'after_title' => '</h3>',  
));
```

上面的代码可以增加两个侧边栏的小模块。以此类推，你可以添加无限多侧边栏的小模块。添加完上面的代码后，你需要把下面的代码添加到你要出现这边小模块的地方。

```
<?php if ( !function_exists('dynamic_sidebar') || !dynamic_sidebar('MiddleSidebar') ) : ?>  
    <!--Default sidebar info goes here-->  
<?php endif; ?>
```

注意：侧边栏并不一定需要出现在 sidebar.php 文件里面。

## 15, 优化 WordPress 博客的 RSS

如何在 RSS 里面加入版权链接？如何在 RSS 加入广告？针对国内互联网的现状，在 RSS 里面加入版权尤为重要，广告倒是次要的。

除了插件（Better Feed）以外，可以采用以下的方法来实现。

```
function wpbeginner_postrss($content) {  
    if(is_feed()){  
        $content = 'This post was written by Syed Balkhi ' . $content . 'Check out WPBeginner';  
    }  
    return $content;  
}  
add_filter('the_excerpt_rss', 'wpbeginner_postrss');  
add_filter('the_content', 'wpbeginner_postrss');
```

## 16, 给 RSS 添加缩略图

缩略图一般是在正常的博客页面上用来起到美化界面的作用。当然，如果需要的话，也可以给 RSS 内容增加一个缩略图。要做到这一点，只需要在 functions.php 里面加入如下代码：

```
function rss_post_thumbnail($content) {  
    global $post;  
    if(has_post_thumbnail($post->ID)) {  
        $content = '<p>' . get_the_post_thumbnail($post->ID) .
```

```
'</p>' . get_the_content();  
}  
return $content;  
}  
add_filter('the_excerpt_rss', 'rss_post_thumbnail');  
add_filter('the_content_feed', 'rss_post_thumbnail');
```

## 17，开启 WordPress 评论嵌套功能

评论嵌套功能是 WordPress 自身带有的最好功能之一，只可惜很多 WordPress 模板都不支持。很多文章都有提到过修改的方法，但一般都涉及到修改 comments 文件和 header 文件。事实上，通过修改 functions.php 文件来修改是最简便的，而且一劳永逸。

```
// enable threaded comments  
function enable_threaded_comments(){  
    if (!is_admin()) {  
        if (is_singular() AND comments_open() AND (get_option('thread_comments') == 1))  
            wp_enqueue_script('comment-reply');  
    }  
}  
add_action('get_header', 'enable_threaded_comments');
```

## 18，移除 WordPress 登陆面板的错误提示

当你输入的密码或者用户名错误的时候，WordPress 登陆界面会给出相应的提示。但如果碰到黑客的话，这些提示反而给了他们更好的提示，让他们更容易破解用户名和密码。因此，处于安全性考虑，移除 WordPress 登陆面板的错误提示是非常必要的。

```
add_filter('login_errors', create_function('$a', "return null;"));
```

## 19，关闭 WordPress 的搜索功能

当把 WordPress 当做 CMS 系统来使用的时候，WordPress 自带的搜索功能实用性就不是太强了。一来增加数据库查询次数，二来 Google 自定义搜索会是更好的替代。因此，你只需要通过以下的代码就可以关闭 WordPress 的搜索功能。

```
function fb_filter_query( $query, $error = true ) {
```



```

if ( is_search() ) {
    $query->is_search = false;
    $query->query_vars[s] = false;
    $query->query[s] = false;

    // to error
    if ( $error == true )
        $query->is_404 = true;
    }
}

add_action( 'parse_query', 'fb_filter_query' );
add_filter( 'get_search_form', create_function( '$a', "return null;" ) );

```

## 20，启用 WordPress 简码功能

Google AdSense 算是博客的标配之一了，很多 CMS 经常会在模板选项里面预置 Google AdSense 的广告位。假如你的模板不支持，你可以通过以下的方法来解决：

```

function showads() {
    return '<div id="adsense"><script type="text/javascript"><!--
    google_ad_client = "pub-XXXXXXXXXXXXXXX";
    google_ad_slot = "4668915978";
    google_ad_width = 468;
    google_ad_height = 60;
    //-->
    </script>

    <script type="text/javascript"
    src="http://pagead2.googlesyndication.com/pagead/show_ads.js">
    </script></div>';
}

add_shortcode('adsense', 'showads');

```

## 21，不通过.htaccess 将 rss 地址唯一化

WordPress 本身提供好几个不同版本的 rss 地址，加入你又使用了 FeedBurner 或者 feedsky 的话，RSS 地址就会更多。太多的 RSS 容易分流订阅客户，而且也不利于品牌推广。

一般的修改方法是通过更改.htaccess 来进行，此外，还可以通过以下的代码来实现。

```
function custom_feed_link($output, $feed) {

    $feed_url = 'http://feeds.feedburner.com/wpbeginner';

    $feed_array = array('rss' => $feed_url, 'rss2' => $feed_url, 'atom' => $feed_url, 'rdf' => $feed_url,
    'comments_rss2' => "");
    $feed_array[$feed] = $feed_url;
    $output = $feed_array[$feed];

    return $output;
}

function other_feed_links($link) {

    $link = 'http://feeds.feedburner.com/wpbeginner';
    return $link;

}

//Add our functions to the specific filters
add_filter('feed_link','custom_feed_link', 1, 2);
add_filter('category_feed_link', 'other_feed_links');
add_filter('author_feed_link', 'other_feed_links');
add_filter('tag_feed_link','other_feed_links');
add_filter('search_feed_link','other_feed_links');
```

## 22. 启用 paypal 捐赠简码

当你写完一篇以后，可以在文章里面插入 paypal 捐赠按钮，方便读者捐赠。以下的代码可以让你非常轻松的做到这一点。

```
function donate_shortcode( $atts ) {
    extract(shortcode_atts(array(
        'text' => 'Make a donation',
        'account' => 'REPLACE ME',
        'for' => "",
    ), $atts));

    global $post;
```

```

if (!$for) $for = str_replace(" ", " ", $post->post_title);

return '<a class="donateLink"
href="https://www.paypal.com/cgi-bin/webscr?cmd=_xclick&business='.$account.'&item_name=Donation
for '.$for.'">'.$text.'</a>';

}

add_shortcode('donate', 'donate_shortcode');

```

### 23. 设定文章从发布到出现在 RSS 中的时间长短

通过 RSS 订阅来阅读博文的朋友可能都会有这个体验：经常发现 RSS 中的文字或者细节有错误，而返回到页面的时候却发现错误已经没有了。这种情况最有可能是因为

RSS 最大的好处是快捷、直接，但这个最大的好处有时候对作者来说却会引发某些尴尬。所以，有时候有必要让文章发布后到读者从 RSS 中接到有一个小小的时间差，方便作者排查某些问题。以下的代码可以做到以下几点：

```

function publish_later_on_feed($where) {
    global $wpdb;

    if ( is_feed() ) {
        // timestamp in WP-format
        $now = gmdate('Y-m-d H:i:s');

        // value for wait; + device
        $wait = '10'; // integer

        // http://dev.mysql.com/doc/refman/5.0/en/date-and-time-functions.html#function_timestampdiff
        $device = 'MINUTE'; //MINUTE, HOUR, DAY, WEEK, MONTH, YEAR

        // add SQL-syntax to default $where
        $where .= " AND TIMESTAMPDIFF($device, $wpdb->posts.post_date_gmt, '$now') > $wait ";
    }
    return $where;
}

add_filter('posts_where', 'publish_later_on_feed');

```

这段代码设置的时间是 10 分钟，你可以把 10 改成任何你想要的时间。

## 24, 自定义摘要输出时的符号

一般设定自动摘要输出，你会经常在 WordPress 博客的首页看到“ [。..] ”这样的符号。为了界面的美观，或者是个性化的需要，你可以把这个默认的符号改变为其他的符号。而以下的代码就是为了实现这个而写：

```
// custom excerpt ellipses for 2.9
function custom_excerpt_more($more) {
    return '...';
}
add_filter('excerpt_more', 'custom_excerpt_more');

/* custom excerpt ellipses for 2.8-
function custom_excerpt_more($excerpt) {
    return str_replace('[...] ', '...', $excerpt);
}
add_filter('wp_trim_excerpt', 'custom_excerpt_more');
*/
```

## 25, 自定义摘要输出的文字长度

假如你比较懒，不想在撰写文章的时候每篇文章都输入摘要，就可以让系统自动截取一定长度的文字来作为摘要输出。下面的代码默认是 100 个字节，也就是 50 个汉字。你可以把数值修改成符合你需要的数字。

```
function new_excerpt_length($length) {
    return 100;
}
add_filter('excerpt_length', 'new_excerpt_length');
```

## 26, 显示精确评论数

WordPress 默认是把 trackbacks 和 pings 都算作评论的，因此当你设置不显示 trackbacks 和 ping 的时候，评论数看起来总是不对头。以下的代码则以让 WordPress 只计算评论的数量，而不把 trackbacks 和 pings 也计算进去。

```
add_filter('get_comments_number', 'comment_count', 0);
```

```
function comment_count( $count ) {
    if ( ! is_admin() ) {
        global $id;
        $comments_by_type = &separate_comments(get_comments('status=approve&post_id=' . $id));
        return count($comments_by_type['comment']);
    } else {
        return $count;
    }
}
```

## 27，取消 RSS 输出

对于某些博客而言，或者因为被太多人采集了，或者因为不想让别人通过 RSS 订阅，想取消 RSS 输出。WordPress 默认是没有这个功能的，但你可以通过以下的代码来取消 RSS 输出。

```
function fb_disable_feed() {
    wp_die( __('No feed available, please visit our <a href="'. get_bloginfo('url') .'">homepage</a>!') );
}

add_action('do_feed', 'fb_disable_feed', 1);
add_action('do_feed_rdf', 'fb_disable_feed', 1);
add_action('do_feed_rss', 'fb_disable_feed', 1);
add_action('do_feed_rss2', 'fb_disable_feed', 1);
add_action('do_feed_atom', 'fb_disable_feed', 1);
```

## 28，显示 Twitter 的订阅数以及其他资料

Twitter 系统以及很多第三方的客户端都可以让你在 WordPress 博客的侧边栏暂时 Twitter 的订阅数以及一些其他的资料。这种做法往往很多时候都没办法跟博客已有的界面结合的很好。而以下的代码则可以让你自定义 Twitter 在博客上的显示外观。

```
function rarst_twitter_user( $username, $field, $display = false ) {
    $interval = 3600;
    $cache = get_option('rarst_twitter_user');
    $url = 'http://api.twitter.com/1/users/show.json?screen_name='.urlencode($username);

    if ( false == $cache )
        $cache = array();

    // if first time request add placeholder and force update
```

```

if ( !isset( $cache[$username][$field] ) ) {
    $cache[$username][$field] = NULL;
    $cache[$username]['lastcheck'] = 0;
}

// if outdated
if( $cache[$username]['lastcheck'] < (time()-$interval) ) {

    // holds decoded JSON data in memory
    static $memorycache;

    if ( isset($memorycache[$username]) ) {
        $data = $memorycache[$username];
    }
    else {
        $result = wp_remote_retrieve_body(wp_remote_request($url));
        $data = json_decode( $result );
        if ( is_object($data) )
            $memorycache[$username] = $data;
    }

    if ( is_object($data) ) {
        // update all fields, known to be requested
        foreach ( $cache[$username] as $key => $value )
            if( isset($data->$key) )
                $cache[$username][$key] = $data->$key;

        $cache[$username]['lastcheck'] = time();
    }
    else {
        $cache[$username]['lastcheck'] = time()+60;
    }

    update_option( 'rarst_twitter_user', $cache );
}

if ( false != $display )
    echo $cache[$username][$field];
return $cache[$username][$field];
}

```

把上面的代码复制到 `functions.php` 后，再把下面代码复制到你想出现的地方即可。

Then place the following code where you want to display the count in your theme file:

```
echo rarst_twitter_user('wpbeginner', 'name'). ' has ' .  
    rarst_twitter_user('wpbeginner', 'followers_count'). ' followers after ' .  
    rarst_twitter_user('wpbeginner', 'statuses_count'). ' updates.';
```

## 29. WP 全自动外链插件

在这里，向大家隆重推荐WP全自动外链插件：[Automatic Backlink Creator](#)插件。

这款软件本人使用过，效果非常不错，所以今天在这里推荐一下，希望可以节省大家做外链的时间和精力！

Automatic Backlink Creator 主要针对 wordpress 程序建立的网站。热衷于 WP 的站长朋友，尤其是针对做外贸的站长朋友们，主做 Google、Yahoo 搜索引擎 SEO，应该是一个非常好的消息！

这款软件类似于 WP 插件，是一个 WP 站外链的完美解决方案！只需要在网站后台轻松安装，就可以采用对搜索引擎有好的方式，让 WP 网站自动增加高权重外链。

最近本软件的官方网站，Automatic Backlink Creator 的价格只需要 37 美元，可以使用信用卡或者 paypal 付款，在国外卖的非常火爆！购买的同时，还赠送 MetaSnatcher 插件，这款插件可以自动跟踪排在谷歌前几名的竞争对手网站核心关键，并且自动返回到软件，省去大量关键字分析的时间。Spin Master Pro 插件。这款插件相当于是 WP 离线伪原创和发布插件，安装这款插件之后，可以在自己的电脑进行内容伪原创并离线发布，节省大量时间。同时，本软件提供 60 天不满意退款保证。

[点击查看](#)

这款软件的开发者是一群 SEO 高手，结合谷歌和雅虎的外链算法，开发出了这款强大优秀的外链软件，在外链的 PR、OBL、FLAG 等方面考虑到了极致。并且通过这个系统可以产生稳定的、持续增加的高质量反链，例如.edu、.gov 等等网站的外链。

这款软件在官网的结果是\$29 美元，同样可以使用信用卡或者 paypal 付款，购买的同时还赠送这个团队总结的最新高质量外链方法，针对谷歌和雅虎的排名提升，会有质的突破！

同时，本软件提供 60 天不满意退款保证。

下载：[最经典的SEO链轮解决方案](#)