



Lesson #23

# Do Almost Anything With Conditional Tags

By

**Lynette Chandler**  
**Tech Based Training**

<http://TechBasedTraining.com/>

**NOTICE: You Do NOT Have the Right  
to Reprint or Resell this Report!**

You Also MAY NOT Give Away,  
Sell or Share the Content Herein

If you obtained this report from anywhere other than **Tech Based Training.com** , you have a pirated copy.

Help stop internet piracy by letting me know. Send email to  
[customerservice@techbasedsupport.com](mailto:customerservice@techbasedsupport.com)

© 2010 Copyright Lynette Chandler

Legal Disclaimer: Keeping things simple here, I have to say that I cannot promise you success. I can give you direction and advice based on my experiences and good internet marketing practices. What you do with this information is up to you. As a Tech Based Training member you agree to not hold me responsible for your results.

## **Introduction & Recap**

Over the past few lessons, we've dipped our toes into conditional tags. Maybe you have already begun to see the power of using conditional tags. We've applied it to the archive pages so that the heading on the archives change when they visit a yearly, monthly, author or category archives.

Those are very practical and common uses. There are many more places and situations you can apply this to. In this lesson, we will explore some of them.

## **Adding Javascript On Only Specific Pages, Posts Or Categories**

Whenever you find a cool Javascript, you are usually told to copy a certain part of it into the page's header or in between the `<head></head>` of your HTML code. In WordPress, you simply have to add it into the header.php template file, between `<head>` and `</head>`.

When add anything in the header (or the footer), it gets called on every single page that gets viewed on your site. But what if you are only using the code on the home page? For example, newsletter signup popups. This is when conditional tags can help.

In this example, I will be using a Javascript code that I picked from Aweber after creating a list and web form. I created a 'Light box' pop-over that would display the sign up box and fade away after a while. I only want this on the home page of the blog.

The instruction from Aweber is to add my Javascript code between the `<body>` and `</body>` tags. For optimal performance, I'm going to put it just before the closing `</body>` tag. There's a reason for this. If you put it – or any Javascript code immediately after the opening `<body>` tag, then it gets loaded before your content. If Aweber hangs up for any reason, your page may not load and it'll sit there waiting for Aweber before it gives up and then loads your content. Your

visitors will see a blank page or half loaded web page for a while. Your web page will appear to be 'slow' and sometimes even make people click away.

By putting it just before the closing `</body>` tag, your content gets loaded first and the Javascript is the last thing that does get loaded. Even if Aweber hangs up, my content will already be loaded. This is my preferred method. Marketing aside, usability and load time should be of utmost importance to your visitor. So think about them first.

Not all Javascripts work that way. Some require the code to be pasted immediately after the opening `<body>` tag. In that case you have little choice.

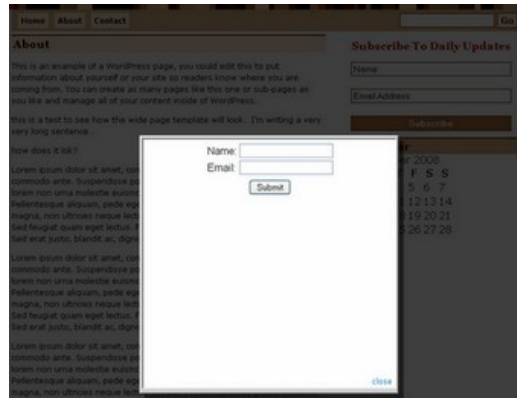
Since my decision is to put it in the closing `</body>` tag and it is located inside my footer.php file, that's the file I'll be working with. If you want to put it right after the opening `<body>` tag, then you'll need to work with header.php since that's where the `<body>` tag is.

Here it is in my footer.php



```
0      10      20      30      40      50      60      70
1      <div id="foot">
2      <p>Copyright &copy; 2008 <?php bloginfo('name') ?><br />
3      Design by <a href="#">BlogThemeBootcamp.com</a></p>
4      </div>
5
6      </div>
7      <script type="text/javascript" src="http://forms.aweber.com/form/15/
8      985980315.js"></script>
9      </body>
10     </html>
```

Let's check to see if it works first and it does. It also works on each and every page on the site which is not the intended effect.



Now, I'll add a conditional tag around the Javascript code. This is the code.

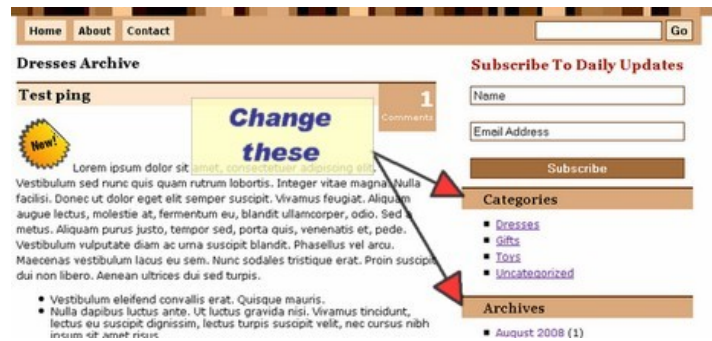
```
<?php if (is_front_page () ) { ?>
Your Javascript code here
<?php } ?>
```

```
1 <div id="foot">
2 <p>Copyright &copy; 2008 <?php bloginfo('name') ?><br />
3 Design by <a href="#">BlogThemeBootcamp.com</a></p>
4 </div>
5
6 </div>
7 <?php if (is_front_page () ) { ?>
8 <script type="text/javascript" src="http://forms.aweber.com/form/15/
9 985980315.js"></script>
10 <?php } ?>
11 </body>
```

Now, only the home page will display the light box.

## **Displaying Different Stylesheets On Different Categories**

In a similar vein, sometimes you want the stylesheet (CSS) to change slightly for each category. You can do that with conditional tags too. For this lesson, I'm going to change the sidebar header colors to a light green in one of my categories.



My first step is to create a copy of style.css and name is something relevant. In this case, I'll rename it style3.css because my category ID is 3. We'll get to finding category ID in a minute.

I'll open style.css in PsPad, then click File Save As, change style.css to style3.css then save into the same folder. If you want to keep things organized you can actually create a folder just for your additional style sheets if you wish. For simplicity's sake, we'll not do that. Instead, we save it in the same place as style.css.

Now, I will go into style3.css to find the code that controls the sidebar header background color. If you can't remember or don't know, just use Firebug ☺.

I happen to find the code around line 239. I'll go ahead and switch out the color code and save the file.

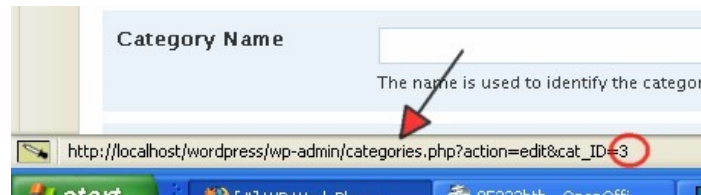
```
238
239 #sidebar h2 {
240     background-color: #DCAA7C;
241     border-top: 2px solid #5C2A04;
242     padding: 3px 2px 3px 25px;
243     margin: 0; From this
244 }
```

```
239 #sidebar h2 {
240     background-color: #DADD8F;
241     border-top: 2px solid #5C2A04;
242     padding: 3px 2px 3px 25px;
243     margin: 0; To this
244 }
```

We're not done yet. Now, we need to hook it up to header.php. Why header.php because that is where style sheets are called from. Before we can do that, we need to first find out what the ID of the category we want to change the color is. That is actually quite simple.

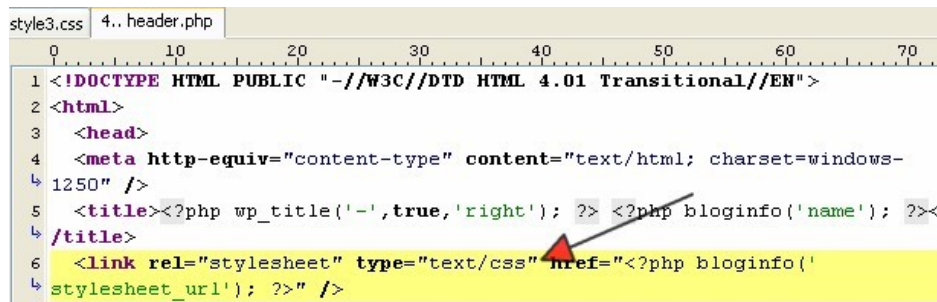
Log in to your WordPress admin, go to Manage >> Categories or if you're using WordPress 2.7 and up, click the arrow next to Posts to expose the options. Click Categories.

Find the category you are interested to work on, then hover your mouse over the category name. At the bottom of your browser, you should see a URL. In that URL, you will see cat\_ID=XX



The XX number right after cat\_ID= is your category ID. Now that I have the category ID number, I'll modify header.php. The original code is:

```
<link rel="stylesheet" type="text/css" href="<?php
bloginfo('stylesheet_url'); ?>" />
```



The modified code is this:

```
<?php if (is_category('3')) { ?>
<link rel="stylesheet" type="text/css" href="<?php
bloginfo('template_url'); ?>/style3.css" />
<?php } else { ?>
<link rel="stylesheet" type="text/css" href="<?php
bloginfo('stylesheet_url'); ?>" />
```

<?php } ?>

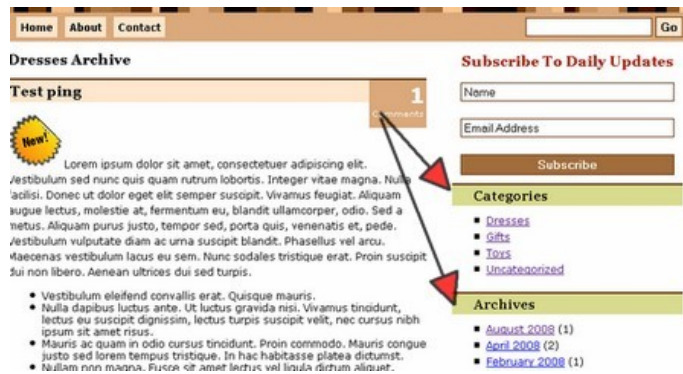
```
4 <meta http-equiv="content-type" content="text/html; charset=windows-  
5 1250" />  
6 <title><?php wp_title('-',true,'right'); ?> <?php bloginfo('name'); ?><  
7 /title>  
8 <?php if (is_category('3')) { ?>  
9 <link rel="stylesheet" type="text/css" href="<?php bloginfo('template_url'); ?>/style3.css" />  
10 <?php } else { ?>  
11 <link rel="stylesheet" type="text/css" href="<?php bloginfo('stylesheet_url'); ?>" />  
12 <?php } ?>  
13 <?php wp_head(); ?>
```

Let's talk about the code a little bit. The first part tells WordPress to pay attention to category 3 – if category 3 archive is being displayed, then use style3.css. Otherwise, use the default. You can keep adding to it. For example, you have category 4 which you want to use style4.css on, then you'd do this.

```
<?php if (is_category('3')) { ?>  
    <link rel="stylesheet" type="text/css" href="<?php  
    bloginfo('template_url'); ?>/style3.css" />  
<?php } elseif (is_category('4')) { ?>  
    <link rel="stylesheet" type="text/css" href="<?php  
    bloginfo('template_url'); ?>/style4.css" />  
    <?php } else { ?>  
    <link rel="stylesheet" type="text/css" href="<?php  
    bloginfo('stylesheet_url'); ?>" />  
    <?php } ?>
```

All you need to do is keep expanding on it. So here are my results. When I view category 3, this is what I see.





I'm going to provide a link to the theme which contain the changes I've made at the end of the lesson. Please note that it will not appear the same on your sandbox. That is because you may not have a category 3. Which is why you should only make these changes after the theme is developed and you are ready to apply it to a site.

## ***Different Sidebars In Different Categories Or Pages***

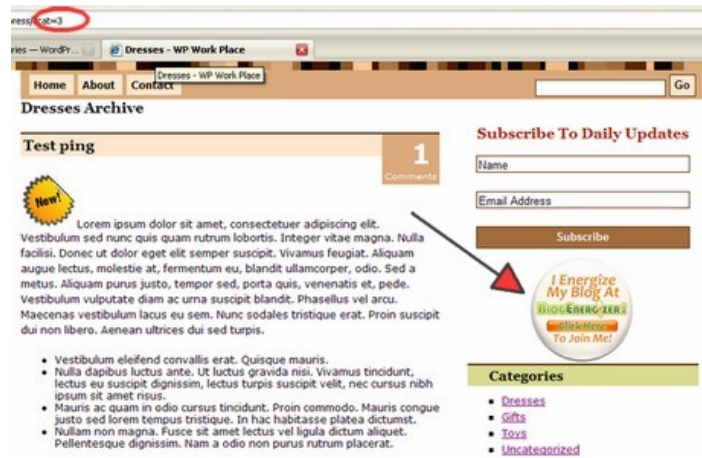
If you have worked with WordPress for a while, you know that you can create different sidebars for different categories. The simplest, most lo-tech method is to create separate template files for each category. Another method is to use conditional tags.

Let's say I have a banner ad that is relevant to one specific category. I want to display this banner ad in the sidebar when an individual post or category is being viewed. As with the previous example, I would first find out the category ID number.

Once I have that, I will open up sidebar.php in PsPad. Right under the newsletter sign up box, because it simply looks better there, I'll add this code:

```
<?php if (in_category ('3')) { ?>
    <div style="text-align: center;"><a
href="http://blogenergizer.com"></a></div>  
  
<?php } ?>
```

You may see something familiar here. If in\_category 3 simply tells WordPress to pay attention and add the button if the post is filed in category 3. Now, when I view category 3 archives or any post that is filed in category 3, the button will appear.



Here's a warning note. Different sidebars with different categories only work well if you do not file each posts in different categories. For example, you have a post that you put in category 1 and category 2. WordPress cannot decide which sidebar to display so it will choose one – which may not be the one you want. Because of this, the best thing is always to file your posts in one category only. There are also SEO benefits to filing your posts in just one category.

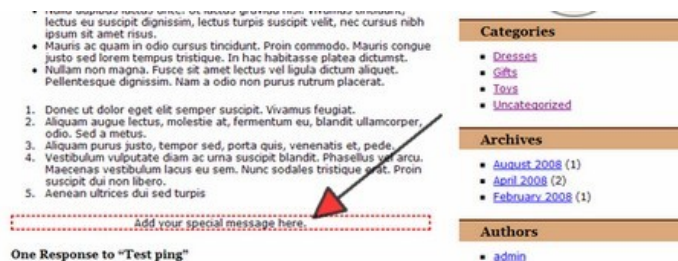
## **Special Messages Or Banner When Post Belongs To A Category**

Sometimes, you want to display a relevant banner or message when people view a post that is filed in a specific category. Here's what you do. Open single.php because that is the template that is used to display individual posts. Then add something like this where you want the message or banner to display.

```
<?php if (in_category ('3')) { ?>
<div style="text-align: center; border: 2px
dashed red;">Add your special message here.</div>
<?php } ?>
```

```
15 <h1><a href="<?php the_permalink() ?>"><?php the_title(); ?></a></
16 by <?php the_author(); ?> on <?php the_date(); ?>
17 </div>
18 <?php the_content(); ?>
19
20 <?php if (in_category ('3')) { ?>
21 <div style="text-align: center; border: 2px dashed red;">Add your
22 special message here.</div>
23 <?php } ?>
24
25 <?php comments_template(); ?>
26 </div>
27 <?php endwhile; ?>
```

This puts a message like this under my post only when viewing a post that is filed under category 3



Once again, the same problem will rear its ugly head if your post is filed under two or more categories. When using this method, ensure that the user only files each post in one category only.

## Conclusion

So there you go. Pretty interesting stuff huh? So how do you know which conditional tag to use and when? Thankfully, WordPress Codex has got a [list of different conditional tags](#) available for us to use and an explanation when to use each. I highly suggest you experiment. Test them out. If your client requests something you think a conditional tag may solve, tell them you will try some

known solutions and test them. That's the best way to know which tags to apply in each situation.

Here's the theme package for this lesson. Download and review its contents if you get stuck.

[Click to download.](#)

## **Next Lesson...**

Did you know, when building Pages in WordPress, you can attach a special template just for that page? This allows you to create a Page that looks entirely different or just a little different from the rest while maintaining a general look and feel of the site. We'll discover the world of Page Templates next.

*Lynette Chandler*

Lynette Chandler

**Tech Based Training**