

# Taking Control of Your Websites With PHP Template Tricks

- A Basic Tutorial -

Presented by:  
GuerrillaArticleProfitSystem.com



Your Competition Is Out To Get You  
[Read This Report Quick](#)

GuerrillaArticleProfitSystem.com

## Getting Started

Designing your websites with PHP templating is useful for managing your sites navigation, header and footer files independently from your sites actual content allowing you to **make instant changes across your entire website by changing the one file**. We design a template page with "php include" tags that will insert the content from whatever page it is set to call where ever you place the tag on your template page.

This is a php tag that would insert the contents from the file named left-menu.html into your template page.

```
<?php include("left-menu.html"); ?>
```

From now on, if we added pages or needed to change our navigation; we would just edit the left-menu.html file directly and the updated navigation menu would show instantly across our entire website, even if we had hundreds of thousands of pages.

## Sample 1

### Basic - Header, Menu, Footer Template

Let's dive right in and design a simple php templated page. In this example, we will create a separate header, left navigation menu and a footer. The code for that is below.

**You can find all the files for this template in the included *Sample-1* folder.**

```
<html>
<head>
  <meta content="text/html; charset=ISO-8859-1"
  http-equiv="content-type">
  <title></title>
</head>
<body>
<div align="center"><?php include("header.html"); ?>
<br>
<table width="100%">
  <tbody>
    <tr>
      <td><?php include("left-menu.html"); ?></td>
      <td>Our Main Body Content is Here</td>
    </tr>
  </tbody>
</table>
<?php include("footer.html"); ?>
</div>
</body>
</html>
```

We'll name this file `index.php` , it will be our main template for our website. **The file extension `.php` is necessary** so that our server knows to look for php code on this page and to process it accordingly.

If we prefer not to name our pages with the `.php` file extension, **we can name them `.html` or whatever extension we choose and tell the server to look for php code in files with those extensions. We will go over that in a later chapter.** For now, we will be using the `.php` extension for our main index page.

Now that we've designed our first, simple template, we need to

design our header.html, left-menu.html and footer.html. So Let's do that now. We'll keep these very basic as well for our first sample.

## Header

Our Sample 1, header will be a simple line of text.

```
<div align="center">
<h1>MY AWESOME WEBSITE</h1>
</div>
```

We paste this code into notepad and save it as header.html.

## Left Menu

Our Sample 1, left navigation menu is a basic three cell table with three links. Remember, since our main template has the head, body, meta and other main tags that we don't need them again so our files that we will include needs to only have the exact html you want inserted on your page.

```
<table>
  <tbody>
    <tr>
      <td><a href="http://www.yoursite.com/link1.html">Link
One</a></td>
    </tr>
    <tr>
      <td><a href="http://www.yoursite.com/link2.html">Link
Two</a></td>
    </tr>
    <tr>
      <td><a href="http://www.yoursite.com/link3.html">Link
```

```
Three</a></td>  
</tr>  
</tbody>  
</table>
```

We paste this code into notepad and save it as left-menu.html

## Footer

Our Sample 1, footer will be a simple line of text

```
<div align="center">THIS IS MY FOOTER</div>
```

We paste this code into notepad and save it as footer.html

That's it! The above is the basic outline for a simple php template.

Now, I want you to upload the Sample-1 folder to your server and type [www.yourdomain.com/Sample-1/](http://www.yourdomain.com/Sample-1/) into your browser. You will see your new page with the header, left navigation menu, content area and footer.

## Let's Practice!

Open up your favorite html editor and select your index.php file. Save this file as index2.php, index3.php, index4.php and index5.php and upload them to your server.

Open up left-menu.html and add a few more rows to the table. Change the link text to Main, Index 2, Index 3, Index 4 and Index 5 then hyperlink them to the pages you just created. The index2.php, etc.. link the word Main back to your index.php

Make sure before you save the file that it only contains the html and not head tags, meta tags, etc...

Upload the left-menu html and go back to [www.yourdomain.com/Sample-1/](http://www.yourdomain.com/Sample-1/) or refresh the page if you are still there.

You should now see a page with your new menu. Click on the navigation links and you should go to those pages. They will look the same because we don't have any other content on the page yet but the address in your browsers address bar will change.

Now, open your header.html file and go ahead and put anything you like in there. Do the same thing with your footer.html file. Go ahead and upload these two files.

Refresh your browser and click on the navigation links. You will see that your new header and footer now show the updates on all pages! Simple huh?

Go ahead and practice with Sample-1 for a bit. Add some graphics if you like, table borders and colors, etc.. until you feel comfortable making site wide changes.

One thing to pay attention to is your table settings. You will want to **define table widths**, either on your main template page or your left-menu.html page so that things fit nicely. Notice that our "Content" is centered and not aligned to the left. These types of layout characteristics needs to be set when creating your main template page. After all, you wouldn't want to have to go through all of your newly created pages and make individual changes. That would defeat the purpose of templating.

## Advanced Templates

If you will now look at your Sample-Advanced folder you will see a file named startwiththis.html

So, let's say that you designed your template. This is how your website will look across all of it's pages. We now want to go through the html on that page and break it apart into sections so we will be able to change our header, navigation and footer across our entire website with ease.

We are also going one step further with this one and will include a php include for the "Hot Price Update!". It would be nice to be able to change this any time we like and have that change reflected site wide.

So let's tear into it.

What I find easiest is to start by actually just copying the different sections and pasting into my wysiwyg editor like Frontpage or NVU.

For this example, I selected the table and pasted it into a new page. I then stripped out the head and body tags leaving only the html for table itself.

Because some text editors can be a little finicky and add the main tags back in, I'll copy the code for the table and paste into notepad and save that as header.html.

I then repeat for each section of the page until I am left with my "template" or skeleton.

Open up the Sample-Layout.php file in your html editor to see how your page would look at this point. This would be your main starting out page except as noted on that page, you wouldn't actually have your placeholders displaying.

For the finished version, open up Template.php

**You would always keep this file unchanged. It is your websites primary template.**

When creating a new page, you would open up your template, add in your content, change your page title, add keywords and description, save it as whatever.php (whatever being whatever name you decide) and uploading it.

*It will automatically insert the header, navigation, footer, etc.. in the proper place.*

## Setting Up Categories

To take what we've learned so far and take it a step further, **you can have different templates for different parts of your site.**

You could have one template which pulled different navigation pages and/or headers than that of your main site. This can be useful for building sites that focus on different niches while still sharing a common navigation system.

```
<?php include("common-menu.html"); ?>
```

For surfers to navigate throughout the various categories of your website while having specialized, niche navigation menus to use within the structure of those categories.

```
<?php include("niche-menu.html"); ?>
```

This way when you add to or take away from content directly related to a particular niche, you can just edit the niche-menu to make the



change throughout all pages in that category. If you were to make changes that you wanted to show site wide, like a new category or niche being added, you could edit the common-menu.html to show those changes.

The name of the files are irrelevant. You could use any naming structure that you would be able to easily understand and work with like...

*articlewriting-menu.html for all pages within that niche*  
*productlaunch-menu for all pages within that niche*

Upload the Sample-Advanced folder and access it now at <http://www.yoursite.com/Sample-Advanced/>

This page looks like one page but it is actually the template with all the pages being inserted into the page via the php include tags. Go ahead and click on some of the links just to get a feel of it.

## **Advanced Practice**

For practice, open up header.html and change "MyWebPage" to your site name, save the file (make sure your editor didn't add the <head> tags back into it and then upload it into the Sample-Advanced folder, overwriting the existing header.html file.

Now go back and click on the links again. Every page will now show your site name. If it doesn't, make sure you've saved the file, uploaded into the correct folder and refresh the page so you aren't looking at a cached version.

Now, go ahead and open up the about.php file and type some information into it, save and upload it. Open up the hotprice.html and type something there, save and upload it as well.

Now go back to your site and view the changes.

Is it getting easier??

Something to keep in mind when building sites using templates is your paths. If you've already been building websites then you are no doubt aware that when linking to an image or page that such a page has to exist.

For instance, all the files above contain links like

```
<a href="page5.php">
```

This works because page5.php is in the same folder. If it is in a subfolder, you would have to either adjust the path in your link like this...

```
<a href="subfolder/page5.php"> if in a subfolder or <a href="../page5.php"> if in the directory above the current directory or <a href="../subfolder/page5.php"> if in a subfolder of the directory above the current directory.
```

Sounds a little confusing, if at all unsure then you can always just supply the full link and then it won't matter, like

```
<a href="http://www.yoursite.com/folder/page5.php">
```

The same thing applies when using the php includes tags

```
<?php include("common-menu.html"); ?>
```

The above will work when the file you are including is in the same folder, as they all were in our samples but when doing a large site, you may have a complicated structure with many levels of directories and subdirectories. You can always be safe by storing your files to be included in one main directory like "includes"

and then using the full path in your tags, like...

```
<?php include("http://www.yoursite.com/includes/common-menu.html"); ?>
```

That concludes the templating part of this tutorial, practice makes perfect so go ahead and play around with the sample files above until you're comfortable with it. When you are ready, start by using the files above and just paste your own content into them to make it your own or start from scratch with a simple header and footer being added and then add more php include components to it

## **Set Up Banner Rotation**

Now that we are able to quickly make changes to our header, navigation menu, footer, etc.. in minutes, we would be remiss if we didn't apply that same technology to our banners and advertising links.

Imaging being able to switch out a poorly converting sponsor site wide in just seconds or add a new sponsor into the mix.

Announce the newest, hottest product site wide in seconds. The uses for this are only limited by your imagination.

For this example, I've made a copy of the Sample-Advanced folder and named it Banner-Rotation. Please upload it now.

To implement this, we will need one file which for this example will be named banner.php , If you open it now with your text editor, you will see lines of code that look the same as what you would see if looking at your code when using your html editor.

As a matter of fact, that's the easiest way of getting the code. Just

use your html editor as you normally would, then just copy the code and paste it into your banners.php file.

Here is one line of banner code from our banners.php file

```
<a href="http://www.yoursponsor.com"></a>
```

Notice for this example the src (source) for banner1.gif is in the images folder. This is where that banner has been uploaded to. Something else to note is that I am not using the full path to the banner1.gif because I didn't know what your domain would be and I knew that the images folder existed inside the folder we are working with, like we discussed above about the paths.

Because you could possibly use this on many directory and subdirectory levels, it would be best to upload your banners into a main banners folder at the top level of your website, then in this code you will be adding to your banners.php file, use the full path to the banners folder so your code would look like this.

```
<a href="http://www.yoursponsor.com"></a>
```

You'll also notice several lines of text links. You can pretty much put anything here to display on your page that you like as long as you keep the code for each instance on it's own line.

Make sure when editing this file that your "word wrap" is off. In notepad, you would find this by clicking "format" on the top Grey tool bar and then unchecking "word wrap".

To display the banners on the page

```
<div align="center">
<?php
$banners = file("banners.php");
$banner = rand(0, sizeof($banners)-1);
echo $banners[$banner];
?>
</div>
```

Don't forget to **use your full path** to the banners.php if it will be used on different directory and subdirectory levels

```
$banners = file("http://www.yoursite.com/banners.php");
```

## Keep Your Site Fresh

You aren't limited to advertising with the above code. Instead of banners and links, here are a few other ways I use it to keep sites looking fresh and updated to returning visitors.

### Motivational Quotes

Motivational quotes provide a way to send powerful messages to your visitors who may be hesitating. After all, who can say it better than some of the worlds most successful.

Open quotes.php for a few examples like the one below...

```
We are the creative force of our life, and through our own decisions  
rather than our conditions,<br>if we carefully learn to do  
certain things, we can accomplish those goals</font>.</b><font  
color="#FF0000"> <br>~Stephen Covey</font></font></p>
```

Just format the quotes in your html editor as you would like for them

to appear then paste the code onto one line inside your quotes.php file.

Insert these quotes into your page by placing the following code where you would like them to appear.

```
<?php
$quotes = file("quotes.php");
$quote = rand(0, sizeof($quotes)-1);
echo $quotes[$quote];
?>
```

## Rotate Images On Your Site

Set it up exactly as we did the banners.php except you don't have to link them, though you certainly can, use box covers, pictures of cars, planes, your favorite artwork, whatever you like.

Paste the code inside images.php (can be named anything) and then paste the following code where ever you would like a random image to appear.

```
<?php
$images = file("images.php");
$image = rand(0, sizeof($images)-1);
echo $images[$image];
?>
```

By now you are probably starting to notice a pattern.

In the codes above you will see \$quotes, \$images and \$banners. You can paste code for anything you would like randomized in a new file and name it anythingyouwant.php , just change the code used to insert it into your pages to the correct file name.

**When changing the code, pay attention to whether the word is singular or plural. You must keep that aspect of it the same.**

For example, let's say you wanted to create a new randomized file.

Open up a text editor and on separate lines, write what you would like to have displayed on the site.

like...

Great Day We're Having!  
Are You Ready For Christmas?  
Stop Thinking About It and Just DO IT!

We would then save that txt file as blurbs.php and upload it.

Then we would take the code above for images and change it

```
<?php
$blurbs = file("blurbs.php");
$blurb = rand(0, sizeof($blurbs)-1);
echo $blurbs[$blurb];
?>
```

And paste that code where we would like for it to appear. **Don't forget to use your full path** to the blurbs.php if it will be used on different directory and subdirectory levels

```
$blurbs = file("http://www.yoursite.com/blurbs.php");
```

## **Categorize Your Banners**

The banner code we used above was pasted in the footer.html file which means it will be displayed on every page throughout the site.

This is great for broad reach but what if we have different niches within our site and we would like to target certain products to visitors on related portions of our site.

Simply create separate separate banner files for each niche and name them appropriately, change the code to insert the banners into your pages to reflect the correct file as shown above and paste that into your pages where you would like the banners to appear.

There are a couple of ways to "categorize" your niches.

You can either start with your primary Template.php file and make changes to that like adding in a niche-menu.php, niche-header, niche-footer, etc.. and then saving it as your Niche-Template.php so every page you create within that niche, you will just start with your primary Niche-Template.php.

Because we would still like our main site navigation and probably a few of our other site wide includes to remain, we can actually place a sub-footer by creating a new niche-footer.php and pasting the include for it right above the existing global footer

```
<?php include("niche-footer.html"); ?>  
<?php include("footer.html"); ?>
```

You can do the same with your main site navigation and header files so that if you make a change that affects your entire site regardless of niche, just edit the global, site wide files and if you just need to make instant changes across a niche, edit the related files for that and it's done in seconds.



## Manage Your Linking Codes

Using this templating system, you can quickly manage links across tens of thousands of pages as well.

Let's say you have been promoting sponsor XYZ and they suddenly shut down. Well, if you're like me, you probably have hundreds of links scattered throughout your site pointing to sponsor XYZ and now they are useless.

Before this would have meant going through and searching out those links and manually changing them or possibly if you are really brave you might try to do a search and replace on your server, risky business to say the least.

Now you can manage you linking one of two ways.

The first way I'll describe here is utilizing the php include

To use this method, you should create a folder to store all of your sponsor codes.

Create a file named sponsorxyz.html and place your linking code there.

```
<a href="http://www.mysponsor.com">Visit my sponsor</a>
```

Then, where you want you sponsor link to appear, place...

```
<?php  
include("http://www.yoursite.com/sponsors/sponsorxyz.html"); ?>
```

If you want a little more control over your links, you could alternatively place only the sponsors linking code in the sponsorxys.html file and insert it directly into the link you place on your page like...

```
<a href="<?php  
include("http://www.yoursite.com/sponsors/sponsorxyz.html"); ?  
>">Visit my sponsor</a>
```

The second way is using the banners.php type templating like we used above.

The only reason you may prefer to use this method rather than the first is that it would give you the opportunity to pseudo split test.

Because the results are random, the results should not be viewed as entirely conclusive but depending on the difference in stats, could help you decide which is better.

To utilize this method, set up your sponsorxyz.php like above and place two or three sponsor links there.

Make sure to set up your link in your html editor and copy the entire code for the link including formatting. Edit the code as we did above, being careful with the path and **keeping the singular and plurals the same** and place that on your page where you would like for your links to appear.

To ease the utilization of this method, you may place either of these linking codes directly into the header, footer or other includes you have for that page, so you can instantly add your new link across all pages or category pages in minutes.

## Other Useful Ways to Use

Have you ever worried about your adsense account getting terminated? It happens.

Or maybe you would like to compare result between the different ppc sponsors.

Set up multiple pages for your different size ads

468x60.html  
250x250.html  
300x250.html

and so on. Paste whatever ppc sponsor code you want displayed into the corresponding size and insert in your pages or your other include files like header.html, footer.html, navigation.html, etc.. using the php include code...

```
<?php include("http://www.yoursite.com/sponsors/468x60.html"); ?>
```

## Setting Up Your Server To Parse .html Pages

If you do not want to use the file extension .php or if you would like to add code to .html or .htm pages that already currently exist on your server, you can accomplish this yourself if you have the ability to use .htaccess on your site.

Open up your .htaccess file or create a new one by opening a blank text page and add the following line to the top of the file

```
AddType application/x-httpd-php .php .html .htm
```

Save or if this is a new file, save as .htaccess and upload to your server. This tells your server to check any page with the extensions .html or .htm for and execute any php code.

If you do not have the ability to use .htaccess, contact your host and they may make changes on their end that will allow this.

**That concludes this tutorial. I hope that you have found the information useful and that you were able to understand it and also that you will put it to good use.**

