



The Handbook

Codex

http://codex.wordpress.org/Template_Tags/Anatomy_of_a_Template_Tag

Version Date

9 August 2005

Template Tags/Anatomy of a Template Tag

Introduction

This document provides a brief examination of the animal known as the WordPress [template tag](#), to help those who may be new to WordPress and [PHP](#) understand what template tags are and how they're used.

A WordPress template tag is made up of three components:

- [A PHP code tag](#)
- [A WordPress function](#)
- [Optional parameters](#)

These are explained below.

PHP code tag

WordPress is built with the PHP scripting language. Though you certainly don't need to be a PHP developer to use it, knowing a little about the language can go a long way in getting the most out of WordPress. Here we provide a tiny bit of that PHP knowledge:

```
<?php ?>
```

The above shows the opening (`<?php`) and closing (`?>`) tag elements used to embed PHP functions and code in a [HTML](#) document, i.e. web page. There are a number of ways to embed PHP within a page, but this is the most "portable," in that it works on nearly every [web server](#)—as long as the server supports PHP (typically a document's filename also needs to end with the extension `.php`, so the server recognizes it as a PHP document).

Anything within this tag is parsed and handled by the PHP interpreter, which runs on the web server (the interpreter is the PHP engine that figures out what the various functions and code do, and returns their output). For our purposes, the PHP tag lets you place WordPress functions in your page template, and through these generate the dynamic portions of your blog.

WordPress function

A WordPress or template function is a PHP function that performs an action or displays information specific to your blog. And like a PHP function, a WordPress function is defined by a line of text (of one or more words, no spaces), open and close brackets (parentheses), and typically a semi-colon, used to end a code statement in PHP. An example of a WordPress function is:

```
the_ID();
```

[the_ID\(\)](#) displays the ID number for a blog entry or post. To use it in a page template, you slip it into the PHP tag shown above:

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

This is now officially a WordPress template tag, as it uses the PHP tag with a WordPress function.

Optional parameters

The final item making up a template tag is one you won't necessarily make use of unless you want to customize the tag's functionality. This, or rather these, are the parameters or arguments for a function. Here is the template function [bloginfo\(\)](#), with the **show** parameter being passed the 'name' value:

```
<?php bloginfo('name'); ?>
```

If your blog's name is **Super Weblog**, the `bloginfo()` template tag, when using 'name' as the **show** parameter value, will display that name where it's embedded in your page template.

Not all template tags accept parameters (`the_ID()` is one), and those which do accept different ones based on their intended use, so that [the_content\(\)](#) accepts parameters separate from those which [get_calendar\(\)](#) can be passed.

Further reading

See the following Codex pages for more information on WordPress templates and template tags:

[Templates](#)

[How to Pass Tag Parameters](#)