

This hands-on workshop will be administered using Virtual Box's free virtualization environment, the files for this workshop are available via (5) USB disk drives that will be handed out. To begin, copy the *wordpress.zip* and *<platform>.zip* file to your local workspace. Where *<platform>* is either Windows or OS X dependent on your system. If you already have Virtual Box installed on your system then you need only copy the *wordpress.zip* file. If you are using Linux then in an apt enabled environment you can install Virtual Box by reviewing:

[https://www.virtualbox.org/wiki/Linux\\_Downloads](https://www.virtualbox.org/wiki/Linux_Downloads)

Once everyone has installed Virtual Box we can begin by starting the virtual machine located in the *wordpress.zip* compressed folder.

To begin I will describe the base virtual image's configuration. These steps have already been taken therefore you do not need to follow through with these steps, I am just explicating them for your reference.

Initially Ubuntu Server 16.04 was installed from an ISO file downloaded directly from Canonical's website. It was used as the install media for the virtual image. After running a standard install with the addition of SSH support, the

image was further enhanced by installing the following package via apt:

```
lamp-server^
```

Please note that the server has the following user / pass configuration:

```
administrator / simple
```

Please note that MySQL has the following user / pass configuration:

```
root / simple
```

Essentially, this is all that is needed to run a most basic Wordpress site.

Next I used *wget* to get the latest version of Wordpress. A script file *get-wordpress.sh* is available in the *home* directory for *administrator* to get the latest source and uncompress it. At this point all pre-configurations have been made and we will begin the hands-on aspect of this workshop.

**Note:** *Due to a lack of built in SSH capability in Windows I am running all commands directly in the Virtual Box virtual machine head window rather than connecting via SSH from terminal. If you would prefer to connect via SSH that is your choice, provided you can handle the connection yourself.*

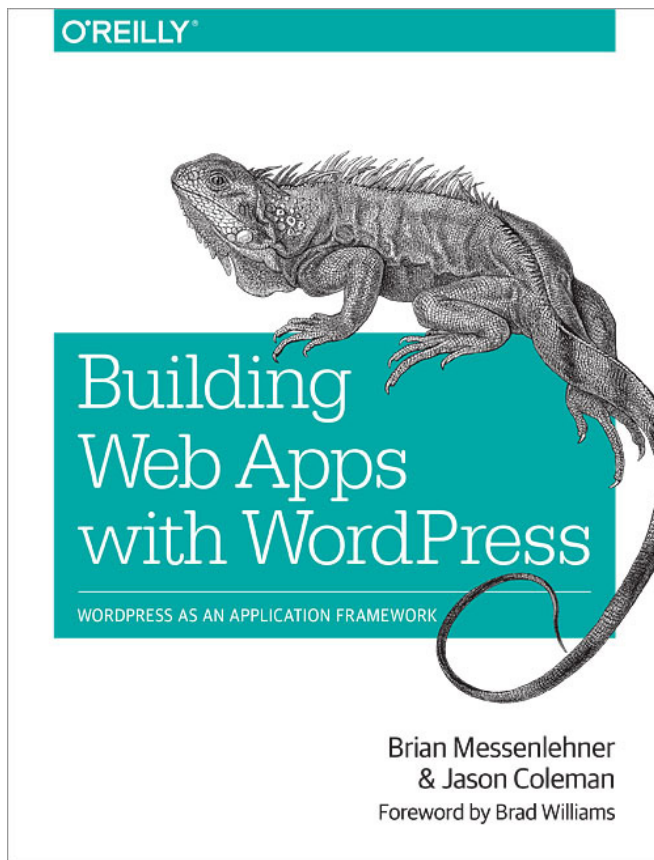
Let's get started!



---

Presenter:  
Phil Parzygnat - CS MA 2017  
<http://pmptechnology.com>





Above is a great text on the subject of Wordpress Development. I learned many valuable things reading this text. Below are also some additional links that could be useful as you get more familiar with Wordpress.

### Wordpress Theme Developers Official Docs

<https://developer.wordpress.org/themes/getting-started/>

### Wordpress Plugin Developers Official Docs

<https://developer.wordpress.org/plugins/intro/>

### Wordpress Developer Tools Plugin

<https://wordpress.org/plugins/developer/>

### Wordpress Theme Unit Test XML

[https://codex.wordpress.org/Theme\\_Unit\\_Test](https://codex.wordpress.org/Theme_Unit_Test)

### Bootstrap Starter Theme

<https://wordpress.org/themes/bootstrap-canvas-wp/>

### Bootstrap Documentation

<http://getbootstrap.com/getting-started/>

### Github Repository for this Workshop

<https://github.com/pmptechology/wpqc/archive/master.zip>

## WP-Loop Basics:

The Loop is PHP code used by WordPress to display posts. Using The Loop, WordPress processes each post to be displayed on the current page, and formats it according to how it matches specified criteria within The Loop tags. Any HTML or PHP code in the Loop will be processed on each post.

When WordPress documentation says "This tag must be within The Loop", such as for specific Template Tags or plugins, the tag will be repeated for each post. For example, The Loop displays the following information by default for each post:

Title (`the_title()`)

Time (`the_time()`)

Categories (`the_category()`).

You can display other information about each post using the appropriate Template Tags or (for advanced users) by accessing the `$post` variable, which is set with the current post's information while The Loop is running.

For a beginner's look at The Loop, see The Loop in Action.

## Using the Loop:

The Loop should be placed in `index.php` and in any other Templates used to display post information.

Be sure to include the call for the header template at the top of your Theme's templates. If you are using The Loop inside your own design (and your own design is not a template), set `WP_USE_THEMES` to false:

```
<?php define( 'WP_USE_THEMES', false ); get_header(); ?>
```

The loop starts here:

```
<?php if ( have_posts() ) : while ( have_posts() ) : the_post(); ?>
```

and ends here:

```
<?php endwhile; else : ?>
    <p><?php _e( 'Sorry, no posts matched your criteria.' ); ?></p>
<?php endif; ?>
```

This is using PHP's alternative syntax for control structures, and could also be expressed as:

```
<?php
if ( have_posts() ) {
    while ( have_posts() ) {
        the_post();
        //
        // Post Content here
        //
    } // end while
} // end if
?>
```