

Overview and Installation

Bloggging has been part of the web landscape for over a decade now. From personal journals to big corporate marketing, the medium has matured to become a ubiquitous mode of live communication. It is estimated that there are more than 145 million blogs, with over 250,000 new blogs created daily and 2 million new postings per day ([Technorati](#), 2010).

What is a Blog?

A “Blog” is a shortening of the words web and log. For you and me, it’s a website for personal or professional use, usually maintained by one person, that is updated frequently, and is set up in such a way that other people on the internet can read it, and comment on the materials posted. A blog can contain not only text, but also audio, video, and photos. Blogs started out being used as journals, and they still are, a place where you can share your thoughts and what you’re doing with the world at large.

The Different Types of Blog Architectures

There are many different types of blogs out there. Most of them fall under a few categories.

The Typical Blog

This is what you normally find on the Internet. The design is nice and clean, it’s easy to find what you’re looking for and (for the blogger) it’s very easy to setup and maintain.

The Photo Blog

Another popular type of blog that is very common on the Internet is what’s known as a “Photo Blog”. The benefit of having a photo blog is that you can share your latest images with the world. These can be paintings, 3D renders, photography, 2D renders, images of sculpture, logo designs, etc.

The point is showcasing your work in a very visual way. Some services for storing your images online include: [Zoomr](#), [Flickr](#), [Google’s Picasa Web](#), [imageshack](#) or [SmugMug](#).

The Video Blog

Last on the list is a video blog. People use video blogs in many different ways to serve a variety of purposes. If you’re an actor you can post readings and clips of you acting and use that for your promotion. If you teach people how to cook, you can make quick videos and show them on your blog.

You can review [Hulu.com](#) as an example of viewing TV shows for free (you can also embed those videos onto your blog if you want). The [Daily Show](#) has a blog that shows the latest episodes of the Daily Show, and you can view those for free. Other people post their favorite videos from YouTube on their blog and that is their entire blog.

In this class, I will be focusing on creating a typical blog while showing you how to incorporate a little bit of the other two in showcasing images and video.

Blogging Terminology

If you are going to be a blogger, you must know a few common terms; or else you'll be pulling your hair out with frustration.

At the heart of any blog are its **posts**. When you write something on your blog, that is a post. When you post something on your blog, WordPress creates something called a **permalink**, which is a fancy term for a permanent link to your post. You can give this URL to anyone so they can then view that specific post.

When you write a post, (provided you have **comments** enabled), someone can **comment** on your blog. What this means is underneath your post, you will see a comments section. Anyone on the web will be able to post a comment about your post. It's up to you if you want this functionality on your blog or not.

Another term you may not be familiar with is **trackbacks**. The way trackbacks work are, you write a post on your blog, then another blogger reads that post and decides to comment on it on their blog so their readers can benefit. So they post a "trackback" (which WordPress does automatically by linking to a blog) to your post.

Under that post you will notice a new comment that looks like:

[...] various text here [...]

The "..." mean there is something before or after this comment. You should be aware that spammers abuse this system to advertise on your blog. So you may want to keep checking on your comments so you can stop this if you are getting spammed.

Another somewhat recent term is **RSS**. What RSS does is allow someone to subscribe to your site, and read your material from their RSS reader, opposed to visiting your website.

[RSS](#) stands for "Really Simple Syndication". What this means for the end-user is, you can go to a website, and subscribe to their RSS feed. When you subscribe to a RSS feed, you get to see all the latest content, without having to go to the website. RSS is very important for both the end user and the content provider. With your RSS feed active, people who enjoy your website can check out the feeds and see all your latest content.

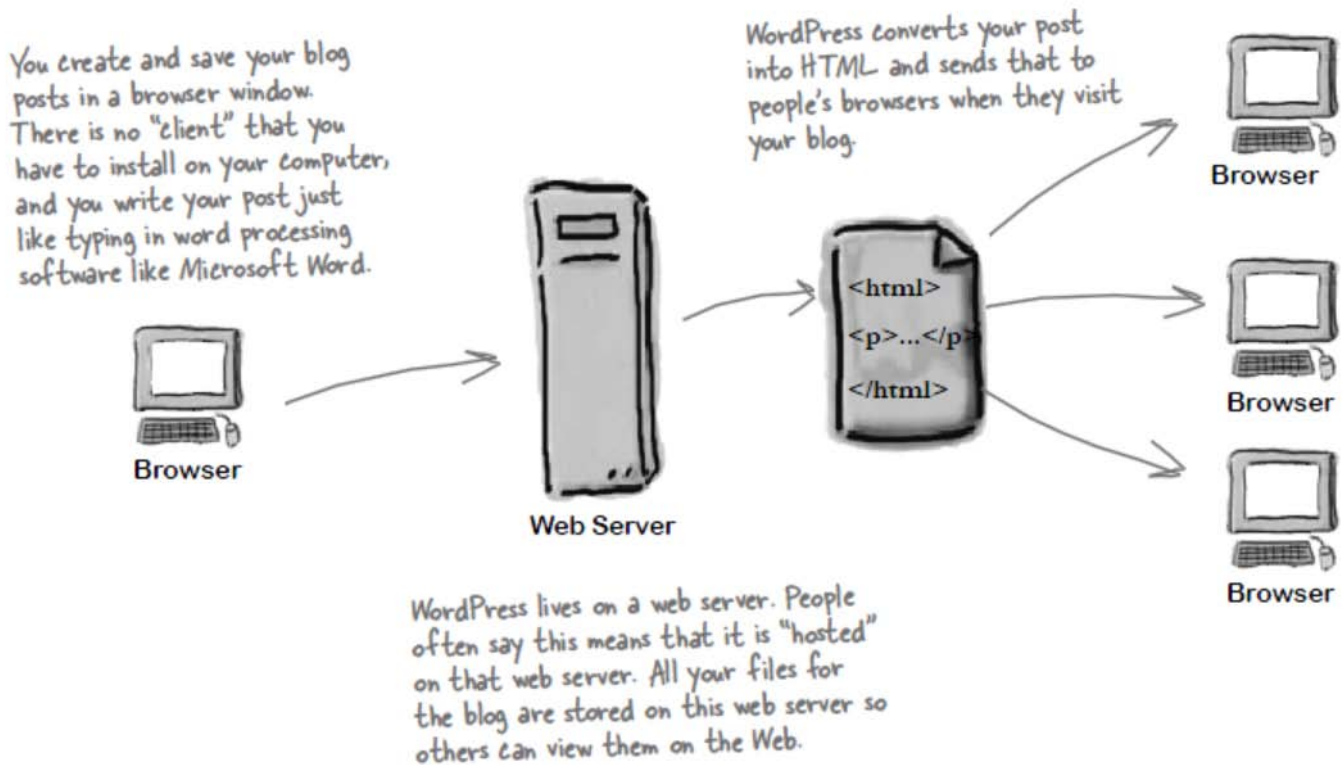
For users of RSS, you get to check out your favorite websites from inside one program, or reader, and see if there is any new content.

A Short History: WordPress

WordPress started similarly to many other popular open source software packages: Some talented developers saw a need to create a powerful, simple tool based on an existing project licensed under the [GPL](#). Built by Matt Mullenweg and Mike Little in 2003 using the MySQL open source database for persisting content and PHP as the development platform.

As with other systems written in PHP, it is self-contained in the sense that installation, configuration,

operation, and administration tasks are all contained in PHP modules. Beyond getting to a first post, WordPress was designed to be extended. WordPress today is supported by a handful of core developers and just under 100 key contributors.



WordPress as a Content Management System

Blogging systems have their roots in simple content management operations: create a post, persist it in stable storage such as a file system or database, and display the formatted output based on some set of temporal or keyword criteria. As the richness and types of content presented in blog pages expanded, and the requirements for sorting, searching, selecting, and presenting content grew to include metadata and content taxonomies, the line between vanilla, single-user-targeted blogging software and enterprise-grade content management systems blurred.

Content management systems (CMS) handle the creation, storage, retrieval, description or annotation, and publication or display of a variety of content types. CMS also covers workflow tasks, typically from an editorial or publishing perspective, but equally including actions such as approval and marking content for additional editing or review. The WordPress Dashboard provides those elements of workflow management and editorial control. WordPress isn't the only open source content management system in widespread use today; the Drupal and Joomla projects are equally popular choices.

Drupal and Joomla start from the perspective of managing content repositories; they handle a variety of content types, multiple authors in multiple roles, and getting the content to a consumer that requests it.

WordPress is at its heart a blogging system, and the end focus is on displaying content to a reader.

The Two Different WordPress Services

WordPress offers two different “classes” of service.

WordPress.com - This version is hosted at wordpress.com, as such; you don't need to pay anything. It's a free turnkey solution. You go to wordpress.com, create an account and you're running. So in other words WordPress.com is a service that runs the WordPress software. Though the main drawback is that the account you create is very limited for any type of customization.

WordPress.org - Home of the software, in the form of an open source project. This is where you can chat to other people about WordPress, download the latest install, find plugins and themes to access the official help and documentation – also known as the Codex. (note: This is the version that will be covered in this class.)

<i>Feature</i>	<i>WordPress.org</i>	<i>WordPress.com</i>
Cost	Free	Free
Software download	Yes	No
Software installation	Yes	No
Web hosting required	Yes	No
Custom CSS control	Yes	\$15/year
Template access	Yes	No
Sidebar widgets	Yes	Yes
RSS syndication	Yes	Yes
Access to core code	Yes	No
Ability to install plugins	Yes	No
WP themes installation	Yes	No
Multiauthor support	Yes	Yes
Unlimited number of blog setups with one account	Yes	Yes
Community-based support forums	Yes	Yes

Getting What You Need

Before I walk you through installing the WordPress software (the screenshots may vary to the version you download) you will need to gather some files and tools, and double check some settings. WordPress requires certain programs to be available on your remote server (web host).

A Web Host

You need a Web-host before you can install WordPress. There are many web hosts to choose from so it is up to you if you do not already have one.

The web host you choose must have three things:

- Linux (Unix) operating system.
- PHP Version 4.3 or later. (WordPress itself is written in PHP).
- MySQL version 4.0 or later. (All your posts, users and settings will be stored in this database).

Local Tools

You will need an FTP client to transfer the files to your web host. There are lots free FTP clients that are available such as [FileZilla](#), [Core FTP LE](#), [Free FTP](#) and others.

The WordPress Code

After you settled on a suitable web host you need the raw materials: the WordPress files.

These files can be found at <http://www.wordpress.org/download>. You will see a big blue box on the right for downloading the latest version of WordPress.

Download the .zip file and extract the files to a folder (Figure 1).

Name	Size	Type
wp-admin		File Folder
wp-content		File Folder
wp-includes		File Folder
index.php	1 KB	PHP File
license.txt	16 KB	Text Document
readme.html	8 KB	Firefox Document
wp-app.php	40 KB	PHP File
wp-atom.php	1 KB	PHP File
wp-blog-header.php	1 KB	PHP File
wp-comments-post.php	4 KB	PHP File
wp-commentsrss2.php	1 KB	PHP File
wp-config-sample.php	3 KB	PHP File
wp-cron.php	2 KB	PHP File
wp-feed.php	1 KB	PHP File
wp-links-opml.php	2 KB	PHP File
wp-load.php	2 KB	PHP File
wp-login.php	20 KB	PHP File
wp-mail.php	7 KB	PHP File

Figure 1 – Extracted files from download

Setting up the MySQL Database

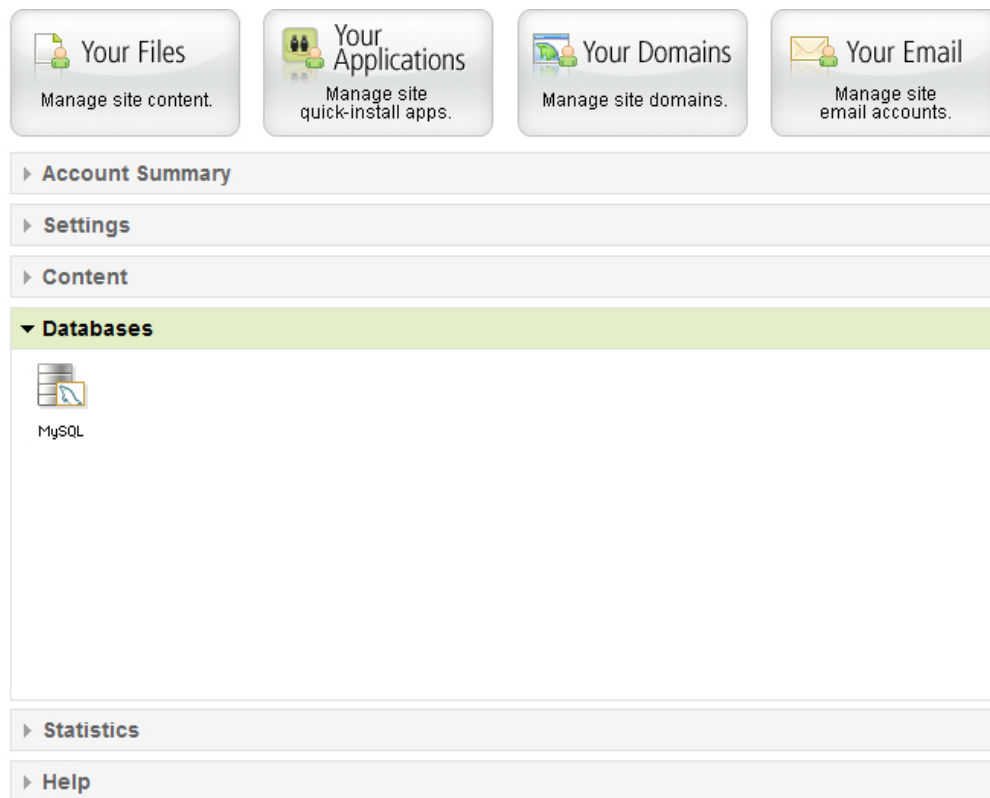
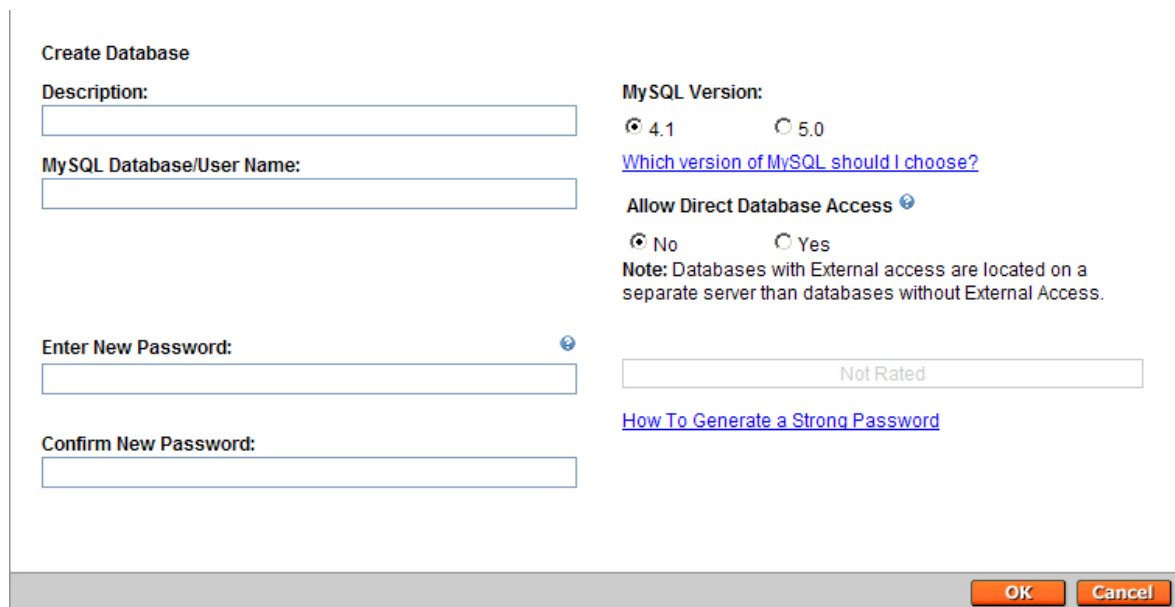


Figure 2 – Web host (GoDaddy) Control Panel

It's almost time to install WordPress, but first you need to create a MySQL database first. The installation will fail if you don't set the database up beforehand.

Step 1: In your Control Panel (Figure 2) there will be a Database section for MySQL; create a new database.



Create Database

Description:

MySQL Database/User Name:

Enter New Password:

Confirm New Password:

MySQL Version:
 4.1 5.0
[Which version of MySQL should I choose?](#)

Allow Direct Database Access No Yes
Note: Databases with External access are located on a separate server than databases without External Access.

Not Rated

[How To Generate a Strong Password](#)

OK Cancel

Figure 3 – Creating a new MySQL database

If your web host has options for specifying the MySQL version and/or Allow Direct Database Access (Figure 3) leave the default selections. You normally just need to enter a user name and password when creating a new MySQL database. The database name itself will usually be created for you, otherwise name it something descriptive.

Note: It may take a little time for the new database to be created (Pending Status first).

MySQL Databases Used: 1 of 10		Displaying: 1-1 of 1		
Version	Database Name	Description	Status	Action
4.1	fsjay (Open Manager)	Understanding WordPress Databas...	Setup	 

Figure 4 – Created MySQL database

Editing the wp-config File

Now that all the prep work is done, you're ready to get your hands messy with a little WordPress code.

Look inside the folder that you extracted your files to (Figure 1), and open the file (using [any text editor](#)) called **wp-config-sample.php**.

You should see a bunch of code, along with some very helpful comments.

The wp-config-sample.php file has four sections, which I'll call MySQL settings, KEY, Languages, and "Not for Editing". You only have to worry about the first two.

Note: Remember to enter all your values between quotation marks; otherwise, your install will fail.

```
17 // ** MySQL settings - You can get this info from your web host ** //
18 /** The name of the database for WordPress */
19 define('DB_NAME', 'enter your databas name');
20
21 /** MySQL database username */
22 define('DB_USER', 'enter your user name');
23
24 /** MySQL database password */
25 define('DB_PASSWORD', 'enter your database name');
26
27 /** MySQL hostname */
28 define('DB_HOST', 'enter you host name);
```

Figure 5 – Inside the wp-config.sample.php file

MySQL Settings

- **DB_NAME** is the name of your database.
- **DB_USER and DB_PASSWORD** are where you enter the user name and password of the MySQL database that you created.
- **DB_HOST** is the name of the computer that is running your MySQL database.

Note: Web hosts give you a specific host name, but sometimes it is hard to find. Normally, it found under the Domain settings of your account, if not, a quick email to their support should give you the answer.

- **DB_CHARSET and DB_COLLATE** both have to do with the character set that your MySQL database is using.

You can leave these values alone.

KEY

define('AUTH_KEY', 'put your unique phrase here');

Change this to a unique phrase (anything you want).

define('SECURE_AUTH_KEY', 'put your unique phrase here');

Change this to a unique phrase (anything you want).

define('LOGGED_IN_KEY', 'put your unique phrase here');

Change this to a unique phrase (anything you want).

The KEY section is all about making your installation of WordPress more secure. You may be tempted to skip this section because it optional (WordPress will work just fine if you don't assign three unique values here), but it's a great way to secure your blog that it will be well worth a few seconds of your time.

Setting these keys lets WordPress hash (scramble) those values to make it harder for someone to get any information from the cookies that WordPress uses to save your login information when you login to your WordPress blog. These keys are also used in your MySQL database to make the passwords stored there harder to decipher.

You never have to remember these values again, don't worry. ☺

Languages

```
define ('WPLANG', '');
```

This is where you would identify what language you want to blog to display. I am assuming we want your blog in English. If you wanted another language, you would need to specify this here as well [include a .mo file containing the translation](#).

“Not for Editing”

```
if ( !defined('ABSPATH') )  
    define('ABSPATH', dirname(__FILE__) . '/');
```

```
/** Sets up WordPress vars and included files. */  
require_once(ABSPATH . 'wp-settings.php');
```

The wp-config-sample.php file acts as a repository for settings that another file wp-settings.php uses to do all the heavy lifting of the WordPress installation. If you change any of this code the installation will production errors and not work. So back away, slowly. ☺

Once you're done setting up your wp-config-sample.php file, save it as **wp-config.php** and not wp-config.sample.php.

Choosing Your Blog's URL

Before you upload the files, you have one more thing to think about; your blog's URL structure.

Suppose that you've registered the domain www.understandingwordpress.com, and you plan to host your WordPress blog there.

You have a few options, including these:

- If you plan for your blog to be the primary content of your domain, you can upload the WordPress files directly to the site's root folder. This way, people who go directly to your URL will be greeted by your blog.
- If you plan to have a landing page or some other content living at the root of your site, you can upload the WordPress files to a subdirectory. To get to your blog, people would have to enter the URL like *www.understandingwordpress.com/blog*. (**note:** This example would require you creating a subdirectory called *blog* at the root of the site and then upload all the WordPress files into that subdirectory)

A third option (assuming your host supports it) is to create a [URL redirect](#). A URL redirect allows you to create an alias so when people type in your domain they get “redirected” to a subdirectory in your root folder. So for example, you can create a subdirectory of *blog* in the root of your site, but people would only have to go to *www.ilovewordpress.com* and they would get redirected to *www.understandingwordpress.com/blog* automatically.

Check with your host for more information on how to do this.

Note: If your web host does not support this, you can do this in code quite easily:

In an `index.php` in the root folder, create a file with just this code (change to your domain):

```
<?php
header("HTTP/1.1 301 Moved Permanently");
header("Location: http://www.understandingwordpress.com/blog/");
exit();
?>
```

Uploading the WordPress files

Step 1 (optional): Create a folder on our host account using FTP to put our WordPress files:

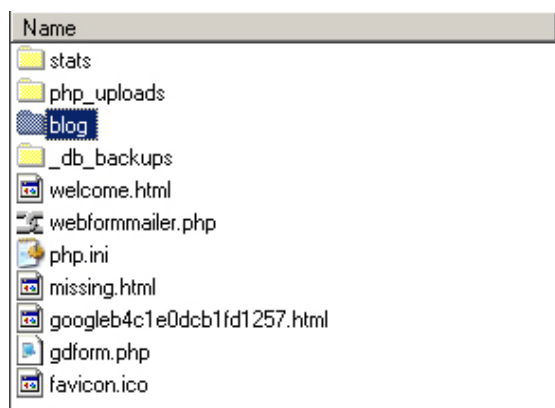
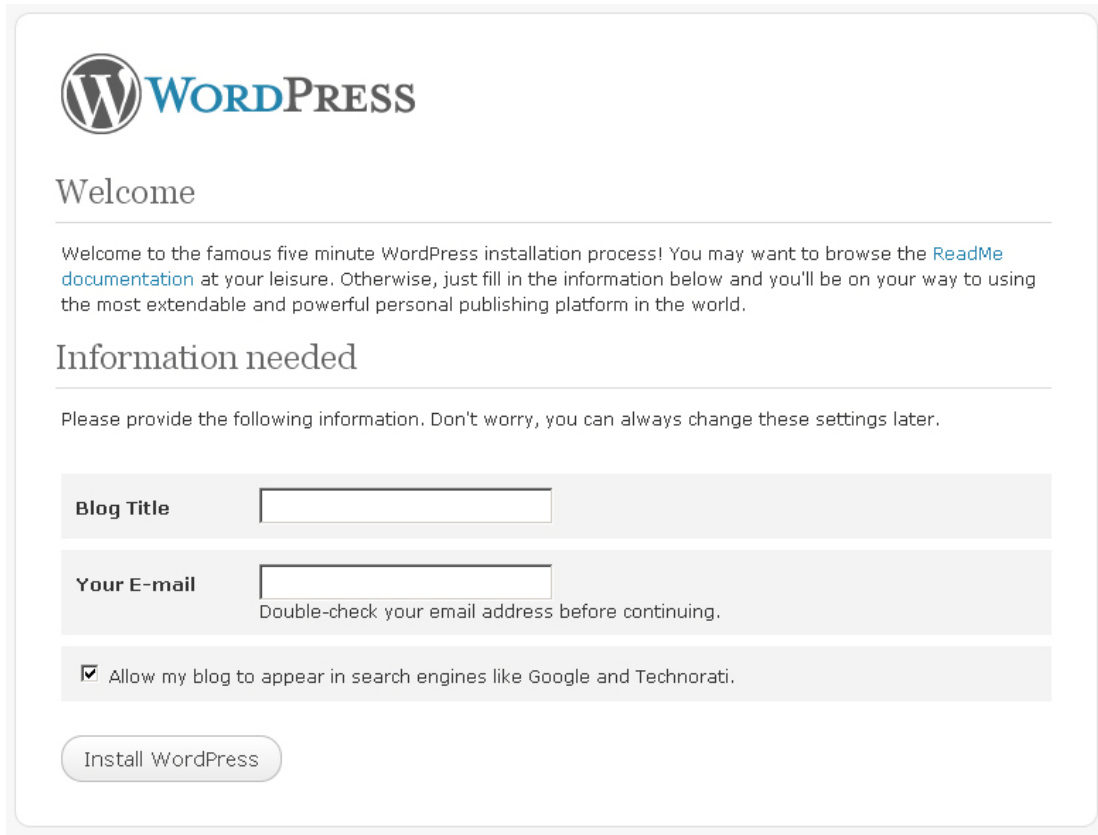


Figure 6 – In my web host, I created a blog folder to put my WordPress files

Step 2: Upload all the files and folders to your web host.

Step 3: Type in your domain (and optional folder) in your web browser. (i.e. **www.yourdomain.com/blog**)

You will be automatically redirected to **http://www.yourdomain.com/blog/wp-admin/install.php**



The image shows the WordPress installation page. At the top left is the WordPress logo. Below it is the heading "Welcome". A paragraph of text follows, mentioning the "famous five minute WordPress installation process" and a link to "ReadMe documentation". Below this is the heading "Information needed". A line of text says "Please provide the following information. Don't worry, you can always change these settings later." There are three input fields: "Blog Title" with an empty text box, "Your E-mail" with an empty text box and a note "Double-check your email address before continuing.", and a checkbox labeled "Allow my blog to appear in search engines like Google and Technorati." which is checked. At the bottom is a button labeled "Install WordPress".

Figure 7 – The WordPress Installation page

This is what you'll see (Figure 7) when you enter the location of all the files you uploaded in your web browser. The installation of WordPress will be complete after you enter these fields and click the button.

Note: Both of these field values can be changed later so no need to worry.



Figure 8 – WordPress Installation complete

You will also get this administrative information (username and password) in an email. Once you log in to your blog you can change it to an easier password to remember.

Step 4: Log into your WordPress blog: www.yourdomain.com/blog/wp-login.php (blog folder optional)



Figure 9 – The WordPress login page

Enter the account information that was emailed to you (if you forgot to jot down that weird password). You will then be transported to the administrative page. You've officially installed WordPress. Congrats! 😊

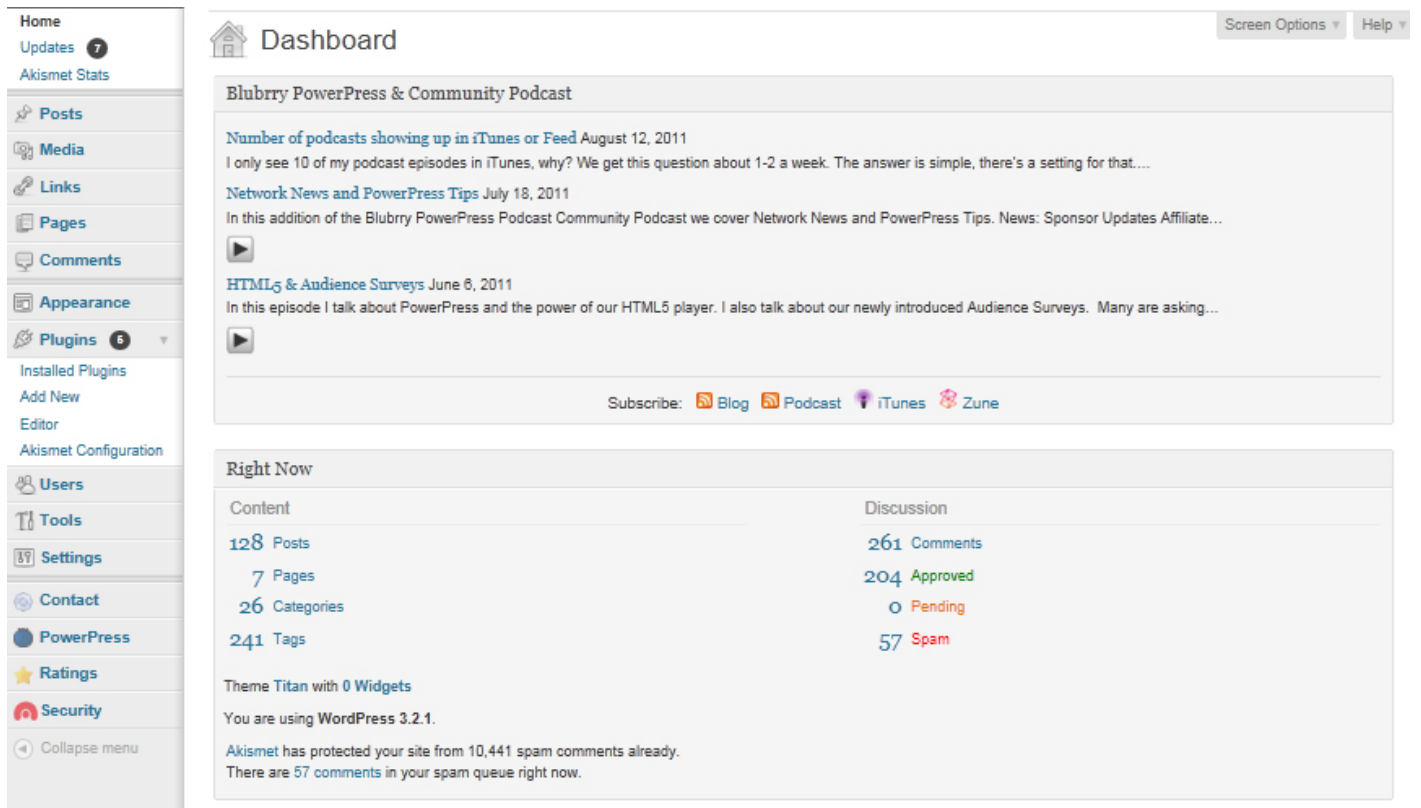


Figure 10 – The WordPress administrative page (Dashboard)

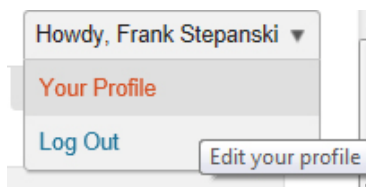
“Wow!”, you might be saying to yourself as you look at this page. I know there is a lot of stuff to take in, but don’t worry I will cover this in the next lesson. But feel free to play around a bit before then. 😊

Note: Every version update of WordPress tweaks the install screens so don’t be worried if yours is a little different than mine.

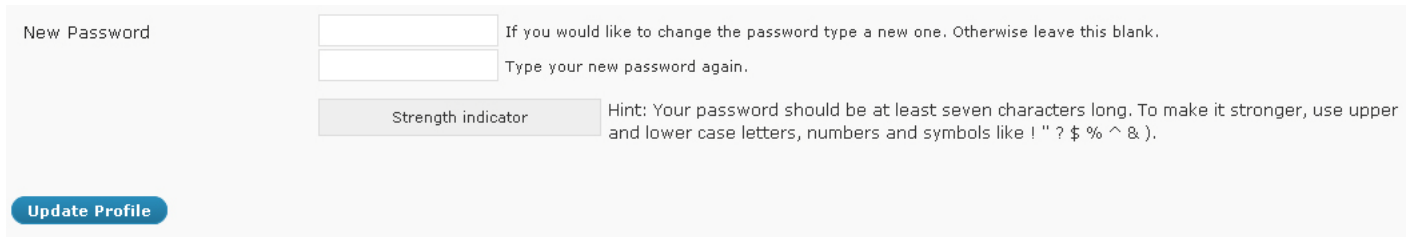
Changing Admin Password

Let’s change our password to our “admin” login to something a little more reasonable than that weird random text.

In the top right corner you will see your name (or the name you gave it during installation):



Click your name and then scroll to the bottom of the page. Here (Figure 11) you can create a new password to the “Admin” user account.



New Password If you would like to change the password type a new one. Otherwise leave this blank.

Type your new password again.

Strength indicator Hint: Your password should be at least seven characters long. To make it stronger, use upper and lower case letters, numbers and symbols like ! " ? \$ % ^ &).

[Update Profile](#)

Figure 11 – Changing the administrative password

Note: If you ever forget your Admin password, you can request a new password by email (Figure 12).



 **WORDPRESS**

Please enter your username or e-mail address.
You will receive a new password via e-mail.

Username or E-mail:

[Get New Password](#)

[Log in](#)

Figure 12 – Forgot your Admin password?

OK, how about we look at our newly created WordPress blog before we wrap up this lesson (Figure 13)?



Best Link to Design and Development Web Sites

Posted on [June 2, 2011](#) by [Frank Stepanski](#)

Keeping track of all your web design and development web sites can be annoying and time consuming. Personally I have a hard time remembering more than a few, and going through my own links no matter what tool I use is still a hassle.

I found one link, that has links to the best sites for learning and keeping up to date on the web:

The Smashing Network:

<http://www.smashingmagazine.com/the-smashing-network/>

Posted in [Web references](#) | Tagged [best site links](#), [web links](#) | [Leave a comment](#) | [Edit](#)

Drupal Speak

Posted on [May 1, 2011](#) by [Frank Stepanski](#)

Archives

- [June 2011](#)
- [May 2011](#)
- [April 2011](#)
- [March 2011](#)
- [January 2011](#)
- [December 2010](#)
- [November 2010](#)
- [September 2010](#)
- [July 2010](#)
- [February 2010](#)
- [December 2009](#)
- [October 2009](#)
- [September 2009](#)
- [August 2009](#)
- [July 2009](#)
- [June 2009](#)
- [May 2009](#)
- [April 2009](#)
- [March 2009](#)
- [February 2009](#)

Figure 13 – Your new WordPress blog

Warning:

In the Admin Panel, under Setting->General you will see this screen:

General Settings Help ▾

Site Title

Tagline In a few words, explain what this site is about.

WordPress address (URL)

Site address (URL) Enter the address here if you want your site homepage [to be different from the directory](#) you installed WordPress.

Figure 14 – General Setting section in Admin Panel

Once you have completed a successful installation, you may need to move your blog to a new location. You may need to move it to a different folder on your existing web host or to a different domain or an entirely different web host.

Do not change the values in either the WordPress address (URL) or Blog address (URL) text boxes.

These issues are not fully covered in this class, but a more detailed information can be found at:

http://codex.wordpress.org/Moving_WordPress

Note: I will review this issue somewhat in the next lesson since many students have asked me this in the past, but I don't recommend doing this until you have gone through the entire course and are comfortable with WordPress first.

Troubleshooting Errors:

Even though you follow these directions exactly, you may find yourself getting an error message before or during the installation process.

One thing to troubleshoot is making sure you **have** and **have full permissions** (777) to the [.htaccess file](#) in your root directory (or the subfolder) you are installing WordPress.

Most web hosts by default give you this access so you should not have any problems, but if you do that is the first thing you should check with your web host. This can also prevent you from modifying your [permalinks structure](#) which is talked about in a future lesson.

Error Message	Common Cause	Solution
Error Connecting to the Database	The database name, username, password, or host was entered incorrectly.	Revisit your MySQL database to obtain the database name, username, and password, and reenter that information.
Headers Already Sent Error Messages	A syntax error occurred in the <code>wp-config.php</code> file.	Open the <code>wp-config.php</code> file in a text editor. The first line should contain only this line: <code><?php</code> . The last line should contain only this line: <code>?></code> . Make sure that those lines contain nothing else — not even white space. Save the file changes.
500: Internal Server Error	Permissions on PHP files are set incorrectly.	Try setting the permissions (<code>chmod</code>) on the PHP files to 666. If that change doesn't work, set them to 644. Each Web server has different settings for how it lets PHP execute on its servers.
404: Page Not Found	The URL for the login page is incorrect.	Double-check that the URL you're using to get to the login page is the same as the location of your WordPress installation (such as <code>http://yourdomain.com/wp-login.php</code>).
403: Forbidden Access	An <code>index.html</code> or <code>index.htm</code> file exists in the WordPress installation directory.	WordPress is a PHP application, so the default home page is <code>index.php</code> . Look in the WordPress installation folder on your Web server. If there is an <code>index.html</code> or <code>index.htm</code> file in there, delete it.

Figure 15 – Common WordPress Installation Problems

Additional Lesson Resources

1. Download latest version of WordPress: <http://wordpress.org/download/>
2. Websites that use WordPress: <http://wordpress.org/showcase/>
3. Blogging basics: [http://codex.wordpress.org/Introduction to Blogging](http://codex.wordpress.org/Introduction_to_Blogging)

Installation

1. Editing wp-config.php: http://codex.wordpress.org/Editing_wp-config.php
2. Instructions: [http://codex.wordpress.org/Installation#Detailed Instructions](http://codex.wordpress.org/Installation#Detailed_Instructions)

After Installation

1. Directory Structure: [http://codex.wordpress.org/Giving WordPress Its Own Directory](http://codex.wordpress.org/Giving_WordPress_Its_Own_Directory)
2. Moving WordPress: [http://codex.wordpress.org/Moving WordPress](http://codex.wordpress.org/Moving_WordPress)

Troubleshooting

1. htaccess: [http://codex.wordpress.org/htaccess for subdirectories](http://codex.wordpress.org/htaccess_for_subdirectories)
2. File Permissions: [http://codex.wordpress.org/Make a Directory Writable](http://codex.wordpress.org/Make_a_Directory_Writable)

Assignment for Lesson 1

1. Read over this lesson.
2. Create your own blog.
 - a. Decide on what type of content you want to put in your blog as well as a domain name that matches that type of content.
 - b. Sign up for a web hosting account (if you don't already have one)
 - c. Create a MySQL database
 - d. Download the WordPress files and modify the **wp-config-sample.php** as described.
 - e. FTP the WordPress (you downloaded earlier) files to your web host.
 - f. Install your WordPress blog.
 - g. Change your Admin password.
 - h. Create your first real post.

Copyright 2011 © Frank Stepanski

Lessons, files and content of these classes cannot be reproduced and/or published with out the express written consent of the author.