

GETTING TO KNOW
WORDPRESS

JASON YINGLING

| | |
|----------------------------------|----|
| A Brief History of WordPress | 3 |
| The Community | 6 |
| How WordPress Works | 10 |
| Where WordPress Runs | 14 |
| The Dashboard | 18 |
| What Are Posts | 21 |
| What Are Pages | 24 |
| Media | 26 |
| Comments | 30 |
| Themes | 33 |
| Customizing WordPress Appearance | 37 |
| Plugins | 41 |
| Users | 46 |
| Settings | 50 |
| Tips, Tricks, and Tools | 61 |
| Outro | 65 |

A Brief History of WordPress

The goal of this book is two-fold. I'm hoping to give people an introduction to what they get with a brand-spanking-new WordPress website, and improve my own skills at explaining the inner-workings of WordPress. So join me and let's understand WordPress together.

What is WordPress?

WordPress can be many things. It's a blogging platform. It's a content management system (CMS). It's an open source software. It's an ecommerce platform. It's a social network platform. It's a web application platform. It's a service as a software solution (SaaS). And the list goes on. But what is WordPress to you? Chances are if you're reading this you're looking to use WordPress to build your website.

To you WordPress is a tool you're using to create a website. Whether it's a site to sell products, market yourself, or post pictures of your cat in a variety of costumes, WordPress can get it done.

Let's look at how WordPress.org explains WordPress on it's landing page.

"WordPress is web software you can use to create a beautiful website, blog, or app. We like to say that WordPress is both free and priceless at the same time."

WordPress certainly isn't the only tool you can use to build a website, and it also isn't the only tool you'll need to accomplish all of your goals. Except for the aforementioned cat costume site. It can handle that right out of the box.

Oh yeah, and you may have noticed that "free" line in the description. That word tends to jump off the page at people. WordPress is free to use. It's an Open Source project maintained by the people that use it. That means there are thousands of developers across the world working to make the WordPress Core software as feature-rich and bug-proof as possible.

That's a big reason why if there is something you want to do in WordPress, chances are someone in the WordPress community has already done it, tested it, documented it, and created the plugin.

Who really maintains WordPress?

Surely with software as sophisticated and complex as WordPress there has to be someone in charge of it. [WordPress.org](https://wordpress.org) is completely 100% free under a GPL license. You'll see GPL licenses throughout all kinds of open source software projects. In simplified terms, it is essentially a license that let's you modify and use software in anyway you please personally or commercially.

But you've been to [WordPress.com](https://wordpress.com) and you've seen paid options.

That's actually run by a company called Automattic. They were founded by one of WordPress's founding developers Matt Mullenweg to make the WordPress core software easier to use for the masses. They own and maintain [WordPress.com](https://wordpress.com), but they don't own [WordPress.org](https://wordpress.org). Confused yet? Let's break that down.

The Flavors of WordPress

As you look into WordPress you're likely to come across two different versions. The [WordPress.com](https://wordpress.com) version and the [WordPress.org](https://wordpress.org) version. [WordPress.org](https://wordpress.org) is where you can go to download your own copy of the WordPress core software and install it on any web host you choose. This also offers the flexibility to edit it, add themes, add plugins and tweak it to your hearts content.

Sound complicated? It can be if you're not familiar with installing websites on servers and don't have the desire to learn. Enter [WordPress.com](https://wordpress.com). [WordPress.com](https://wordpress.com) is essentially hosting specifically for vanilla WordPress sites. You can sign up and have a working site in minutes. With the free version your site URL would look something like yourdomain.wordpress.com.

The best way to think about it is [WordPress.org](https://wordpress.org) is self-hosted WordPress. It's the most feature rich version of the WordPress core software and it's 100% free. You can take it and repurpose it in anyway for your own fun and profit.

[WordPress.com](https://wordpress.com) is a stripped down version of WordPress limiting your theme and plugin options. [WordPress.com](https://wordpress.com) was created to offer a quick and easy way to set up a WordPress site and get started blogging. [WP Beginner](https://wpbeginner.com) offers a nice comparison of [WordPress.org](https://wordpress.org) vs a free [WordPress.com](https://wordpress.com) site.

Still want the flexibility of [WordPress.org](https://wordpress.org), but don't want to use [WordPress.com](https://wordpress.com). No problem. With the proliferation of WordPress across the internet many hosting providers offer one click installs for WordPress sites. There are even hosting providers that are tuned to run WordPress sites exclusively, which often leads to better overall site performance and security.

Check out the official [WordPress web hosting providers](#) on [WordPress.org](#) for a few good options. I use SiteGround personally and WP Engine professionally. I've had good success with both.

Who Uses WordPress?

Excellent question! As of this writing something like 26% of the web uses WordPress ranging from huge brands like CNN and Beyonce (Queen B!) to unknown brands like View From The Pine. And WordPress is used in a variety of ways from ecommerce to web games to social networking. To see more uses of [WordPress check out the Showcase](#).

And soon to be you!

The Community

As you continue your dive into WordPress, you'll likely come across the term "The Community". As much as it sounds like a sci-fi totalitarian regime, it really is not. It's a group of developers, designers, and WordPress users who are just trying to make WordPress work for their particular business or personal needs. Let's break down who makes up the community in a way that everyone is familiar. Lunch room tables!

Let's find where you sit within the WordPress cafeteria, but don't worry you'll likely sit at many tables in your time with WordPress.

The Starters (General Users)

When you walk into the cafeteria with your rectangle pizza and juice box, the first set of tables you'll come across are the general WordPress users. You'll find all sorts of interesting people at these tables. This includes people that have got a WordPress site and a theme that they're using to blog or run a pretty basic, no-frills site. These people are content creators that are using WordPress as a way of getting their content on the internet for all to see.

Some General Users:

People using only the core functionality of WordPress. You may find someone like Sally at this table. Sally runs a blog posting pictures of her cats doing cat things (napping and being a jerk) to share with her family and friends.

The Power Users

As you continue to explore the WordPress cafeteria you'll come across the next set of tables. The Power Users.

These are WordPress users that use the WordPress.org software to run more complicated setups.

Let's say Sally from the General Users group above has a post of her cat pushing things off a table go viral and she wants to capitalize on it by selling merchandise. She may install WooCommerce and some other plugins to turn her simple blog into a fully-fledged ecommerce store. She may need to get a new, more advanced theme setup too.

Suddenly Sally is doing a lot more with WordPress and has turned her small photo-blog into a business. She's become a WordPress Power User. The problem is she may not know how to do a lot of that herself, and may be looking to hire some help.

Power Users to Follow:

Chris Lema - @chrislema

Post Status - @post_status

Dan Norris - @thedannorris

The Customizers

The next set of tables you'll come to are The Makers. These are the people who use WordPress to build solutions for their clients. They can be designers, developers, content creators, or all of the above. They can also vary in size from a freelancer putting in a couple of hours at night, to fully-fledged development companies with staffs and building WordPress sites for many clients.

This group uses WordPress as a tool to meet client needs in a more efficient manner. By building off of WordPress, The Customizers can create completely unique WordPress builds that don't function like an out of the box WordPress site at all, or they could be brought in to make modifications to an existing theme or plugin. The Customizers are those that customize WordPress to fit their specific needs.

Customizers to Follow:

Red8 Interactive - @red8interactive

Tom McFarlin - @tommcfarlin

Matt Report - @matterport

The Contributors

Finally as you get to the center of the room you'll come across another group of tables. This group is The Contributors, and it can be broken down into a couple smaller groups.

The Extenders

First you've got your plugin and theme authors. These are the people that build the tools that can be used to extend WordPress and push it from a blogging platform to a full content management system.

They do so by building plugins and themes that can be installed on your WordPress site to add extra functionality. Things like ecommerce, extended security, portfolios, galleries, sliders, SEO, and the list goes on and on. These users create products that are both free and paid. Chances are if you have an idea for something you'd like to do on your site, you can find a plugin or theme that already supports it.

Extenders to Follow:

Plugins - Pippin Williamson - @pippinsplugins

Themes - Carrie Dils - @cdils

The Core Contributors

Then right in the middle of the cafeteria you've got the Core Contributors. These are the users who spend their time solving bugs and adding features to the open-source WordPress software. They're not even paid to do it!

Every time you see that little orange dot with a "1" in it on your Updates page, that's the core contributors making WordPress faster, safer, and more feature rich for you. Every tool you use when you download WordPress is the result of countless hours spent by the Core Contributors to make WordPress great. So if you see one give them a hug, or a coffee because they've probably been up all night trying to solve a bug.

Core Contributors to Follow:

Helen Hou Sandi - @helenhousandi

Aaron Jorbin - @aaronjorbin

Outside the Cafeteria

You may be wondering how you can get involved with the WordPress community outside of our hypothetical cafeteria (which is actually pretty real at WordCamps). The good news is the WordPress community is one of the biggest friendliest web communities. Outside of a few bad apples here and there everyone in the WordPress community is willing to lend a helping hand whenever they can. Here's a few venues where you can dip your toes into the community.

Local WordPress Meet Ups

Find your local WordPress meet up on [Meetup.com](https://www.meetup.com). There's a good chance that if you live anywhere near other people there is a WordPress meet up within driving distance of your home. These are typically groups that come together weekly or

monthly to chat about WordPress and share their knowledge with presentations or open discussion.

WordCamps

WordCamp can be one of the best introductions to WordPress. These weekend long conferences are often less than \$40 to attend and typically get you two days of speakers and lunches. Plus tons of swag. This is probably the best return on investment you can get as far as conferences go.

Speakers often come to WordCamp's from across the nation and world. Check the upcoming WordCamp list to find one near you.

Not so keen on attending right off the bat? Some WordCamp's offer live-streaming tickets of their lectures now too. But you'll miss out on the hallway track. Where a lot of the best learning happens.

Slack Groups

Slack is the new hotness among messenger apps, and it has proven to be very useful for giving communities a place to chat. If you're interested at all in contributing to WordPress the Make WordPress Slack channel is a must. Even if you just creep in the background for a few months on the Wednesday core meetings (me).

I've also been a fan of the Post Status Slack channel. There's a lot great WordPress people in there, and it's very active throughout the day. It does require a subscription to Post Status, which is totally worth it if you're a WordPress power user.

Finally, the St. Louis WordPress community has it's own Slack channel that's open to the public. We're just growing it at this point, but hoping to get a wider audience. There may be other local meet ups with their own Slack channels as well. Contact the organizers to find out.

Now that you're familiar with the roles of the WordPress community we can start delving deeper into how WordPress works.

How WordPress Works

WordPress began its life as a blogging platform, but has since morphed into fully-fledged Content Management System (CMS from here on out). What exactly is a CMS, you ask?

A content management system is an application (WordPress core software) or set of programs (think plugins) that work together to allow a user to create and manage digital content.

That's nice and all, but how exactly does WordPress take that text about "10 Potato Chips That Look Like Abraham Lincoln," and display it to a user across the world when someone goes to the URL buzzfeed.com/10-chips-lincoln? How does WordPress maintain that content and display it. There's a few layers to this story.

The WordPress Core

The WordPress Core is where much of the base content managing happens. The Community has already taken it upon themselves to create the structure for creating and saving content. At its base, WordPress is a PHP application that saves data to a MySQL database. Which is why those are the only two real [requirements of WordPress](#).

The Database

When you first download WordPress and perform the famous 5-minute install, the software creates a SQL database that contains 11 database tables. These tables store data for site settings like the default URL (wp_options), user information (wp_users & wp_usermeta), post content, images, and post metadata (wp_posts & wp_postmeta), category and tag information for posts (wp_terms, wp_term_relationships, wp_term_taxonomy), comments on posts (wp_comments & wp_commentmeta), and of course the deprecated links feature (wp_links). Although it can be turned back on.

The table that is likely to get the most use in your WordPress install is the wp_posts table. This is the table where all of your content is stored. Anytime you create a post in the WordPress backend, the software is creating a new entry in the wp_posts table with a unique numerical identifier. The post ID.

This is where things can get a bit confusing due to the terminology WordPress carries over from starting as a blogging platform. Any bit of content created is

typically referred to as a post. When you hit Add New under the Posts tab in the WordPress dashboard you are creating a post of type Post. When you hit Add New under Pages, you are creating a post of type Page. Confused yet? Just wait until we throw custom post types into the mix. The main takeaway is that Post, Page, and Custom Post Type data is stored in the `wp_posts` table.

At its most basic, when a user and a CMS love each other very much, the user enters data into the CMS and, after several edits, a post is formed. That may be the worst metaphor ever.

For more information on what exactly is stored in each table of the WordPress database check out [the Codex](#).

How a Post Becomes a Webpage

At this point you've used the WordPress software to add data into a SQL table somewhere on a server. Your content is thoroughly managed, and ready to spring forth into the the world. It may seem difficult, but don't worry the WordPress core comes fully stocked with functions for delivering your content to the web.

You Lost Me At Function

Functions are blocks of code that perform a specific task. WordPress has a function for nearly anything you could imagine. Want to get the posts from the database? You could use the [get_posts](#) function. (That's not necessarily the best way to do it, but it is an easy example to grasp the terminology and purpose.) If you're editing your theme and you want to put a post or page title in a certain spot of your HTML you could use the [the_title](#) function. That function gets the title of the current post and outputs it onto the page. That current post is determined by where we are in The Loop.

The Loop, There It Is

Ah the Loop. The ever beating heart of WordPress. This is a nifty little PHP function created for one thing, and one thing only. Getting your content from the database and allowing it to be displayed onto a PHP based webpage. There's a bit more to it than that, so let's break it down.

"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." - [WordPress.org](#) Codex

(https://codex.wordpress.org/The_Loop)

The Loop is an if / while loop. Meaning it first checks *if* we have any post data in our database. Then *while* we have post data it performs any HTML or PHP we have within the the while for each individual post. That allows us to use predefined functions from WordPress for displaying the title or post content, or even create our own functions to run.

```

1 <?php if ( have_posts() ) : while ( have_posts() ) : the_post(); ?>
2   <?php // post content goes here ?>
3 <?php endwhile; else : ?>
4 <p><?php _e( 'Sorry, no posts matched your criteria.' ); ?></p>
5 <?php endif; ?>

```

The first part of the Loop is an if statement. It is using the `have_posts()` function to determine if your current WordPress query has found any posts in your database. If there are posts, `have_posts()` will return true. Making our if statement pass and move into our while statement.

The while statement also checks if we `have_posts()`. Any code within the while statement will run once for each post found. Essentially if we have 5 posts returned, `have_posts()` will return true for all 5 posts, then switch to false and end our while loop. If we didn't have that condition our while loop would never end and we'd crash the browser.

The last function you'll see in The Loop is [the_post](#). This is function gets the next post and sets up its data. This little function allows WordPress to know we're in The Loop and properly pull in data from template tags like `the_title` and `the_content`.

Make Me Pretty

Now that we know how WordPress handles data we need to get it displayed on our site so a user can see it.

The theme handles displaying WordPress content. It is the coat of paint on the WordPress data. When you first install WordPress you'll be active with the newest twenty[year] theme that ships with the software. If you didn't have a theme, you would have nothing to display to a user.

Most of the editing you'll do to alter WordPress for your site will be in the theme, but there is another area of expandability, plugins.

Plugins extend the WordPress core functionality. Want to sell something on your site? You would use a plugin like WooCommerce or Easy Digital Downloads to get

that done.

There are nearly 50,000 plugins available on the [WordPress.org](https://wordpress.org) repository that do everything under the sun from securing your site to allowing you to sell products. Plugins can range from full blown applications to a couple lines of code for adding a message to the admin dashboard.

That's the basics of how the database, WordPress core, themes, and plugins work together to allow you to create and display content to anyone with an internet connection. If you want to dig even deeper into the inner workings of WordPress check out *Up and Running* by [WPShout](#). They have a fantastic metaphor explaining [WordPress as a factory](#).

Where WordPress Runs

In the last section we briefly touched on the requirements for running the WordPress.org software. The two main requirements are PHP 5.6 or greater and MySQL 5.6 or greater *OR* MariaDB version 10.0 or greater. MariaDB in its most basic sense is a newer version of MySQL. Check the MariaDB site for a [more thorough comparison](#).

There is also the server. We need a server to run PHP and MySQL on. Any sever configuration that can run those two services will do, but WordPress recommends either Apache or Nginx.

Apache has been around forever, in terms of the internet, and in doing so has benefited from years of documentation and development. If you're looking to setup your own server, and unless you're a server tech I don't recommend it, Apache is the quickest setup to get WordPress working on. That's just in my limited server setup experience. See the server tech line.

Nginx on the other hand was developed to handle some of the issues Apache has had trouble dealing with. Particularly with handling tens-of-thousands of concurrent requests to the web server. Nginx is gaining popularity among WordPress hosting due to its ability to scale to larger traffic websites on minimal hardware when compared to Apache. Basically, it's faster.

For a more detailed comparison of Apache vs Nginx check out this article from [Digital Ocean](#).

Not to say that Apache can't be fast. A good server configuration can get a WordPress website running silky smooth on Apache or Nginx. That's why we're going to focus on hosting for the rest of this section. But before we do a little bit on self-hosting.

Self-Hosting

If the developer in you wants full control over your server configuration self-hosting is the way to go. Of course, you'll be on the hook for fixing any kind of server issues outside of the server farm going down. Unless you host your own server farm. Then you're on your own and probably aren't reading this anyway.

My experience with self-hosting has been hit and miss. It's been relatively easy to get things setup thanks to pre-made server configurations for WordPress available

through Digital Ocean and Amazon Web Services. But after the setup, fine-tuning things and tracking down issues has proven much more difficult and time consuming.

The price certainly can look good on a self-hosted server, but if you're not sure what you're doing the time-cost lost to fixing issues and debugging certainly makes that a little less appetizing. I've switched all of my sites off Digital Ocean at this point due to just that.

That being said Digital Ocean and Amazon can make for great hosting which is affordable and very scalable. Amazon provided some of the best server speed I've experienced on a site. Once we got it to a point it wasn't going down every other day due to the configuration.

If you go self-hosted be prepared to spend a lot of time learning server configuration, or have someone on hand that you can trust to be there to get your site back up in the middle of the night. I'd recommend the latter.

General Hosting vs Managed WordPress Hosting

General Hosting

General Hosting providers are going to be your Bluehosts, your GoDaddys, and your SiteGrounds. These are hosts that provide servers for all kinds of web applications. Not just specifically WordPress. Though all 3 of these hosts do have pseudo-WordPress hosting. Meaning they offer quick and easy setup for hosting WordPress sites and many of them have tools for easily configuring things like WordPress caching, auto-updates, and more.

SiteGround and Bluehost are both in the [WordPress recommended hosts](#). GoDaddy has been the red-headed step child of the developer community for some time now. However, they're very active in the WordPress community and all of the GoDaddy-ers I've met at WordCamps have been very knowledgeable. So things there may be on the up swing.

The general host's "WordPress" specific hosting tends to be more their regular hosting dressed up in WordPress branding for marketing. But this pseudo-WordPress hosting represents a nice bridge between self-hosting and managed WordPress hosting as it gives you a little bit more control over certain areas of your server, and includes some tools designed for WordPress.

I personally use SiteGround to host my sites.

Typically that is done through the cPanel. If you've hosted a website before, you've probably dealt with the cPanel. And if you're not really a developer you probably haven't used much in the cPanel aside from File Manager. That's where managed WordPress hosting can help simplify things.

Managed WordPress Hosting

Managed WordPress hosts are popping up all over the place. These are hosting providers that offer servers fine-tuned to host WordPress and WordPress only. If you know you're going to use WordPress, I'd highly recommend getting a managed WordPress host. They're a bit more expensive than general or self-hosting. But that extra money gets you a server that is specifically tuned for WordPress and support agents that deal with WordPress issues everyday, 24/7.

The user experience is often cleaner in the user portal for managed WordPress hosting, as they don't have to bother with as many of the features cPanel requires. Getting setup with SFTP, accessing your database, and more tends to be a breeze with managed WordPress hosting.

The biggest reasons to go managed WordPress though are the speed benefits of having a server configured only for WordPress and the support. Having a helpful support team that is available 24/7 is a life-saver when problems arise on your site. General hosting will have 24/7 support too, but they deal with all kinds of issues not related to WordPress sites.

Some managed WordPress Hosts:

- WP Engine
- Pagely
- FlyWheel
- Pantheon

The Final Word

Your WordPress hosting choice really comes down to what you're looking for as a user or business. If you don't want to worry about server configurations and know you're using WordPress, then managed WordPress hosting is likely the way to go. If you want to set up your server from the ground up, then self-hosted is your best bet. If you land somewhere in the middle you may want to consider a general WordPress host.

For more information on picking the right kind of hosting for you, check out this

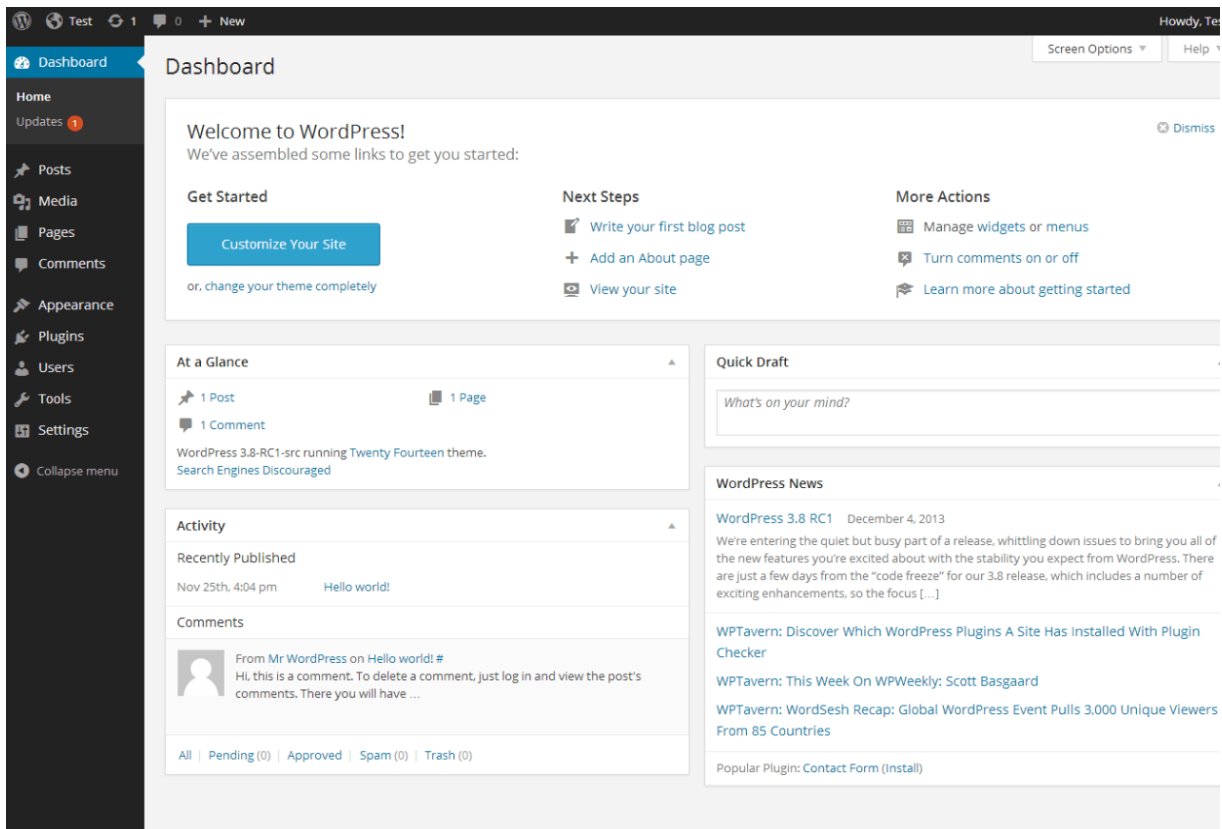
[article by Chris Lema.](#)

The Dashboard

After you've completed your famous 5 minute install and logged into WordPress for the first time you'll land on The Dashboard. Also known as the the admin screen, the backend, the underutilized on-boarding area. Ok, that last one is more of a personal term. Essentially this is the gateway to creating content in your WordPress website.

The Dashboard is technically a certain section of the more encompassing [Administration Screens](#), but for keeping things simple we can refer to the backend as the Dashboard. Administration Screen just doesn't have as nice of a ring to it.

When you land at the Dashboard you'll see an admin screen broken into 4 sections.



At the top is the Toolbar. Formerly known as the Admin Bar. It has context sensitive links dependent on what screen you're viewing and is persistent for logged in users across the backend and front-end of your site. If you're viewing a specific post there will be links for jumping directly to the edit screen of that post. Or if you're editing a post in the backend you'll see a link to view that post on the front-

end.

On the left you have the main navigation which you can use to navigate to your Posts, Pages, Users, Settings, and more. Plugins and Themes can both add their own menu items to the main navigation, or even add sub-menu items to existing menu items. All of this is available with easy to use [WordPress functions](#).

At the bottom of the page is the footer. This consists of a link to WordPress and your current version number.

The right side of your Dashboard is the work area. This is where any information is displayed tied to the particular menu item you have selected. Select “Posts” and this displays all of the posts on your site.

When we first land on the Dashboard the work area displays what are known as widgets. WordPress comes loaded with 5 of these dashboard widgets, but plugins and themes can easily add their own as well. These are different than the Widgets you’ll see under Appearance in the main navigation. Those are for your sidebars, and we’ll cover them in more detail soon. The starting 5 dashboard widgets are:

At a Glance

This widget displays some basic information about your WordPress site such as the number of posts, pages, and comments. It also displays your current WordPress version and the currently active theme.

Activity

The Activity widget shows you the most recent posts on your site as well as the most recent comments and controls to moderate them. Feel free to mark your enemies’ comments as spam with ease.

Quick Draft

The Quick Draft widget is perfect if you’ve got an idea for a post so pressing you don’t have time to click Add New under Posts in the main navigation.

WordPress News

Wonder what has been published recently on the [WordPress.org](#) blog? Wonder no more! The WordPress News widget puts these links just a click away from your Dashboard.

Welcome

This is the WordPress on boarding screen. In this widget you’ll find links to Customize your site, make your first post, create a page, manage widgets, and my

personal favorite “Dismiss”.

I troll a bit, but the WordPress on boarding process could see some improvement. Smarter people than me are working on it as we speak. With the REST API coming into WordPress I expect this area to see significant improvement in coming versions of WordPress.

All of the widgets on the dashboard can be easily rearranged, collapsed, and removed via “Screen Options” at the top of the screen.

The Dashboard menu item in the main navigation also houses the Updates work area. This is where you can easily see all of the updates available for your plugins, themes, and WordPress core. Keeping everything up to date is one of the best and easiest things you can do to keep your site secure. Check this area often and perform the updates that are available. Do not fear the update.

Now that we know the WordPress Dashboard we’ll continue our journey following along with the main navigation. Which means next up: Posts.

What Are Posts

WordPress began its life as a blogging platform. That means the terminology it started with is rooted in the blogosphere. As WordPress has developed and grown into a CMS, a lot of that early terminology has hung on. Which means some of that terminology can get very confusing.

Posts in WordPress can mean a couple things. First, posts (purposely all lowercase) could be considered any entry (row) in the `wp_posts` table. That's really any type of content that gets put into WordPress. Posts, Pages, Custom Post types. Those entries are all stored in the `wp_posts` table and can be considered posts.

[Up and Running](#) has the best explanation of this I've yet come across. It suggests thinking of each entry as a post of a certain type. For example, creating a Post in the main navigation of the Dashboard would be making a post of type Post. Notice the purposeful capitalization of the post type. In the previous example "Post". In this sense you would also have a post of type Page. If you create a custom post type for Reviews, an entry there would be a post of type Review.

It is a useful way of thinking about posts vs Posts and how WordPress stores data. If you're still confused, and I don't blame you, check out [Up and Running](#).

Now let's dig into what makes up a post of type Post.

A default WordPress post of type Post consists of the title, the permalink, the content, categories, tags, featured image, and post formats. You'll see these fields on your first WordPress post.

The Title

The title is easy enough. It is your post's title. Plain and simple. No tricks to this one. Set your title and WordPress automatically will set your permalink to a web-safe version. But don't worry you can still edit it.

The Permalink

This is your post's address on the web. The URL a visitor will see in their browser when they visit your site. If you would like you can spruce up the format a bit through Settings > Permalinks in the Dashboard's main navigation.

The Content

This is the meat of your post. Anything entered inside the TinyMCE WYSIWYG (What You See Is What You Get) editor becomes part of the `_content`. I've written

the_content like that for good reason. That's the function name you use within your theme to pull in this data.

It is also the filter themes and plugins will use to add items before or after the_content. Those sharing plugins that allow for social sharing buttons to be added before or after your post? Those are hooking into the_content filter to prepend or append content as needed. One of the many things that makes WordPress so flexible.

Categories & Tags

Categories and Tags are known as Taxonomies within WordPress. While the term may seem commonplace to seasoned WordPress vets it isn't that common among the average user. So what is a taxonomy?

"Taxonomy is one of those words that most people never hear or use. Basically, a taxonomy is a way to group things together." - [WordPress Codex](#)

Essentially this is how we can structure and group data together within WordPress. It allows for us to create archives of our posts related to certain key terms. Which can be either a Category or a Tag.

Categories are a hierarchical taxonomy. Meaning they can have parent and child relationships. For example, you could create a Movie Category that has sub-categories such as Review or Preview. Categories are typically more broad than Tags. As a general practice it's best to limit to 1 or 2 categories for a post.

Tags are non-hierarchical. They're more free-form. Using our Movie example above a tag could be the actors in the film. Say you're writing a review of Point Break, you could use the Tags Keanu Reeves, Patrick Swayze, or 90s Surfer Bank Heists. Ok that last one may be a bit too specific, but you get the point. They tend to be more freeform than Categories.

One note about Tags: They aren't going to give your post significant Search Engine Optimization (SEO) juice. For the same reason Google has lessened the value of meta keywords in the past, which is you can just make up whatever you want there. Instead of trying to use terms for the Google Bot, focus on using terms that are best going to improve the user experience. That will help your site more in the long run.

Post Formats

The last item we'll cover with our posts is Post Formats. Post Format is actually another taxonomy, like Categories and Tags, that can be used for presenting content

to the user. They allow theme developers to create custom layouts for the different post formats available within a theme.

While WordPress supports the aside, gallery, link, image, quote, status, video, audio, and chat formats a theme may not make use of them. In this case a good theme developer would remove those options. But that isn't always the case.

Put all of that together and you have yourself your first WordPress post.

A Note on Custom Post Types and Custom Taxonomies

What makes WordPress great is its ability to expand. One such area for that is Custom Post Types. These allow for further content organization by allowing a means for creating entirely different sets of data. Want to create a Portfolio section separate from your blog? A post type of Portfolio could be an extremely useful tool for building that.

Custom Taxonomies then allow for the creation of new Taxonomies that can be tied to one or many post types. This allows for the data structure of WordPress sites to get very complex, but also very useful for managing content.

What Are Pages

Let's jump out of order going down the WordPress Administration main navigation real quick and cover Pages instead of Media.

Pages are the very similar to Posts. Using our "post of type" phrasing, Pages would be a post of type Page.

Posts and Pages differ in the content they are meant to display. Posts are for content that is intended to show up chronologically in a feed. Pages are intended for more static content that isn't as time sensitive.

For example, a Post is likely to be used for making an announcement about a new product your company is releasing. Where a Page would be used to display your company's mission statement and history.

Aside from content there are a few other areas that Pages differ from Posts.

Page Hierarchy

Unlike Posts, Pages are allowed to have a parent-child relationship with other pages. This is useful for grouping page content as well as creating URL structures that show this relationship. Let's say you have an About page on your site that you would like to break down further into more detailed pages. You could create a History page and set it's parent to the About page. WordPress would then give the History page the URL domain.com/about/history. This also groups the parent and child pages together in the Pages Administration Screen.

Page Templates

This section is only relevant until WordPress 4.7 comes out in December 2016 when custom templates will be added to Posts.

Currently, only Pages can have custom templates. These are PHP files created in the theme that allow for Pages to display content in different ways. Developers can use these templates to give the user different layouts to choose from when creating a Page. A basic example would be a template for a Page layout with a sidebar and another for a layout without a sidebar. However, the customization can get much more sophisticated than that simple example.

Template Hierarchy

In order to determine what PHP file to use to display content when custom templates are used, WordPress relies on the template hierarchy. Essentially this is the order in which WordPress looks for specifically named PHP template files when certain content is accessed by the user.

The WordPress template hierarchy is complex enough for it's own post. For a quick primer check the [Codex](#).

Taxonomies

Pages do not come with Category support out of the box. Pages instead rely on the parent-child relationship for organization. Pages also have an Order attribute, which can be used to control the order in which pages are displayed when using functions like [wp_list_pages](#).

That's a brief introduction to Pages, also known as, posts of type Page.

Media

Now that we've covered Posts and Pages we can move back up a spot on the Dashboard main navigation to Media. Also known as the Media Library, this is where WordPress stores all of the files you upload through the Media Uploader.

Let's break that down a little bit.

Media

The term Media is pretty generic. In WordPress, Media, is actually a post of type Attachment. The post type holds data such as the description, name, and url of the file. It also allows the uploaded file to be tied to metadata stored in the wp_postmeta table. For file uploads like images that could include dimensions and EXIF data[*note*]The bits of information stored in the image file, like location, time, and that fun stuff. We don't need to get into that at all.[/*note*].

One thing you'll notice about Attachments is that the work area you land on is different than what you've seen for Posts and Pages. This area is called the Media Library.

The Media Library

When you first click on the Media navigation item you'll land on the Media Library Grid View. This view of your Attachment posts requires Javascript to function properly. Essentially it shows you all of the images and files you've uploaded to your WordPress site in a nice thumbnail grid view. Switching to List view gives a layout closer to that of Posts and Pages, which just isn't as useful for view images. Though it can help when scanning your images looking for a specific name.

The Media Uploader

In order to upload any new files to WordPress you'll use the Media Uploader tool.

Click Add New on the Media work area and you'll be presented with an area to drag and drop a file, or a button to browse your computer. Once you've selected a file and begun the upload process WordPress handles everything else, including cropping the image to predefined sizes set by themes, plugins, or the default [WordPress image sizes](#).

Attachments can also be inserted on the individual Post editing pages through the Media Uploader. You'll see a button for it right above your `the_content` WYSIWYG editor.

Any attachment uploaded through the Media Uploader is going to be stored within the "uploads" directory in your WordPress install. This can be found at the path "/wp-content/uploads".

Editing Attachment Information

Uploading attachments isn't the only thing you can do from the Media Library. You're also able to edit file meta information to make your site more accessible and edit the actual image's dimensions or rotation.

Attachment Details

Selecting an Attachment in the Media Library will open the Attachment Details panel. This panel will display information about the Attachment such as the file name, file type, date uploaded, file size, dimensions, and URL of the file on your server.

You can also edit the image title, caption, alt text, and description.

Title

The image title is descriptive text that provides additional information about the image. This can be used by WordPress to sort images in the Media Library which makes searching for images easier on you once you have hundreds of images in your library.

Alt Text

The image's alt text, or alternative text, is displayed when the image doesn't load for a user. The alt text allows you to give information about what the image is for users that may be on a bad connection where images aren't loading or maybe they have stopped images from loading. Alt text is also used by screen readers to help visually impaired users navigate your site.

Caption

The caption can be used to display information directly below an image. This can be used to explain the image and provide detail to help the user understand what they are seeing.

Description

Since Attachments are a post type they can have their own single attachment page. The description allows you to enter any information you find relevant to the picture that you would like displayed on the attachment page.

Common Problems

Media Upload Size Limit

When uploading a new attachment you may notice a line that says something like “Maximum upload file size: 32 MB.” This upload file size limit will vary based on your hosting. For 90% of use cases the default limit on your server will be enough. However, it can be a bit of a problem if you’re uploading video or audio files. Even large images can be an issue, but if you’re uploading images larger than 1 MB and you aren’t a photographer offering full-size originals I would suggest you resize. That’s a lot of data to force on your poor user, especially on a mobile device.

There is a fix though. It involves editing your php.ini file or your .htaccess. If you’re not familiar with either it is best to contact your hosting provider and ask them to increase the upload size limit.

If you’re on shared hosting of any sort you most likely won’t have access to edit your php.ini file, but if you do you’ll want to find the lines ``upload_max_filesize`` and ``post_max_size`` and increase them.

If you don’t have access to your php.ini you may be able to increase the file size through your .htaccess file. The .htaccess file is what WordPress uses to rewrite URLs when you have pretty permalinks on. Editing the .htaccess can be dangerous. I’ve taken many a site down by forgetting a single character in an .htaccess.

If you’re ok with the risk (you backed it up before editing, didn’t you?) you would want to add the lines ``php_value upload_max_filesize 64M`` and ``php_value post_max_size 64M`` replacing “64M” with your desired upload size in MB. Typically I put these at the bottom of the .htaccess outside of the WordPress section.

Permissions Errors

The always fun permissions error. These errors are great because WordPress really doesn’t give you any information as to what is happening. You may just get an “Upload failed” message. Fun!

This can typically happen if you’ve installed WordPress through FTP. Essentially

the owner of the files is the FTP user instead of the server which causes the WordPress software to not be able to properly write files to the uploads directory. Some hosts provide an option to Reset File Permissions (WP Engine for example), other hosts you may have to contact, or if you have SSH / Terminal access you can run the following command:

```
`chown -R www-data:www-data /var/www/public_html`
```

In the example above “www-data:www-data” is setting the user and group (user:group). Wwww-data is commonly used for the server user:group, but you may need to check what your server uses. That is a bit more complex than we’ll be getting though. Also in the example “/var/www/public_html” is the root directory of the WordPress install. You’ll need to replace that with your server’s path to your WordPress install.

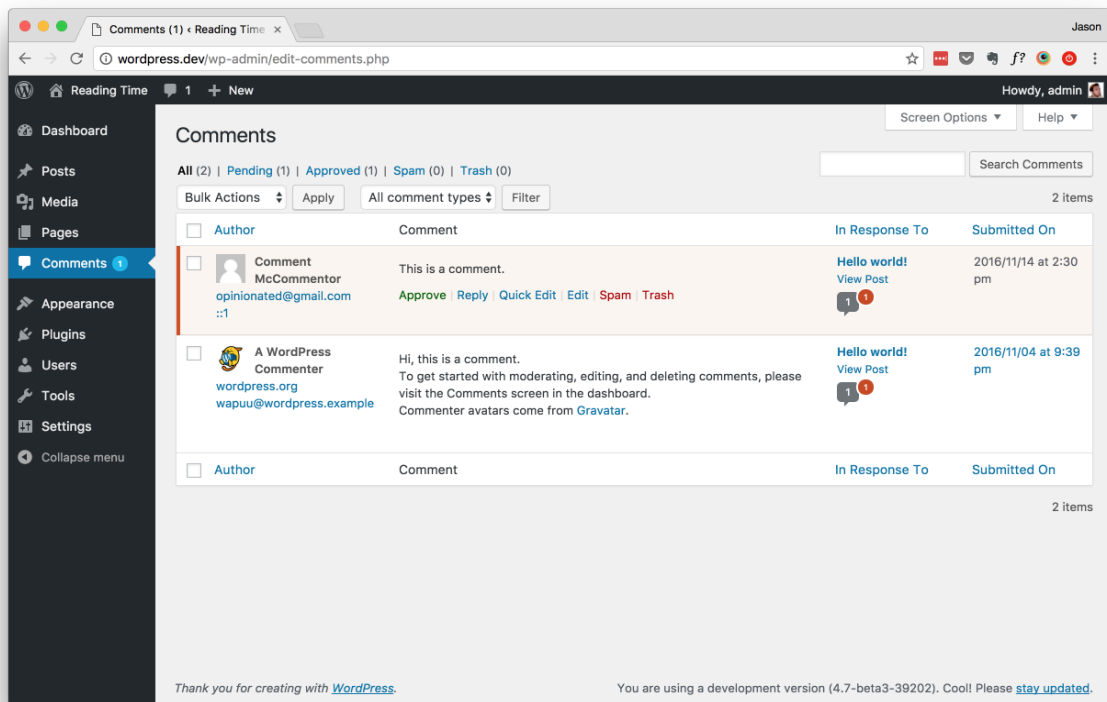
Hopefully now you have a better understanding on what exactly the WordPress Attachment post type is. Next we’ll move on to Comments.

Comments

Comments can be a great way to engage with your audience in an open discussion on the topic of an article. They can also be the bane of the content creator's existence. See most YouTube comments or any comments on a political post.

By default, WordPress comments will need to be approved before they show under your content. This is to prevent the ever dreaded spam. And watch out for spam. You're going to get spam.

Luckily WordPress comes with plenty of settings and tools for making sure the discussion on your site doesn't turn into bots and trolls battling each other for supremacy.



Moderating Comments

When a new comment is posted to your site you'll see a circle with a number in it next to "Comments" in the main navigation of your Dashboard. That indicates how many new comments you have pending approval.

Once in the Comments screen, hovering over any comment will give you some

links to Approve, Reply, Edit, mark as Spam, or Trash the comment. Replying will let you approve and submit a reply in one fell swoop. Way to go! You're a mod now! Redditors hate you!

Protecting Against Spam

WordPress comes with some built in settings to help prevent comment spam. A lot of the settings will be found under Settings > Discussion from your WordPress Dashboard main navigation. By default WordPress will hold any comment with more than 2 hyperlinks in queue for moderation. It's typical for spam comments to contain multiple links. WordPress will also require a user to have a previously approved comment for any following comments to be auto-approved.

And by default users will be required to enter a name and an email. Of course neither of those have to be real, but it can help. You can make it easier or more difficult for spam to get through by playing with the options available at Settings > Discussion.

WordPress also comes with the Akismet plugin installed by default. Akismet runs each comment to your site through a web service to better detect if the comment is spam or just a poorly worded argument. You'll need to activate the plugin and get a free API key to take advantage of Akismet, but it can go a long way to keeping your comment section squeaky clean.

The #\$%@ is a pingback?

You may notice you get some comments that are identified as pingbacks. Essentially pingbacks are a way of alerting you that your post has been linked to on another WordPress site that has pingbacks enabled.

Think of it like this. You write a post and link to your friend's post. Both of you have pingbacks enabled. When you publish your post with your friend's link in it, your friend will receive a pingback comment.

Essentially this is a way for tracking the discussion of your post when it is done off of your site. Of course it only works if both sources have ping backs enabled.

Disabling Comments

There may be the case where you don't want to moderate comments or allow pingbacks. Luckily disabling all of this can be done through a few WordPress settings.

Pingbacks are easy. Under Settings > Discussion uncheck the option that says “Allow link notifications from other blogs (pingbacks and trackbacks) on new articles”.

Comments is a bit more difficult. You have to check two boxes if you already have any open posts. Under Settings > Discussion you’ll want to uncheck “Allow people to post comments on new articles”. This will prevent comments from being open by default on any future posts on your site.

If you already have posts you’ll also want to check “Automatically close comments on articles older than __ days” and set it to 1 day(s). That way any post older than 1 day will have its comments closed.

What happens if you have a great article down the road and want to enable comments only for that post? Easy. When editing the single post click “Screen Options” at the top of the page and make sure “Discussion” is checked. Down below the content editor you’ll see a Discussion box with options to “Allow comments” and “Allow trackbacks and ping backs on this page”. Check which ever you’d like to enable and you’re good to go!

Now that we know all about creating and moderating content within WordPress we can get into how to fancy it up with the theme.

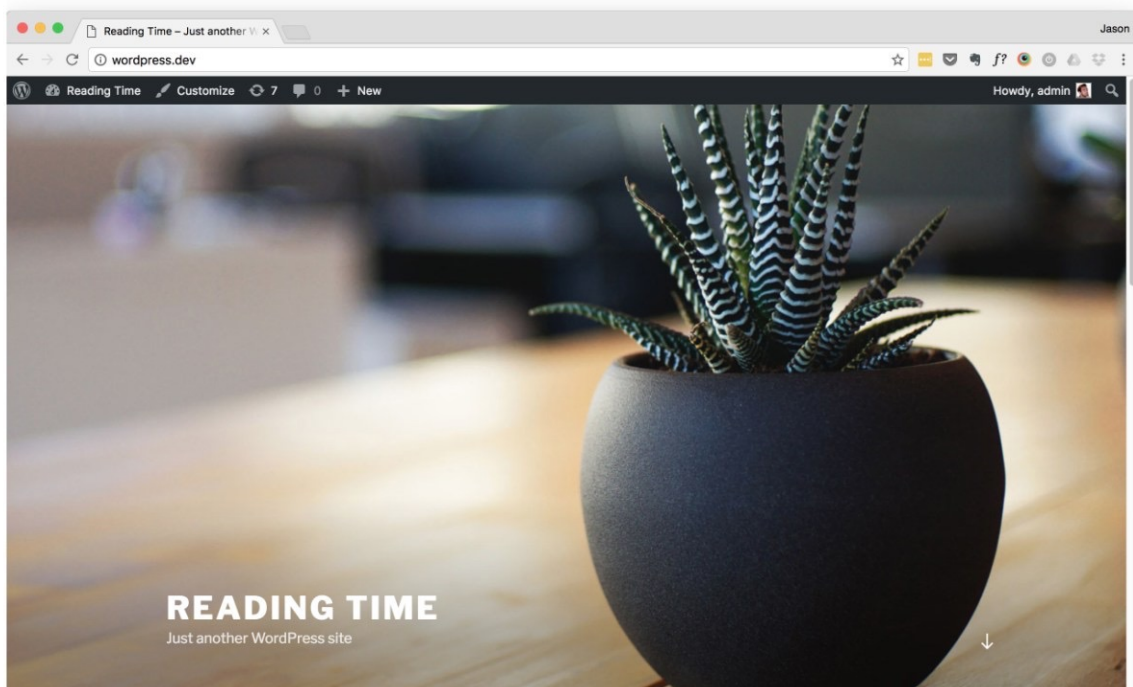
Themes

As we move down the WordPress Dashboard we'll come to the Appearance tab. Which brings us to our theme.

The theme is the face of your WordPress install. When you start up a fresh WordPress install you'll be greeted with one of the twenty{something} themes. As of this writing that would be twentyseventeen. You'll also have a few other twenty{year} options under the Themes menu, but what exactly is a theme.

What is a Theme

The WordPress theme is the face of you site. It's the combination of PHP, CSS, and JavaScript files that works together to make your WordPress site come to life.



Your WordPress theme could consist of only an `index.php` and `style.css` file. With `style.css` file declaring your theme name and `index.php` containing your WordPress loop. Your site is going to be pretty lame, but it could be done.

A theme could also be as complex as having several different PHP files that combine to form a single PHP template for a post.

A typical theme is likely going to have at least the following files:

index.php - The main loop for displaying posts on the site.

single.php - The main layout for a single post of type Post.

page.php - The main layout for a single post of type Page.

style.css - The styles and theme name declaration.

functions.php - This file is options, but can be used for adding functionality to WordPress sites. More on this shortly.

Template Hierarchy

How WordPress determines which files from the theme to use to display certain post is known as the template hierarchy. This is a set of rules WordPress uses to determine which PHP files to look for first when outputting a post.

It can get pretty complex. For now we just want to understand the concept of the template hierarchy instead of delving to deeply into which templates follows which. If you'd like to dig into it yourself check out the [Codex](#). For now just know that everything falls back to index.php.

Functions.php

Most themes you will come across will make use a functions.php file. This file acts like similarly to a plugin, but can be used to add features and functionality specific to a theme. The functions.php file is good for adding Javascript libraries for certain theme functionality via the `wp_enqueue_scripts` action. Or adding theme support for things like custom logos or the new-fangled video header functionality[[note](#)]As of WordPress 4.7.[/[note](#)].

I'll use this point now to tell you that you shouldn't put Custom Post Types or Custom Taxonomies in the the functions.php, or anywhere else in the theme files for that matter. Unless it really is specific to that particular theme.

Why? Because if you switch themes, suddenly that Post Type isn't registered anymore and you lose all that data from the WordPress dashboard. It would still be in the database, but you wouldn't have a way to edit it in the backend as WordPress wouldn't recognize it as a Post Type anymore.

Of course there are two sides to every matter. WP Beginner does a good job laying out the differences between [registering Post Types in your theme vs. a plugin](#).

Child Themes

Child themes are themes that pull in the styles and functionality from another parent theme allowing you to modify the parent without editing files in the parent theme directly.

This is extremely useful if you're making modifications to a theme that receives updates from the theme developer. If you were to edit files directly in the parent theme directory, anytime you would update the parent theme you would lose all of your changes.

A child theme allows you to create a separate file directory with a style.css declaring the child theme name and parent theme template, and a functions.php used for enqueueing the styles from the parent theme.

You can then duplicate a file such as single.php from the parent theme directory and edit it to your hearts content in the child theme directory. WordPress is smart enough to know to find the template file in the child theme first.

If down the road the theme developer issues an update that improves the single post display, you could delete your child theme's single.php and it would revert to using the parent theme's file.

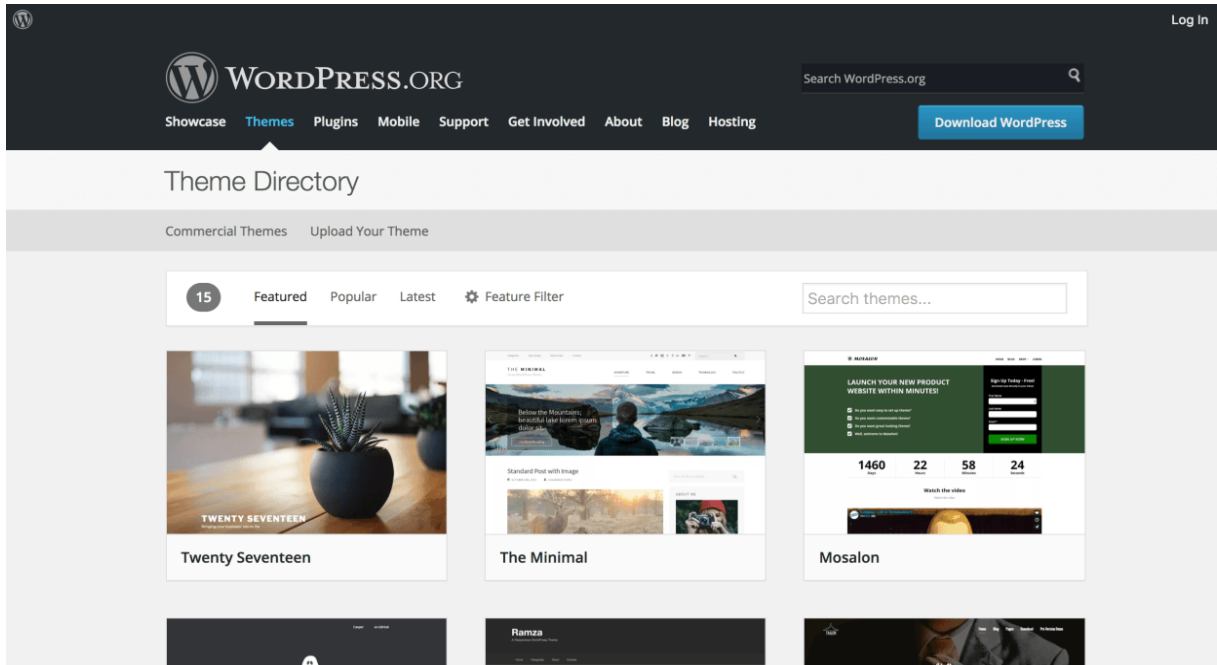
Neat!

Where to Get Themes

The "twenty" themes are those that come with the Core WordPress software, but there are a few other flavors of themes: free and premium.

Free Themes

WordPress.org has a wide variety of [free themes](#) to choose from. They can range from pretty great to pretty not great. But the theme repository does have standards a theme has to meet to be included, so for the most part all of these themes should get the job done. These won't have a lot of features though. For that you'll want to check to Premium Themes.



Premium Themes

WordPress is great because anyone with an internet connection and some know how can create a custom theme. That means there are tons of WordPress themes available on the web for purchase at various prices.

Themeforest and other marketplaces have driven a culture of including every bell and whistle in a theme for next to nothing costs. Which causes developers to focus on adding more features than the quality of features in a lot of these themes. I find themes tend to work much better when developed for a specific purpose. The common user probably ends up using a small percentage of the features in these do-it-all themes.

Not to say there aren't some great themes on Themeforest. There is just a lot of noise to sort through to find a quality bug-free theme. If such theme as a bug-free theme exists.

One of my favorite theme developers is [Array Themes](#). You can really tell the thought and time that goes into developing each of their themes to fit specific segments of the theme marketplace.

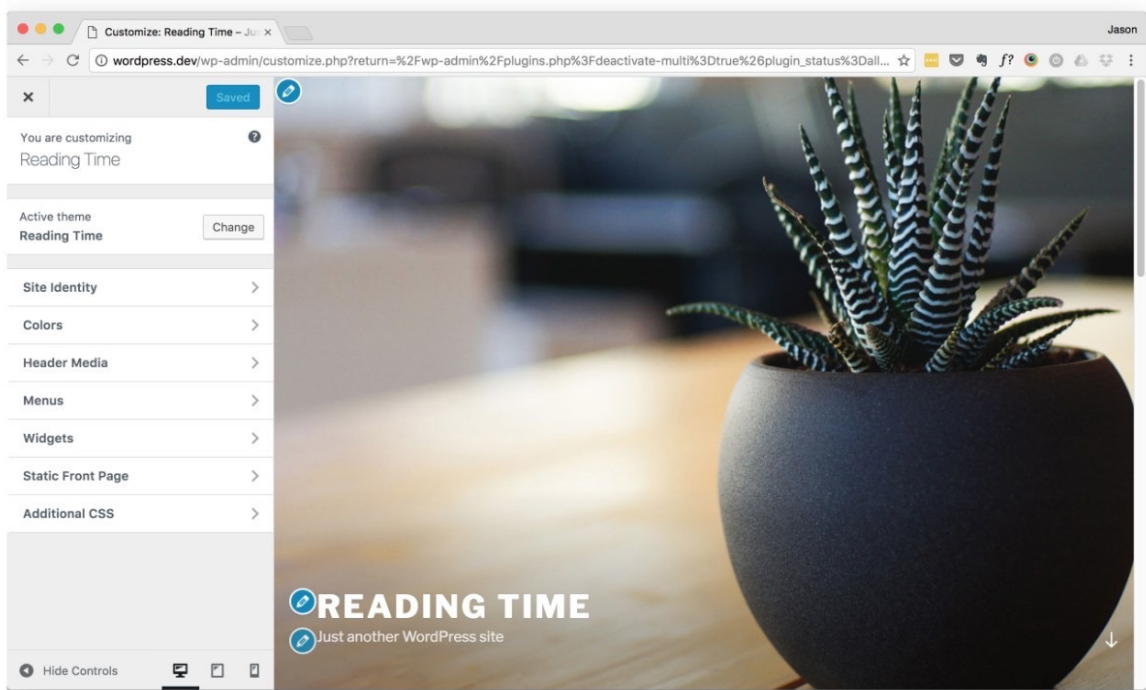
I've also developed a theme for podcasting and blogging named [The Pine](#). I use it for my personal site[[note](#)]As of writing in December 2016[/[note](#)], and another sports blog I help run for fun on the side.

Customizing WordPress Appearance

One of the biggest benefits of running a WordPress site is the ability to customize it and make it yours. WordPress gives us a few tools for making your site your own.

The Customizer

The Customizer is what those in the know call the [Appearance > Customize](#) screen within the WordPress dashboard. By default you can edit the Blog Title, Tagline, Menus, and Widgets within the Customizer. But the true strength of the customizer is in the API.



Customizer API

The Customizer API is a framework allowing developers to add options for editing themes into the Customizer screen. What you can edit depends on the theme. Some themes may let you control the color of every single element on the page. Some themes may only let you select from a set of preset color schemes.

No one Customizer is the same on a theme-by-theme-basis. In any new theme it's good to take a look and see what type of customization panels your theme

provides you.

Widgets

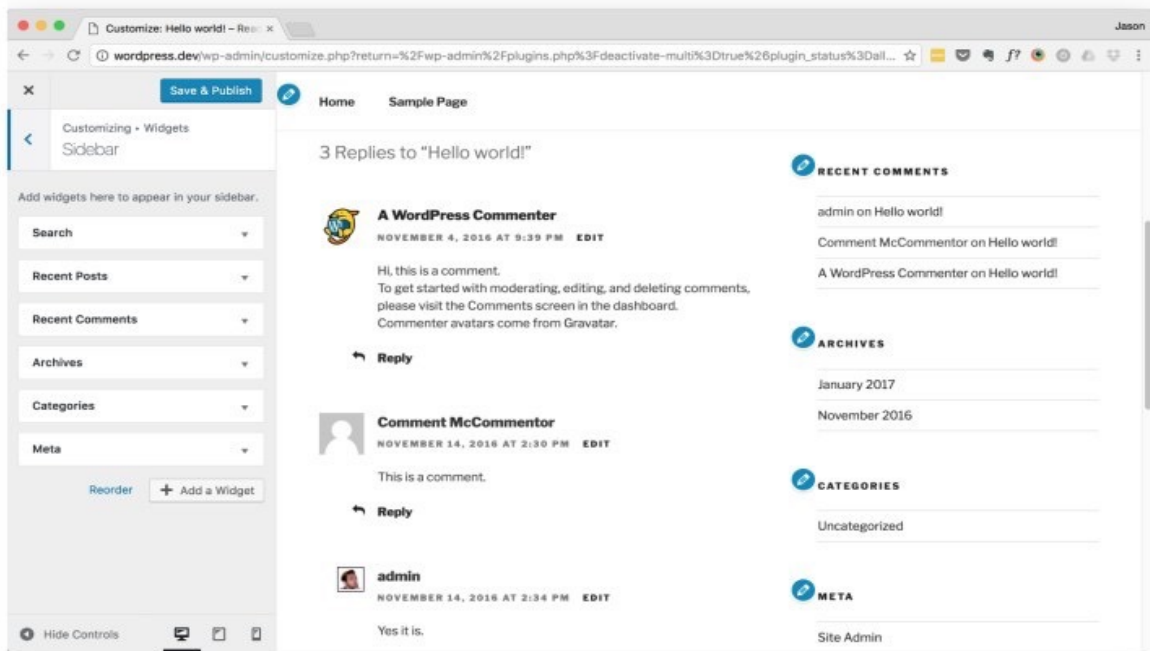
WordPress widgets allow you to add content and features to widgetized areas. In your typical theme a widgetized area will be sidebars. However, many themes put the power of widgets to use in other areas throughout the site. A theme could include widgets in the footer or before and after posts.

Where widgets are used is dependent on the theme. In the backend you'll see separate panels for each existing widgetized area. These widgets will show up throughout your site anywhere the widgetized area is included in the theme.

Fret not though dear user! There are plenty of plugins out there that offer more fine-grained control. [WooSidebars](#) is one example and there are more great examples on the [Plugin Repository](#).

WordPress comes with a few widgets out of the box. There's a widget for displaying a search input, your recent posts, your blog's categories, and even a calendar of your posts. Each of these widgets typically have options for things like widget titles, how many posts to display, or extra info to show.

Not enough for you? Well, WordPress comes through again. It is easy for developers to create their own widgets. Often these will be included in plugins. WooCommerce offers plenty of widgets specific to running an ecommerce store.



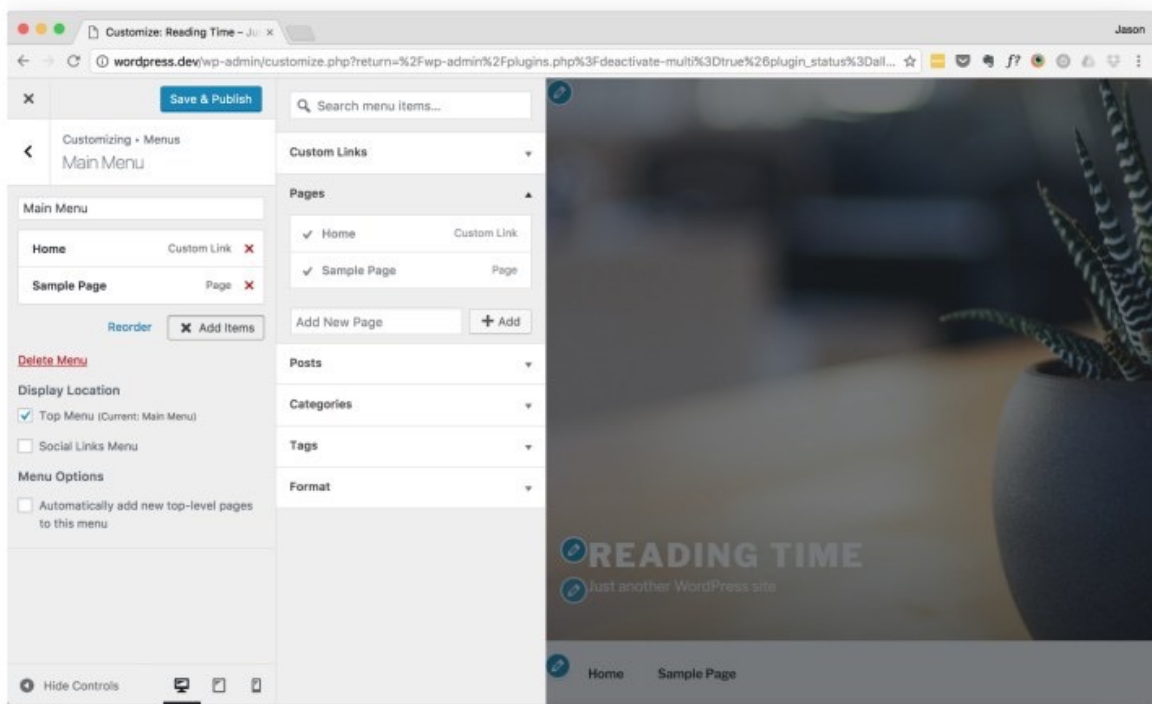
Menus

Almost every site consists of more than one page, and if you're running a WordPress site there's a good chance that includes your site. WordPress allows you to add various Pages or Posts to your navigation.

Again this is another item that will depend on the theme for placement. Most of the time you're going to have a header menu and a footer menu. Though the latter isn't always true.

When you navigate to Appearance > Menus or the Menus tab on the Customizer, you'll want to create a new menu. Once you've defined your menu you can add links to it and select a theme location for it. The locations available will depend on the theme.

When adding links to WordPress menus you'll be given options for the navigation label, title attribute, CSS classes, and description. If you're editing from the Customizer you should see all that, if you're editing from Appearance > Menus you may need to check the Screen Options tab and activate CSS classes and the sort.



A Note On The Editor

There are a couple other options you might see under Appearance in the dashboard. Header which will drop you into the Customizer to edit the Header where available. The other option you may see is the Editor. I don't recommend using this. As a matter of fact I recommend developers turn it off for their site's or use plugins like iThemes Security that can turn it off for you.

The Editor is just what it says, an editor. But this particular editor let's you edit code in the theme file. This could really only be useful if you don't have access to edit the code directly through a text editing program. You're skipping out on benefits of testing code locally or in a staging environment first. Oh and you could also white screen your entire site by forgetting to add a ";" in the right spot. Avoid this.

That's a few of the ways you can customize your WordPress install directly within the Dashboard.

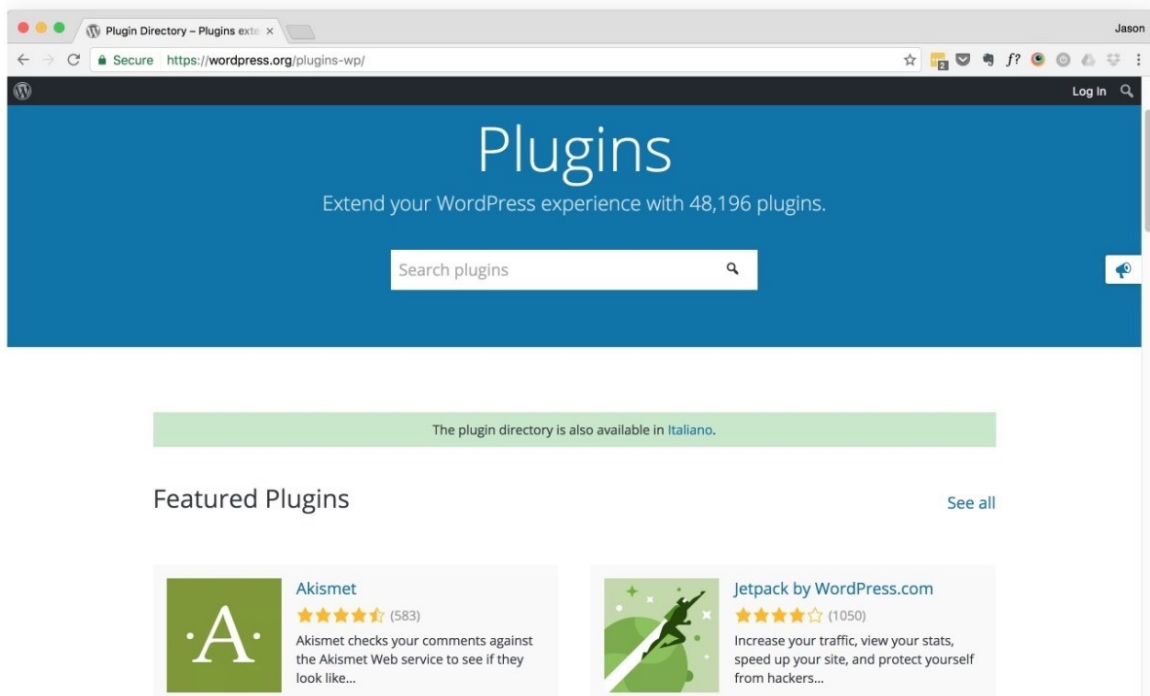
Plugins

What is a WordPress Plugin

WordPress plugins are bits of code that can range from full scale applications to just a few lines of code. They allow you to extend and add to the WordPress core functionality.

The WordPress core is purposefully designed to be as lightweight as possible. Not everyone is going to need a full-fledged ecommerce system. Including one would just add bloat to site's that don't use it. Plugins allow for developers to create new functionality. This even allows for developers to cater their plugins even more specifically to their target.

For example, there are several ecommerce plugins available for WordPress. All offering different features. Two of the more popular are WooCommerce and Easy Digital Downloads. WooCommerce is a full-fledged ecommerce system with shipping built in. Easy Digital Downloads on the other hand focuses on exactly what it's name implies. It offers an easy way to set up a digital download store. No extra bloat from shipping info you wouldn't need.



The Default Plugins

A fresh WordPress install comes with two plugins installed. Akismet and Hello Dolly. Akismet is a useful plugin for detecting and managing comment spam on your site. Hello Dolly is, well, it was the first official plugin for WordPress. It adds a lyric from the song *Hello, Dolly* to the upper right of your Administration Panels page. Feel free to delete it.

There's a few things WordPress doesn't do out of the box that you might be interested in. Here are some of the go to plugins I like to use.

Go To Plugins

SEO is very important in building a strong audience for your website. [Yoast SEO](#) is one of the leader's in SEO for WordPress sites. This plugin makes it easy to make sure you're getting the most out of your keywords.

Forms are pretty important on any website and WordPress doesn't have any form functionality built in. [Contact Form 7](#) gives an interface for building forms that can be easily set to deliver to any email address you prefer. There are some premium plugins that are pretty reliable in this field too such as [Gravity Forms](#), [Ninja Forms](#), and [Formidable Forms](#). Most have free options available as well.

Ecommerce is a big area for plugins. Two of the market leaders mentioned earlier were [WooCommerce](#) and [Easy Digital Downloads](#). WooCommerce is probably the most popular ecommerce option on WordPress and will likely continue grow. Easy Digital Downloads is very developer friendly and while focused on digital products it does have extensions for adding in shipping. Both plugins operate under a freemium model. Meaning you get the core feature for free, but things like additional payment processors or added functionality are available through paid extensions.

One of my personal favorite plugins is Advanced Custom Fields. This plugin is more for developers and allows for creating easy to use fields to store custom data within Posts and Pages. It also comes with a variety of functions for displaying that data within a theme. You'll need to be familiar with PHP to get the full power of this plugins. But without it I may not have ever dug as deep into WordPress as I have now.

There are many, many more plugins available with WordPress. Want to start a podcast? I'd recommend using PowePress by blubrry. Want the most bang for your free download? Jetpack offers a lot of useful features for sharing your content on

social networks, adding galleries to your posts, site security, and much more.

How to Pick Good Plugins

WordPress's open source nature is both a blessing and a curse. Mostly a blessing, but being open source does allow for anyone with enough know how to make a plugin and submit it to the WordPress repository. This is great for getting almost any functionality you could think of covered, but it isn't always great for things staying up to date and free of security issues.

Picking good WordPress plugins is something I've [written on in the past](#). But here are a few simple guidelines to help you get the best plugins available from data available in the [plugin repository](#). Note: no plugin is perfect. Even the best get hit with security issues.

Active Installs

This shows the number of WordPress sites the current plugin is active on. The higher the better for this as the more people that use it, the more likely the developers are to keep up support on it. These will even begin to develop their own ecosystem with users building extensions to the plugins on their own.

| | |
|------------------|------------|
| Version: | 4.1 |
| Last updated: | 7 days ago |
| Active installs: | 1+ million |
| Tested up to: | 4.7.1 |

Tags:

| | |
|-----------------------|-----|
| Content analysis | |
| google search console | |
| Readability | seo |
| xml sitemap | |

Ratings

[See all >](#)



| | | |
|---------|--|-------|
| 5 stars | | 7,478 |
| 4 stars | | 252 |
| 3 stars | | 92 |
| 2 stars | | 71 |
| 1 star | | 400 |

Actively Supported

There are a few other ways to determine how actively a plugin is supported. The repository gives you info on when the plugin was last updated, what version of WordPress it has been tested up to, and even the number of support requests that have been solved on the forum in the last two months.

Rating

This one is a no-brainer. The higher the rating the more useful all of the others using the plugin have found it. If you're looking at a plugin with over 1,000,000 active installs and it's sporting a rating of 4+ stars, it is likely a pretty good plugin.

Premium Plugins

If you're looking into premium plugins you won't have as much data to go off of. Although a lot of premium plugins may have a free version on the repository so check that out. Otherwise check out the support forums for the plugin, if it has one and see what people are saying about it. Honestly, you can judge some premium plugins by their cover. If they've got a superb site there is a good chance they're

also putting a lot of effort into their actual plugin. Don't let that be the only factor though.

For more info on picking good plugins and troubleshooting them if they break check out [my article that goes more in depth](#).

Users

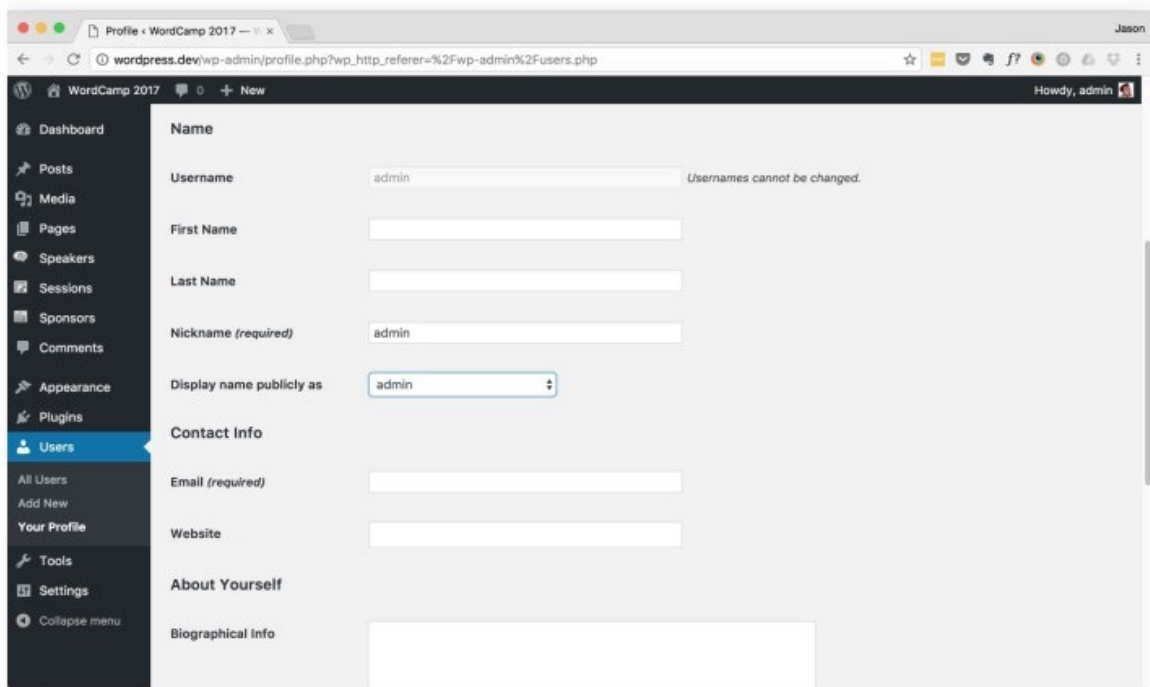
Our journey through the WordPress Administration Screens continues to the [Users Screen](#).

Users in WordPress can take on a great many roles. A user can be the site administrator, an editor, or someone that simply subscribes to the blog. The Users screen is only going to deal with people that have registered for your site.

Exactly what a User is allowed to do on the site is determined by roles and capabilities. By default WordPress comes with several roles with varying capabilities. Ranging from the most capable to the least capable are: Administrator, Editor, Author, Contributor, Subscriber. There's also a Super Administrator role once you dig into multisite setups, but we won't get into that today. Let's break the roles down a bit.

Administrator

The Administrator role has capabilities to do everything within WordPress. An Admin can edit all posts on a site regardless of the author. They can add and delete plugins, change themes, and manage theme options. Typically it's good to try to keep only 1 administrator on a site. As there is less chance for someone malicious to gain access.



Editor

Editors can publish and edit posts including those written by other users. They lack the access to edit things like plugins and themes. This is a great role for someone that is only needing to be able to edit posts, but shouldn't need to change site options.

Author

The Author role can publish and edit their own posts, but not those written by other users. This role is useful if you're going to have users that can write their own posts and you would also like them to be able to publish the posts on their own.

Contributor

The Contributor role can write posts, but they cannot publish them. When a contributor writes a post instead of "Publish" they'll see a button to "Submit for Review". This allows you to give users the ability to create new posts, but to have them only be published by an Administrator or an Editor.

Subscriber

This role only has the ability to manage their profile. They don't have access to do any editing on the site.

Other Roles

Many plugins include their own roles and capabilities. WooCommerce for example creates a Customer role to store account information for users that make purchases on your site.

You can also create your own roles within WordPress using plugins like [User Role Editor](#), or using the [add_role function](#). Before you dig into that, you'll want to know a little more about capabilities.

Capabilities

Capabilities are the powers that make up a user role. There are different capabilities for creating posts, editing posts, editing others posts, and much more. This allows you to create unique user roles that provide each role with specific access to the different features of WordPress. You could create a moderator role that only has access to moderate and edit comments. The possibilities are endless.

You can also create custom capabilities. Lots of plugins have their own capabilities as well, or you can make them with the [add_cap function](#). You can then check if the current user can perform that capability in order to return specific functionality to the user on the frontend or backend.

Registering Users

By default WordPress has the setting for anyone to register for the site deactivated. You can turn it on via Settings > General in the Administration Screen by checking the box next to "Anyone can register". Users are then able to register at [yourdomain.com/wp-login.php?action=register](#). The [wp_register function](#) allows you to place a link on the page that returns a "Site Admin" link to logged in users or a "Register" link to non-logged in users.

Again, plugins can also offer their own user registration options. Not beating the WooCommerce horse again, but it provides it own options for registering users as Customers with a custom login and register page.

User Options

All users will need to provide an email, username, and password. Users also

have options for including their website, Twitter, Google+, and Facebook profiles. As well as biographical information.

As far as profile pictures, WordPress makes use of [Gravatars](#) to display an avatar when users comment on your site or within their profile. Gravatars are images tied to the user's email address that can be used across the internet anywhere that Gravatar is used. Since it's included in WordPress, that's about 27% of the web at this point.

That's a brief introduction to users, roles, and capabilities within WordPress. Like many other features of WordPress this is just the tip of the iceberg as far as user management and creation is concerned.

Settings

We've almost reached the end of our WordPress journey through the Dashboard. We've got just a couple stops left. Settings and Tools.

The Settings Administration Screen contains just what you'd think it would, settings for your WordPress site. These are settings that are going to be tied to your site and not just the active theme. There is quite a bit to work with here so all those fine Core Contributors have done us a favor and separated them out into several sections for us.

General Settings

The Settings General Screen contains some basic, though important, settings for your site. It is here that you can set your site title, tagline, who can register, date and time displays, and your admin email address. Let's break those down.

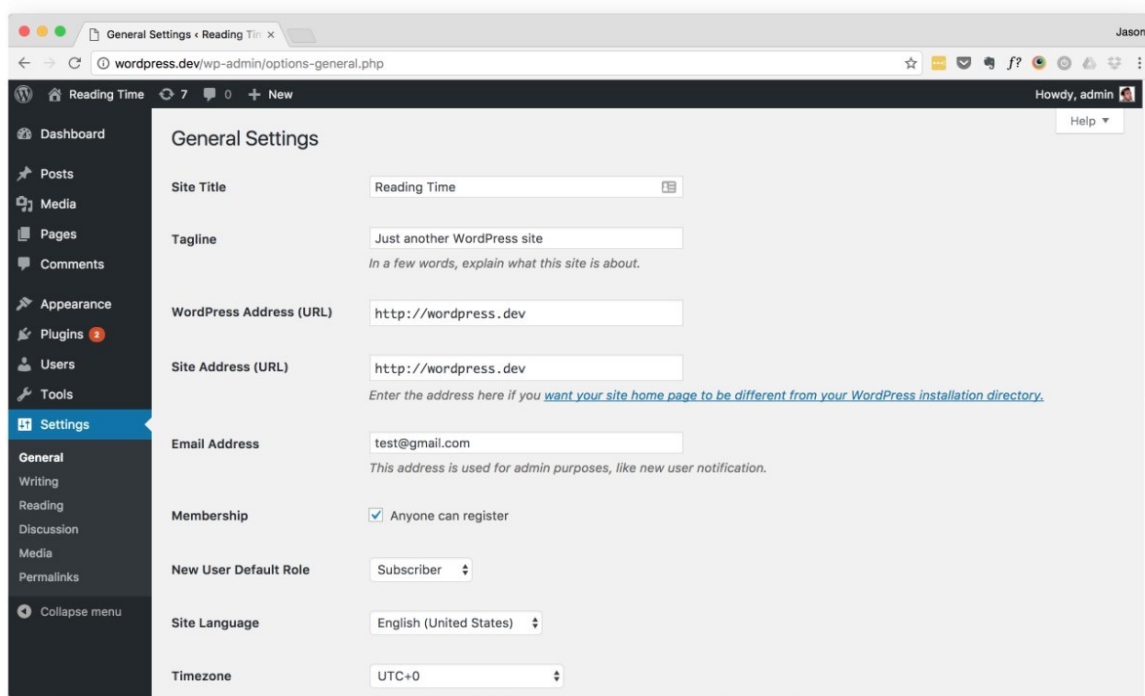
- **Site Title** - This should be fairly self explanatory. This is the title of your team. If you specialize in animal based positive reinforcement gifs, this would be *Stay Pawsitive*. Typically this will show in at the top of your site to users or be included as part of the tab title in the browser. Some themes may not use this and only offer a logo upload. Which may be a sign to avoid that theme. Just sayin'.
- **Tagline** - The tagline gives you a place to put a brief, and I mean very brief, description of your site. In our animal gif example this could be "Gifs to make you pawsitively happy". But maybe don't double up on the puns. It will be theme dependent if this shows on the front-end, but by default it will show in the browser tab title bar along with the site title.
- **WordPress Address (URL)** - This is the full URL where the WordPress application is installed on your server. Most of the time this will be <http://yourdomain.com>, or if you installed WordPress as a subdirectory such as "blog" it would be <http://yourdomain.com/blog>. This will automatically be filled in based on the domain you enter in the Five Minute Install. It can also be defined within the wp-config.php file using WP_SITEURL.
- **Site Address (URL)** - This is the address you want users to enter into their browser to access this site. You have to own the domain of course and have it pointing to the server. So no trying to swipe google.com. The only time this would be different from your WordPress Address is if you're giving WordPress its own directory. Like the WordPress Address the Site Address can also be set in the wp-config.php with WP_HOME.
- **Email Address** - This is the email for the site administrator. New user

notifications and the like are sent to this email. Most form plugins, such as Gravity Forms, will also default to have new form entry notifications send to this email address. This is different than the admin user email you set during the Five Minute Install. Though it will use that email by default, they can both be changed separately.

- **Membership** - Checking this box allows anyone to register as a user on your site. When this is checked and a new user registers they are created with the role selected in the **New User Default Role** dropdown. Subscriber is probably your best bet if using his option.

The remaining settings in the General Settings relate to dates, times, and language display for your site. You can set your timezone as well as the format you would like the date and time to appear throughout your site in the backend and front-end. The week starts on field allows you to set which day shows first on the WordPress calendar widget.

Finally, **Site Language** allows you to set the language for the dashboard of your site.



Writing Settings

The Writing Settings Screen provides a few options to help you with writing

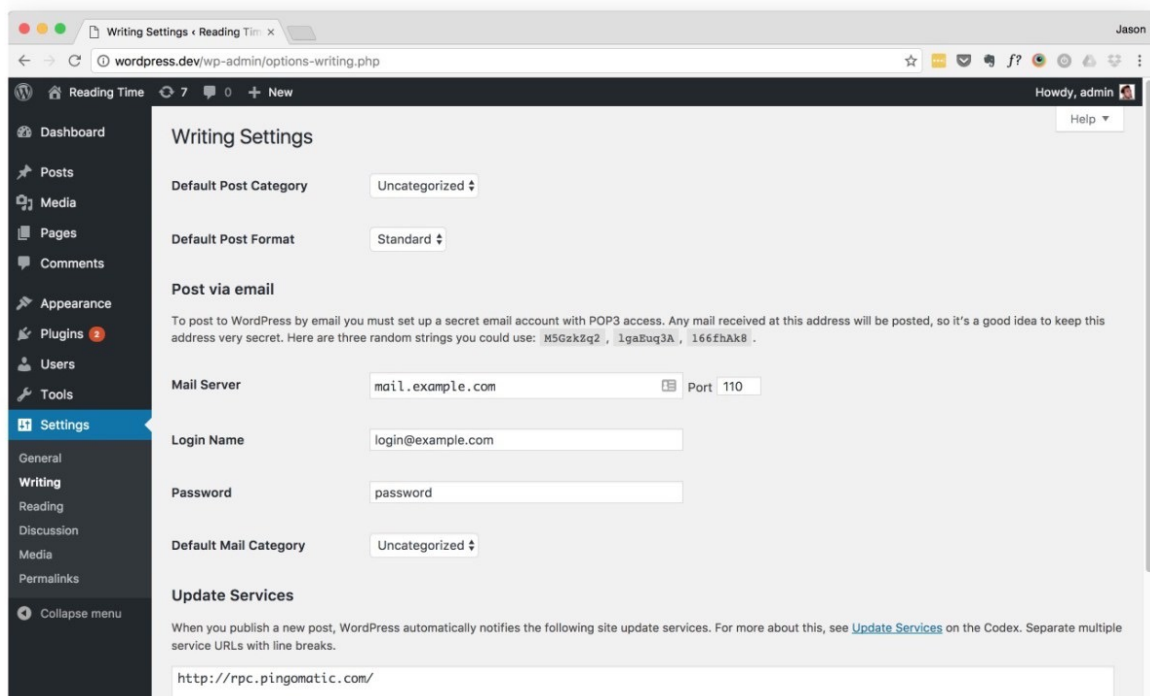
posts.

The first few options give you some choices to set how WordPress displays some content. Such as everyone's favorite, emoticons. The Codex has [much more on using smileys](#).

You're also able to set some of the defaults for your Posts. Such as the default Post category and Post format from those available within your theme.

The Settings Writing Screen also gives you the option of setting up an email so you can post by simply sending an email to a preset email list. Of course you'll want to keep this email very secret as anything sent to it would be posted to your site. Definitely not one you want getting spam. With the ability to use Jetpack and the WordPress mobile app and Calypso applications to connect your [WordPress.org](#) sites to a [WordPress.com](#) account this feature is likely going the way of the dodo. Both those options provide easy and powerful methods for posting to your WordPress site.

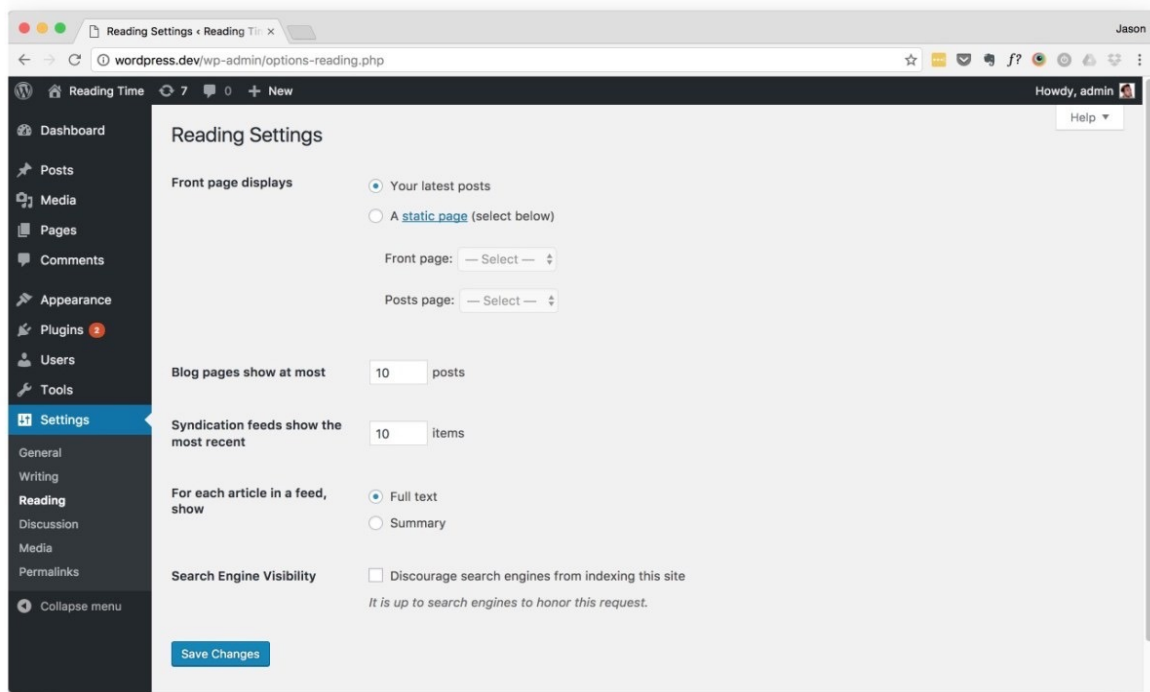
[Update Services](#) allows you to set services to be pinged anytime you create or update a post. This is useful if you need to ping things like a FeedBurner to update automatically.



Reading Settings

The Reading Settings allow you to set some options for how your site displays content to readers.

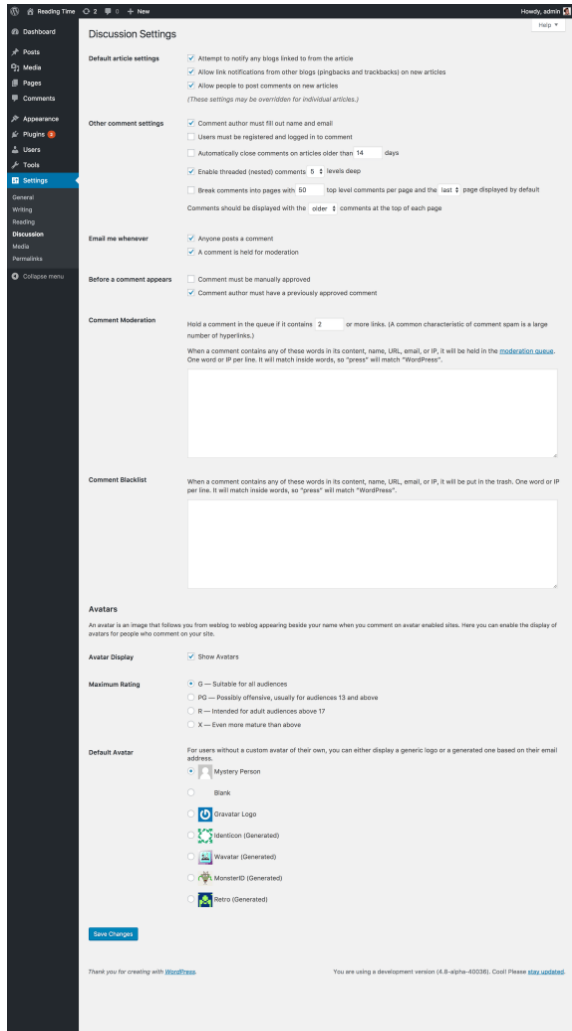
- **Front Page Displays** - This setting allows you to select what is displayed on the home page when a user visits your site. Either a list of your latest posts or a static Page (post of type Page).
 - **Your latest posts** - If this option is selected your theme will use the index.php file to display the front page of your site. Typically index.php is your generic loop.
 - **A static page** - Selecting a static page gives you two further options by allowing you to select from your Pages what to use for both the front page and the posts page. One thing to note here, whatever page you set as the Posts page here will use the index.php file to display. While whatever you set as Front page will use whichever file the selected page hits in the template hierarchy. If a theme uses a front-page.php file that template will always be used for the front page, regardless of what is set as the Static page.
- **Blog pages show at most** - This setting let's you select the number of posts to show on a page before using pagination. This defaults to 10 which would mean your posts page shows the latest 10 posts then would display a pagination link to view older posts. This can be overwritten within theme or plugin code using the posts_per_page query parameter.
- **Syndication feeds show the most recent** - Similar to the above setting this allows you to set the number of items to show in syndications feeds.
- **For each article in a feed, show** - This setting allows you to control whether a post's full text or summary shows in syndication feeds.
- **Search Engine Visibility** - Checking to **Discourage search engines from indexing this site** creates a robots.txt file within the root of your site that helps prevent robots like those provided by Google from indexing your site. This is useful for site's you're trying to keep off the grid, like development or staging environments of live sites.



Discussion Settings

Next let's discuss the Discussion Settings[[note](#)]I'll see myself out.[/[note](#)].

The Discussion Settings are all about allowing users to interact with your blog by either commenting or pinging your site. Let's break down the settings section by section.



Default Article Settings

- **Attempt to notify any blogs linked to from the article** - This setting controls your blog sending out pings to other sites letting them know you used one of their links within your post. If you've got an article with hundreds of links, this setting could slow down your posting process as WordPress attempts to contact all of those links during the publishing process.
- **Allow link notifications from other blogs (pingbacks and trackbacks) on new articles** - Checking this allows for your site to accept ping backs and trackbacks from other WordPress users. Any ping backs or trackbacks can then show in your comments section for the post.
- **Allow people to post comments on new articles** - This setting will disable comments for any future articles posted. However, you can still edit to allow or disallow comments per each individual post. Used in conjunction with the option to close posts after a certain number of days can help you fully

disable comments on your site.

Other Comment Settings

- **Comment author must fill out name and email** - The home security system sign in the yard of the house that doesn't have a home security defense of spam. This setting requires a name and email to be entered prior to commenting, but there is no verification to check if the name or email are legitimate. Still it can slow down some spammer bots.
- **Users must be registered and logged in to comment** - A little more security to protect against comment spam. This requires users be registered to comment on your posts. You'll want to allow anyone to register if you're using this setting.
- **Automatically close comments on articles older than [X] days** - Setting this closes the comments on posts older than a given number of days. If set this to 1 day and disallow people to post comments on new articles, you'll essentially disable all commenting on your site.
- **Enable threaded (nested) comments [X] levels deep** - This allows for nested comments to a given number [X] of levels. Think of this as replies to a main comment. Typically these are indented up to a point and then continue down inline.
- **Break comments into pages with [X] top level comments per page and the [last/first] page displayed by default. Comments should be displayed with the [older/newer] comments at the top of each page** - This setting allows pagination for your comment section as well as giving you some control as to how the comments are ordered. This is very handy if you're getting tons of comments. As loading all comments on one page can eat up a lot of load time.

Email Me Whenever

- **Anyone posts a comment** - Enabling this will cause WordPress to send an email to the admin email from Settings > General any time a new comment is made on your site.
- **A comment is held for moderation** - Likewise you can have WordPress email you anytime a comment is held for moderation and needs to be approved.

Before a comment appears

- **Comment must be manually approved** - Setting this will cause all comments made to the site to need to be moderated and approved by an administrator before they are posted to the site.

- **Comment author must have a previously approved comment** - This setting allows for comments to be approved if they have a previously approved comment from the same email address.

Comment Moderation

- **Hold a comment in the queue if it contains [X] or more links** - Setting this will cause any comments with more than a given number [X] of links to be held for moderation. Typically spam features multiple links. This can help you prevent spam from showing on your site.
- **When a comment contains any of these words in its content, name, URL, e-mail, or IP, it will be held in the moderation queue. One word or IP per line. It will match inside words, so "press" will match "WordPress"** - The longest setting name ever. This section allows you to add words, IPs, emails, and more to a list that will be used to check comments against to determine if they should be held for moderation.
- **When a comment contains any of these words in its content, name, URL, email, or IP, it will be put in the trash. One word or IP per line. It will match inside words, so "press" will match "WordPress"**. - Much like the last setting except this time they comment goes straight into the garbage.

Avatars

No, not the James Cameron epic 3D adventure, but the previously mentioned Gravatar image that can be tied to a user's email address. This section allows for a few settings on how avatars are displayed within your comments section.

- **Avatar Display** - Checked shows avatars in comments. Unchecked does not.
- **Maximum Rating** - This allows you to set the rating limit allowed for avatars that will display on your site. This can help prevent showing avatars that are inappropriate.
- **Default Avatar** - This setting allows you to select a default image to use if a user does not have a Gravatar image set.

Media Settings

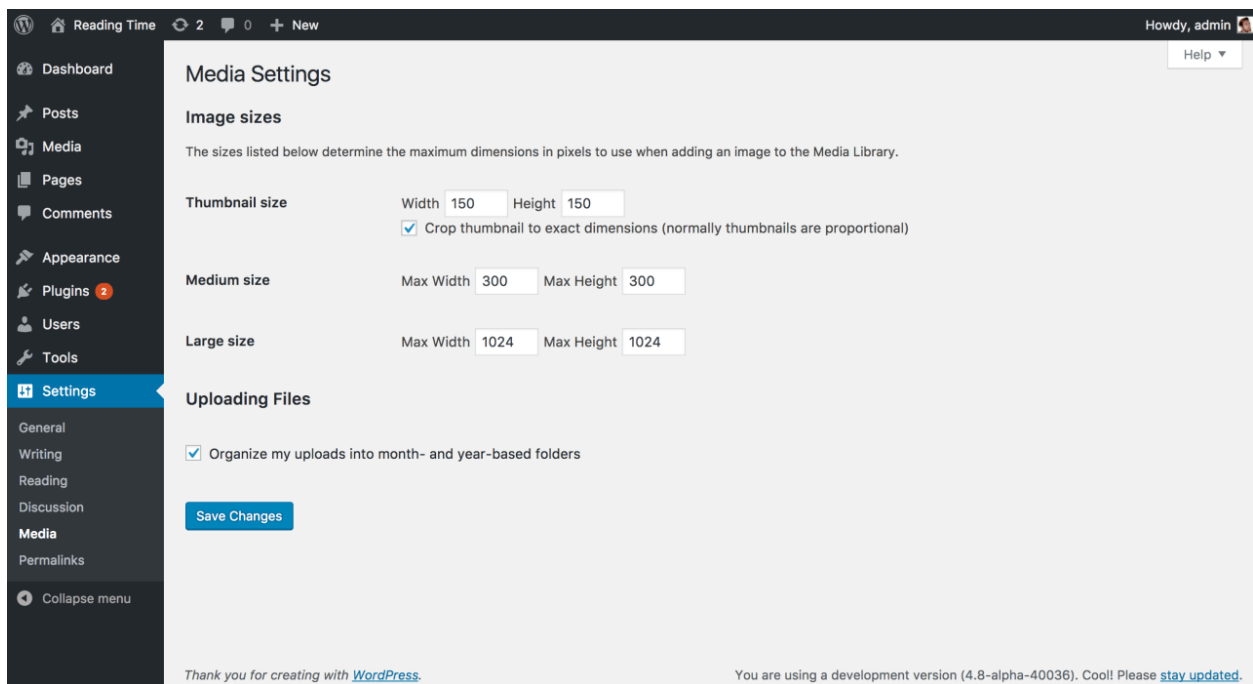
Out of the box WordPress will crop images you upload to a few sizes, if the image is larger than those dimensions to begin with. These are the image sizes you can choose when inserting an image into a post. The Media Settings screen allows you to adjust those default sizes to better fit your site's needs.

The three sizes are thumbnail, medium, and large. The main difference being

that thumbnail can be cropped to exact dimensions, whereas medium and large take a max-width and max-height. For example, if you were to upload an image at 1280 x 600 it would crop the large size to 1024 x 480 and the medium size to 300 x 141.

With the addition of [responsive images in WordPress 4.4](#), a medium-large size was added with a 768px width by default and no height limit. It cannot be edited within the Media Settings page though. In order to edit the size for it you would need to use the `update_option()` core function.

There is one other setting within the Media Settings Screen. This option, **Organize my uploads into month- and year-based folders**, allows you to select whether your `wp-content/uploads` folder will be broken down into sub-directories for the year and month the media was uploaded or just leave all uploads in a single level directory.



Permalinks Settings

Permalinks are the links for your posts. Typically you don't want your URLs to change. You want them to be permanent. See what they did there?

The Permalinks Settings page allows you to pretty them up and make them human readable. Instead of [domain.com/?p=234](#), you can get [domain.com/my-awesome-post](#). In the past WordPress defaulted to the plain `?p=[post id]` example.

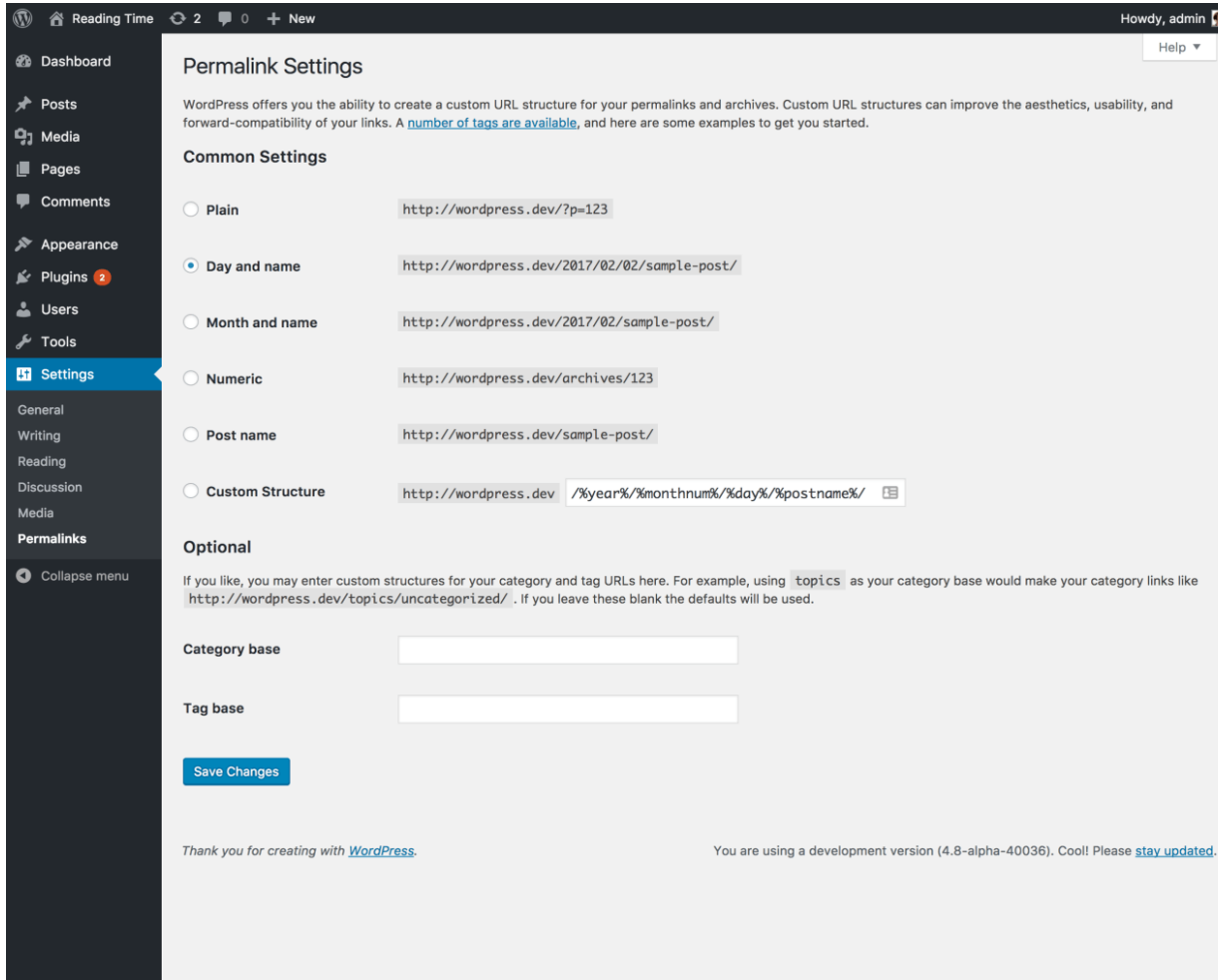
Now days a new WordPress install is going to come set with Post name selected.

There's a few presets you can choose from. Simply switch to the format you want to use and click Save Changes at the bottom of the page and WordPress will handle the rest. But choose early. You don't want to change this after you've got links out to your posts or Google has indexed them. If you switch and someone navigates to an old link they could hit the dreaded 404 page. No one wants that.

If you want to get a bit more funky with your permalink structure there are [tags available](#) for setting a custom structure.

On the Permalinks Settings Screen you can also change the slug used for category and tag links. For example, if you prefer your category archive for a certain term shows as [domain.com/topic/category-name](#) instead of [domain.com/category/category-name](#), you can set the Category base as "topic" and save the changes. The same applies for tags.

There are a whole host of options available in WordPress for more [advanced customizations](#).



Custom Settings Screens

That is it. That is all there is for the Settings Screens. Until you install some plugins. And then they may add their own host of Settings Screens to your site. And a lot of plugins will. If you're looking for where you should setup a new plugin, under the Settings menu is a good place to start. If of course it doesn't add its own menu item entirely.

Ok, now that is really it for Settings.

Tips, Tricks, and Tools

We have reached the last part of our journey through a basic WordPress install. There is a lot to take in. That's why we need to take a look at some tips, tricks, and the tools screen we skipped over.

Tools

First, let's take a brief look at the Tools screen.

When you first land on the Tools Screen you drop into the Available Tools Screen. As of 4.7 that will give you two items. Press This and the Categories and Tags Converter. The Categories and Tags Converter let's you do what it sounds like, convert categories to tags and tags to categories.

Press This is a little more interesting. Press This is a bookmarklet you can add to your browser that lets you easily share a page you're viewing on your blog with a brief preview.

JANUARY 10, 2017 BY ADMIN

Handling Comments on Your WordPress Site | Jason Yingling

Allowing comments on your WordPress site is a great way to engage users. Before you do you will want to know how to moderate and manage comments.

Source: [Handling Comments on Your WordPress Site | Jason Yingling](#)

Import Tools

The Import Tools screen provides you with several options for importing content from a different site. There are tools for importing from Blogger, Tumblr, RSS, or another WordPress site.

Export Tools

The Export Tools allows you to export your WordPress site content into an XML file that can be used to import into another WordPress site.

There is one tricky thing about WordPress's export tool. You won't get your posts featured images unless you use the All Content option. Just exporting Posts won't give you their featured images as featured images are stored as Media Posts.

Tips

Screen Options

One thing you may come across frequently is a guide telling you to edit a field you may not see on your editor. This can be extremely frustrating. Before hulking out and snapping your computer in two and throwing it into the ocean, check the Screen Options tab at the top right corner of the screen and see if there is an unchecked box hiding what you're looking for. They vary from screen to screen.

Bulk Edit

Need to change the author on 100 posts? Rather than individually open every post, change the author, and click update, you can select multiple posts at once and edit them all. Some of the things you can edit through the Bulk Edit tool are categories, tags, author, comments, status, format, pings, and sticky or not sticky.

The screenshot shows the WordPress admin dashboard with the 'Posts' section active. The 'Bulk Edit' dropdown is open, showing options for Author, Comments, Status, Format, Categories, and Tags. The 'Categories' dropdown is set to 'Uncategorized'. The table below shows two posts: 'Handling Comments on Your WordPress Site | Jason Yingling' and 'Hello world!'. The 'Hello world!' post has a comment count of 3.

| <input type="checkbox"/> | Title | Author | Categories | Tags | | Date |
|-------------------------------------|---|--------|---------------|------|---|----------------------|
| <input checked="" type="checkbox"/> | Handling Comments on Your WordPress Site Jason Yingling | admin | Uncategorized | — | — | Published 2017/01/10 |
| <input type="checkbox"/> | Hello world! | admin | Uncategorized | — | 3 | Published 2016/11/04 |

Finding all posts for a specific category or tag

When you first spin up a WordPress site your posts will have default to being categorized under Uncategorized. It's an unbecoming look for your posts. Let's say you never got around to adding categories because your old theme didn't display them[*note*]Bad theme.[*/note*]. But now you've upgraded and don't want "Uncategorized" showing on all your posts.

In order to get a list of all the posts under Uncategorized you'll want to navigate to Posts > Categories. Then in the table of Category terms you'll want to click the number under count. Boom, all your posts categorized as Uncategorized. Click the checkbox next to Title and it'll select all on the current page view. Then from the Bulk Edit dropdown hit Edit and you'll be able to select a new Category.

The screenshot shows the WordPress admin interface for managing categories. On the left is a dark sidebar with navigation links: Dashboard, Posts (highlighted), All Posts, Add New, Categories, Tags, Media, Pages, Comments, Appearance, Plugins (2), Users, Tools, Settings, and Collapse menu. The main content area is titled 'Categories' and includes a search bar, 'Screen Options', and 'Help' buttons. Below the search bar is a table with one item, 'Uncategorized', with a count of 2. The 'Count' column is highlighted with a red box. The 'Add New Category' form is visible on the left, with fields for Name, Slug, Parent, and Description. A note at the bottom right states: 'Deleting a category does not delete the posts in that category. Instead, posts that were only assigned to the deleted category are set to the category Uncategorized. Categories can be selectively converted to tags using the [category to tag converter](#).' The version number 'Version 4.7.2' is displayed in the bottom right corner.

| Name | Description | Slug | Count |
|---------------|-------------|---------------|-------|
| Uncategorized | | uncategorized | 2 |

Those are just a few of the tricks and tips available in a fresh WordPress install.

Outro

That's it. Thanks for following along with this introduction to WordPress. Hopefully you've got a better understanding of what WordPress offers you out of the box. If you found it interesting or helpful please feel free to leave a comment and subscribe to my blog for more content.