

# WORDPRESS THEMES AND CUSTOM POST TYPES

By: William Johnson

Harness the power of word press through theme creation and the ability to create custom post types

# Who Am I?

Will Johnson

[William.c1.johnson@gmail.com](mailto:William.c1.johnson@gmail.com)



@will\_\_da\_\_beast



# Agenda

- Cover the basic fundamentals of theme development
- Build the core template files
- Create the “single” template file
- Create the functions file
  - ▣ Create all necessary functions
  - ▣ Add necessary functions as actions
  - ▣ Add necessary filters
- Discuss how to include external function files to help maintain modularity
- Create a meta box to allow site owners to enter custom data
- Create the custom post renderer

# Before We Start...

- This is a live demo, there will be coding involved if you choose to follow along.
- The template files can be downloaded here:  
[https://dl.dropboxusercontent.com/u/24449420/wordpress\\_demo.zip](https://dl.dropboxusercontent.com/u/24449420/wordpress_demo.zip)

# Theme Development

- A Theme provides a separation of concerns between the page content and how the content is rendered.
- Themes define how content is displayed and provide admins with the ability to edit content within the scope of the theme.
- Advantages:
  - ▣ Creates a unique look for your website
  - ▣ Utilizes templates and the WordPress Loop
  - ▣ The ability to easily switch between different layouts
  - ▣ Use custom post types

# Other WordPress Theme Advantages

- Separation of presentation style and template files from the system files help ensure that upgrades will go relatively smoothly
- Allows for quick changes of the visual design and layout of a WordPress site.
- Alleviates the need for the site owner to learn CSS, HTML, PHP, Javascript, etc.

# Theme Fundamentals

- Themes live in the wp-content/themes folder
- Most themes contain four components:
  - ▣ The Header – Static across most pages
  - ▣ The Post Loop – Loops through and renders your posts
  - ▣ The Sidebar – Static across most pages
  - ▣ The Footer – Static across most pages
- Themes contain three types of files:
  - ▣ A stylesheet (style.css) – contains the presentation styles
  - ▣ Template files – control how the site generates information from the WordPress database
  - ▣ Optional - a functions.php file to add actions and filters

# Theme Stylesheet

- Every theme is required to have a `style.css`
- Defines the styles to control the presentation of the data
- The `style.css` must contain meta data at the top of the file describing the theme. The most important meta data to include is the **Theme Name**.
- It's a good idea to use a css framework such as twitter bootstrap as it will define a set of standards to follow and encourages responsive design.



# Template Files

- Used to generate pages based on data in the database
- There are several template types:
  - `index.php` – The main template file. It must be present.
  - `comments.php` – Comments template
  - `front-page.php` – The front page template
  - `home.php` - The home page template
  - `single.php` – Used when a single post is queried
  - **`single-{post-type}.php` – Used for custom post types.**
  - `page.php` – The page template
  - `category.php` – Category template
  - `tag.php` – Used when a tag is queried
  - `taxonomy.php` – Used when a term in a custom taxonomy is used
  - `author.php` – Used when an author is queried
  - `date.php` – Used when a date is queried

# Template Files (Continued)

- Archive.php – The archive template
- Search.php – The search results template
- Attachment.php – Used when viewing an attachment
- Image.php – Image attachment template
- 404.php – The 404 not found template
  
- Depending on the post query, the corresponding template will be used to load the page.
- See the [http://codex.wordpress.org/Template\\_Hierarchy](http://codex.wordpress.org/Template_Hierarchy) to reference the rules defining how templates are overridden.

# Functions

- Functions define custom logic, which can be executed through actions and defined as filters.
- Functions should be placed in the **functions.php** file
- The functions.php file acts as a plugin for your theme
- The functions file can include external php files. It is a good practice to modularize your code such that your functions are grouped together in related files.

# Functions (Continued)

- Useful functionality to add to your functions file:
  - ▣ Enable sidebars, navigation menus, post thumbnails, post formats, custom headers, custom backgrounds, etc
  - ▣ Define functions used in your template files
  - ▣ Setup an options menu, giving site owners options for colors, styles, etc
- Use the [Function Reference](#) to see what types of functionality you can add to your theme.

# Demo - Core

Adventures In WordPress Themes

## Welcome to the Demo

Welcome!

Sample Page

Contact

## Adventures in WordPress Themes

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

# Recap

---

- Wordpress themes provide a separation of concerns between the presentation of your site and its content.
- To build a theme, you need to create a `style.css` and an `index.php`.
- Depending on the page you are navigating to, the associated template for that page will be used when building the page.

# Recap Continued

- You can include external files within your php files to create more manageable code.
- The functions.php file allows you to define functions that can be used in your other php files. You should register actions and filters here. You should also place common functions in this file.
- Meta boxes provide site authors with a form, allowing them to enter any custom information.

# Recap Continued

---

- The information stored in a meta box can be rendered using a template. You define how this information should be rendered through your templates.
- Custom post types provide a mechanism for defining your own custom posts. This can be handy depending on the type of pages you want to display on your site.



# Recap Continued

---

- If you want your custom post types to be associated with the “get\_pages” call, you will need to add a filter that first retrieves your custom post types and then merges them with the current set of pages.

# Resources Continued

- [http://www.amazon.com/Professional-WordPress-Development-Brad-Williams/dp/111844227X/ref=pd\\_sim\\_b\\_1](http://www.amazon.com/Professional-WordPress-Development-Brad-Williams/dp/111844227X/ref=pd_sim_b_1)
- <http://codex.wordpress.org/>
- <http://generatewp.com/>

# Questions?

