

Child Themes Made Easy

Manchester WordPress Meetup

May 2013

Presented by:
Jonathan May
President, YourPresenceOnTheWeb.com



Agenda

- Themes De-mystified
- What is a Child Theme?
- Why Create Child Themes?
- Create a Simple Child Theme in 3 Easy Steps
- Modifying Templates with Child Themes
- Resources for Digging Deeper



Themes De-mystified

- Q: What does a theme do? A: Establishes the look and basic functionality of a Wordpress site
- Q: What comprises a theme? A: A collection of files
 - **PHP files**
 - Known as “templates”, they determine how page elements are laid out and function
 - At minimum, every theme must have an **index.php** file. Typically, other “template” files as well, e.g., header.php, sidebar.php, footer.php
 - **CSS files** (Cascading Style Sheets)
 - Define how various elements on the page are styled, e.g., text font, size and color, background colors/images, spacing between elements, borders around elements, etc etc
 - At minimum, every theme must have a **style.css** file. May be additional style sheets as well
 - Optionally, other files, e.g., image files, Javascript files, language files



Themes De-mystified

- A theme establishes the **look and basic functionality of a Wordpress site**
- A theme is comprised of a **collection of files**
 - PHP files – have .php as the extension
 - A theme must have an **index.php** file, which is first file that WordPress tries to load when someone visits the site.
 - Most themes also have other “template” files which determine how page elements are laid out and function, e.g., header.php, sidebar.php, footer.php, functions.php, etc.
 - CSS files (Cascading Style Sheets) – have .css as the extension
 - Define how various elements on the page are styled, e.g., text font, size, color and decoration, background colors/images, spacing between elements, borders around elements, and so on.
 - *At minimum*, every theme must have a **style.css** file. May be additional style sheets as well.
 - Many themes also have other files, e.g., image files, Javascript files, language files



Themes De-mystified

- Where do these files reside?
 - Each theme has its own sub-directory located in the /wp-content/themes directory where WordPress is installed
 - By convention, the sub-directory is named after the theme, e.g., the sub-directory for the Twenty Twelve theme is named /wp-content/themes/twentytwelve



What is a Child Theme?

- A **child theme** is a theme that inherits the functionality of another theme, called the **parent theme**, and allows you to modify, or add to, the functionality of that parent theme.
- Instead of editing a theme's files directly, you can create a child theme.
- A child theme inherits all of the templates and functionality from its parent theme, but allows you to modify the behavior to the parent theme because code in the child theme overrides the code in the parent theme.



Why Create Child Themes?

- It is the safest way to modify the behavior an existing theme.
 - If you modify an existing theme and it is updated, your changes will be lost. With a child theme, you can update the parent theme (which might be important for security or functionality) and still keep your changes.
- It is the fastest way to create one or more variations of an existing theme.
 - Once you get the hang of it, you can easily create many variations of the same parent theme.
- It's a great way to get started if you are just learning WordPress theme development.



Create a Child Theme in 3 Easy Steps

- To create a simple child theme that alters the ***styling*** of a given parent theme:
 - Step 1: Create a directory for the child theme files
 - Step 2: Create a basic style.css file
 - Step 3: Add some custom CSS code to the style.css file



Create a Child Theme in 3 Easy Steps

- **Step 1: Create a directory for the child theme files**
 - Locate the directory in the /wp-content/themes directory
 - By convention, name the directory the same as the parent theme's directory, with the suffix “-child”
 - Example: If you are creating a child theme for the Twenty Ten theme, name the directory:

`/wp-content/themes/twentyten-child`



Create a Child Theme in 3 Easy Steps

- **Step 2: Create a basic style.css file**

- Use a plain text editor (e.g., Notepad, Bluefish) to create the file
- Add a comment block at the top of the file identifying, **at minimum:**

```
/*  
Theme Name: A Bubbly Version of Twenty Ten  
Description: Jonathan's first child theme  
Author: J. May  
Author URI: http://yourpresenceontheweb.com/  
Version: 1.0  
Template: twentyten  
*/  
@import url('../twentyten/style.css');
```

- **Theme Name:** This name will appear in the **Appearance => Themes** screen
- **Template:** Must match *the directory name of the parent theme*
- Following the comment block, include an **import** command that directs WordPress to include the parent theme's style sheet
 - The URL value needs to point to the location of the parent theme's style.css file



Create a Child Theme in 3 Easy Steps

- **Step 3: Add some custom CSS code to style.css**
 - Code must follow the **import** command
 - Example customizations:
 - Modify styling of text elements
 - Apply a background image

```
/*  
Theme Name: A Bubbly Version of Twenty Ten  
Description: Jonathan's first child theme  
Author: J. May  
Author URI: http://yourpresenceontheweb.com/  
Version: 1.0  
Template: twentyten  
*/  
@import url('../twentyten/style.css');  
  
h1,h2,h3,h4,h5,h6 {  
    text-transform: uppercase;  

```



Modifying Templates with Child Themes

- With a child theme you also can modify the behavior of a parent theme, e.g., how pages are laid out, what content displays in an area of the page
 - **Override an existing template file in a parent theme** with a template file of the same name in the child theme
 - **Add a brand-new template file** to the child theme.
 - **Add a functions.php template file** to the child theme, which WordPress loads *in addition to* the parent theme's functions.php file



Modifying Templates with Child Themes

- Example of overriding an existing template file:
 - 1) Copy the footer.php file for the parent theme into the child theme directory
 - 2) Edit the customizations into the new file

Original:

```
<div id="site-generator">
<?php do_action( 'twentyten_credits' ); ?>
<a href="<?php echo esc_url( __( 'http://wordpress.org/', 'twentyten' ) ); ?>"
title="<?php esc_attr_e( 'Semantic Personal Publishing Platform', 'twentyten' ); ?
>"><?php printf( __( 'Proudly powered by %s.', 'twentyten' ), 'WordPress' ); ?
></a></div><!-- #site-generator -->
```

New:

```
<div id="site-generator">
<?php do_action( 'twentyten_credits' ); ?>
<a href="<?php echo esc_url( __( 'http://yourpresenceontheweb.com', 'twentyten' )
); ?>" title="<?php esc_attr_e( 'We are...All Things Wordpress!', 'twentyten' ); ?
>"><?php print("All Things WordPress!"); ?></a></div><!-- #site-generator -->
```



Resources for Digging Deeper

- General information about how themes work:
http://codex.wordpress.org/Theme_Development
- General information about child themes:
http://codex.wordpress.org/Child_Themes
- Template files explained:
http://codex.wordpress.org/Stepping_Into_Templates
- Functions file explained:
http://codex.wordpress.org/Functions_File_Explained
- WordPress PHP functions:
http://codex.wordpress.org/Function_Reference



Resources for Digging Deeper

- Books on theme development (or some material thereof):
 - *WordPress All-In-One for Dummies*, L. Sabin-Wilson, et al., 2011
 - *Professional WordPress: Design and Development*, B. Williams, 2013
 - *Web Designer's Guide to WordPress*, Friedman, 2012
- Learn the fundamentals of HTML, CSS and PHP
 - <http://www.w3schools.com/> (Free)
 - <http://lynda.com> (not free)
 - <http://tutsplus.com> (not free)

