

CUSTOMIZING YOUR WORDPRESS SITE

**THE WHAT, WHERE, HOW, AND WHY OF
MAKING YOUR SITE YOURS**

Nate Finch

WHO'S THIS NATE GUY??...

- WP for ~6 years, ~5 years full-time
- The Digital Ring, toptal, Codeable, and other clients for fun and profit
- Currently residing in Madison
- Texas >> Kansas >> Spain (a little time in Morocco) >> France >> Illinois >> S. Korea >> and now Madison
- “I’d be a professional student if I could...”
- Married w/ 1.5 year old daughter



THE QUESTIONS

- Where do I make changes to my theme?
- Should I just put everything in `functions.php`?
- How can I add a JS library to my site?
- How can I update my site safely and not worry about losing customizations?
- How should I be thinking about customizations?
- What is the general structure of WordPress?

THE GOAL

- Beginner to Intermediate
- Framework with some specific examples
- Empower you to make decisions

I: THE INITIAL PROBLEM

DEMO

AXIOMS

- Themes handle Form, Plugins handle Function
- Code should go in Codebase
- Data should go in Database
- Media Files should be processed through the Media Library (and reside in the `wp-content/uploads` directory)
- And...

ABC

ALWAYS BE CLOSING...

WAIT...

Always Be

**CheckingYouHaveABackupOfYourCodeAndDatabase
BeforeYouRunAnyUpdatesOrChangeAnything**

II: UNDERSTANDING THE STRUCTURE AND INTENT

"Form and Function"

THEMES HANDLE *FORM*

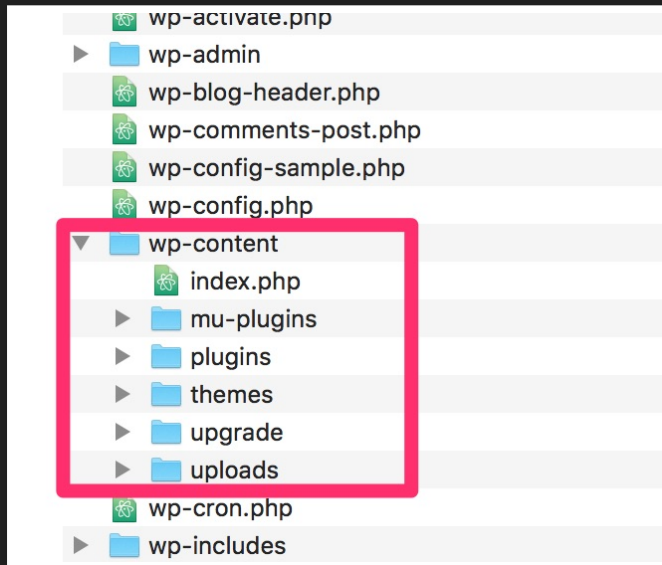
- CSS
- Layout
- Theme supports
- Post Types (maybe)
- Whatever is specific to the current layout and style

PLUGINS HANDLE *FUNCTION*

- Basics: widgets
- Custom Fields/Meta Boxes
- CRM/Marketing
- E-commerce
- Memberships
- Anything that extends what's available "out of the box"

WORDPRESS STRUCTURE

WHERE DOES ALL THIS STUFF LIVE?



- Customizations go in **wp-content**
- Use the "wp-content/* namespace"
- *Don't hack core!*

LET'S BUILD SOMETHING!

A STYLED AFTER-POST BOX



- override and hard code the single.php template file in child theme
- use an action hook in your theme's function.php file
- create a customizations plugin

BEFORE STARTING

1. Make sure you have a code and database backup
2. Work on a Local Environment or Staging Environment ([Local](#) or [VWV](#))
3. Maybe even look at the [Pantheon Workflow](#):

Code Moves Up, Content Moves Down



III. MAKING STYLE OR THEME-BASED CHANGES

1. Make a Child Theme
2. *"Are the changes I want to make Form or Function?"*
3. Add Form changes to theme
4. Use Child Theme's `style.css`
5. [Overriding templates](#)

IV. MAKING STYLE OR THEME-BASED CHANGES

1. Make a Child Theme
2. *"Are the changes I want to make Form or Function?"*
3. Add Form changes to theme
4. Use Child Theme's `style.css`
5. Overriding templates

SOME FAQs...

Should I use the Custom CSS in the customizer?

- You can if you don't want to make a child theme.

Should I use a Custom CSS/JS plugin?

- You can if you don't want to make a child theme or custom plugin.

Should I use a Custom PHP plugin?

- You can, but you probably shouldn't. 🤪

ALL OF THE ABOVE BREAK THE AXIOMS THAT CODE SHOULD BE IN THE CODEBASE AND DATA SHOULD BE IN THE DATABASE.

III. MAKING FUNCTIONAL CHANGES

Using a plugin allows us to:

- keep this functionality after a site redesign
- keeps function separated from form
- allows us to "separate concerns"

DEMO

1. Create a single file plugin
2. Activate it in wp-admin
3. Copy over the template, leave the styles
4. Use action hook (or add your own!!)

BONUS POINTS

1. Create a multfile plugin with inc. PHP, JS and CSS
2. Don't add your own version of jQuery

RESOURCES AND NEXT STEPS

LOCAL/STAGING DEV

- [Local](#)
- [VWV](#)
- [Pantheon Workflow](#)

WORDPRESS CODEX

- [Child Themes](#)
- [Plugin Headers](#)
- [Template Hierarchy](#)
- [Template Files](#)
- [add_action](#)
- [Template Tags](#)
- [Conditional Tags](#)

TUTORIALS AND LEARNING

YOUTUBE

- [Customize Genesis Child Theme](#)
- [More Katrinah Videos](#)

PAID

- [KnowTheCode.io](#)
- [Morten on Lynda.com](#)
- [Brad on Udemy](#)