WEB DESIGN II

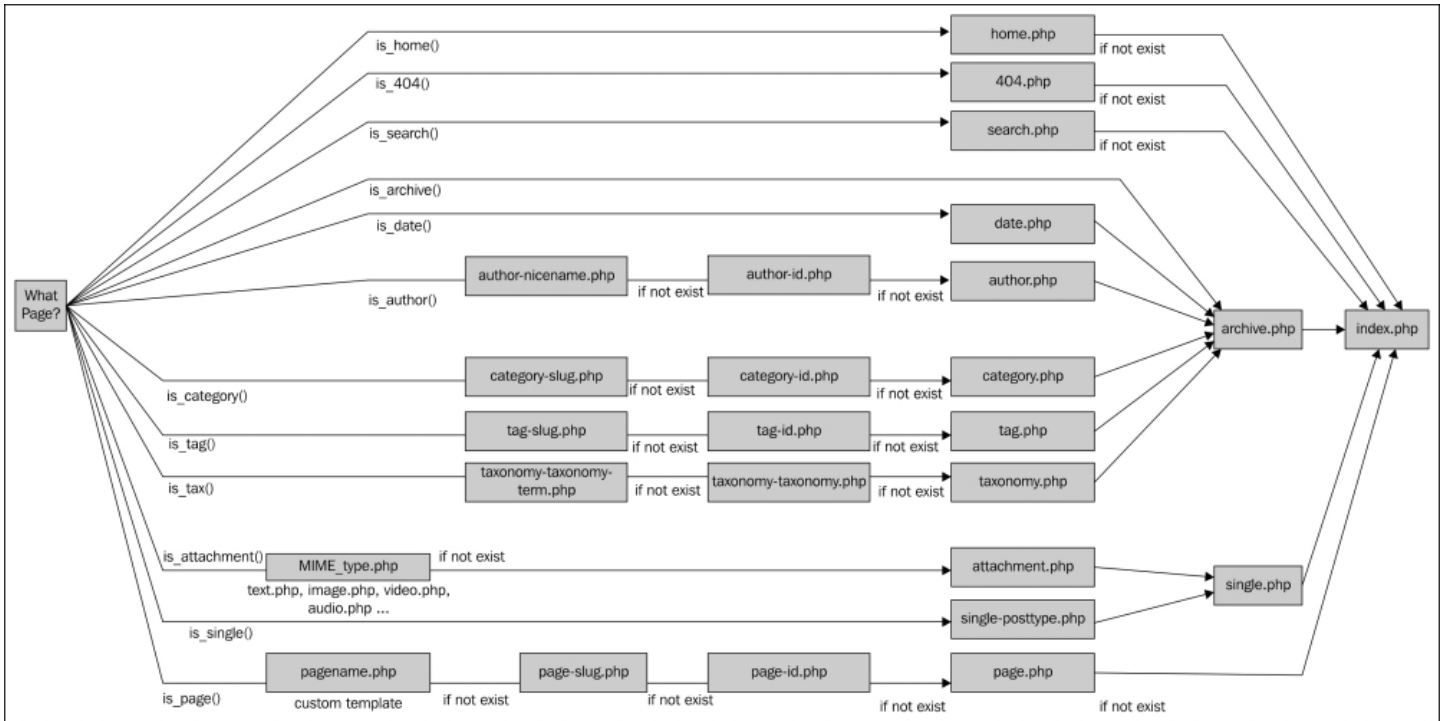## TIPS & TRICKS: Wordpress Themes, Part 1

Background

Working with themes in wordpress can be simple or complex depending on the site and the design you have in mind. Before you can work with your own custom themes, be sure you have completed the following:

- Create design comps for the pages of the site and complete HTML/CSS build of those comps. It is often simpler to work from existing source than to start from scratch.
- Download the latest version of Wordpress from wordpress.org and install it on a functional webserver. This might be your own local testing server, or a remote server. Complete the installation and then log in and create at least some placeholder content that matches elements form your design.
- Establish a smooth connection to your webserver in your coding software of choice so that you can easily test your solutions as you work.

Setting up a Custom Theme

First we need to set up the appropriate fileset for our custom theme. With the site open in a code editor:

1. Add a new folder for your custom theme under `wp-content/themes`.

2. Copy your existing webpage reconstruction files into this folder, being sure that all the images, styles, and any scripts remain linked correctly given the new location of these files.

3. Rename `index.html` to `index.php`, or create `index.php` and place your homepage markup into it. Optionally, rename other files you've prepared that correspond to pages for your site based on the following chart, including these common files:

   » `category.php` — a page for a feed of posts from a given category (this can usually be accomplished just as well with `index.php` unless there are significant differences in the surrounding conent).
   » `single.php` — a template for a single post (this can usually be accomplished just as well with `index.php` unless there are significant differences in the surrounding conent).
   » `page.php` — a template for a static page.

McCollin, R., and Sliver, T.B. (2013). *Wordpress Themed Development Beginner's Guide.* Packt Publishing.

4. Ensure that your main stylesheet is named precisely `style.css`. Then add the following code to the top of your main CSS stylesheet:

```
/*
Theme Name: __YOUR_THEME_NAME__
Theme URI: __YOUR_DEVELOPMENT_URL_IF_APPLICABLE__
Author: __YOUR_NAME__
Author URI: __YOUR_WEBSITE_IF_APPLICALBE__
Description: __DESCRIPTION_OF_THEME__
Version:
Tags:
*/
```

5. In any theme PHP files, find the stylesheet path you had entered and replace it with this:

```
<link type="text/css" rel="stylesheet" href="<?php bloginfo('stylesheet_url'); ?>" />
```

6. If you have any images placed directly in any of these template files, you should update their src attribute path to match the following (being sure to match your own local directory for images and the image's filename:

```
<img src="<?php bloginfo('template_directory'); ?>/path-to-file" />
```
Images in your stylesheet should render correctly on their own as long as you're using relative paths to point to them.

7. Log in to your Wordpress site, go to Appearance > Themes and click Activate on your custom.

8. You should now see your existing webpage build, but it will not yet show actual content from WordPress.