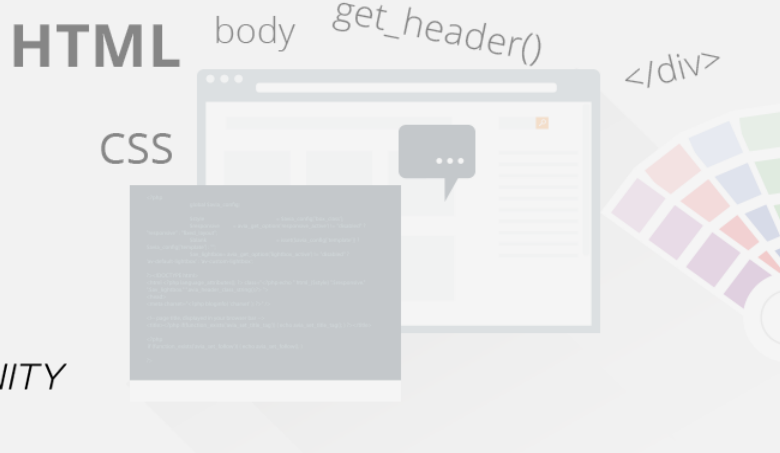


+



# “ WordPress theme building from scratch ”

*A Web Developer approach*

Panagiotis Halatsakos (Web Developer)  
Vasilis Baimas (Web Designer / Web Developer)



“WordPress makes it drop-dead easy to start a site. Take my advice and go do it.”

– John Battelle



# Topics to cover

- Template Hierarchy 101
- How to create a minimal WordPress Theme
- Structure of page Template
- The Loop: Bringing data into frontend [1]
- The Loop: Bringing data into frontend [2]





# Templates and their Hierarchy



## Templates :

- They are basically files that tell WordPress how to display different types of content.

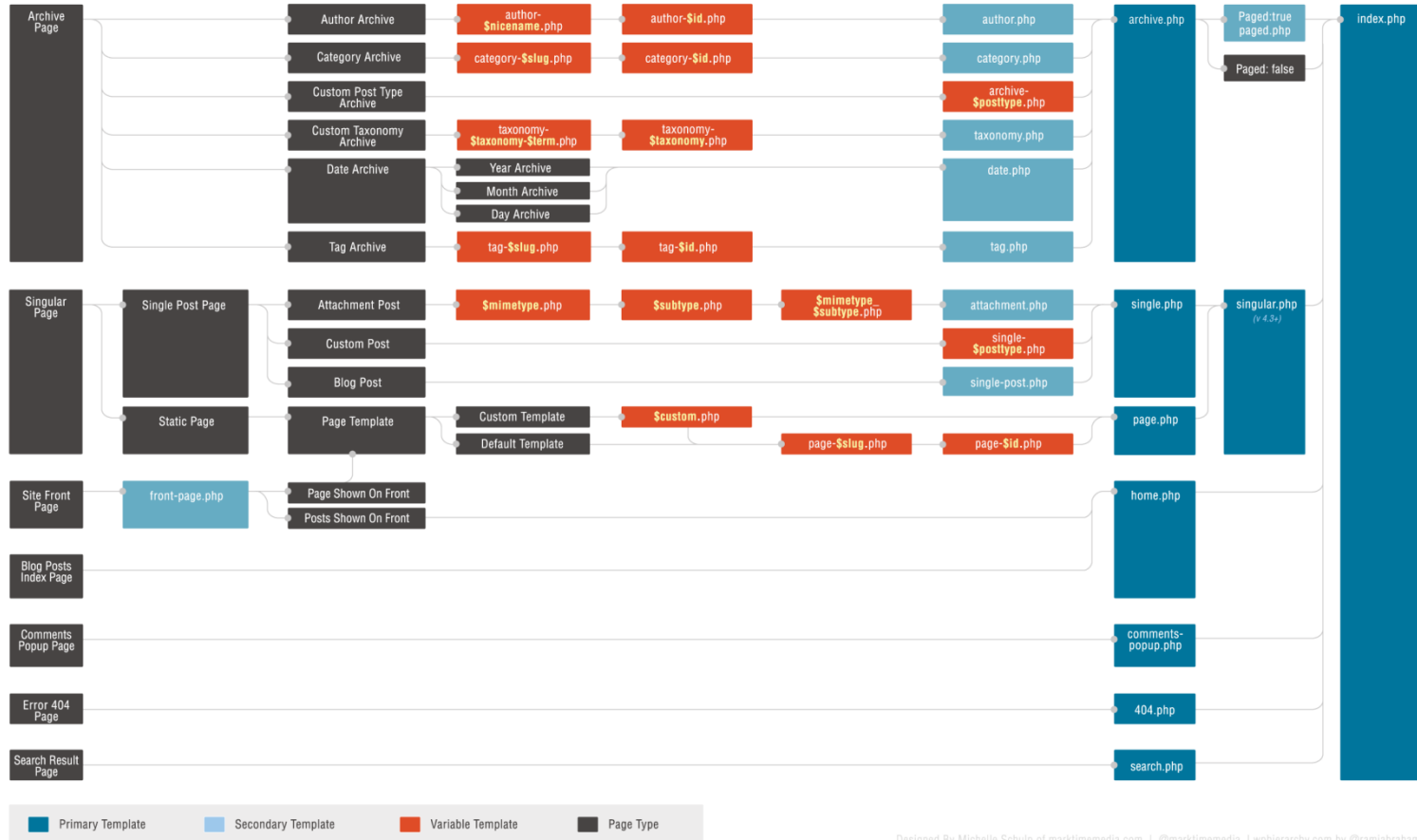


## WordPress Template Hierarchy :

- It determines the order in which template files are chosen.

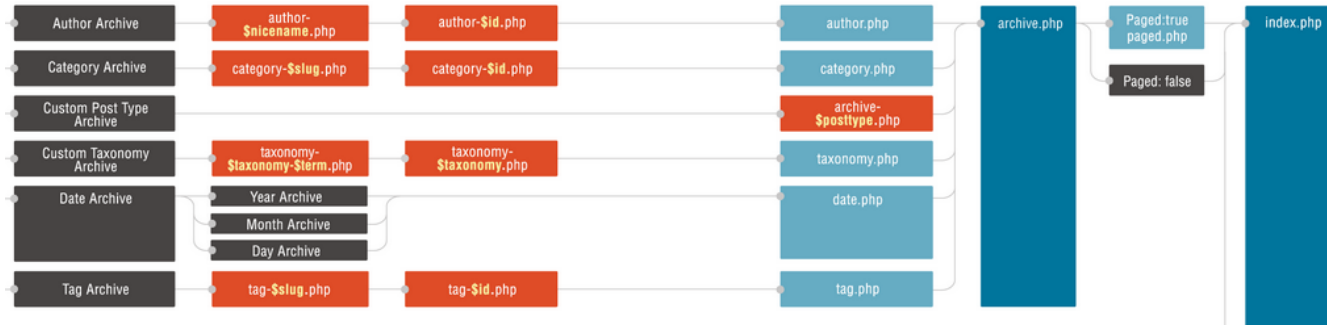


# WordPress Template Hierarchy





# Specific Example of Template Hierarchy (author template)



**Example :** *When we create an author user*

- First, WordPress will search for the template labeled `author-[slug].php`.
- If that file doesn't exist, then it looks for `author-id.php`.
- If that file doesn't include, then it looks for `author.php`
- Going up still, it next looks for `archive.php`.
- Finally, if this is not found it uses `index.php` to render the page.



# Specific Example of Template Hierarchy

- We create the news as a category in our dashboard

<input type="checkbox"/>	ID	Name	Description	Slug	Count
<input type="checkbox"/>	2	news		news	0

- The WordPress looking in the Theme to find the related file, using the Hierarchy.

- The steps :

`category-news.php -> category-2.php -> category.php -> archive.php -> index.php`



# Which Template File Will WordPress Use?

PAGE TYPE	tries first > tries next > tries last
404	404.php > index.php
SEARCH	search.php > index.php
TAXONOMY	taxonomy-{tax}-{term}.php > taxonomy-{tax}.php > taxonomy.php > archive.php > index.php
HOME	home.php > index.php
ATTACHMENT	{mime-type}.php > attachment.php > single.php > index.php
SINGLE	single-{post-type}.php > single.php > index.php
PAGE	{custom-template}.php > page-{slug}.php > page-{id}.php > page.php > index.php
CATEGORY	category-{slug}.php > category-{id}.php > category.php > archive.php > index.php
TAG	tag-{slug}.php > tag-{id}.php > tag.php > archive.php > index.php
AUTHOR	author-{author-nicename}.php > author-{author-id}.php > author.php > archive.php > index.php
DATE	date.php > archive.php > index.php
ARCHIVE	archive.php > index.php



# How do we create a “minimal” WordPress theme ?





# Process – Least requirements

Design (Concept) + HTML files

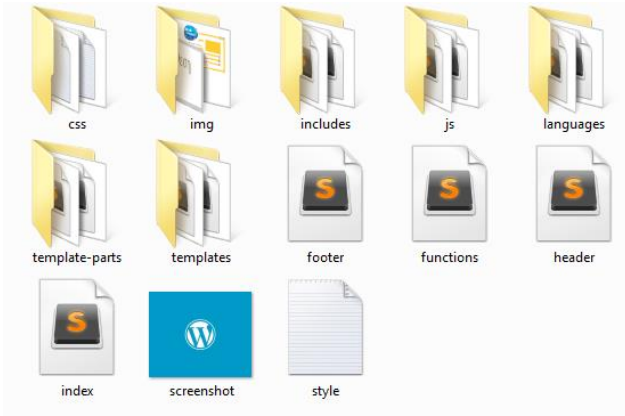
- Navigate to wp-content/themes
- Create a new folder
- Create the `style.css`
  - ☛ Define in the form of “*comments*” the following :

```
style.css
1 @charset "utf-8";
2 /* CSS Document
3 | Theme Name: My Theme Name
4 | Theme URI: www.yoursiteurl.com
5 | Description: The Basic information of website
6 | Version: 1.0
7 | Author: The Author Name
8 | Author URI: http://authorname.com
9 | License: GNU General Public License v2 or later
10 | License URI: http://www.gnu.org/licenses/gpl-2.0.html
11 | Tags: black, brown, orange, tan, white, yellow, light, one-column
12 | Text Domain: My Theme Name
13 /*
```

- Create the `header.php`, `footer.php`, `index.php` and «cut» the html
- Activate the theme from : **Appearance / Themes**



# Guidelines for Naming WordPress Theme Folders and Files



File and folder names should be used to clearly identify their contents!

`/css`: All CSS files other than the `style.css` file like CSS libraries or resets.

`/img`: includes any image files.

`/includes`: Any PHP files that are not part of the WordPress templating system.

`/js`: All javascript files.

`/languages`: Any files related to internationalization and translation.

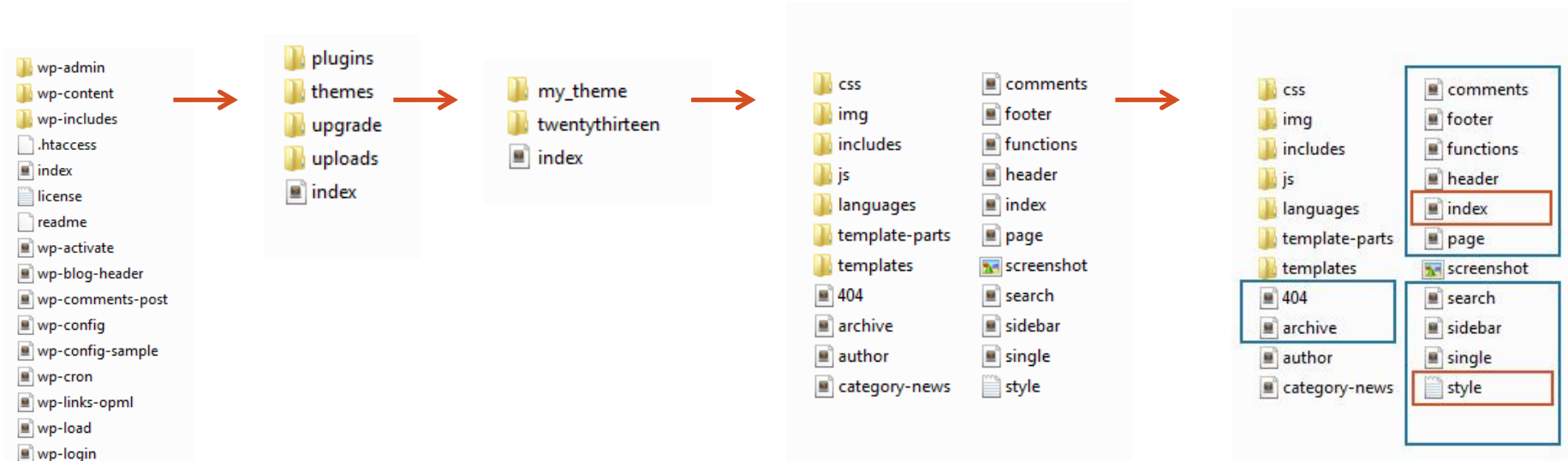
`/templates`: If the theme includes any page templates.

`/template-parts`: Any files called by `get_template_part()`.



# WordPress Theme Structure

## Basic and Core Files



Basic files

Core files



# WordPress Template File List [1]

- `header.php` : Contains everything you'd want to appear at the top of your site.
- `index.php` : The core file that loads your theme.
- `archive.php` : The template file used when viewing categories, dates, posts by author, etc
- `single.php` : The template file that's used when viewing an individual post.
- `single-{post-type}.php`. The single post template used when a single post from a custom post type is queried.
- `sidebar.php` : Contains everything you'd want to appear in a sidebar.
- `footer.php` : Contains everything you'd want to appear at the bottom of your site.

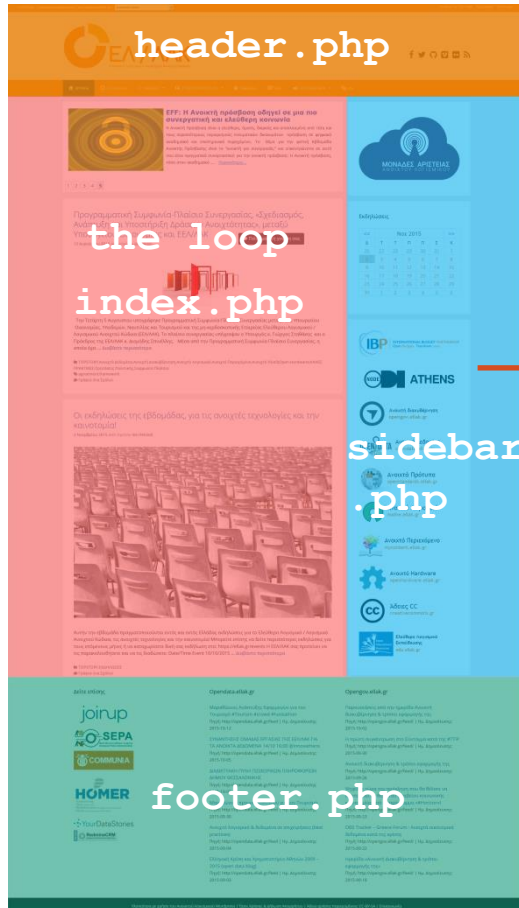
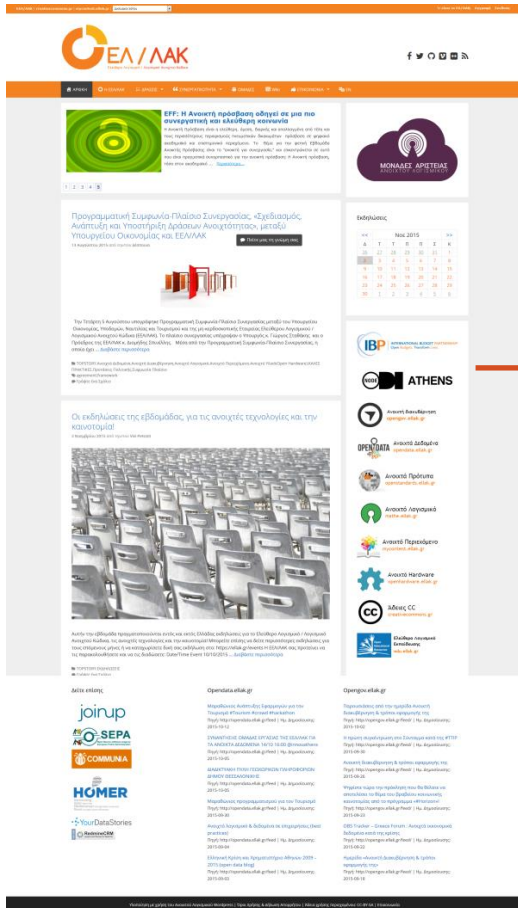


# WordPress Template File List [2]

- `page.php` : Similar to `single.php`, but used for WordPress pages.
- `search.php` : The template file used to display search results.
- `404.php` : The template file that displays when a 404 error occurs.
- `comments.php` : Called at the bottom of the `single.php` file to enable the comments section.
- `style.css` : All the styling for your theme.
- `functions.php` : A file that can be used to configure the WordPress core, without editing core files.



# Basic Page Anatomy (e.x. https://ellak.gr)



# What is the Magic With Loops?







# The loop : Bringing data into frontend

From the Back-End to the Front-End

What is the loop? Do I need one?

- A simple example of the Loop
- Data binding with template tags

Template Hierarchy, Incorporated

- Displaying the details of a post and a page (example)
- Displaying items based on a category (example)

```
<?php if ( have_posts() ) : while ( have_posts() ) : the_post(); ?>

<article class="post">

<div class="post-details">
  <h2><a href="<?php the_permalink(); ?>"><?php the_title(); ?></a></h2>

  <ul class="post-meta">
    <li>Categories: <?php the_category(' '); ?></li>
    <li>Date posted: <?php the_date(); ?></li>
    <li>By: <?php the_author(); ?></li>
  </ul>
</div> <!-- end post details -->

<div class="post body">
  <?php the_content(); ?>
</div><!-- end post body -->

</article> <!-- end post -->

<?php endwhile; else: ?>
<p><?php _e('Sorry, no posts matched your criteria'); ?>
<?php endif; ?>
```





# WordPress as CMS



- Wordpress has become a Content Management System (CMS), apart from being a blog engine.
- You have to have a content structure (or an idea of that) before laying your theme.
- So far, your data are on the database and you see them in the Back - End



# The Loop: Bringing data into frontend

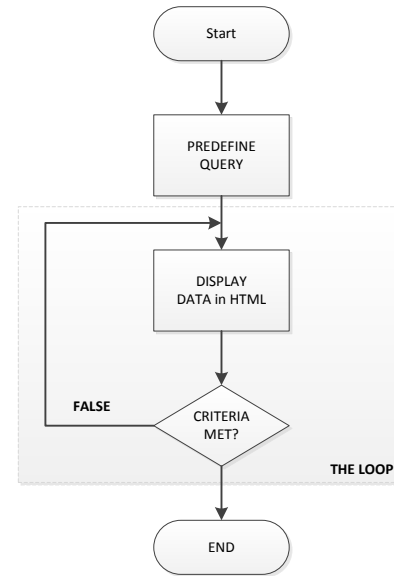
Άρθρα Νέο άρθρο

Όλα (366) | Δημοσιευμένα (361) | Προσχέδια (3) | Απόρρητα (2)

Μαζική επεξεργασία | Εκτέλεση | Όλες οι ημερομηνίες | Όλες οι κατηγορίες | All SEO Scores | Φιλτράρισμα

366 αντικείμενα 1 από 19

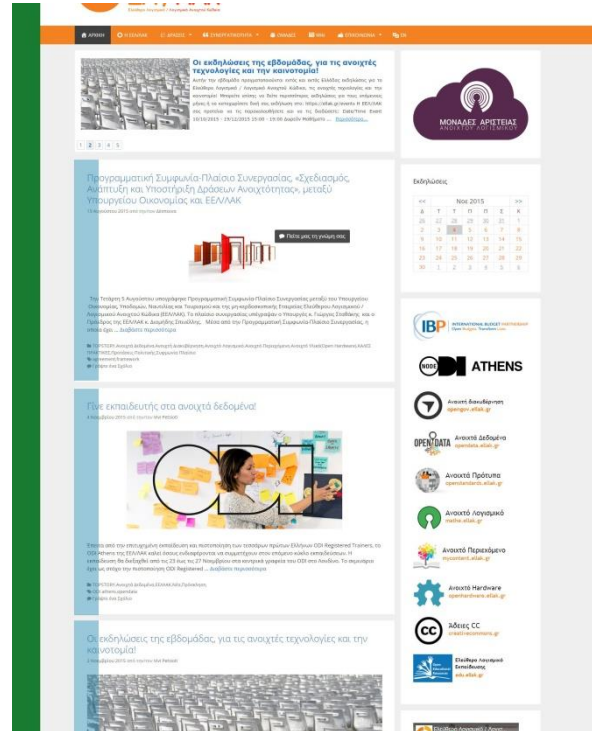
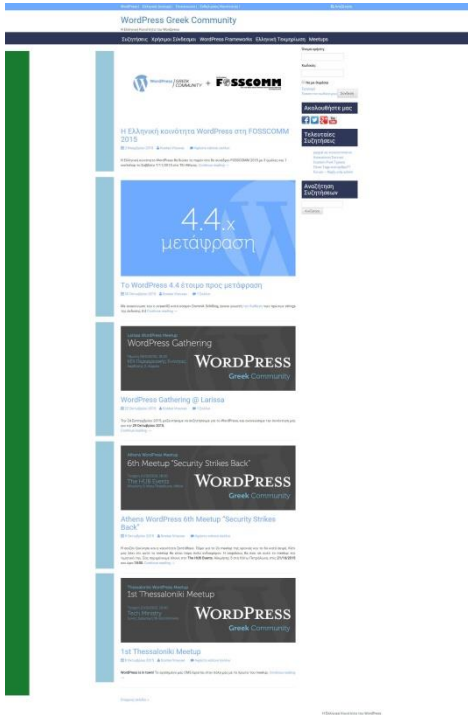
<input type="checkbox"/>	Τίτλος	Συντάκτης	Κατηγορίες	Ετικέτες	Ημερομηνία	SEO
<input type="checkbox"/>	Η Ελληνική κοινότητα WordPress στη FOSSCOMM 2015	Kostas Vrouvas	Ανακοινώσεις, Εκδηλώσεις	—	02/11/2015 Δημοσιεύτηκε	●
<input type="checkbox"/>	To WordPress 4.4 έτοιμο προς μετάφραση	Kostas Vrouvas	Μετάφραση WordPress	—	30/10/2015 Δημοσιεύτηκε	●
<input type="checkbox"/>	WordPress Gathering @ Larissa	Kostas Vrouvas	Εκδηλώσεις	—	22/10/2015 Δημοσιεύτηκε	●
<input type="checkbox"/>	Athens WordPress 6th Meetup "Security Strikes Back"	Kostas Vrouvas	Εκδηλώσεις	—	08/10/2015 Δημοσιεύτηκε	●
<input type="checkbox"/>	1st Thessaloniki Meetup	Kostas Vrouvas	Ανακοινώσεις	—	08/10/2015 Δημοσιεύτηκε	●
<input type="checkbox"/>	Athens WordPress Meetup 5 "Election Heat"	Kostas Vrouvas	Εκδηλώσεις	—	02/09/2015 Δημοσιεύτηκε	●
<input type="checkbox"/>	WordPress Meetups	dgyer	Εκδηλώσεις	—	01/09/2015 Δημοσιεύτηκε	●



The loop is responsible for **retrieving** bound **data** from the database, to the **user view**



# The Loop: With the first look



Loop visualized:  
Green  $\Rightarrow$  loop (while...endwhile)  
Blue  $\Rightarrow$  Looped data



# The Loop: Example to fetch 10 posts

## Defining the loop

1. Open `index.php` (homepage)
2. Switch to php mode (using `<?php ?>` tags)
3. If posts exist (recommended), *loop using `while(have_posts())`*
4. Required immediately: `the_post()`
5. Use template tags
6. Use html and CSS to style the data
7. Close the loop
8. Close the check condition

```
...
<section class="content-area">
  <?php
    if (have_posts()):           // 3
      // Loop start
      while (have_posts()):     // 3
        the_post();           // 4
      ?>
  <!-- 5,6 -->

  <article id="article-<?php echo get_the_ID();?>">
    <h2><?php echo get_the_title(); ?></h2>
    <?php the_content(); ?>
  </article>
  <!-- /5,6 -->
  <?php endwhile; // 7, end Loop ?>
  <?php endif; // 8 ?>
</section>
...
```

Business Logic

php / html Code



# Using the Template Tags

- **Template Tags** are used to bind backend data to the frontend.
- There are two “flavors” of template tags.
  1. One brings the raw value (`get_xxxx()` tag),
  2. The other brings the value with format (`the_xxxxx()`);
- **Template Tags** are connected to filters and actions (advanced development)

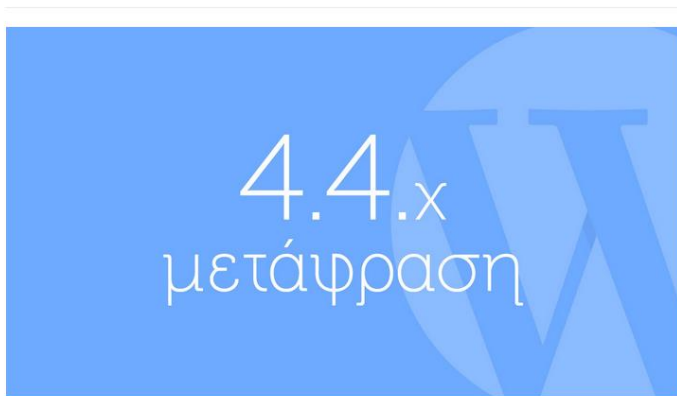


```
<?php my_template_tag() ;?>
```



# Example of Template Tags

(ex : wpgreece.org)



To WordPress 4.4 έτοιμο προς μετάφραση

📅 30 Οκτωβρίου 2015 👤 Kostas Vrouvas 💬 1 Σχόλιο

Με ανακοίνωση του ο *ocean90*, κατά κόσμον Dominik Schilling, έκανε γνωστή την διάθεση των πρώτων strings της έκδοσης 4.4 [Continue reading](#) →

`the_post_thumbnail()` : Displays the featured image.

`get_the_title()` : Displays the title.

`get_the_date()` : Displays the date (published)

`get_the_ID()` : Displays the ID of the post from the DB

`get_permalink(<postid>)` : Acquires the link leading to the detail page

`get_the_author()` : Displays the author's name of post

`get_the_content()` : Displays the content of post



# Using the loops for page!

## Template Hierarchies, Incorporated

- Displaying the content of a page or a post
- For multiple posts we used the loop.

**What do we use for the detail page e.g. when you click on a page or a post?**



An aerial photograph of a bay with a steep, rocky cliff on the left side. The water is a vibrant blue, with a white wake from a boat visible in the foreground. Several other boats are scattered in the bay. The text "HINT! YOU DON'T HAVE TO LOOK FAR!" is overlaid in white, bold, sans-serif font across the middle of the image.

**HINT! YOU DON'T HAVE TO LOOK FAR!**



# The Loop: Define the loop for a post or page

1. Open `index.php` (homepage)
2. Switch to php mode (using `<?php ?>` tags)
3. If posts exist (recommended), *loop using `while(have_posts())`*
4. Required immediately: `the_post()`
5. Use template tags
6. Use html and CSS to style the data
7. *Close the loop*
8. *Close the check condition*

```
...
<section class="content-area">
    <?php
        if (have_posts()):           // 3
            while (have_posts()):    // 3
                the_post();         // 4
            ?>
    <!-- 5,6 -->

    <article id="article-<?php echo get_the_ID();?>">
        <h2><?php echo get_the_title(); ?></h2>
        <?php the_content(); ?>
    </article>
    <!-- /5,6 -->
    <?php endwhile; // 7 ?>
    <?php endif; // 8 ?>
</section>
...
```

Business Logic

php / html Code



## Example of Template Tags (displaying a post or a page detail)

- Displaying the content of a page or a post
- The loop is the same! The underlying engine of Wordpress “decides” whether the loop is referring to a post or a group of posts.

There are a few behavior changes though to some template tags!



# Using Custom Templates

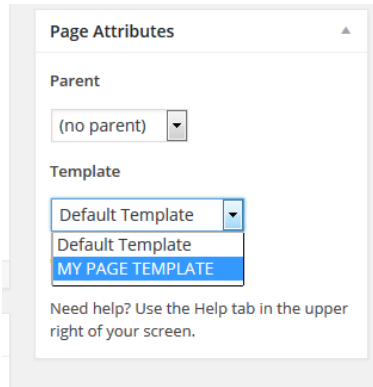
- Using template hierarchies you can create a “common” template (alter the default page) or specialize templates

## Code



```
templates-mytemplate.php x
1 <?php
2 /* Template Name:  MY PAGE TEMPLATE */
3 get_header();
4 ?>
5 <section>
```

## WordPress Dashboard





# Example of Template Tags

Suppose we have 3 categories amongst others:

- `news (slug:news)`
- `projects (slug:projects)`
- `products (slug:products)`

How can we display them in different Look & feel?

ID	Name	Description	Slug
2	news		news
8	projects		projects
1	products		products

QUICK EDIT

Name

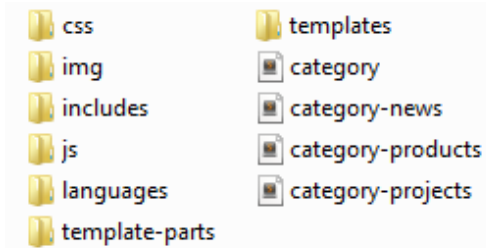
Slug



# Example of Template Tags (category content)

- Create the required files :

- `category.php`
- `category-news.php`
- `category-projects.php`
- `category-products.php`

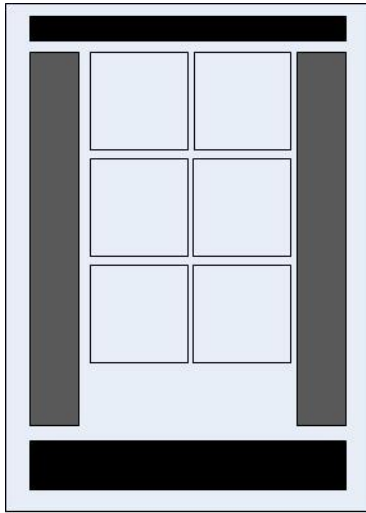


- Use the loop. It's the same as always  
(but different template tags and html might apply!)
- Call the appropriate category from the frontend. Magic!

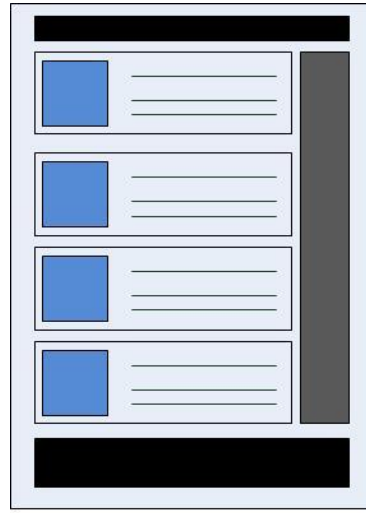


# Example of Template Tags

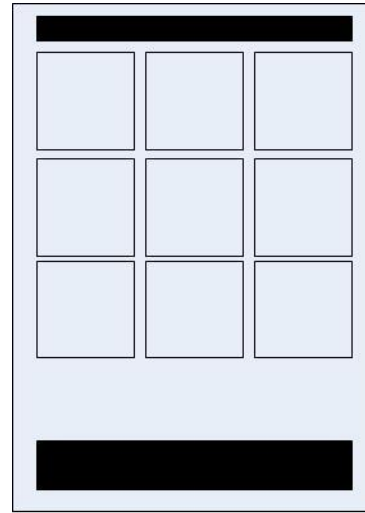
(category content)



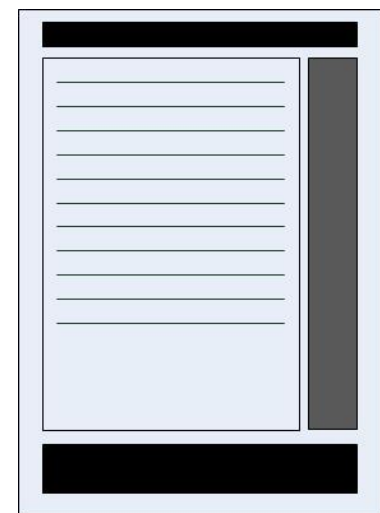
Category-projects.php



category-news.php



Category-products.php



category.php



# Template Hierarchies, Incorporated

## Common Pitfalls

- If there is no default file, using the hierarchy it will backtrack for the next hierarchical template.
- Actually it will search for the `404.php`, but if the `404.php` is absent it will send you back to the homepage with no data query parameters (Freaky things will happen).
- Usually you should check if there are “posts” in the category. If there are not, there will be no data shown.
- Also always remember to use `the_post()`; because it is required to do jumps to the next database record. If it is omitted, then prepare to overload the server!





# Resources

## Further reading!

<https://developer.wordpress.org/themes/basics/template-hierarchy/>

[https://codex.wordpress.org/Theme\\_Development](https://codex.wordpress.org/Theme_Development)

[https://codex.wordpress.org/The\\_Loop\\_in\\_Action](https://codex.wordpress.org/The_Loop_in_Action)

[https://codex.wordpress.org/Function\\_Reference/query\\_posts](https://codex.wordpress.org/Function_Reference/query_posts)

[https://codex.wordpress.org/Stepping\\_Into\\_Template\\_Tags](https://codex.wordpress.org/Stepping_Into_Template_Tags)

[https://codex.wordpress.org/Plugin\\_API/Action\\_Reference/pre\\_get\\_posts](https://codex.wordpress.org/Plugin_API/Action_Reference/pre_get_posts)

[https://codex.wordpress.org/Class\\_Reference/WP\\_Query](https://codex.wordpress.org/Class_Reference/WP_Query)



+



thanks!

Any Questions?

**You can find us at :**

Panagiotis Halatsakos

Facebook : ditikos, | [chibioni@gmail.com](mailto:chibioni@gmail.com)

Vasilis Baimas

twitter : [vbaimas](https://twitter.com/vbaimas) , | [vpbaimas@gmail.com](mailto:vpbaimas@gmail.com)