Wordpress for developers

Loops

In advanced mode

Recall: the loop

HEADER *header.php*	
THE LOOP *loop.php*	**SIDE** *sidebar.php*
FOOTER *footer.php*	

```php
<?php if (have_posts()) : ?>
    <!-- If there is Posts, Displays this part -->
    <?php while (have_posts()) : the_post(); ?>
        <!-- For each Post, displays something -->
    <?php endwhile; ?>
<?php else : ?>
    <!-- If there is'nt any post… -->
<?php endif; ?>
```
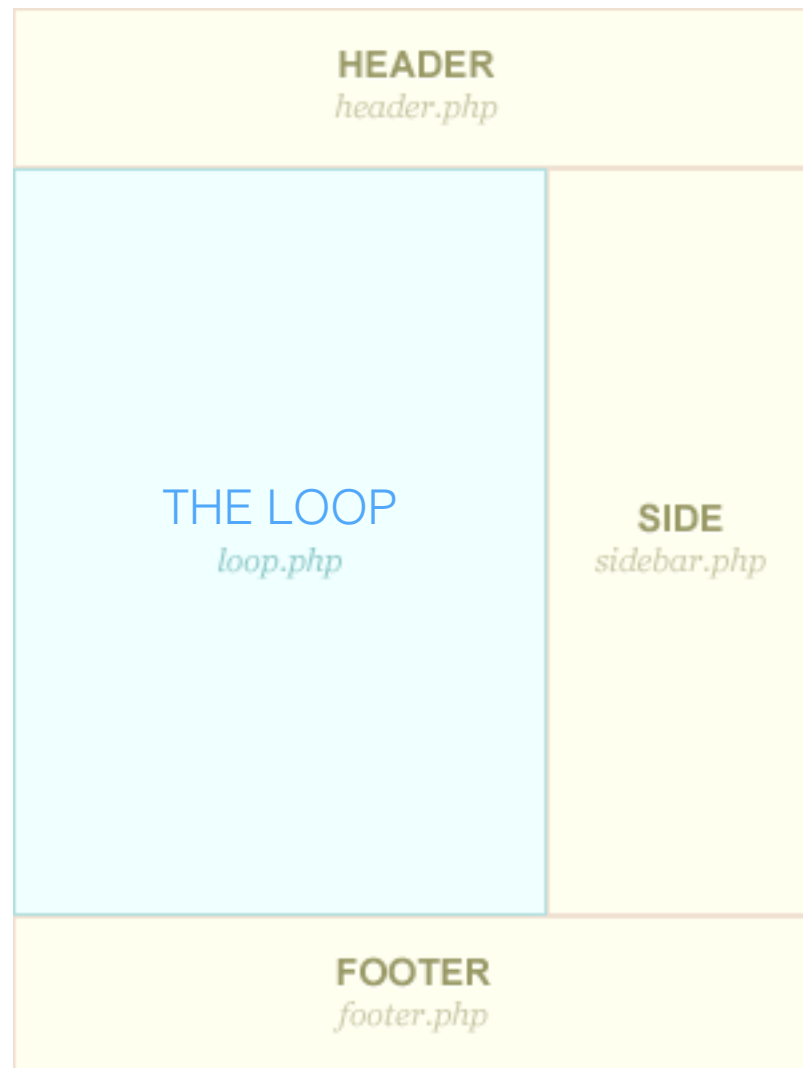
Costum loop
with query_posts()

```html
<!-- Include header.php -->
<?php get_header(); ?>

<div class="content">

  <?php query_posts('posts_per_page=20'); ?>
  <!-- We modify the default Wordpress loop -->
  <!-- We ask for 20 posts per page in the loop -->

  <?php get_template_part('loop'); ?>
  <!-- We include the loop template : loop.php -->

</div>

<?php wp_reset_postdata(); ?>
<!-- After a custom loop, always restore the default behavior -->

<?php get_sidebar(); ?>
<?php get_footer(); ?>
```

https://codex.wordpress.org/Function_Reference/query_posts

Multiple loops
With query_posts()

```php
<!-- latest posts -->
<h4 class="section">Derniers articles</h4>
<ul class="list">
  <?php wp_reset_postdata();
  query_posts('posts_per_page=5');
  while (have_posts()) : the_post(); ?>
    <li><a href="<?php the_permalink(); ?> »>
          <?php the_title(); ?></a></li>
  <?php endwhile; ?>
</ul>
<?php wp_reset_postdata(); ?>
```

https://codex.wordpress.org/Function_Reference/query_posts

Multiple loops
With WP_Query()

```php
<?php
  $args = array( 'post_type' => 'post', 'posts_per_page' => 4 );
  $loop = new WP_Query( $args );
  while ( $loop->have_posts() ) : $loop->the_post(); ?>

<h3><?php the_title(); ?></h3>

<?php
  endwhile;
  wp_reset_postdata();
?>
```

https://codex.wordpress.org/Function_Reference/query_posts

Functions.php

Your main theme « plugin »

Add functionalities
to your theme

add_theme_support

```
add_theme_support('post-thumbnails');
    add_theme_support('html5', ['caption', 'comment-form',
'comment-list', 'gallery', 'search-form']);
  add_theme_support('title-tag');
  add_theme_support( 'post-formats', array( 'video', 'aside',
'gallery' ) );
  add_theme_support( 'custom-background');
```

```
  $args = array(
    'width'         => 560,
    'height'        => 120,
    'default-image' => get_template_directory_uri() . '/images/
default-header.jpg',
    'uploads'       => true,
  );
  add_theme_support( 'custom-header', $args );
```

In the header.php

```
<?php $header_image = get_header_image(); ?>
<img src="<?php echo header_image() ?>" />
```

Add custom image sizes

add_image_size

```
add_image_size( 'custom-size', 220, 180 );
// 220 pixels wide by 180 pixels tall, soft proportional crop mode

add_image_size( 'custom-size', 220, 180, true );
// 220 pixels wide by 180 pixels tall, hard crop mode

add_image_size( 'custom-size', 220, 220, array( 'left', 'top' ) );
// Hard crop left top
```

Widgets areas (sidebars)

functions.php

```php
if ( function_exists('register_sidebar') ) {
  register_sidebar([
    'name'          => __('Sidebar','text_domain'),
    'id'            => 'sidebar-primary',
    'before_widget' => '<section class="widget %1$s %2$s">',
    'after_widget'  => '</section>',
    'before_title'  => '<h3>',
    'after_title'   => '</h3>'
  ]);
}
```

sidebar.php

```php
<?php dynamic_sidebar('sidebar-primary'); ?>
```

Navigation menus

Declare some menu(s) in functions.php

```php
register_nav_menus( array(
    'primary' => 'Primary Menu'
) );
```

Then call it into a template :

```php
<?php
wp_nav_menu( array(
    'menu'            => 'primary',
    'theme_location'  => 'primary',
    'depth'           => 2
); ?>
```

It is possible to redefine the rendered HTML code of a menu with some 'Walkers'

https://codex.wordpress.org/Navigation_Menus

Shortcodes

functions.php

```php
//[foobar]
function foobar_func( $atts ){
    return "foo and bar";
}
add_shortcode( 'foobar', 'foobar_func' );
```

Display this shortcode inside a template :

```php
<?php echo do_shortcode('[foobar]'); ?>
```

Use this shortcode in Wordpress admin WYSIWYG editor

```
[foobar]
```

https://codex.wordpress.org/Shortcode_API

Hooks

2 types of hook:

Action

Add an function call at a specific time during the process of page loading (ex : add a widget, load a script or css).

Filtrer

Filters are intercepting a specific Wordpress function before it's execution and used to modify it's output (ex : add a CSS Class, change HTML tag rendering, modify the excerpt length…).

add_action, do_action, remove_action, add_filter, apply_filters

Action

```
// Add text after the header on the homepage
add_action( 'hook-name' , 'my_action_function' );

function my_action_function() {
  // Check that we are on the front page
  if ( is_front_page() ) {
    echo « <div>New promotion today...</div>";
  }
}
```

https://codex.wordpress.org/Plugin_API/Action_Reference

http://adambrown.info/p/wp_hooks/hook/actions

Filter

```php
//Change WordPress Excerpt length
function mytheme_excerpt_length( $length ) {
    return 200;
}
add_filter( 'excerpt_length', 'mytheme_excerpt_length' );
```

https://developer.wordpress.org/reference/functions/add_filter/

http://adambrown.info/p/wp_hooks/hook/filters

Create your own content types (custom post and custom taxonomies)

Articles, Pages, Medias, Commentaries… **My custom type**

Category, Tags… **My custom taxonomy**

Custom Post Type

Wordpress can handle your own custom posts just
like 'posts' and 'pages'

It's the best solution to administrate a portfolio,
some testimonials, or any other type of content…

With the capability_type property you can
determinate if your new custom content
will act like a post or like a page.

Custom Post Type

```php
register_post_type(
    'projet',
    array(
        'label' => 'Projets',
        'labels' => array(
            'name' => 'Projets',
            'singular_name' => 'Projet',
            'all_items' => 'All the projets',
            'add_new_item' => 'Add a projet',
            'edit_item' => 'Edit projet',
            'new_item' => 'New projet',
            'view_item' => 'See projet',
            'search_items' => 'Search in projets',
            'not_found' => 'No projet found',
            'not_found_in_trash'=> 'No project in the trash'
        ),
        'public' => true,
        'capability_type' => 'post',
        'supports' => array(
            'title',
            'editor',
            'thumbnail'
        ),
        'has_archive' => true
    )
);
```

Custom Taxonomy

Like custom posts, you can add new taxonomy types that will act just like 'categories' and 'tags'

With the hierarchical property you can choose from the two types of taxonomy : category (hierarchical) or tag (not hierarchical).

Custom Taxonomy

```php
register_taxonomy(
  'type',
  'projet',
  array(
    'label' => 'Types',
    'labels' => array(
    'name' => 'Types',
    'singular_name' => 'Type',
    'all_items' => 'All the types',
    'edit_item' => 'Edit type',
    'view_item' => 'See type',
    'update_item' => 'Update type',
    'add_new_item' => 'add a type',
    'new_item_name' => 'New type',
    'search_items' => 'Search for types',
    'popular_items' => 'Popular used types'
  ),
  'hierarchical' => true
  )
);
```

Custom fields

Custom fields allows you to add additional custom meta datas to your contents, whether it's Articles, Pages or Custom Content.

It is handled by a pair of key/value:
- the key represents the metadata **name**
- The value contains the associated **information**

Custom fields

functions.php

```
add_action('wp_insert_post', 'my_custom_fields');

function my_custom_fields ($post_id) {
    if ( $_GET['post_type'] != 'page' ) {
        add_post_meta($post_id, 'my_custom_field', 'my
default value', true);
    }
    return true;
}
```

In a template :

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

Or :

```
get_post_meta($post_id, $key, $unique);
```

https://codex.wordpress.org/Custom_Fields

Advanced Custom Fields with :
(free or premium version)

https://wordpress.org/plugins/advanced-custom-fields/

http://advancedcustomfields.com/

SQL requests with $wpdb

SQL request with $wpdb

Declare the global Class $wpdb
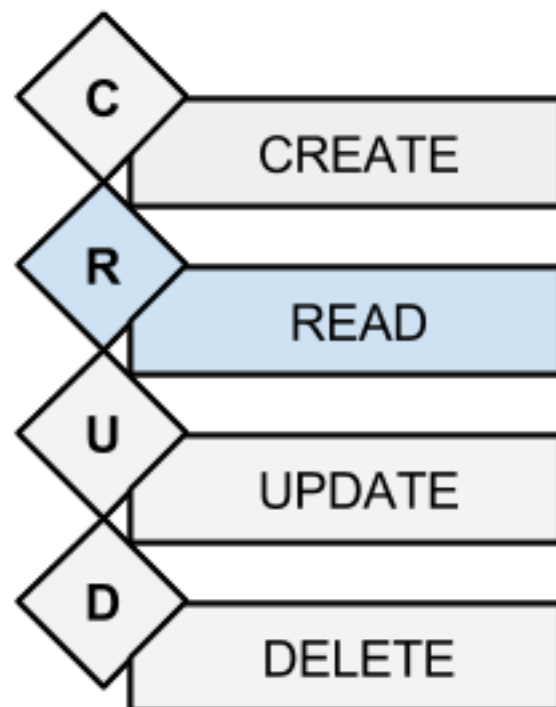
```php
<?php global $wpdb; ?>
```

Get the table prefix

```php
$wpdb->prefix
```

Choose between :

```php
$query = "SELECT * FROM {$wpdb->prefix}comments" ;
$query = "SELECT * FROM $wpdb->comments" ;
```

C	CREATE
R	READ
U	UPDATE
D	DELETE

```php
global $wpdb;

// Requesting the Post table from database
$resultats = $wpdb->get_results("SELECT * FROM {$wpdb->prefix}posts") ;
// Iterate through results
foreach ($resultats as $post) {
  echo $post->post_title ;
  echo '<br/>' ;
}
```

SQL request with $wpdb

Get_results

```php
global $wpdb;

// Request post table from database
$result = $wpdb->get_results("SELECT * FROM {$wpdb->prefix}posts") ;
// Iterates through results
foreach ($result as $post) {
  echo $post->post_title ;
  echo '<br/>' ;
}
```

Get_row

```php
$result = $wpdb->get_row("SELECT post_title, post_date FROM {$wpdb->prefix}posts WHERE ID = 12") ;
echo $result->post_title . ':' . $result->post_date;
```

SQL request with $wpdb

Secure your queries:

```
$id =$_GET['id'];
$result = $wpdb->get_results("SELECT * FROM {$wpdb->prefix}ma_table WHERE id = $id" );
```

```
$id = $_GET['id'];
$result = $wpdb->get_results(
    $wpdb->prepare(
        "SELECT * FROM {$wpdb->prefix}ma_table WHERE id = %d",
        $id
    )
);
```

https://codex.wordpress.org/Class_Reference/wpdb

SQL request with $wpdb

Data manipulation :

$wpdb->insert()

$wpdb->update()

$wpdb->delete()

```php
$wpdb->insert(
    $wpdb->prefix.'my_table',
    array(
        'name' => 'Laetitia',
        'activity' => 'Ski',
        'date' => '2014-12-25',
        'period' => 4
    ),
    array(
        '%s',
        '%s',
        '%s',
        '%d'
    )
);
```

https://codex.wordpress.org/Class_Reference/wpdb

Publish your website

http://www.my-website.com -> http://www.other-website.com

http://localhost -> http://www.my-website.com

```sql
# Change website URL in the database
UPDATE wp_options
SET option_value = replace(option_value, 'http://www.old-
website.com', 'http://www.new-website.com')
WHERE option_name = 'home'
OR option_name = 'siteurl';

# Change GUID URL
UPDATE wp_posts
SET guid = REPLACE (guid, 'http://www.old-website.com',
'http://www.new-website.com');

# Change Medias URL in the posts and pages
UPDATE we_posts
SET post_content = REPLACE (post_content, 'http://www.old-
website.com', 'http://www.new-website.com');

# Change URL in post metas
UPDATE wp_postmeta
SET meta_value = REPLACE (meta_value, 'http://www.old-
website.com','http://www.new-website.com');
```

Other solutions

https://github.com/interconnectit/Search-Replace-DB

https://fr.wordpress.org/plugins/duplicator/

WP security checklist

Backup !
(-> duplicator, BackupBuddy,
BackUpWordPress, Vaultpress…)
Or with your hosting

Use secure password

Change tables préfix

Block Apache directory listing

Disable the default admin account

Hide Wordpress metas in the header (plugin)
and delete the readme.html file
that can give information on your Wordpress
version.

Protect write access to the wp-config.php file
with a good .htaccess

```
<FilesMatch ^wp-config.php$>
        deny from all
</FilesMatch>
```

And with unix correct user rights:

```
chown -R www-data:www-data .
find . -type f -exec chmod 644 {} +
find . -type d -exec chmod 755 {} +
        chmod 644 wp-config.php
```

Hide connexion failed errors :
```
add_filter('login_errors',create_functio
n('$a', "return null;"));
```

Verify your hosting security

Use only verified plugins