

Snippets de estructura

<?php wp_head(); ?> - Indica a Wordpress dónde está el archivo header.php y además ayuda a que los plugins y wordpress inserte el código que requiera para correr algunas instancias como por ejemplo jQuery.

<?php wp_footer(); ?> - Indica a Wordpress cuál es el archivo footer.php y permite que inserte código en esa parte cuando sea necesario cómo cuando un plugin requiera una librería de javascript.

<?php get_header(); ?> - Sirve para conectar los archivos del cuerpo con la header.php del sitio

<?php get_sidebar(); ?> - Conecta el cuerpo con el sidebar.php, si tenemos varias debemos agregar dentro de los parentesis el nombre de la otra barra

Ejemplo:

sidebar-derecha.php se mandaría a llamar como **<?php get_sidebar('derecha'); ?>**

<?php get_footer(); ?> - Conecta al cuerpo con el footer.php

<?php get_search_form(); ?> - Obten el archivo searchform.php y habilita las búsquedas dentro del sitio

<?php wp_nav_menu(array('menu' => 'Top')); ?> - Sirve para mandar a llamar la función Menús, para que funcione hay que añadir la siguiente línea en el *functions.php*:

```
add_theme_support('menus');
```

<body <?php body_class(); ?>> - Asigna una clase específica a cada página de acuerdo a su nombre y a la categoría en la que este

Snippets para obtener información del template:

<?php bloginfo('stylesheet_url'); ?> - Manda a llamar a la hoja de estilos, ésta debe llamarse style.css y estar dentro de la carpeta del template que estemos usando.

<?php bloginfo('template_url'); ?> - Regresará como valor la url donde está nuestro template, ej: <http://misitio.com/wp-content/themes/mitemplate/>

Con esto podremos llamar archivos que estén dentro de la carpeta y mostrarlos en el sitio como hojas de estilo, scripts, imágenes etc.

<?php bloginfo('name'); ?> - Regresa el nombre que le hemos puesto a nuestro sitio al momento de instalarlo

<?php bloginfo('description'); ?> - Regresa el valor que hayamos asignado a la descripción de nuestro sitio.

<?php bloginfo('charset'); ?> - Regresa como valor el charset que esté configurado en el administrador

<?php bloginfo('url'); ?> - Regresa el valor de la url del sitio

<link rel="shortcut icon" type="image/png" href="<?php echo get_template_directory_uri(); ?>/favicon.png"> - Muestra el favicon

<link rel="alternate" type="application/rss+xml" title="<?php bloginfo('name'); ?> Feed" href="<?php echo home_url(); ?>/feed/"> - Enlace al feed rss

Snippets para las notas (loops)

En el loop es dónde se muestran las notas, el siguiente código es el de un loop normal:

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

// Contenido del loop

```
<?php endwhile; ?>  
<?php endif; ?>
```

El siguiente código ayuda a añadir una imagen destacada al post, en caso de no tener una podemos poner una default:

```
<?php if ( has_post_thumbnail() ) {  
the_post_thumbnail();  
} else { ?>  
" />  
<?php } ?>
```

Al igual que con los menús hay que agregar una línea de código en el functions.php:

```
add_theme_support('post-thumbnails');
```

<?php the_title(); ?> - Muestra el título del post

<?php the_permalink(); ?> - Regresa la url del post

<?php the_author_posts_link(); ?> - Muestra el nombre del autor y enlaza a el listado de sus notas

<?php the_category(); ?> - Regresa las categorías en las que está publicado el post, podemos determinar como separa las categorías poniendo un valor entre los paréntesis, ejemplo: **<?php the_category(' '); ?>** regresa: Categoría, Categoría 2, etc

<?php the_time(); ?> - Regresa la hora en la que fue publicado el post

<?php the_date(); ?> - Regresa la fecha en la que fue publicado el post

<?php the_excerpt(); ?> - Muestra un resumen de la nota

<?php the_content(); ?> - Muestra todo el contenido de una nota

<?php posts_nav_link(); ?> - Muestra los enlaces a la siguiente y anterior página de post

Podemos usar un single.php para cada categoría distinta, como si tenemos una categoría de juegos y otra de reseñas y ambas son distintos estilos, para eso usamos el siguiente código en el **single.php**

```
<?php  
if (in_category(1)) { // Si el post pertenece a la cat 1, redirigir a single-1.php  
include(TEMPLATEPATH . '/single1.php');  
} elseif (in_category(2)) { // Si el post pertenece a la cat 2, redirigir a single-2.php  
include(TEMPLATEPATH . '/single2.php');  
} else { // Si el post pertenece a cualquier otra categoría, redirigir a single-3.php  
include(TEMPLATEPATH . '/single3.php'); } ?>
```

También podemos crear templates específicos para cada página, para eso sólo hay que agregar las siguientes líneas al principio del archivo que estemos creando, ya después podremos asignar esa plantilla a la página desde el administrador:

```
<?php  
/*  
Template Name: Nombre de la plantilla  
*/  
?>
```

De igual forma, si dentro de cualquier parte de nuestro tema necesitamos insertar un archivo adicional, cómo por ejemplo un script para mostrar tweets, podemos hacerlo con la siguiente línea:

<?php include(TEMPLATEPATH . '/nombre_del_Archivo.php') ?>

Para mostrar el formulario de comentarios en nuestras notas debemos agregar la siguiente línea dónde queramos que aparezca:

<?php comments_template(); ?>

Podremos editar su estilo en un archivo que llamemos comments.php

<?php next_post_link(' %link ') ?> - Muestra el enlace al siguiente post

<?php previous_post_link('%link') ?> - Muestra el enlace al post anterior

<?php wp_tag_cloud('smallest=1&largest=9&'); ?> - Muestra una nube de etiquetas

Podemos personalizar la información de nuestra plantilla en el archivo style.css con las líneas:

```
/*  
Theme Name: Nombre del tema  
Description: Descripción del tema  
Theme URI: http://urldelsitio.com  
Version: 2.0  
Author: Nombre del autor  
Author URI: http://urldelautor.com  
*/
```

Para registrar los widgets de la barra lateral será necesario poner el siguiente código en el functions.php:

```
if ( function_exists('register_sidebar') )  
    register_sidebar(array('name'=>'nombre_de_la_sidebar',  
        'before_title' => '<div class="nombre_de_la_clase_titulo">',  
        'after_title' => '</div>',  
        'before_widget' => '<div class="contenedor_del_widget">',  
        'after_widget' => '</div>',  
    ));
```

Una vez que ya está hecho podemos poner el widget en nuestro tema con el siguiente código:

```
<?php if ( !function_exists('dynamic_sidebar')  
dynamic_sidebar('nombre_de_la_sidebar') ) : ?>  
<?php endif; ?>
```

Snippets Varios

Loop para Multiples

```
<?php
// The Query
$the_query = new WP_Query( $args );
// The Loop
while ( $the_query->have_posts() ) : $the_query->the_post(); ?>
    <!--Contenido del post -->
<?php endwhile;
// Reset Post Data
wp_reset_postdata();
?>
```

// Limitar tamaño del extracto

```
function excerpt($num) {
$limit = $num+1;
$excerpt = explode(' ', get_the_excerpt(), $limit);
array_pop($excerpt);
$excerpt = implode(" ",$excerpt)."...";
echo $excerpt;
}
```

```
function content($num) {
$theContent = get_the_content();
$output = preg_replace('/<img[^>]+./', "", $theContent);
$limit = $num+1;
$content = explode(' ', $output, $limit);
array_pop($content);
$content = implode(" ",$content)."...";
echo $content;
}
```

// Limitar tamaño del extracto

```
function excerpt($num) {
$limit = $num+1;
$excerpt = explode(' ', get_the_excerpt(), $limit);
array_pop($excerpt);
$excerpt = implode(" ",$excerpt)."...";
echo $excerpt;
}
<?php excerpt('20'); ?>
```

```
function content($num) {  
    $theContent = get_the_content();  
    $output = preg_replace('/<img[^>]+./', "", $theContent);  
    $limit = $num+1;  
    $content = explode(' ', $output, $limit);  
    array_pop($content);  
    $content = implode(" ", $content). "...";  
    echo $content;  
}
```

```
<?php content('20'); ?>
```