

Buenas Prácticas en WordPress

Laura Fernández Lledó

@Softwartive



¿QUÉ SON LAS BUENAS
PRÁCTICAS EN LA
PROGRAMACIÓN?

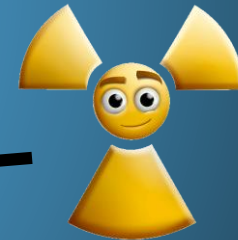


Nivel 1 EI WP - CONFIG

```
// ** Ajustes de MySQL. Solicita estos datos a tu proveedor de alojamiento web. ** //  
/** El nombre de tu base de datos de WordPress */  
define('DB_NAME', 'nombredetubasededatos');  
  
/** Tu nombre de usuario de MySQL */  
define('DB_USER', 'nombredeusuario');  
  
/** Tu contraseña de MySQL */  
define('DB_PASSWORD', 'contraseña');
```



```
define('AUTH_KEY', 'pon aquí tu frase aleatoria'); // Cambia esto por tu frase aleatoria.  
define('SECURE_AUTH_KEY', 'pon aquí tu frase aleatoria'); // Cambia esto por tu frase aleatoria.  
define('LOGGED_IN_KEY', 'pon aquí tu frase aleatoria'); // Cambia esto por tu frase aleatoria.  
define('NONCE_KEY', 'pon aquí tu frase aleatoria'); // Cambia esto por tu frase aleatoria.  
define('AUTH_SALT', 'pon aquí tu frase aleatoria'); // Cambia esto por tu frase aleatoria.  
define('SECURE_AUTH_SALT', 'pon aquí tu frase aleatoria'); // Cambia esto por tu frase aleatoria.  
define('LOGGED_IN_SALT', 'pon aquí tu frase aleatoria'); // Cambia esto por tu frase aleatoria.  
define('NONCE_SALT', 'pon aquí tu frase aleatoria'); // Cambia esto por tu frase aleatoria.
```



```
/**#@*-*/
```

```
/**
```

```
* Prefijo de la base de datos de WordPress.
```

```
*
```

```
* Cambia el prefijo si deseas instalar multiples blogs en una sola base de datos.
```

```
* Emplea solo números, letras y guión bajo.
```

```
*/
```

```
$table_prefix = 'wp_';
```





Nivel 2 La pureza del content

Añadir objeto

Visual HTML

b i link b-quote del ins img ul ol li code more cerrar etiquetas

Número de palabras: 0 Borrador guardado a las 11:22:53.

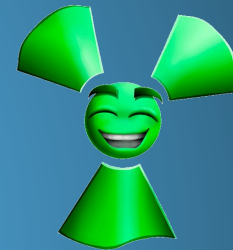


Nota: Tampoco en bloques de Gutenberg



Nivel 3 El functions.php

- Encolar los scripts y estilos
- Funciones que sean del tema
- Child Themes (function_exists)
- Theme supports
- Parciales





Nivel 4 Creando un nuevo tema

- Uso de parciales en las plantillas
- Pre procesadores de hojas de estilos
- Código Estándar
- HTML validado



Código estándar

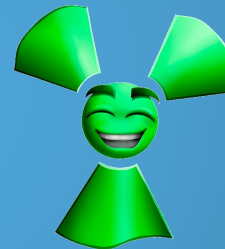
Indentación

Usar tabuladores y no espacios.

Excepciones:

```
1 [tab]$foo = 'somevalue';  
2 [tab]$foo2 = 'somevalue2';  
3 [tab]$foo34 = 'somevalue3';  
4 [tab]$foo5 = 'somevalue4';
```

```
$query = new WP_Query( array( 'ID' => 123 ) );  
  
$args = array(  
[tab]'post_type' => 'page',  
[tab]'post_author' => 123,  
[tab]'post_status' => 'publish',  
);  
  
$query = new WP_Query( $args );
```



De esta forma permite leerse y entenderse mejor.

Además recordad que en un array debe ir cada item en una línea



Código estándar

Abriendo y cerrando PHP

Multilínea:

```
1 function foo() {  
2     ?>  
3     <div>  
4     <?php  
5     echo bar(  
6         $baz,  
7         $bat  
8     );  
9     ?>  
10    </div>  
11    <?php  
12 }
```

Una única línea

```
1 <input name="<?php echo esc_attr( $name ); ?>" />
```

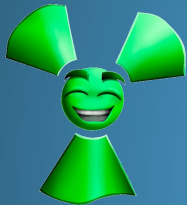
```
1 if ( $a === $b ) { ?>  
2 <some html>  
3 <?php }
```





Código estándar

No uses la forma corta de PHP



```
1 | <?php ... ?>  
2 | <?php echo $var; ?>
```



```
1 | <? ... ?>  
2 | <?= $var ?>
```

Esta forma de PHP podría ser incompatible con algunos servidores



Código estándar

SQL

Las funciones como UPDATE esperan que los parámetros se pasen escapados, por ello es recomendable el uso del Prepare()

```
$var = "dangerous'"; // raw data that may or may not need to be escaped
$id = some_foo_number(); // data we expect to be an integer, but we're not certain

$wpdb->query( $wpdb->prepare( "UPDATE $wpdb->posts SET post_title = %s WHERE ID = %d", $var, $id ) );
```

%s está siendo usado como un string placeholder y %d como enteros.

El beneficio de eso es no tener que acordarse de usar manualmente el esc_sql()



Código estándar

SQL

Intenta evitar tocar la base de datos directamente. WordPress ya tiene funciones que hacen estas tareas de manera óptima.

¡No uses wp-hackers!





Código estándar

Convenciones para nombres

Cosas como usar minúsculas para variables, acciones/filtros y funciones de nombres (nunca camelCase)

Ejemplos:

```
function some_function( $some_variable) { [...] }
```

```
class Walker_Category extends Walker { [...] }
```

```
define ( 'DOING_AJAX', true);
```

```
my-plugin-name.php
```



Código estándar

Argumentos y funciones autoexplicativas

```
function eat( $what, $slowly = true ) {  
  ...  
}  
eat( 'mushrooms' );  
eat( 'mushrooms', true ); // what does true mean?  
eat( 'dogfood', false ); // what does false mean? The opposite of true?
```

```
function eat( $what, $speed = 'slowly' ) {  
  ...  
}  
eat( 'mushrooms' );  
eat( 'mushrooms', 'slowly' );  
eat( 'dogfood', 'quickly' );
```

De esta forma detallamos un poco más a que nos referimos como speed

```
function eat( $what, $args ) {  
  ...  
}  
eat ( 'noodles', array( 'speed' => 'moderate' ) );
```



Código estándar

Hooks dinámicos

Los hooks dinámicos se deben llamar mediante la interpolación de su tag name.

```
do_action( "{$new_status}_{$post->post_type}", $post->ID, $post );
```

Las variables deben ir entre { } y dobles comillas



Código estándar

Condiciones Yoda

```
if ( true === $the_force ) {  
    $victorious = you_will( $be );  
}
```

Si se hace de manera normal y se comete el error de (\$the_force = true) hará una asignación y es perfectamente válido. Pero si está al revés y se causa el error (true = \$the_force) dará un error.



Código estándar

Código Inteligente

Es más importante que sea legible que breve

```
isset( $var ) || $var = some_function();
```

```
if ( ! isset( $var ) ) {  
    $var = some_function();  
}
```

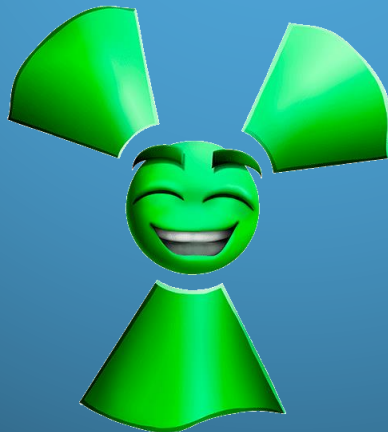
No usar asignaciones fuera de condicionales

```
$data = $wpdb->get_var( '...' );  
if ( $data ) {  
    // Use $data  
}
```




Algunos Consejitos

- Comprueba que todas las imágenes tienen la etiqueta alt
- No uses contrastes de colores que dificulten la lectura.
- Mantén el WordPress actualizado al igual que sus plugins
- Usa PHP 7
- Los comentarios muchas veces ayudan a la legibilidad.
- Usa un sistema de versiones
- Hagas lo que hagas siempre piensa que un código limpio ayuda a futuros cambios y modificaciones tanto si lo haces tu como tu equipo



¿ PREGUNTAS?

