**DESARROLLANDO EN WORDPRESS**

## GUÍA 14

**Plan de trabajo**

|  |  |  |
| --- | --- | --- |
| **TAREA** | **DESCRIPCIÓN** | **FECHA DE ENTREGA** |
| Lectura  | Revisa la siguiente guía que contiene videos y lectura.  |  |
| Test  | Realiza el test para evaluar los conocimientos adquiridos en Google Classroom |  |

## DESARROLLANDO EN WORDPRESS

A continuación se presentan distintos recursos e información para que te familiarices con las temáticas de Wordpress.

**Video Introductorio**

Revisa los siguientes videos para entender el uso de Wordpress.

**Video 1: Estructura de Wordpress**

https://youtu.be/gJZehyPyA4M

## DEFINICIÓN DE CONCEPTOS

Una vez revisados los videos, lee con atención la siguiente información sobre recursos de Wordpress.

**Núcleo**

El núcleo de WordPress es la base del CMS, que aporta la mayor parte de la lógica y las funcionalidades. No se debe modificar, ya que esto podría causar un funcionamiento inesperado. Además, cualquier modificación podría perderse al instalar una nueva actualización.

**El Codex**

Es la guía de referencia oficial mantenida por la comunidad y traducida, al menos parcialmente, a diversos idiomas. Sin embargo, hay que tener en cuenta que a veces la documentación no está lo suficientemente actualizada como debiera. Pese a eso, sigue siendo una referencia ineludible a la hora de trabajar con WordPress. Más referencias en: [**https://codex.wordpress.org/**](https://codex.wordpress.org/)

**Referencia a funciones**

En esta página se puede buscar fácilmente una función y consultar su descripción, los parámetros, el valor devuelto, el changelog y la propia definición de la función. Esta es la misma información que presentan los archivos PHP, y no es casualidad.

**Coding Standards**

Con el objetivo de que WordPress siga siendo un CMS de código legible y sencillo, nacen los Coding Standards.

**Convenciones sobre nombres:**

* Los nombres de las variables se escriben en letras minúsculas.
* Las clases comienzan las palabras en mayúsculas.
* Las constantes están escritas íntegramente en mayúsculas.
* En los tres casos anteriores, las palabras se separan mediante guiones bajos (**"snake\_case"**).
* En los nombres de archivo las palabras se separan con guiones medios.

**Uso de espacios**

* Se deben usar espacios a ambos lados de los paréntesis de los condicionales y bucles.
* Se deben usar espacios en los lados interiores de los paréntesis al definir y llamar a funciones.
* Se deben usar espacios a ambos lados de los comparadores.

**Condiciones "Yoda"**

En las comparaciones lógicas se deben poner las variables en la parte izquierda y las constantes, literales o funciones en el lado derecho. Así, por ejemplo, en caso de olvidar un signo igual **(=)**, al hacer una comparación de igualdad **(==)** se lanzará un error en lugar de asignarse un valor de forma errónea.

## FUNCIONAMIENTO WORDPRESS

Pese a la gran variedad de personalización que se puede hacer en WordPress, lo cierto es que casi todas las páginas trabajan de una forma muy similar.

**La base de datos**

Además del propio código del núcleo, es interesante ver cómo está estructurada la base de datos para comprender cómo se almacenan los datos. La siguiente imagen muestra la estructura de la base de datos de Wordpress, incorporando las siguientes tablas:

* wp\_commentmeta
* wp\_comments
* wp\_links
* wp\_options
* wp\_postmeta
* wp\_posts
* wp\_terms
* wp\_termmeta
* wp\_term\_relationships
* wp\_term\_taxonomy
* wp\_usermeta
* wp\_users

¿Quieres más información? Haz clic [aquí](https://kinsta.com/es/base-de-conocimiento/base-de-datos-wordpress/) para entender más sobre el funcionamiento de la base de datos de Wordpress.



**Query y Loop**

El llamado **"Loop"** es una de las partes principales de las páginas de Wordpress. Es la parte que se usa para mostrar cada uno de los posts, independientemente del tipo de página en la que se encuentre el usuario. Como su nombre lo indica, se trata básicamente de un bucle, que recorre los posts a manejar (ya sean varios o uno solo) y actúa sobre cada uno de ellos de la forma que se precise.

**Hooks**

La forma en la que una pieza de código interactúa con otra en WordPress se denomina hooks. Los usan los plugins para interactuar con el núcleo de la plataforma o con otros plugins, pero también se usan dentro del propio núcleo para relacionar unas partes con otras. Son eventos disparados por acciones y filtros, que permiten asociar las propias funciones.