

Aplicaciones Informáticas – Programación Web

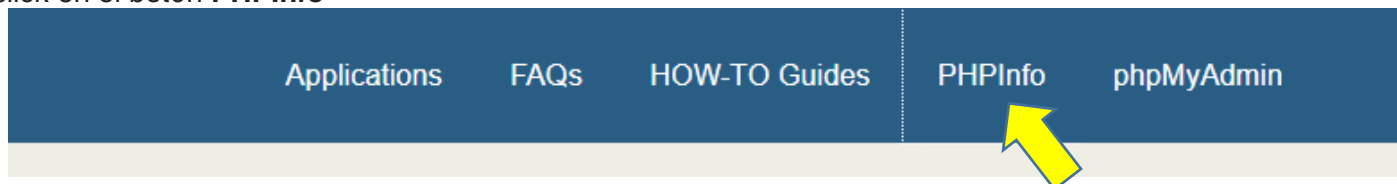
Vamos a comenzar a trabajar con el servidor local **Apache** que pondremos en línea con el Panel de Control de Xampp.

Paso 1:

Arrancamos el Panel de Control de Xampp y ponemos en funcionamiento el Servidor MySQL y el Servidor Apache.

Paso 2:

Entraremos en el explorador de internet y escribiremos localhost y en la pantalla de bienvenida, haremos click en el botón **PHPInfo**



Nos desplazaremos hacia abajo hasta encontrar la sección **Apache Environment**. Ahí vamos a buscar la variable **DOCUMENT_ROOT** y vamos a tomar nota de la carpeta donde está instalado el **Home** de Apache, que es la carpeta **htdocs**, para utilizarlo posteriormente.

Inicio:

Para comenzar a programar en Web, primero es necesario conocer algunas palabras y conceptos básicos con los que nos encontraremos durante este trayecto: Dominio, hosting, maquetación, diseño responsivo, Frontend, Backend, MVC, framework, Diseño en cascada, etc.

Dominio:

Un dominio de internet es la dirección **única** (no puede repetirse) que identifica a un sitio web. Simplificándolo, podríamos decir que es el nombre **único y exclusivo** que tiene tu página en internet. El dominio que está vinculado a tu sitio en internet, permite que pueda ser accedida desde el navegador.

Normalmente los dominios se muestran después del signo @ en las direcciones de correo y después de **www.** en las direcciones web. Por ejemplo, el nombre de dominio *example.com* se podría traducir a la dirección física 198.102.434.8. Otros ejemplos de nombres de dominio podrían ser *google.com* y *wikipedia.org*.



Hosting:

Es el servidor donde se alojan las páginas web. En nuestro caso, el servidor Apache servirá de hosting local para alojar nuestras páginas web.

¿Qué es el Frontend?

Frontend es la parte de un programa o una aplicación a la que accede directamente un usuario para interactuar con ella. El Frontend está compuesto de tecnologías de diseño y desarrollo web que son ejecutadas por el navegador.

Para ser un *Frontend Developer* se debe conocer fundamentalmente HTML y CSS, que son los lenguajes de maquetación que nos permiten definir la estructura y estilos de una página web. También se debe conocer JavaScript, que es un lenguaje de programación que nos permite programar la lógica de nuestra aplicación, recibiendo las solicitudes de los usuarios y enviárselos al Backend. También se encarga de recibir la información desde el Backend y se la envía al Frontend.

Para utilizar estas tecnologías, se pueden utilizar algunos frameworks, librerías o preprocesadores que ayudan a crear todo tipo de interfaces de usuario. Ejemplos de ellos son: React, Vue, Angular, Svelte, Bootstrap, Foundation, Sass, Less, Stylus, PostCSS.



¿Qué es el Backend?

En la arquitectura de programación en capas, Backend es la capa encargada del acceso a los datos. Y contiene la lógica de la aplicación que los maneja. No puede ser accedida directamente por el usuario, sino a través del Frontend o desde otra aplicación. El Backend también accede al servidor y es el encargado de *entender* la forma en que el navegador solicita cosas. Algunos de los lenguajes de programación para Backend son Python, Node.js, PHP, Go, Ruby y C#. Tal como en el Frontend, estos lenguajes tienen diferentes frameworks que te permiten trabajar mejor según el proyecto que estás desarrollando, como Django, Flask, Express.js, Laravel, Symphony Framework, Ruby on Rails y ASP.Net.

¿Cómo se relacionan Frontend y Backend?

Cuando en la barra del navegador colocamos una URL, estamos indicando que queremos ver una página web. El servidor evalúa la petición que le están haciendo. Si nuestra petición requiere buscar datos en la base de datos del sitio, conectará con la base de datos y realizará lo indicado. Con eso, el Backend devuelve la respuesta para que la reciba el Frontend y la acomode en la interfaz del sitio, que es lo que ve el usuario.



Investigar y responder en un documento en el Entregable WEB-001:

- 1- ¿Qué es el lenguaje HTML? ¿Por qué no es un lenguaje de programación?
- 2- ¿Qué son las “etiquetas” HTML?
- 3- ¿Cuáles son las principales etiquetas y para qué sirven?
- 4- ¿Cuál es la estructura mínima de una página HTML? (Etiquetas que no pueden faltar)
- 5- Ir a la carpeta Home de Apache **htdocs** (que tomaste nota en el Paso 2) y crear ahí adentro la carpeta **6b**. Creá un archivo de texto que contenga estas etiquetas:

```
<html>
<head>
<title>Hola Mundo</title>
</head>
<body>
<H1>Hola Mundo!!!</H1>
<H2>Hola Mundo!!!</H2>
<H3>Hola Mundo!!!</H3>
</body>
</html>
```

- 6- Guardalo dentro de la carpeta **6b** con el nombre **holamundo.html**
- 7- Abrió el navegador web y colocá la siguiente URL: **localhost/6b/holamundo.html**
- 8- Hacé una impresión de pantalla y colocala junto a tus respuestas en el entregable de esta actividad.

No olviden colocar el nombre de los integrantes del trabajo

