



# PHP | Intro

## Después de esta lección podrás:

1. Explicar qué es PHP.
2. Ejecutar PHP en tu máquina local.
3. Conocer los conceptos básicos de PHP.

## Presentación

En este curso de nivelación veremos qué se puede hacer con **PHP**, la instalación del intérprete, así como conocer conceptos básicos del lenguaje.

Nos saltaremos muchos conceptos que se darán en el bootcamp, pero servirá como un buen inicio para sentar las bases de este lenguaje.

Tal vez os estaréis preguntando si vale la pena o no aprender este lenguaje de programación, os queremos decir que... **SÍ!!**

Se estima que PHP es usado por el 83,5% de todos los websites cuyo lenguaje de programación del lado de servidor es conocido.

Un ejemplo de sitios que utilizan **PHP** son los siguientes: Facebook, Wikipedia, WordPress, tumblr, Slack, etc... no son precisamente pequeños verdad?

Esa es la gran versatilidad y potencia del lenguaje y su ecosistema, el poder ser usado para crear webs de tamaños más pequeños, hasta proyectos inmensos!

# Introducción

## ¿Qué es PHP?

**PHP** (acrónimo recursivo de PHP: Hypertext Preprocessor) es un lenguaje de código abierto muy popular especialmente adecuado para el desarrollo web y que puede ser incrustado en **HTML**.

Bien, pero ¿qué significa realmente? Lo mejor es ver un ejemplo:

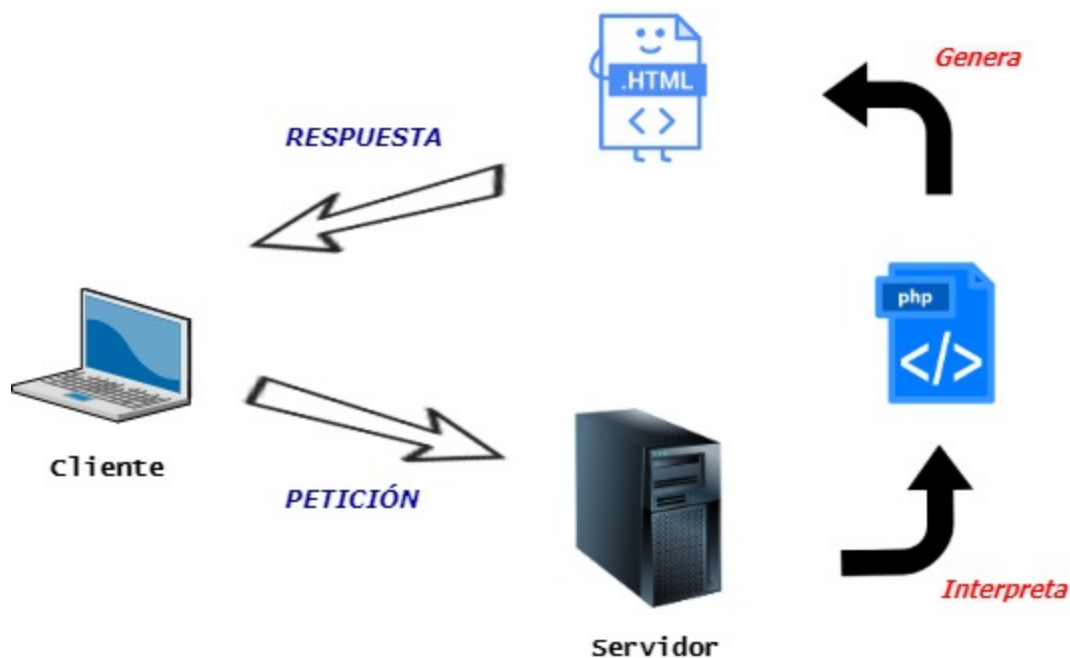
```
<!DOCTYPE HTML>
<html>
  <head>
    <title>Ejemplo</title>
  </head>
  <body>
    <?php
      echo "¡Hola, soy un script de PHP!";
    ?>
  </body>
</html>
```

En lugar de usar muchos comandos para mostrar HTML, las páginas de PHP contienen HTML con código incrustado que hace «algo» (en este caso, mostrar «¡Hola, soy un script de PHP!»). El código de PHP está encerrado entre las etiquetas especiales de comienzo y final **<?php** y **?>** que permiten entrar y salir del «modo PHP».

Lo que distingue a PHP de algo del lado del cliente como Javascript es que el código es ejecutado en el servidor, generando HTML y enviándolo al cliente. El cliente recibirá el resultado de ejecutar el script, aunque no se sabrá el código subyacente que era. El servidor web puede ser configurado incluso para que procese todos los ficheros HTML con PHP, por lo que no hay manera de que los usuarios puedan saber qué se tiene debajo de la manga.

El objetivo final es generar páginas dinámicas. Es decir, lo habitual es que PHP haga consulta a bases de datos, accede a ficheros de texto, haga a su vez peticiones a otros servicios... lo que provocará que una misma petición a una URL con un margen de tiempo de unos minutos, devuelva resultados diferentes. El ejemplo más claro puede ser cualquier portal de noticias, que está continuamente actualizándose, o por ejemplo cualquier red social.

Este sería un esquema que refleja el funcionamiento de PHP:



Lo mejor de utilizar PHP es su extrema simplicidad para el principiante, pero a su vez ofrece muchas características avanzadas para los programadores profesionales.

Aunque el desarrollo de PHP está centrado en la programación de scripts del lado del servidor, se puede utilizar para muchas otras cosas.

## ¿Qué puede hacer PHP?

Existen principalmente tres campos principales donde se usan scripts de PHP.

- **Scripts del lado del servidor.** Este es el campo más tradicional y el foco principal. Son necesarias tres cosas para que esto funcione. El intérprete de PHP, un servidor web y un navegador web. En este escenario el servidor web tendrá integrado el intérprete de PHP (del mismo modo que un navegador web tiene integrado un intérprete de JavaScript). El modo de funcionamiento es el descrito anteriormente: la petición se realiza desde el navegador web, el servidor web procesa dicha petición mediante el intérprete de PHP, y devuelve el resultado final (que típicamente será una página HTML) al navegador web.
- **Scripts desde la línea de comandos.** Se puede crear un script de PHP y ejecutarlo sin necesidad de un servidor o navegador. Solamente es necesario el analizador de PHP para utilizarlo de esta manera. Este tipo de uso es ideal para scripts que se ejecuten con regularidad empleando cron (en \*nix o Linux) o el Planificador de tareas (en Windows). Estos scripts también pueden usarse para tareas simples de procesamiento de texto.
- **Escribir aplicaciones de escritorio.** Sí, has oído bien, se pueden realizar aplicaciones de escritorio en PHP, pero eso NO lo verás en este curso ya que es algo... digamos... raro 😊

PHP puede usarse en todos los sistemas operativos principales y mayoritarios: Linux, Windows y Mac OS X entre otros y admite la mayoría de servidores web de hoy en día, incluyendo Apache, IIS, y muchos otros. Esto incluye cualquier servidor web que pueda utilizar el binario de PHP FastCGI, como lighttpd y nginx.

## Instalación

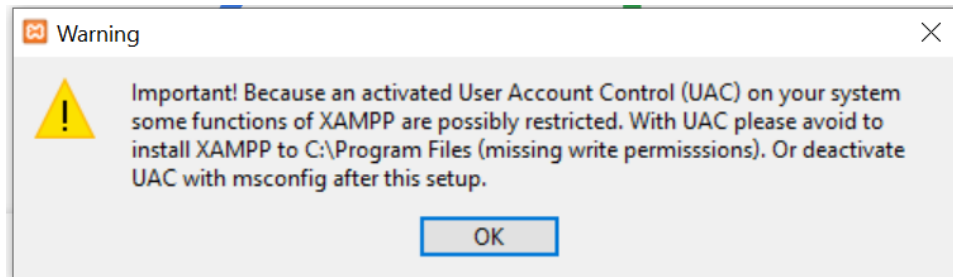
Al contrario que JavaScript, PHP necesita de un proceso de instalación para poder trabajar con él. Este proceso varía dependiendo en qué Sistema Operativo trabajemos.

### Instalación en Windows

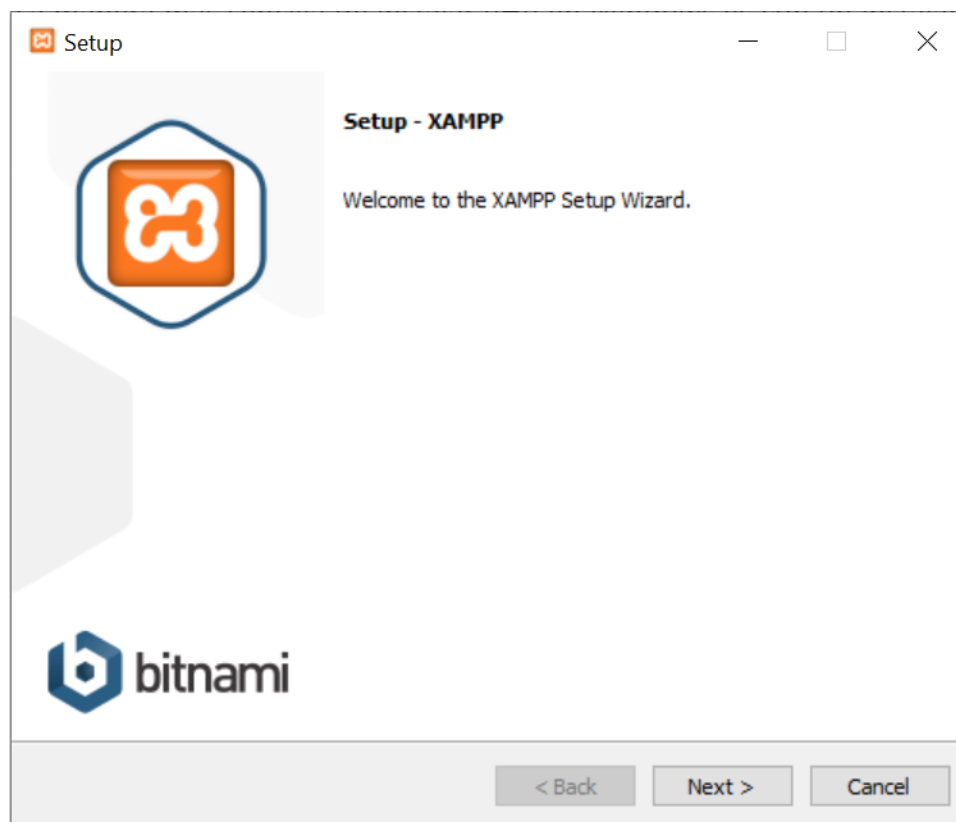
La manera más sencilla de instalar PHP en Windows es con un instalador con ciertas aplicaciones ya preparadas para el desarrollo. Uno de ellos es **XAMPP** de los más famosos entornos de desarrollo, que incluye además del intérprete PHP, un servidor Web (Apache) y una base de datos (MySQL).

A continuación mostramos paso por paso cómo se instala. Por supuesto lo primero a hacer es descargar el programa desde el enlace proporcionado y ejecutarlo.

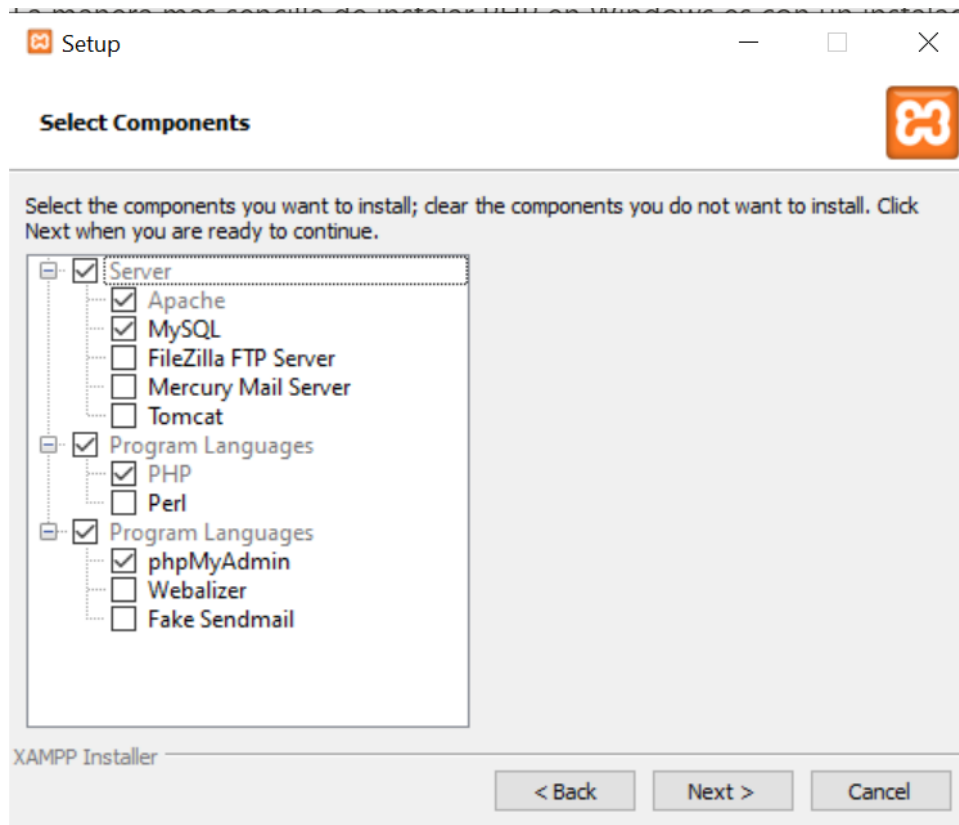
Es posible que os de el siguiente mensaje, pero no hay que alarmarse, simplemente se pulsa en OK y se puede continuar con la instalación.



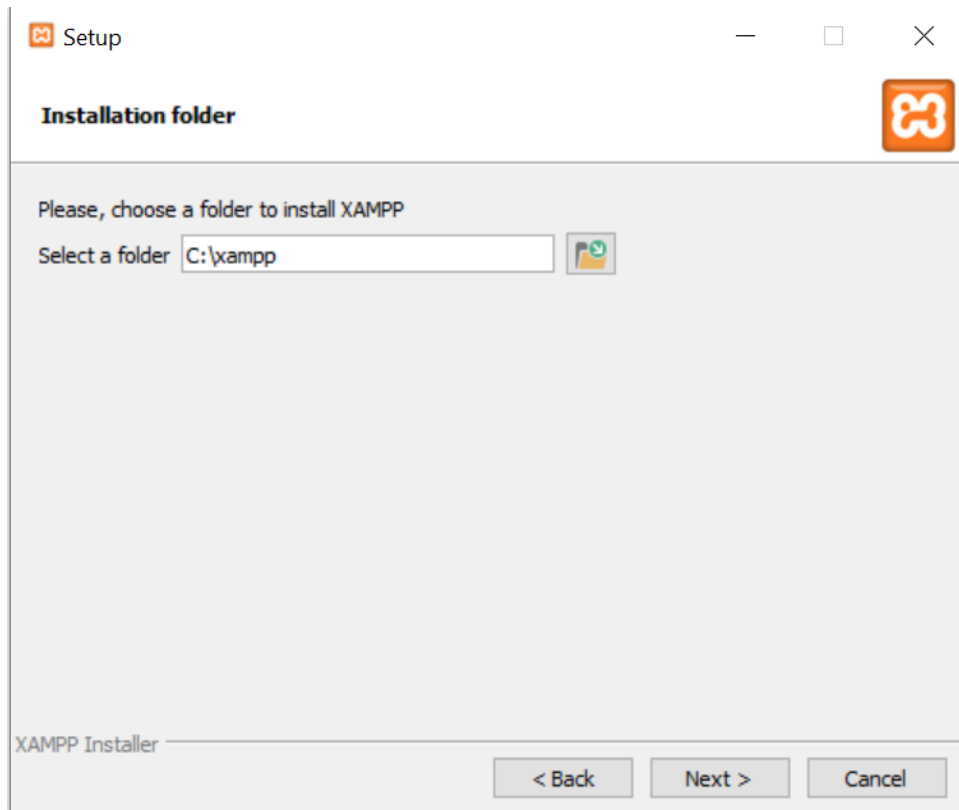
En este momento comenzaremos con la instalación:



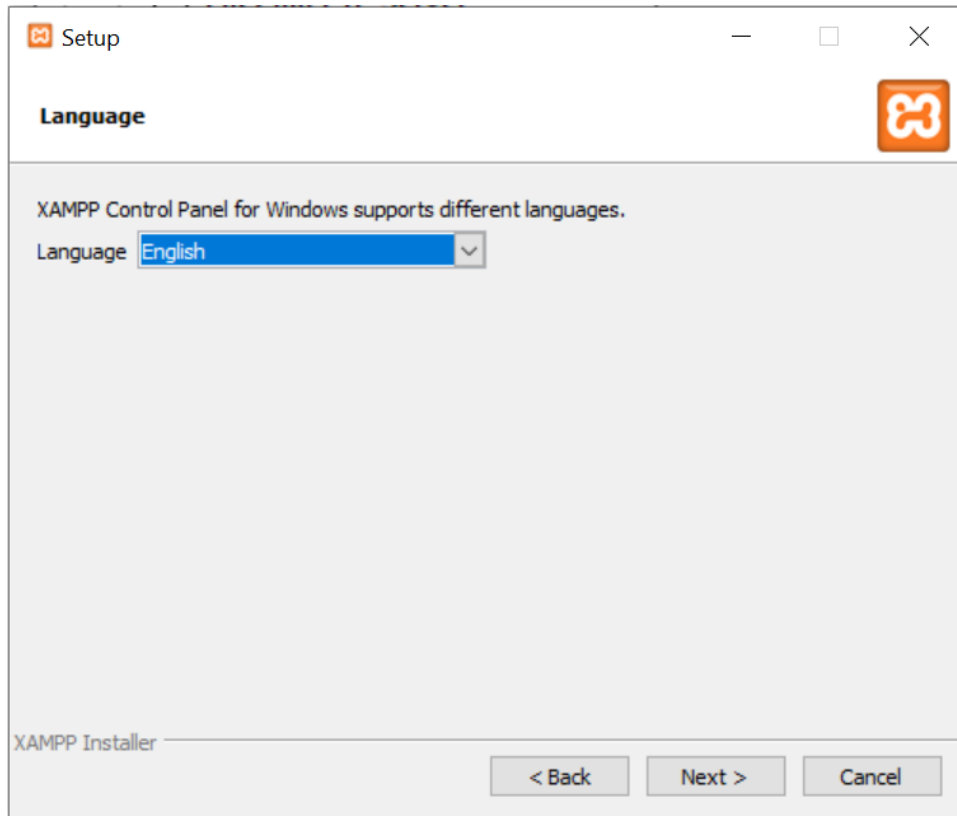
En el siguiente paso mostrará una serie de opciones para que elijamos qué queremos instalar. Por defecto están marcadas todas, y perfectamente podríamos dejarlo así. Sin embargo con que marquemos las 4 opciones mostradas abajo (Apache, MySQL, PHP y phpMyAdmin) sería suficiente.



Ahora preguntará en qué carpeta queremos instalar XAMPP. Nuestra recomendación es que dejes la opción por defecto, pero siéntete libre de cambiar la carpeta si lo ves necesario.

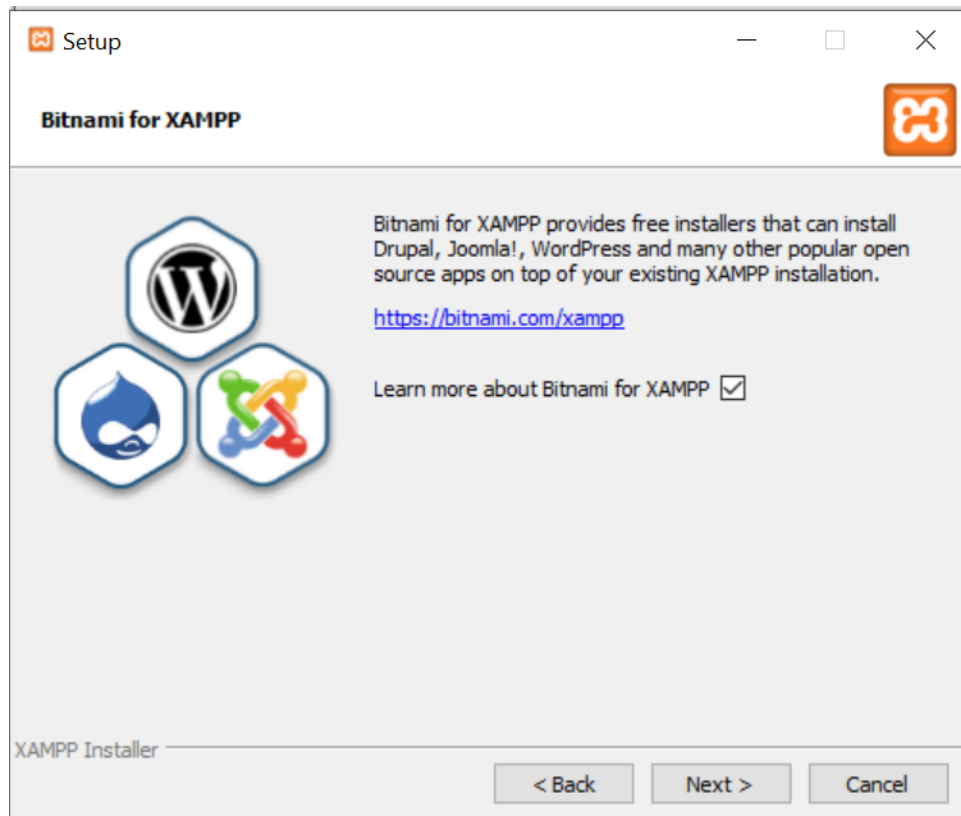


Por último pregunta por el idioma del panel de control, aunque no hay muchas posibles opciones. No te preocupes porque la interacción con XAMPP será mínima.

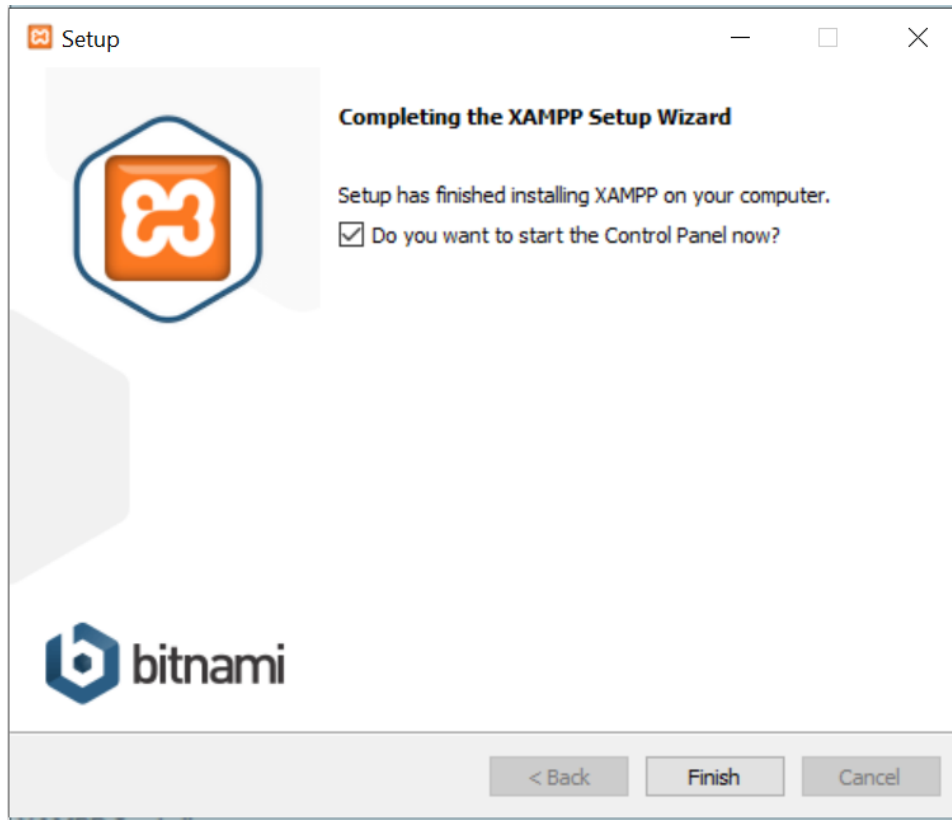


Antes de comenzar el proceso de instalación como tal, te pregunta si quieres saber más sobre Bitnami (la empresa que se encarga del mantenimiento de XAMPP). Simplemente te abrirá su página web en el navegador. Podemos saltarnos este paso simplemente desmarcando la opción de "Learn more about Bitnami for XAMPP".

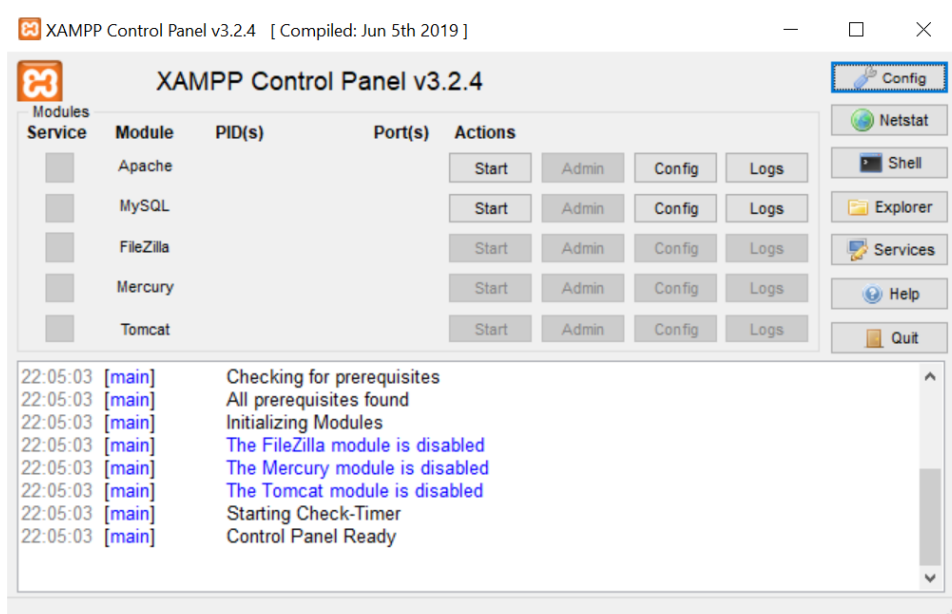




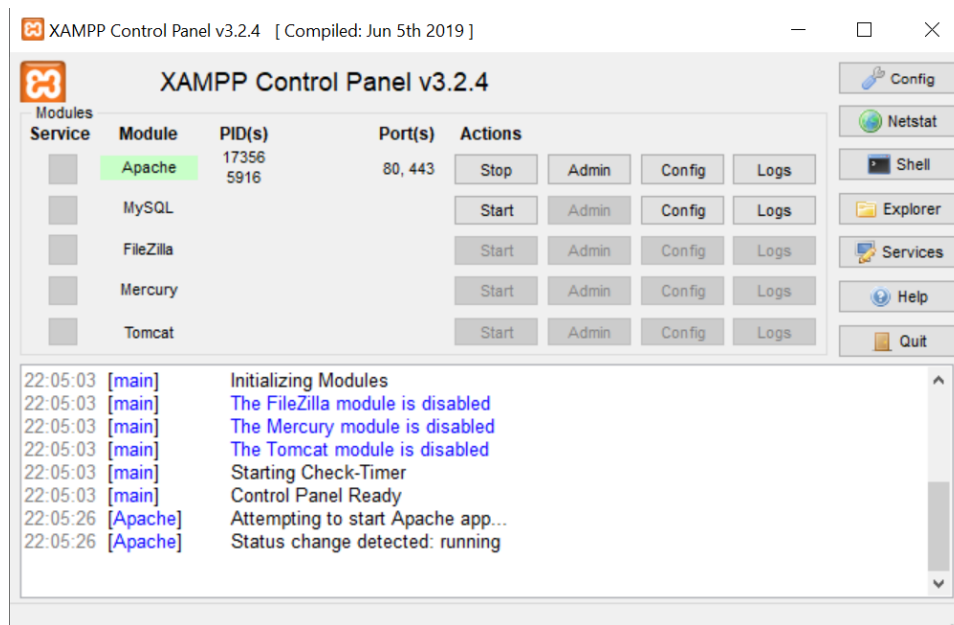
Una vez finalizada la instalación, mostrará la siguiente pantalla donde pregunta si quieres abrir el panel de control. Recomendamos que sí lo abras para dejar funcionando el servidor web.



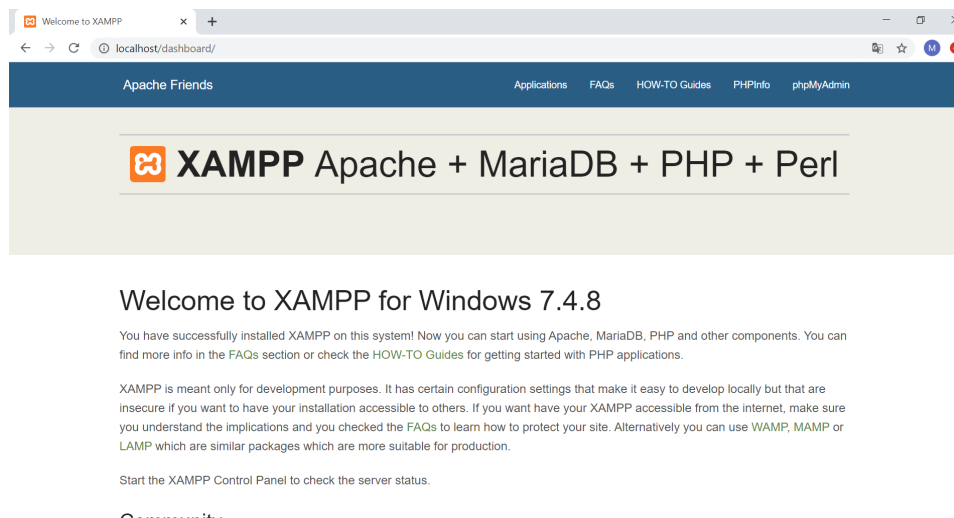
Lo que te aparecerá ahora es el panel de control de XAMPP. Lo único que tienes que hacer es pulsar sobre el botón de Start de Apache.



Si todo ha ido aparecerá la siguiente pantalla en donde indica que Apache está funcionando.

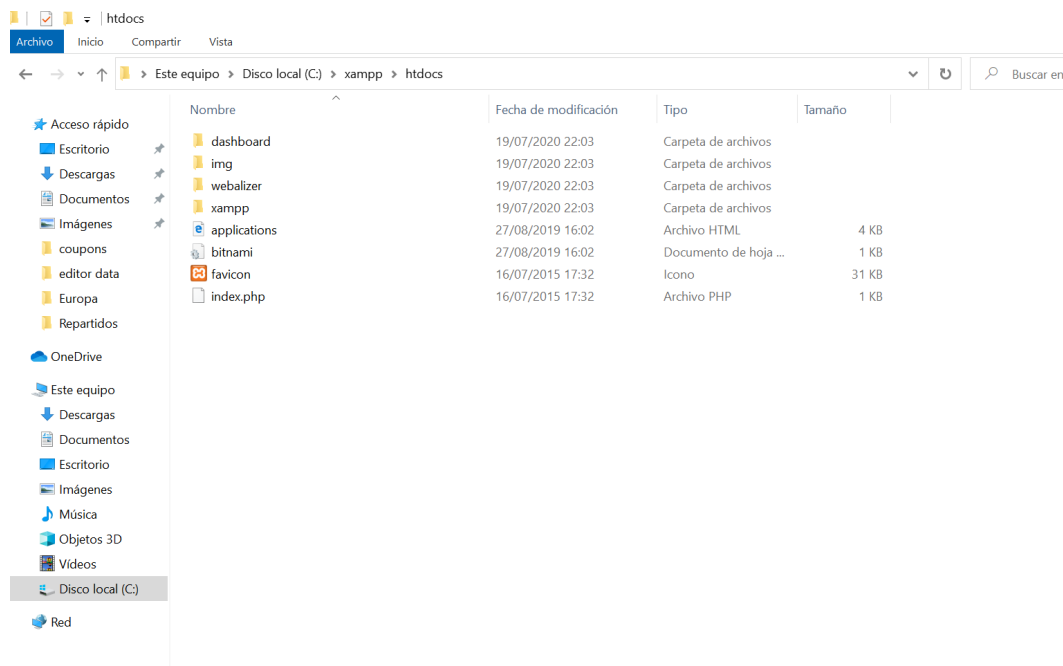


Y para terminar de asegurarnos que todo está correcto, basta con irnos a nuestro navegador favorito (Chrome, Firefox, Safari, Microsoft Edge...) y escribir en la barra superior simplemente "localhost". Si ha ido todo bien aparecerá la siguiente pantalla:



A partir de ahora puedes empezar a crear tus ficheros PHP en la carpeta "C:\xampp\htdocs" (si es que mantuviste la carpeta por defecto en la instalación).

Verás que esa carpeta ya tiene contenido, que será algo como esto:



Pero puedes borrar tranquilamente todo y empezar a crear ahí tus ficheros PHP.

Acuérdate que cada vez que reinicies tu ordenador, tienes que abrir XAMPP y darle a Start en Apache para poder empezar a trabajar en tus ejercicios.

## Instalación en macOS

Hay diferentes maneras de instalar el intérprete de PHP. Una posibilidad es instalar XAMPP del mismo modo explicado anteriormente para Windows. Otra opción es con el gestor de paquetes Homebrew:

```
brew install httpd php74
```

Otra opción por ejemplo es siguiendo de las instrucciones de una empresa que saca siempre versiones actualizadas del mismo, aprovechando la instalación del Apache que viene instalada por defecto en OS X: <https://php-osx.liip.ch/>

## Instalación en Linux

Lo más sencillo es con el gestor de paquetes de la distribución que se esté utilizando, por ejemplo en Ubuntu sería:

```
apt-get install php
```

Esto, instalará también el servidor Apache con el módulo de PHP configurado.

Si se quiere se pueden instalar módulos adicionales como por ejemplo para conectarnos a la base de datos MySQL, hacer peticiones http con curl, etc...

Una vez realizada la instalación en cualquiera de los Sistemas Operativos, para comprobar que la instalación se ha realizado correctamente basta con ejecutar desde la consola, lo siguiente:

```
pi@piedpiper:~$ php -v
PHP 7.2.8-1+0~20180725113544.3+stretch~1.gbp571e56 (cli) (built: Jul 25 2018 11:35:49) ( NTS )
Copyright (c) 1997-2018 The PHP Group
Zend Engine v3.2.0, Copyright (c) 1998-2018 Zend Technologies
    with Zend OPcache v7.2.8-1+0~20180725113544.3+stretch~1.gbp571e56, Copyright (c) 1999-2018, by Zend Technologies
    with Xdebug v2.6.0, Copyright (c) 2002-2018, by Derick Rethans
```

Si te sale algo parecido, enhorabuena!! ya puedes empezar a programar en PHP