

# Introduction to PHP

PHP is a server-side and general-purpose scripting language that is especially suited for web development.

PHP originally stood for **Personal Home Page**. However, now, it stands for **Hypertext Preprocessor**. It's a recursive acronym because the first word itself is also an acronym.

PHP was created by [Rasmus Lerdorf](#) in 1994. It's currently maintained by the [PHP Development Team](#).

## PHP is a server-side language

When you open a website on your web browser, for example, <https://www.phptutorial.net>

The web browser sends an HTTP request to a web server where [phptutorial.net](https://www.phptutorial.net) locates. The web server receives the request and responds with an HTML document.

In this example, the web browser is a client while the web server is the server. The client requests for a page, and the server serves the request.

PHP runs on the web server, processes the request, and returns the HTML document.

## PHP is a general-purpose language

When it comes to the purpose of the programming languages, there are two main types: domain-specific and general-purpose languages.

The domain-specific languages are used within specific application domains. For example, SQL is a domain-specific language. It's used mainly for querying data from relational databases. And SQL cannot be used for other purposes.

On the other hand, PHP is a general-purpose language because PHP can develop various applications.

## PHP is a cross-platform language

PHP can run on all major operating systems, including Linux, Windows, and macOS.

You can use PHP with all leading web servers such as Nginx, OpenBSD, and Apache. Some cloud environments also support PHP like Microsoft Azure and Amazon AWS.

PHP is quite flexible. It's not just limited to processing HTML. PHP has built-in support for generating PDF, GIF, JPEG, and PNG images.

One notable feature of PHP is that it supports many databases, including MySQL, PostgreSQL, MS SQL, db2, Oracle Database, and MongoDB.

## What can PHP do

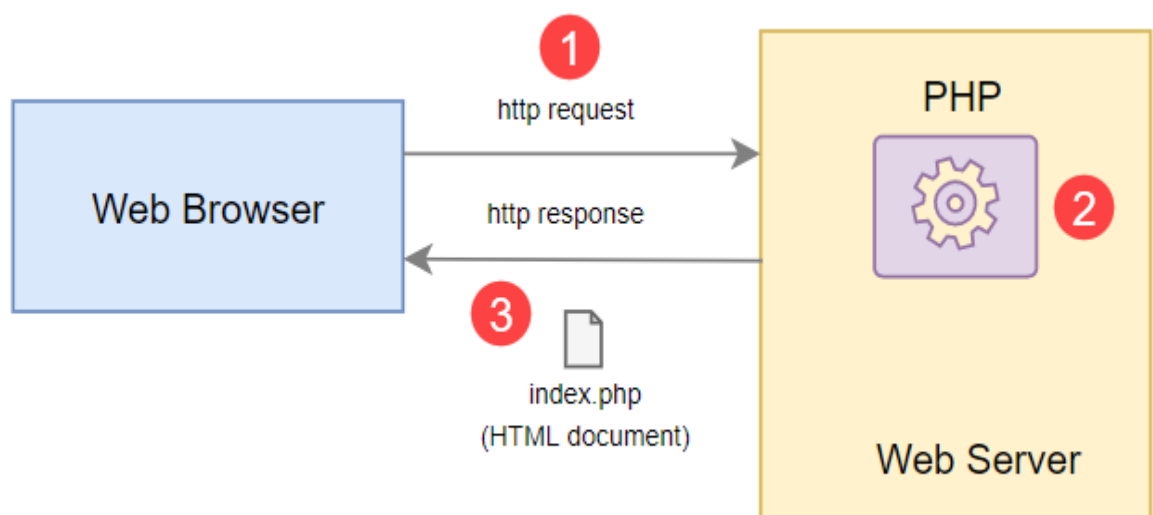
PHP has two main applications:

- Server-side scripting – PHP is well-suited for developing dynamic websites and web applications.
- Command-line scripting – like Python and Perl, you can run PHP script from the command line to perform administrative tasks like sending emails and generating PDF files.

The tutorials on this website mainly focus on server-side scripting.

## How PHP Works

The following illustrates how PHP works:



## How PHP works:

- First, the web browser sends an HTTP request to the web server, e.g., index.php.
- Second, the PHP preprocessor that locates on the web server processes PHP code to generate the HTML document.
- Third, the web server sends the HTML document back to the web browser.

## Advantages of PHP

Since PHP is designed for the web in the first place, it brings many advantages to web development:

- Simple – PHP is quite easy to learn and get started.
- Fast – PHP websites typically run very fast.
- Stable – PHP is stable since it has been in existence for a long time.
- Open-source and free – PHP is open source and free. It means that you don't have to pay a license fee to use PHP to develop software products.
- Community support – PHP has an active online community that helps you whenever you face an issue.

## Install PHP

Installing PHP on your computer allows you to safely develop and test a web application without affecting the live system.

To work with PHP locally, you need to have the following software:

- PHP
- A web server that supports PHP. We'll use the [Apache webserver](#).
- A database server. We'll use the [MySQL database server](#).

Typically, you won't install all this software separately because connecting them is tricky and not intended for beginners.

Therefore, it's easier to find an all-in-one software package that includes PHP, a web server, and a database server. One of the most popular PHP development environments is [XAMPP](#).

XAMPP is an easy install Apache distribution that contains PHP, MariaDB, and Apache webserver. XAMPP supports Windows, Linux, and macOS.

Note that [MariaDB](#) is a fork of the most popular relational database management system, MySQL.

## Download XAMPP

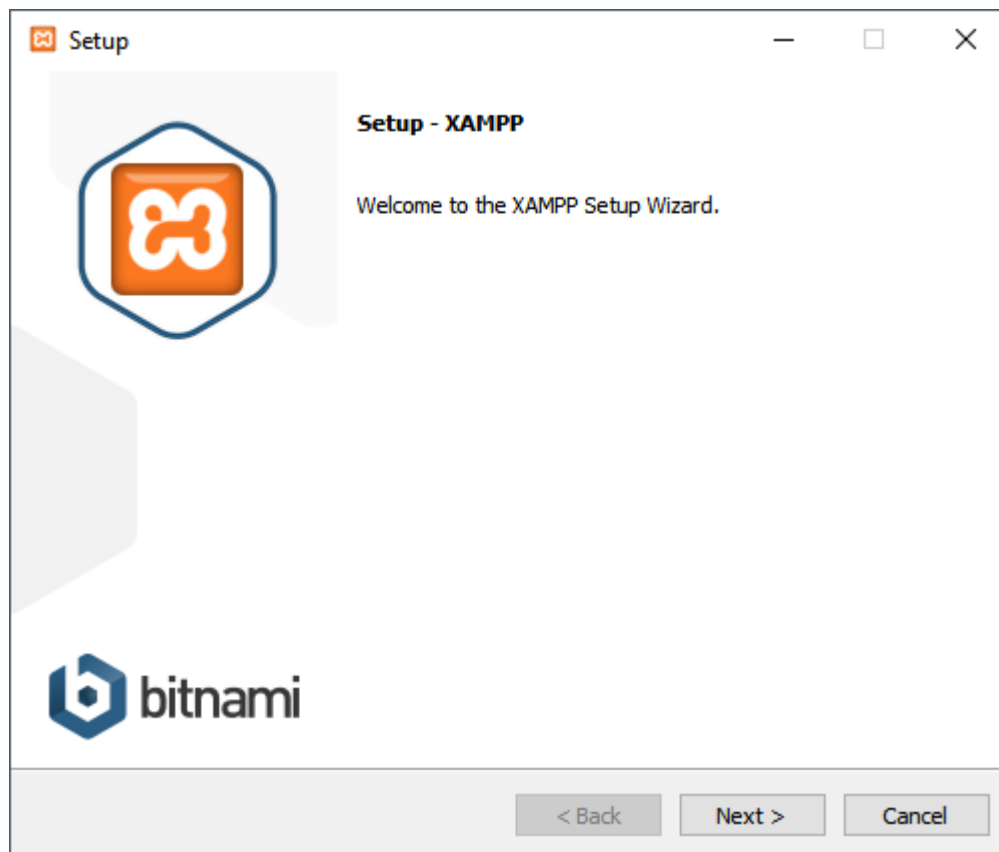
To install XAMPP on windows, you can go to the [XAMPP official website](#) and download the suitable version for your platform.

## Install XAMPP on Windows

To install XAMPP on Windows, you can follow these steps:

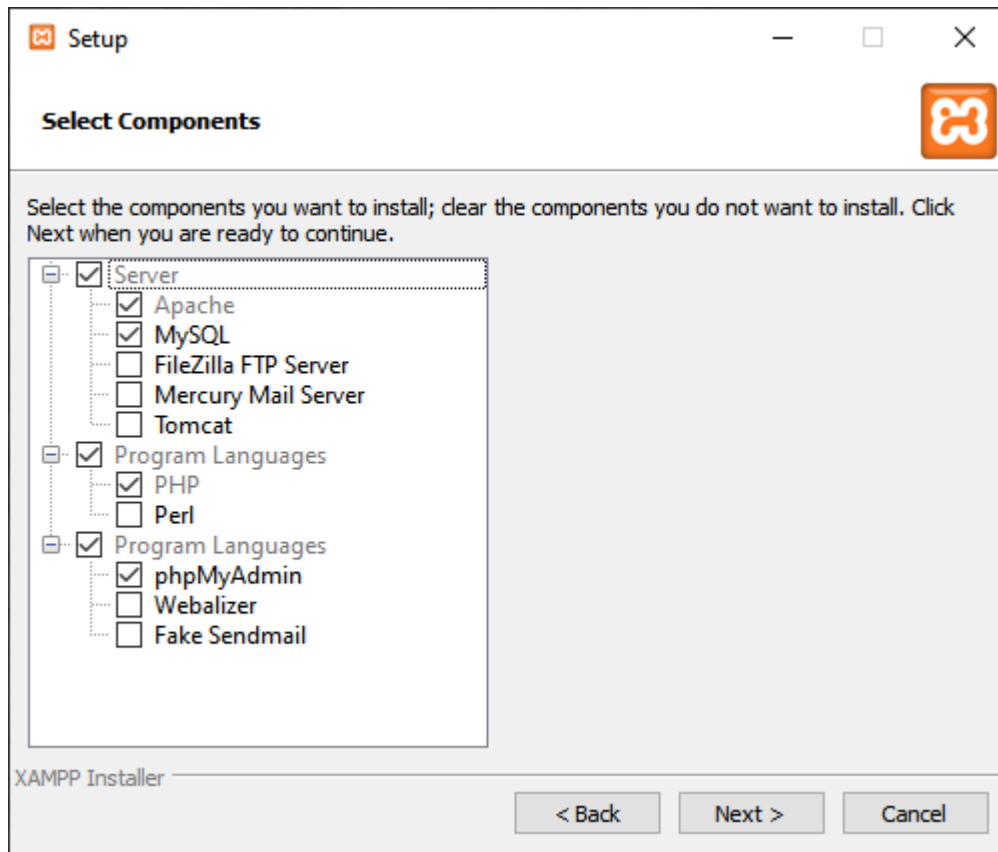
### Step 1. Start the installation

Double-click the downloaded file to start setting up XAMPP:



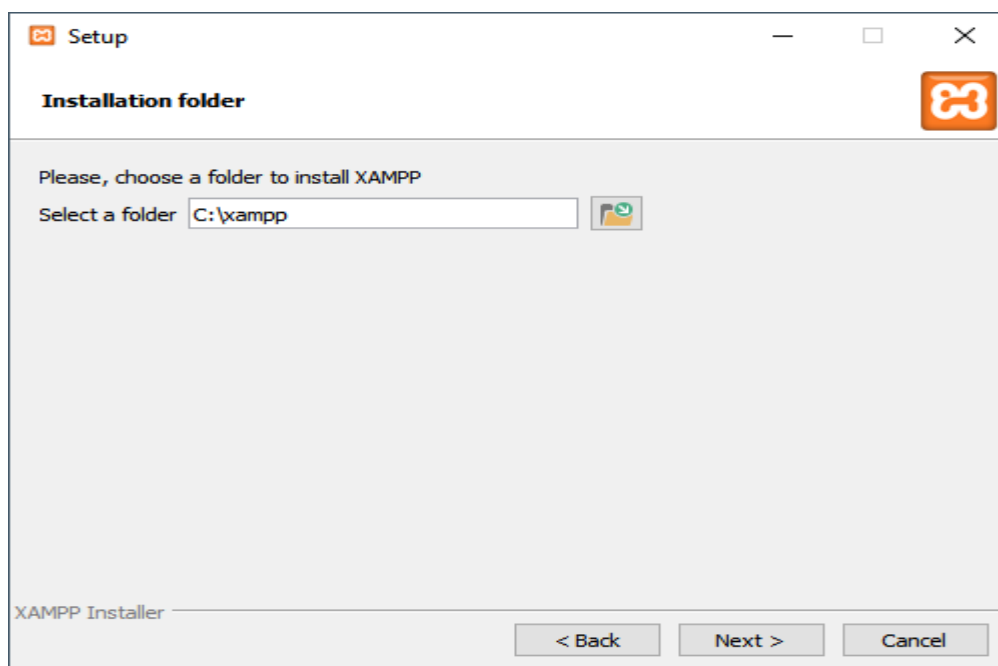
### Step 2. Select components to install

Select the components that you want to install. In this step, you can select Apache, MySQL, PHP, and phpMyAdmin, deselect other components like the following, and click the Next button to go to the next step.



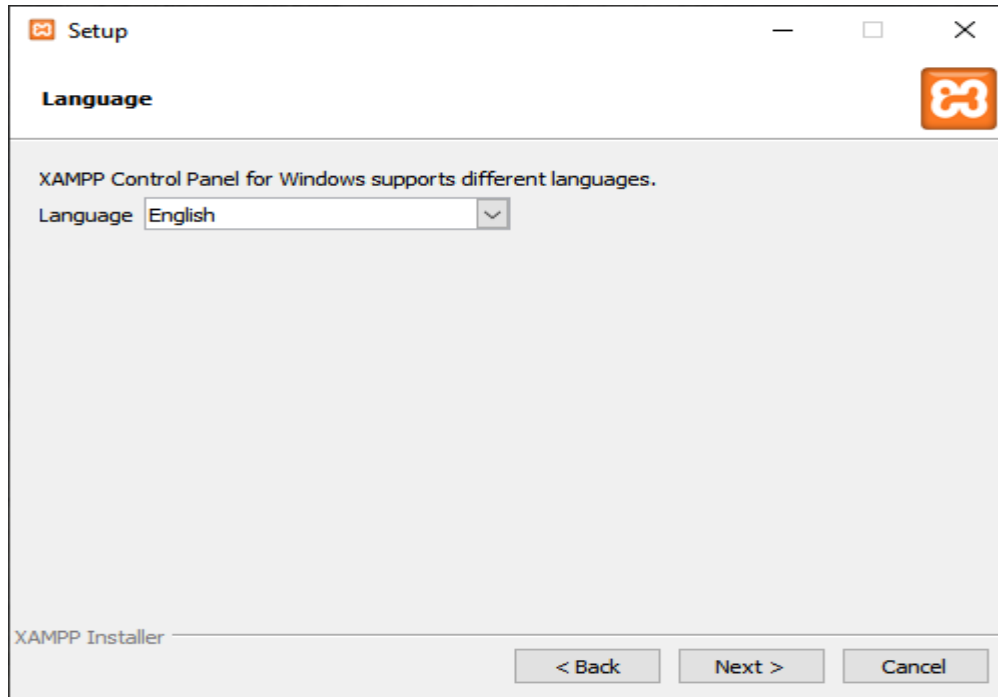
### Step 3. Specifying the installation folder

Select a folder to install XAMPP. It's recommended to install XAMPP in the c:\xampp folder. Click the Next button to go to the next step.



## Step 4. Selecting a language

Select a language for XAMPP Control Panel. By default, it's English. And you can select your preferred language and click the Next button to go to the next step.



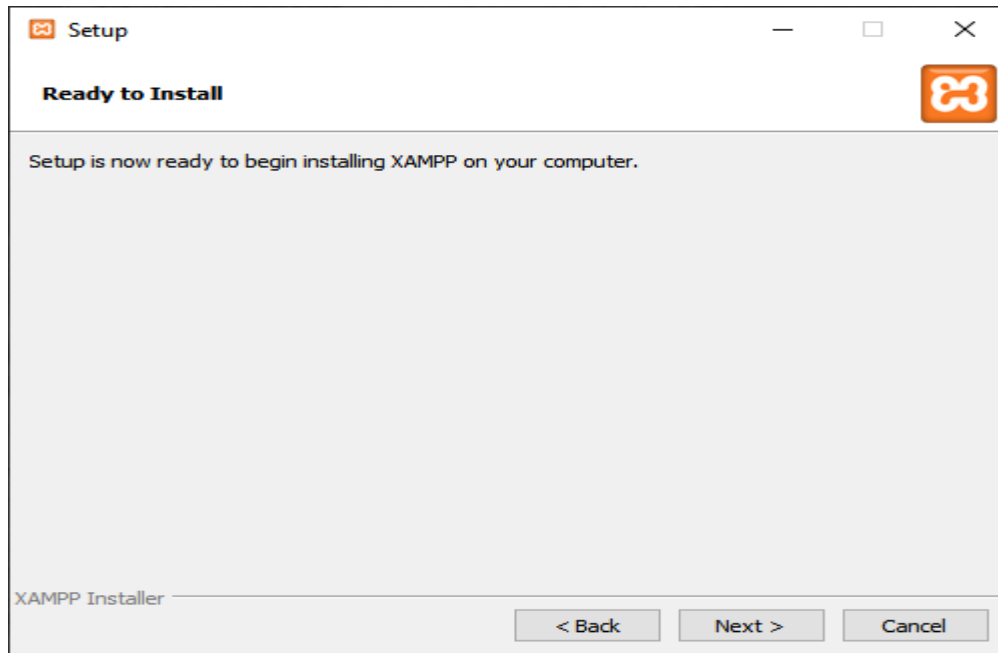
## Step 5. Bitnami for XAMPP

Feel free to skip this step because you don't need Bitnami for learning PHP. Just click the Next button to go to the next step.



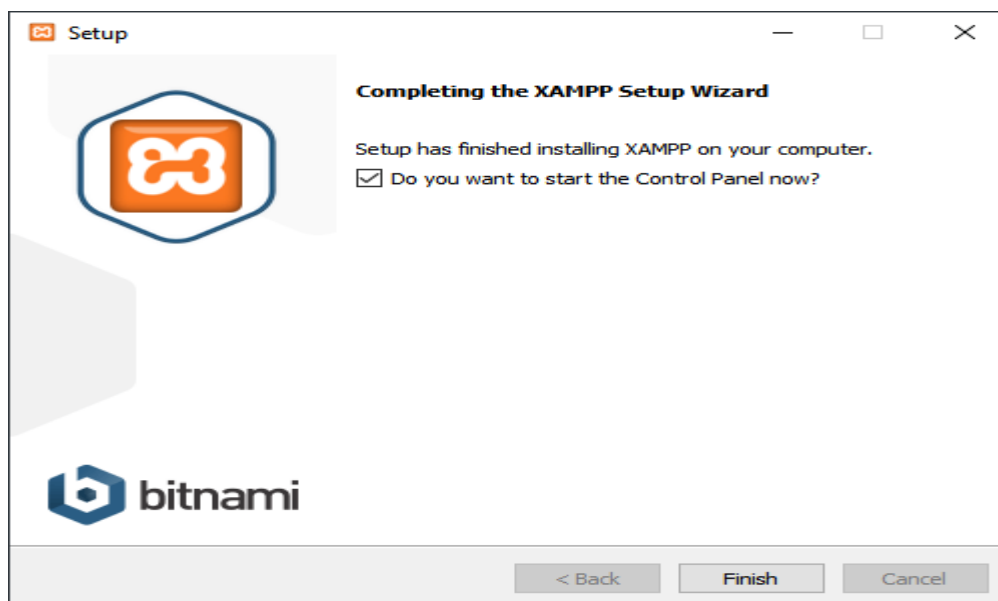
## Step 6. Begin installing XAMPP

And you're now ready to install XAMPP. Click the Next button to start the installation. It'll take a few minutes to complete.

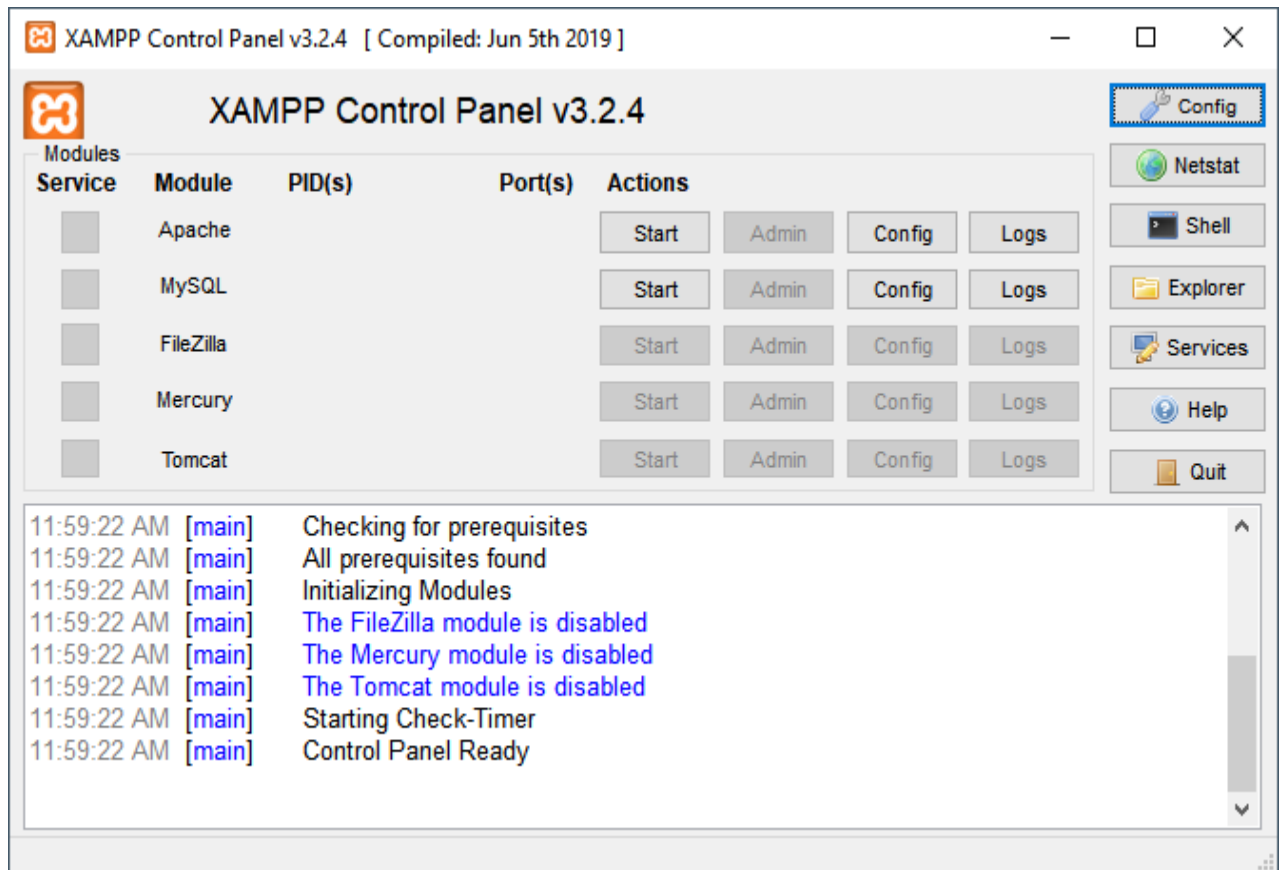


## Step 7. Completing the XAMPP setup

Once completed, the XAMPP setup wizard shows the following screen. You can click the Finish button to launch the XAMPP Control Panel:

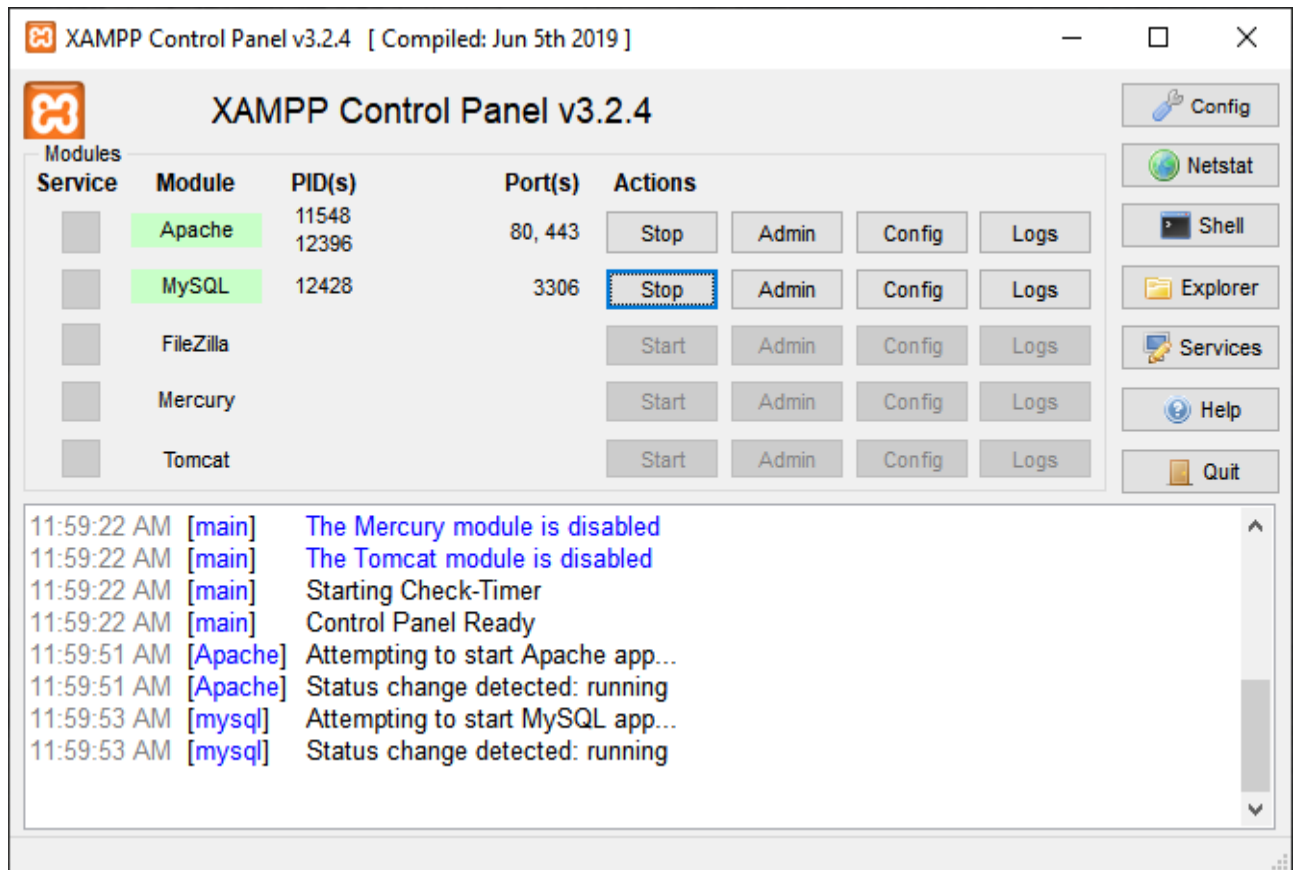


The XAMPP Control Panel lists installed services. To start a service, you click the corresponding Start button:



The following shows the Apache web server and MySQL are up and running. The Apache web server listens on the ports 80 and 443 while the MySQL listens on port 3306:





## Step 8. Launch the XAMPP

Open the web browser and navigate to the following URL: <http://localhost/>. If the installation is completed successfully, you'll see the welcome screen of the XAMPP.

## Troubleshooting

By default, Apache uses port 80. However, if port 80 is used by another service, you'll get an error like this:

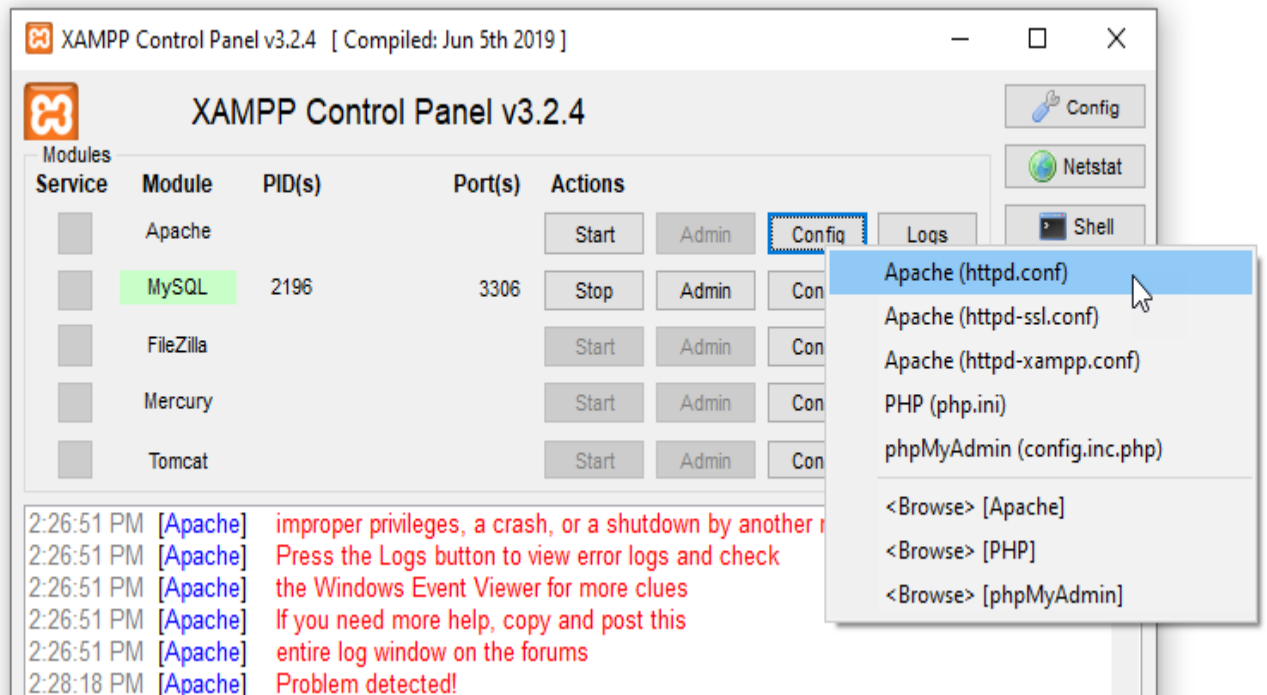
```

Problem detected!
Port 80 in use by "Unable to open process" with PID 4!
Apache WILL NOT start without the configured ports free!
You need to uninstall/disable/reconfigure the blocking application
or reconfigure Apache and the Control Panel to listen on a different port
Code language: plaintext (plaintext)

```

In this case, you need to change the port from 80 to a free one, e.g., 8080. To do that, you follow these steps:

First, click the Config button that aligns with the Apache module:



Second, find the line that has the text `Listen 80` and change the port from 80 to 8080 like this:

```

httpd.conf - Notepad
File Edit Format View Help
# mutex file directory is not on a local disk or is not appropriate for some
# other reason.
#
# Mutex default:logs

#
# Listen: Allows you to bind Apache to specific IP addresses and/or
# ports, instead of the default. See also the <VirtualHost>
# directive.
#
# Change this to Listen on specific IP addresses as shown below to
# prevent Apache from glomming onto all bound IP addresses.
#
#Listen 12.34.56.78:80
Listen 8080

```

Third, click the Start button to start the Apache service. If the port is free, Apache should start properly, as shown in the following picture:

XAMPP Control Panel v3.2.4 [ Compiled: Jun 5th 2019 ]

## XAMPP Control Panel v3.2.4

Modules

Service	Module	PID(s)	Port(s)	Actions
<input type="checkbox"/>	Apache	15696 17640	443, 8080	Stop Admin Config Logs
<input type="checkbox"/>	MySQL	2196	3306	Stop Admin Config Logs
<input type="checkbox"/>	FileZilla			Start Admin Config Logs
<input type="checkbox"/>	Mercury			Start Admin Config Logs
<input type="checkbox"/>	Tomcat			Start Admin Config Logs

2:28:20 PM [Apache] entire log window on the forums  
 2:43:35 PM [Apache] Problem detected!  
 2:43:35 PM [Apache] Port 80 in use by "Unable to open process" with PID 4!  
 2:43:35 PM [Apache] Apache WILL NOT start without the configured ports free!  
 2:43:35 PM [Apache] You need to uninstall/disable/reconfigure the blocking application  
 or reconfigure Apache and the Control Panel to listen on a different port  
 2:43:35 PM [Apache] Attempting to start Apache app...  
 2:43:36 PM [Apache] Status change detected: running