**1: Install Apache**

Apache HTTP Server is the web server running on top of Linux in the LAMP stack. The [web server](https://phoenixnap.com/blog/web-server-vs-application-server) uses HTTP to process requests and transmit information through the internet.

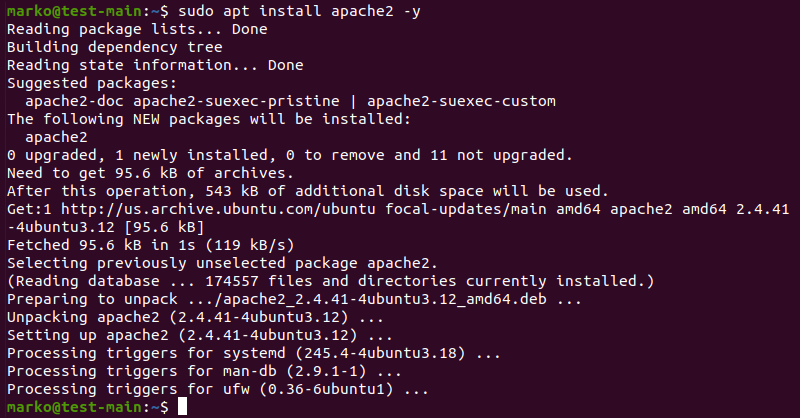
Follow the procedure below to install Apache.

1. Before installing the first LAMP component, ensure the package list on the system is up to date. In the terminal, type:

sudo apt update

1. To install the Apache package, run the following command:

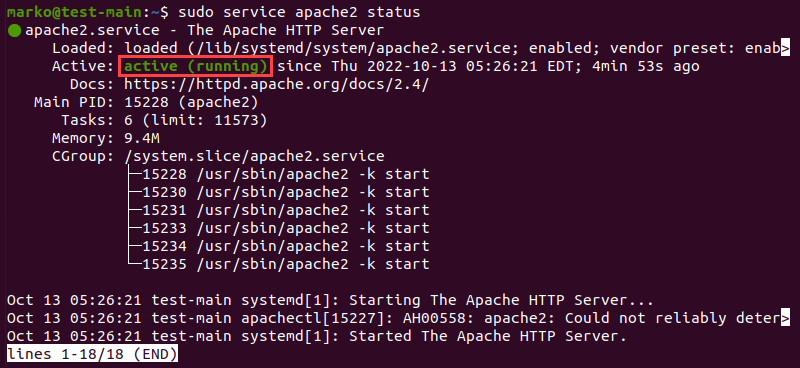
sudo apt install apache2 -y



1. Check if Apache installed correctly by checking the Apache service status:

sudo service apache2 status

The service shows as running in the output:



Exit the status screen by pressing **Ctrl** + **C** on the keyboard.

1. Next, make sure that the [UFW firewall](https://phoenixnap.com/kb/configure-firewall-with-ufw-on-ubuntu) contains the Apache profiles by typing in the following command:

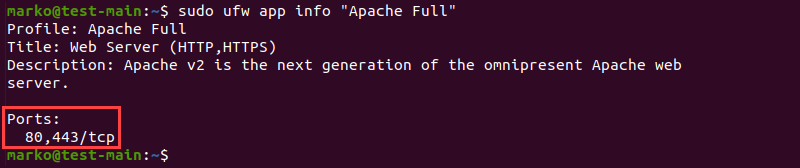
sudo ufw app list



1. Ensure the **Apache Full** profile allows the traffic on ports **80** and **443** by running the command:

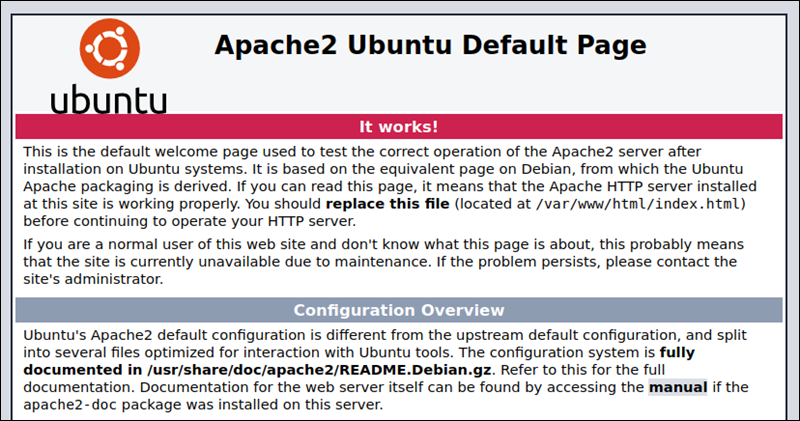
sudo ufw app info” Apache Full”

The output should look similar to the following example:



1. To confirm that Apache is running, enter the IP address of your server in the address bar of an internet browser and press **ENTER**.

The test Apache web server page should display as below.



**Note:** You can also access the Apache test page by typing localhost in the address bar.

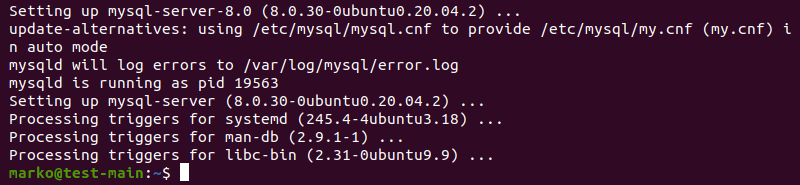
### Step 2: Install MySQL and Create a Database

MySQL is a relational [database management system](https://phoenixnap.com/kb/best-database-software) for creating and maintaining dynamic enterprise-level databases. It is compatible with all major OS platforms, which makes it a good fit for web application development.

**Note:** Refer to our article and find out [what is a relational database](https://phoenixnap.com/kb/what-is-a-relational-database).

Install MySQL by typing the following command:

sudo apt install mysql-server -y

****

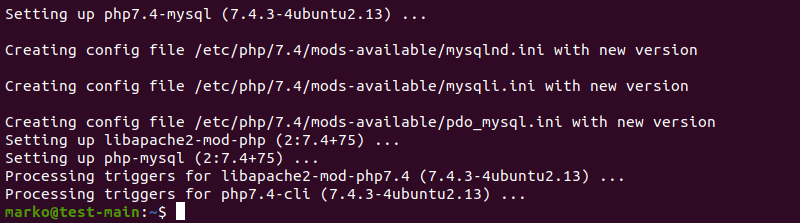
### Step 3: Install PHP

Although other programming languages, such as Python and Pearl, also work well within LAMP, PHP is usually the final layer of the stack because it [integrates well with MySQL](https://phoenixnap.com/kb/connect-mysql-with-php). As a dynamically typed language, PHP embeds into HTML, improving the speed and reducing the complexity of web applications.

Install PHP by following the steps below.

1. Obtain the necessary PHP packages by typing:

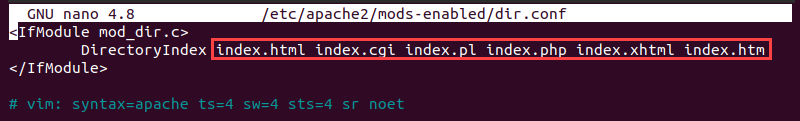
sudo apt install php libapache2-mod-php php-mysql -y



2. Modify the way Apache serves files by opening the dir.conf file in a text editor with root privileges:

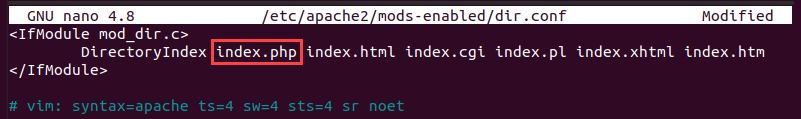
sudo nano /etc/apache2/mods-enabled/dir.conf

The configuration file looks like in the example below:



By default, Apache first looks for an index.html file card.

3. Edit the list so that the index.php file is in the first position:



4. Press **CTRL + X**to save and close the file. Press **y** and **ENTER** to confirm.

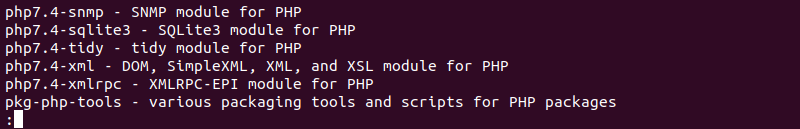
#### **Install PHP Modules (Optional)**

If necessary, add more modules to improve the functionality of PHP. Search, view, and install various libraries and modules by following the procedure below.

1.  Get a list of available PHP modules with:

apt-cache search php- | less

The command pipes the results of the apt-cache search into less to simplify viewing the output.

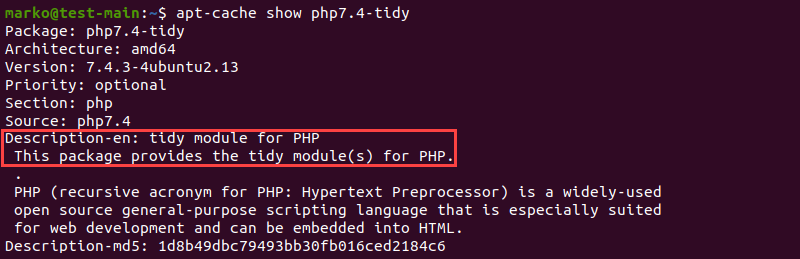


2. Scroll up and down by using the arrow keys to see all the options, including a short description for each module.

3. For example, to find out what the module **php7.4-tidy**does, type:

apt-cache show php7.4-tidy

The output displays the module description.



4. To install the php7.4-tidy package after viewing its description, use the following command.

sudo apt install php7.4-tidy

5. When you finish, press **q** to quit.

### Step 4: Restart Apache

For the changes to take effect, restart the Apache service by typing:

sudo systemctl restart apache2

If the command executes correctly, it returns no output.

### Step 5: Test PHP Processing on Web Server

To test the new LAMP installation, create a basic PHP script and place it in the web root directory located at **/var/www/html/**, then check if the script is accessible via an internet browser. The steps below explain the procedure for performing this test.

1. Create a file in the web root directory by typing the following command:

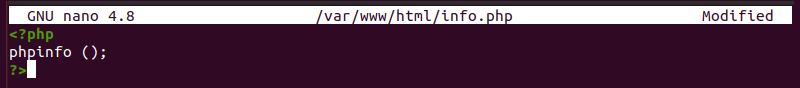
sudo nano /var/www/html/info.php

2. Inside the file, type the PHP code:

<?php

phpinfo ();

?>



3. Press **CTRL + X** to save and close the file. Press **y** and **ENTER** to confirm.

4. Open an internet browser and type the following address:

[server-ip-address]/info.php

Alternatively, type:

localhost/info.php

The output should display the details of the LAMP stack, as seen in the image below:

