

Installation and Setup of LEMP and LAMP

stack in on VM Box .

Objective : Install linux in VM box and setup LAMP and LEMP stack.

INDEX

S.no	Title
1	What is VM Box
2	Installation of VM box in Windows
3	Creation Of Machine
4	Installation of LAMP stack

What is Oracle Virtual Box

Oracle VirtualBox is a powerful open source virtualization software developed by Oracle Corporation. It allows users to run multiple operating systems simultaneously on a single physical machine, making it ideal for developers, testers, and IT professionals. Below is a brief description of its main features and uses:

Key Features :

- **Cross-Platform Compatibility**
- **Guest Operating System Support**
- **Snapshots and Cloning**
- **Seamless Mode and Scaled Mode**
- **Virtual Networking**
- **Extensibility**

Use Cases :

- **Development and Testing**
- **Training and Education**
- **Legacy Application Support**
- **Sandboxing**

Download and Installation Of VM Box in Windows

Step 1 :

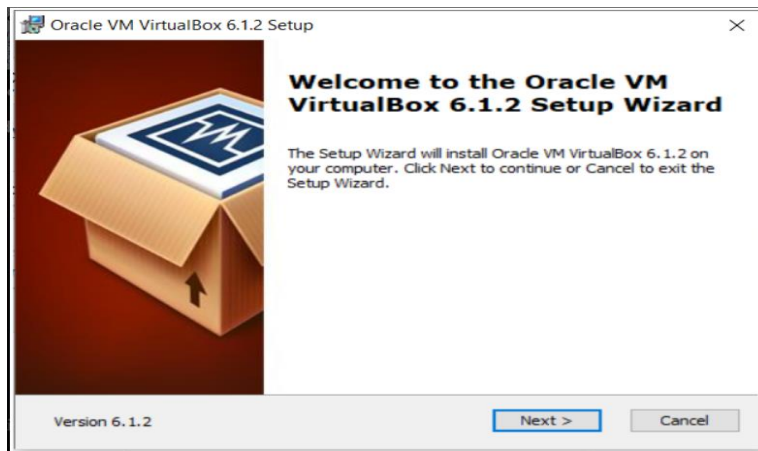
- Search Oracle Virtual Box on Browser.



Step 2 :

- Click on Windows Host , Download(latest version) will automatically start.

Step 3 :Now open the Download folder and open the VM Box application.



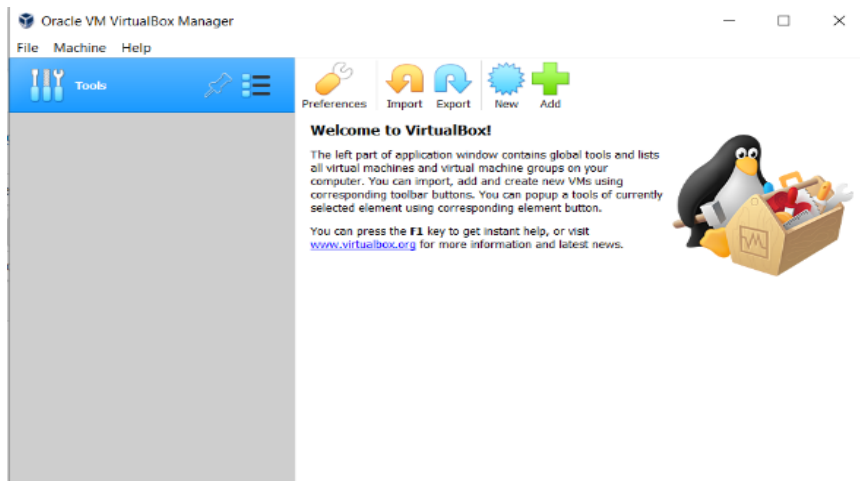
Step 4 :

- Click on next and proceed to allow VM Box to install.

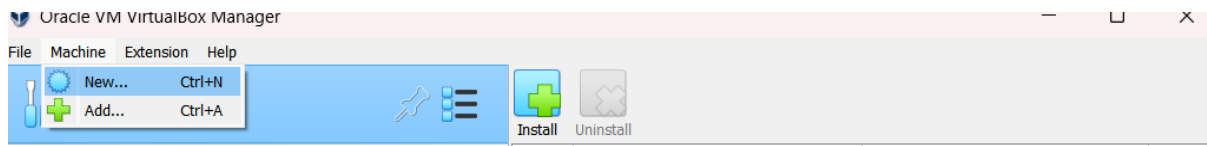


-
- This will be the final interface after successful installation.

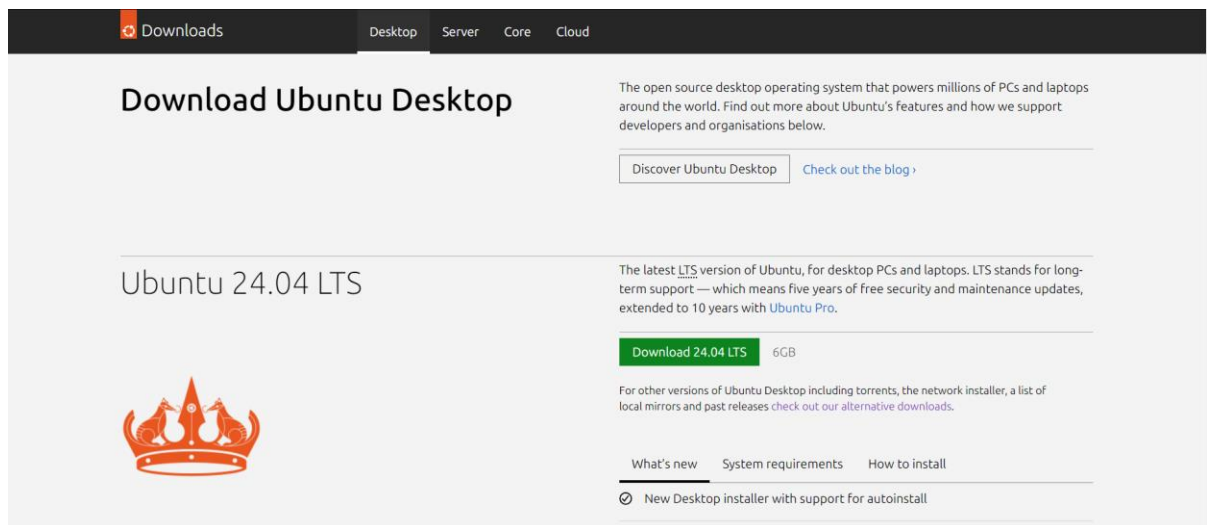
Creation of first machine.



- As I have shown VM box will automatically start after installation , so above shown picture will be interface of application.
- Now click on Machine in the menu bar



- Click on "New" to create your first machine.
- Now download ISO image file of ubuntu from <https://ubuntu.com>.



- Click on Download 24.04 LTS and an ISO image file will be downloaded.

Create Virtual Machine

✓ Name and Operating System ⚠

Name:

Folder:

ISO Image:

Edition:

Type:

Version:

☐ Skip Unattended Installation

> Unattended Install

> Hardware

> Hard Disk

Help Guided Mode Back Finish Cancel

-
-
- **Click on Drop Down button of ISO Image tab and add the downloaded file.**
- **Give the name of the machine you want to give , let it be something meaningful.**

- After that go to Unattended Install button and give the username and password for accessing Ubuntu on the machine.

Create Virtual Machine

> Name and Operating System

> Unattended Install

Username and Password

Username: vboxuser ✓

Password: ••••••

Repeat Password: ••••••

Additional Options

Product Key: #####-#####-#####-#

Hostname:

Domain Name: myguest.virtualbox.org

☐ Install in Background

☐ Guest Additions

Guest Additions ISO: <not selected>

> Hardware

> Hard Disk

-
- You can change the default username password here.
- Now open the Hardware Button clicking on it :

> Hardware

Base Memory: 4 MB 2048 MB 8192 MB

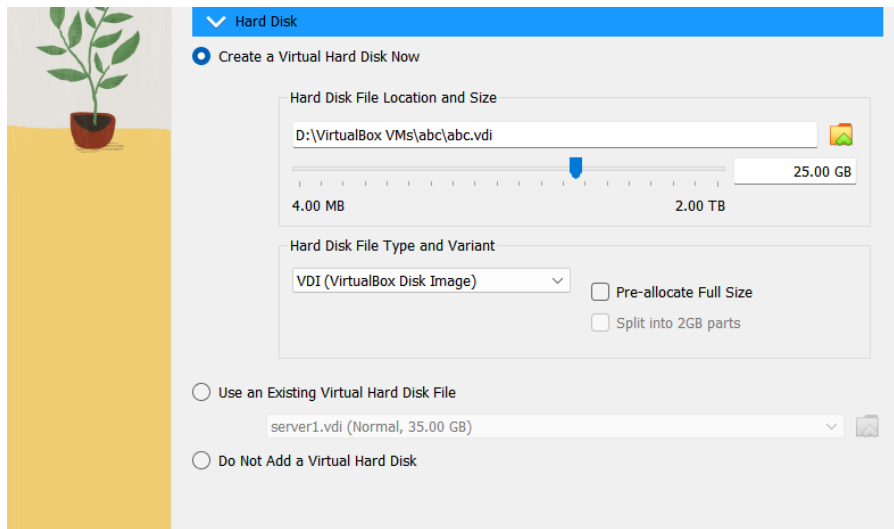
Processors: 1 CPU 8 CPUs

☐ Enable EFI (special OSes only)

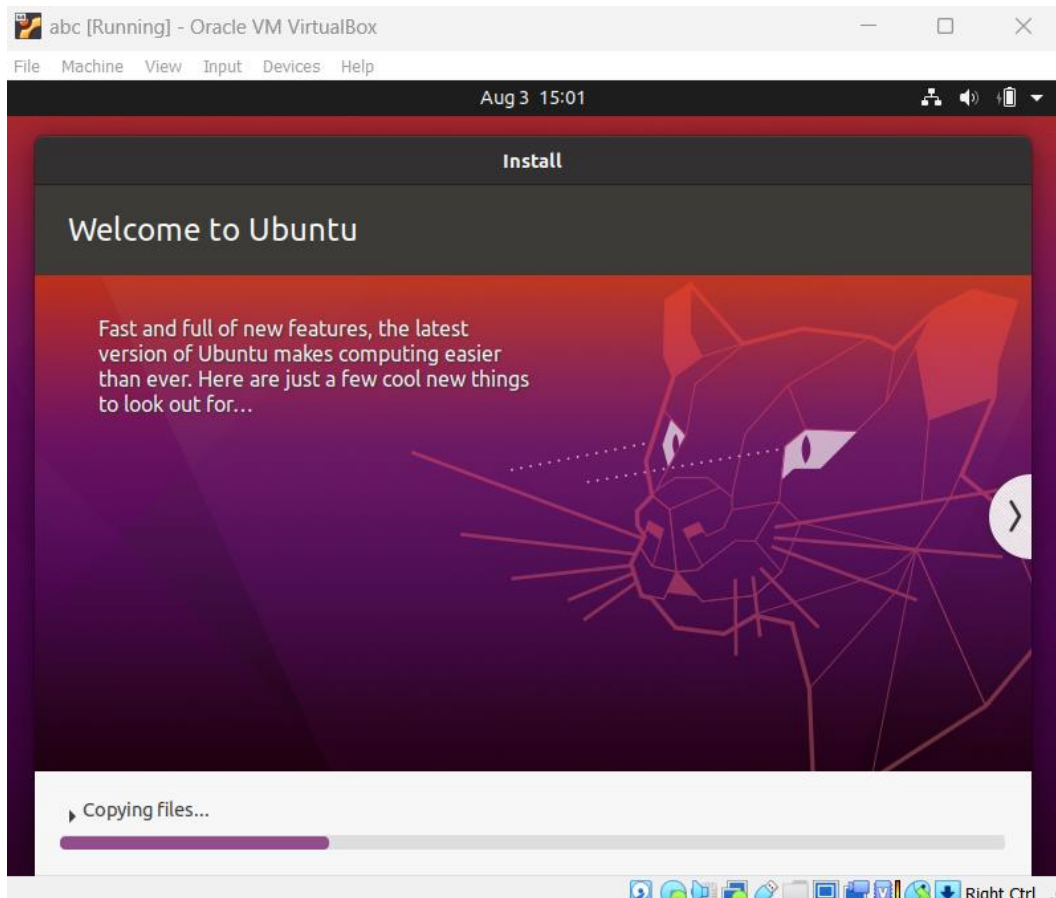
> Hard Disk

-
- Now, it is up to you to give the memory and processor to the Machine , but for installing LAMP and LEMP stack and after that wordpress , so ideal configuration would be 1 GB of base memory and 2 Processors.

- After that come to hard disk button at the last :



-
- 20 - 25 GB of storage would be more than sufficient for the Machine
- Now click on finish button at the bottom and your first machine will be made after some processing time.



- This will be the interface of installation of ubuntu on VM .

Installation of LAMP stack (Linux, Apache, Mysql & Php)

- Brief Introduction of LAMP stack :

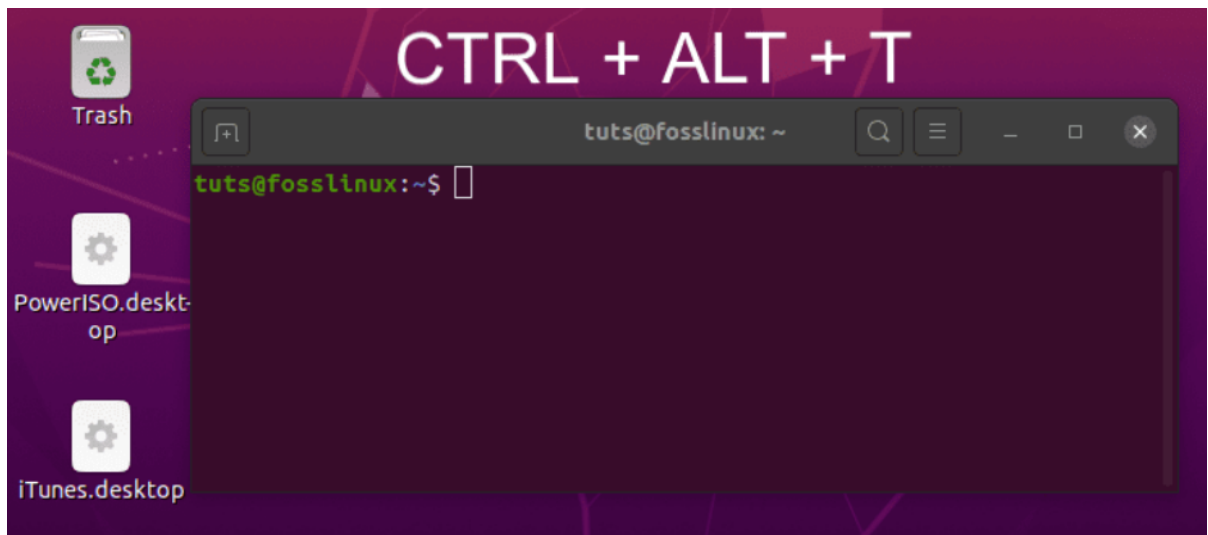
The LAMP stack is a widely used collection of open-source software for web development. It consists of:

1. Linux: The operating system.
2. Apache: The web server.
3. MySQL: The database system.
4. PHP: The programming language (alternatively, Perl or Python).

This combination creates a powerful environment for creating and deploying dynamic websites and applications.

Step 1 :

- As we have installed ubuntu linux already so first part is done of Linux.
- Now open the terminal in Ubuntu by pressing Ctrl+Alt+T.



- Make a sudo user (A sudo user can execute commands with superuser privileges using the sudo command.) , ref to the below image :

```
root@LAMPuser: /home/userismayank
root@LAMPuser:/home/userismayank# sudo adduser username
```

-
- **Create and Confirm the password for the user and all other basic details.**

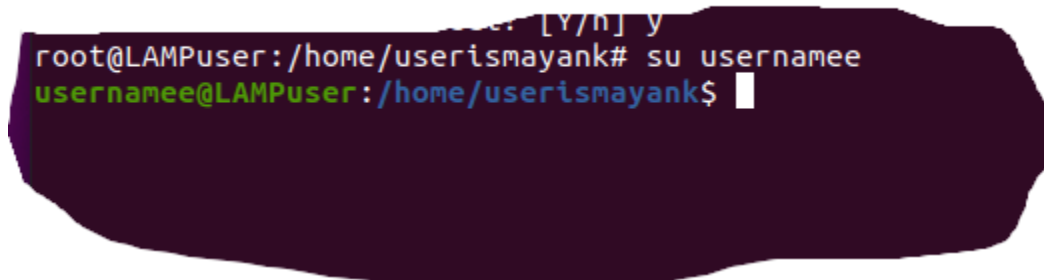
```
root@LAMPuser:/home/userismayank# sudo adduser username
adduser: The user 'username' already exists.
root@LAMPuser:/home/userismayank# sudo adduser usernameee
Adding user 'usernameee' ...
Adding new group 'usernameee' (1003) ...
Adding new user 'usernameee' (1003) with group 'usernameee' ...
Creating home directory '/home/usernameee' ...
Copying files from '/etc/skel' ...
New password:
Retype new password: █
```

GAS_7.0.18

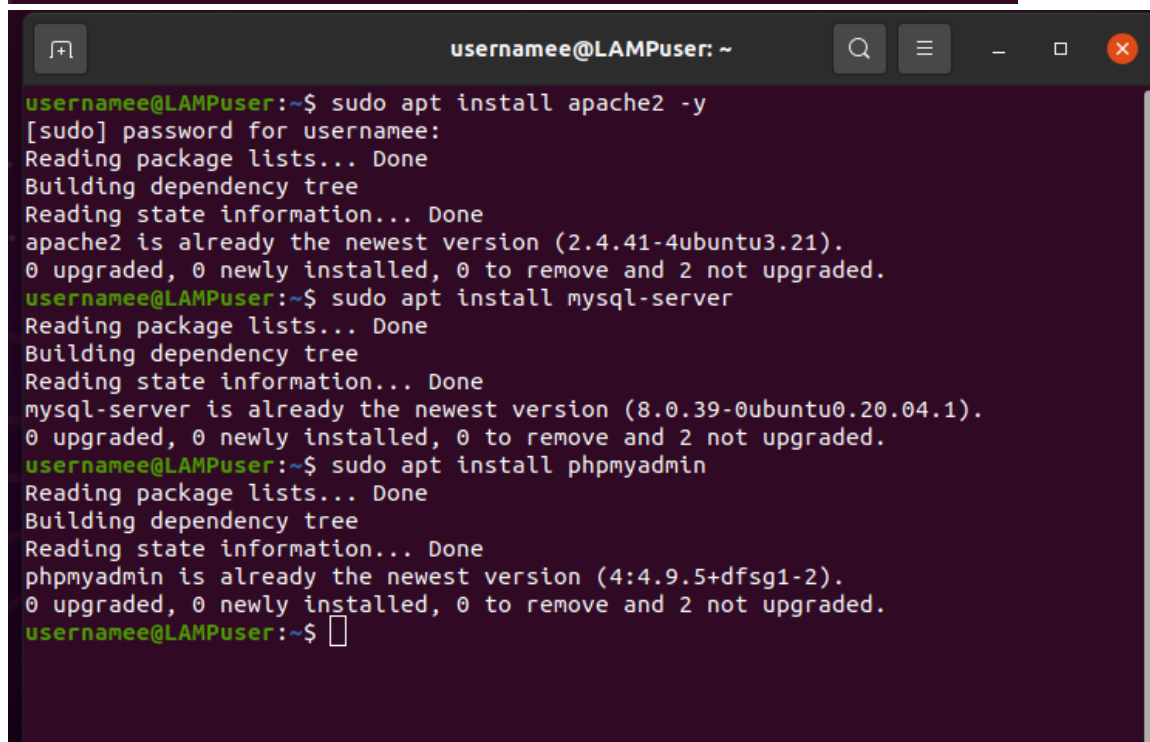
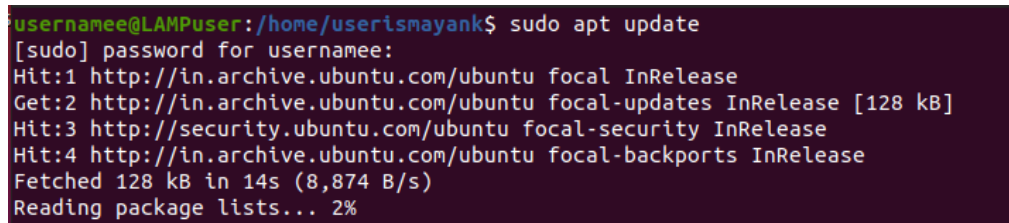
-
- **After filling all the details you will see something like this :**

```
root@LAMPuser:/home/userismayank# sudo adduser usernameee
Adding user 'usernameee' ...
Adding new group 'usernameee' (1003) ...
Adding new user 'usernameee' (1003) with group 'usernameee' ...
Creating home directory '/home/usernameee' ...
Copying files from '/etc/skel' ...
New password:
Retype new password:
Sorry, passwords do not match.
passwd: Authentication token manipulation error
passwd: password unchanged
Try again? [y/N] y
New password:
Retype new password:
passwd: password updated successfully
Changing the user information for usernameee
Enter the new value, or press ENTER for the default
  Full Name []: user1
    Room Number []: 1
      Work Phone []: 1
      Home Phone []: 1
        Other []: 1
Is the information correct? [Y/n] y
root@LAMPuser:/home/userismayank#
```

- Now switch to sudo user by referring following image :



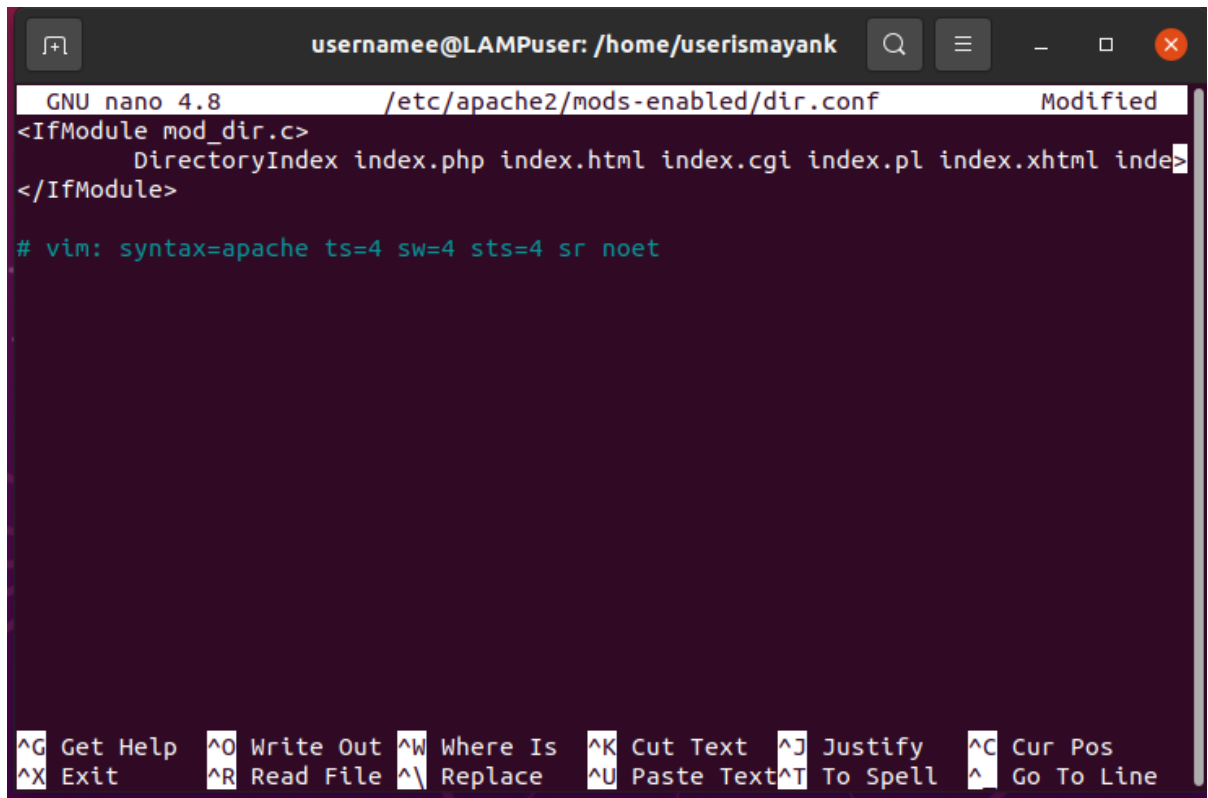
-
- Now apply multiple commands in terminal to install apache, mysql-server and phpmyadmin:



Command to install :

- **Apache :** sudo apt install apache2
- **Mysql :** sudo apt install mysql-server
- **PhpMyAdmin :** sudo apt install phpmyadmin

Make the index.php file first in the queue in the dir.conf file which can be access through nano /etc/apache2/mods-enabled/dir.conf

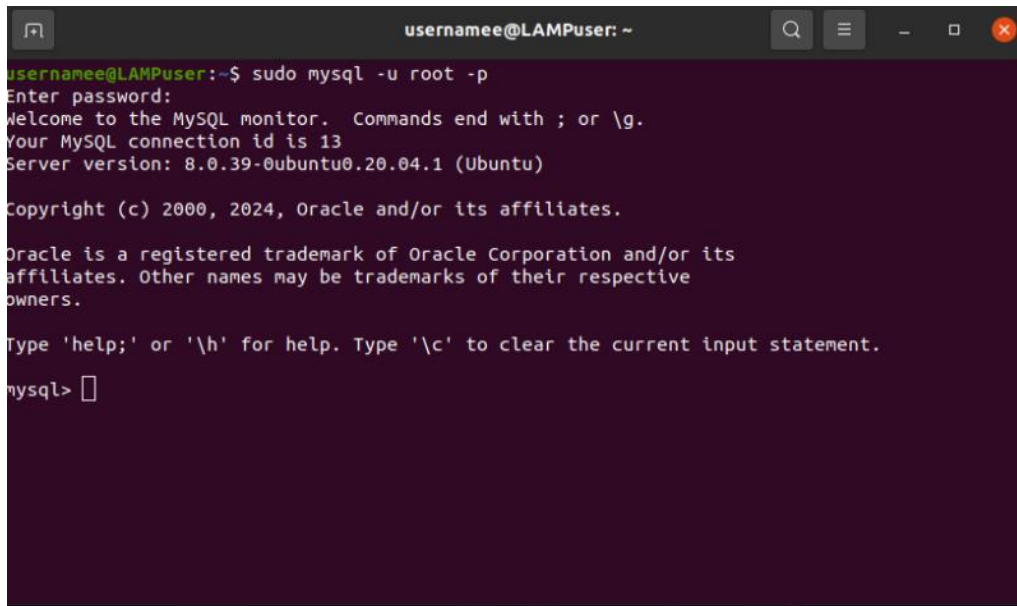


```
GNU nano 4.8 /etc/apache2/mods-enabled/dir.conf Modified
<IfModule mod_dir.c>
    DirectoryIndex index.php index.html index.cgi index.pl index.xhtml index
</IfModule>

# vim: syntax=apache ts=4 sw=4 sts=4 sr noet

^G Get Help  ^O Write Out ^W Where Is  ^K Cut Text  ^J Justify   ^C Cur Pos
^X Exit      ^R Read File ^\ Replace   ^U Paste Text ^T To Spell  ^_ Go To Line
```

-
- Now setup mysql to create a database :



```
usernamee@LAMPuser: ~
usernamee@LAMPuser:~$ sudo mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 13
Server version: 8.0.39-0ubuntu0.20.04.1 (Ubuntu)

Copyright (c) 2000, 2024, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

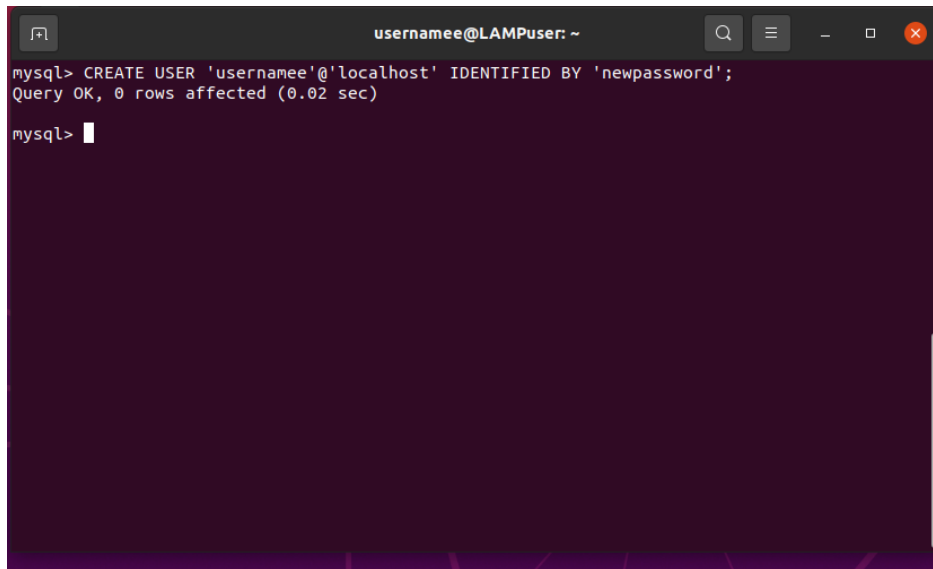
mysql>
```

-

- Access mysql by giving command : `sudo mysql -u root`
-

Create database : `CREATE DATABASE dbname ;`

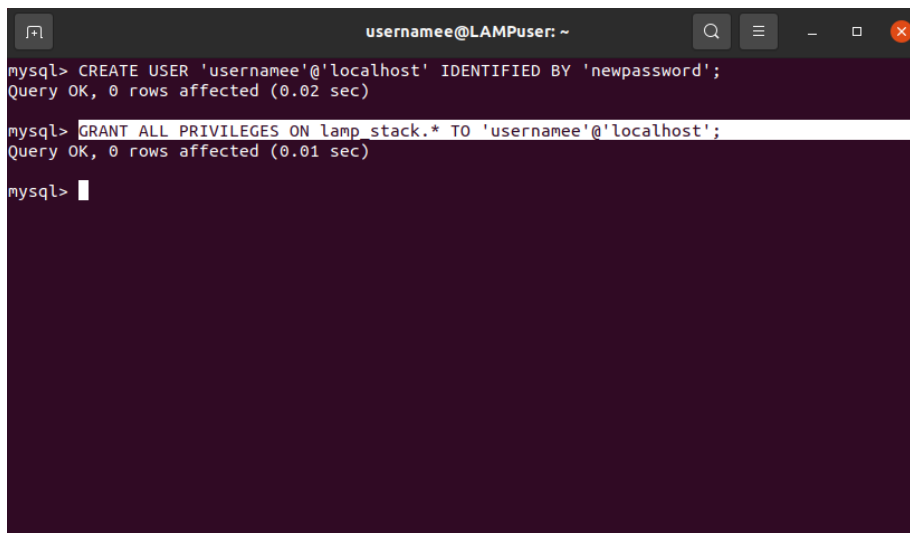
- Now create user :

A terminal window titled 'usernamee@LAMPuser: ~' with search, menu, and window control icons. It shows the execution of the command 'mysql> CREATE USER 'usernamee'@'localhost' IDENTIFIED BY 'newpassword';' which returns 'Query OK, 0 rows affected (0.02 sec)'. The prompt 'mysql>' is shown again with a cursor.

```
usernamee@LAMPuser: ~
mysql> CREATE USER 'usernamee'@'localhost' IDENTIFIED BY 'newpassword';
Query OK, 0 rows affected (0.02 sec)

mysql>
```


- Grant all privileges of the database to created user.

A terminal window titled 'usernamee@LAMPuser: ~' with search, menu, and window control icons. It shows the execution of the command 'mysql> GRANT ALL PRIVILEGES ON lamp_stack.* TO 'usernamee'@'localhost';' which returns 'Query OK, 0 rows affected (0.01 sec)'. The prompt 'mysql>' is shown again with a cursor.

```
usernamee@LAMPuser: ~
mysql> CREATE USER 'usernamee'@'localhost' IDENTIFIED BY 'newpassword';
Query OK, 0 rows affected (0.02 sec)

mysql> GRANT ALL PRIVILEGES ON lamp_stack.* TO 'usernamee'@'localhost';
Query OK, 0 rows affected (0.01 sec)

mysql>
```

- 
- Now reload the grant tables from the disk and clear the cache for faster access by `FLUSH PRIVILEGES` command.

```
usernamee@LAMPuser: ~  
mysql> CREATE USER 'usernamee'@'localhost' IDENTIFIED BY 'newpassword';  
Query OK, 0 rows affected (0.02 sec)  
  
mysql> GRANT ALL PRIVILEGES ON lamp_stack.* TO 'usernamee'@'localhost';  
Query OK, 0 rows affected (0.01 sec)  
  
mysql> FLUSH PRIVILEGES;  
Query OK, 0 rows affected (0.01 sec)  
  
mysql> \q
```

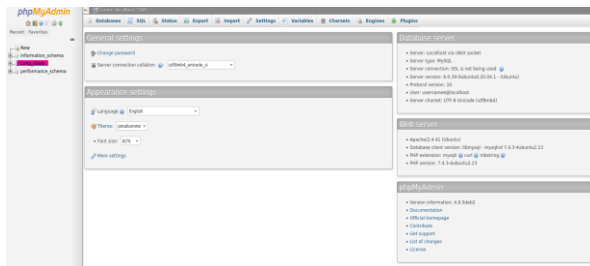
-
- Restart apache server by sudo systemctl restart apache2 command.

```
usernamee@LAMPuser: ~  
mysql> CREATE USER 'usernamee'@'localhost' IDENTIFIED BY 'newpassword';  
Query OK, 0 rows affected (0.02 sec)  
  
mysql> GRANT ALL PRIVILEGES ON lamp_stack.* TO 'usernamee'@'localhost';  
Query OK, 0 rows affected (0.01 sec)  
  
mysql> FLUSH PRIVILEGES;  
Query OK, 0 rows affected (0.01 sec)  
  
mysql> \q  
Bye  
usernamee@LAMPuser:~$ sudo restart apache2  
sudo: restart: command not found  
usernamee@LAMPuser:~$ sudo systemctl restart apache2  
usernamee@LAMPuser:~$ sudo nano /etc/mysql/mysql.conf.d/mysqld.conf  
usernamee@LAMPuser:~$ sudo systemctl restart mysql  
usernamee@LAMPuser:~$
```

-
- Now open phpmyadmin on localhost by : localhost/phpmyadmin/ on browser.



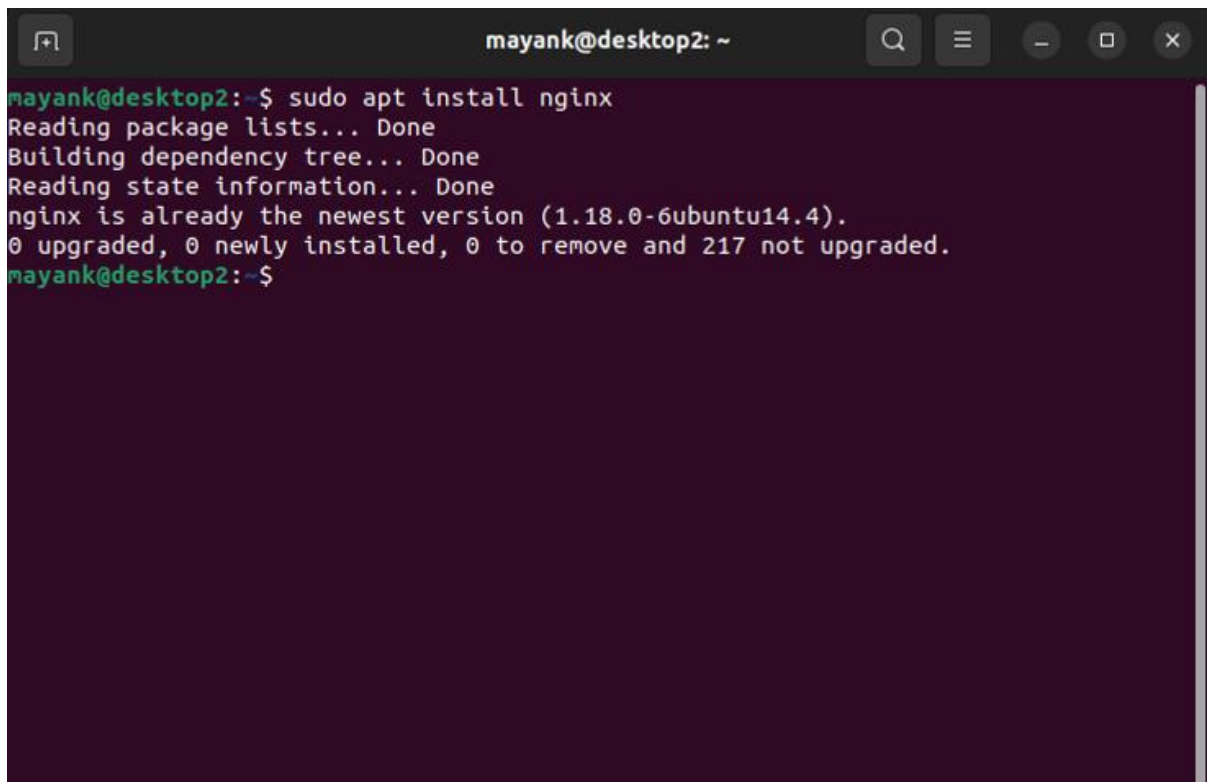
- Access phpmyadmin by entering username and password entered while we were creating user in mysql.



- Now we can see in databases dashboard our database named lamp_stack is created.

Installation of LEMP stack (Linux nginx mysql & Php)

- We will create new machine with same process of existing machine to build LEMP on it in which Linux is already installed.
- Now first we have to install nginx server .

A terminal window titled 'mayank@desktop2: ~' with standard window controls. The terminal output shows the command 'sudo apt install nginx' being executed. The output indicates that the package lists are read, the dependency tree is built, and state information is read. It then states that nginx is already the newest version (1.18.0-6ubuntu14.4) and that 0 packages were upgraded, 0 newly installed, 0 to be removed, and 217 not upgraded. The prompt returns to 'mayank@desktop2:~\$'.

```
mayank@desktop2:~$ sudo apt install nginx
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
nginx is already the newest version (1.18.0-6ubuntu14.4).
0 upgraded, 0 newly installed, 0 to remove and 217 not upgraded.
mayank@desktop2:~$
```

-
-
- To install nginx , give command : `sudo apt install nginx`.
- Now download mysql like we have downloaded in LAMP stack.

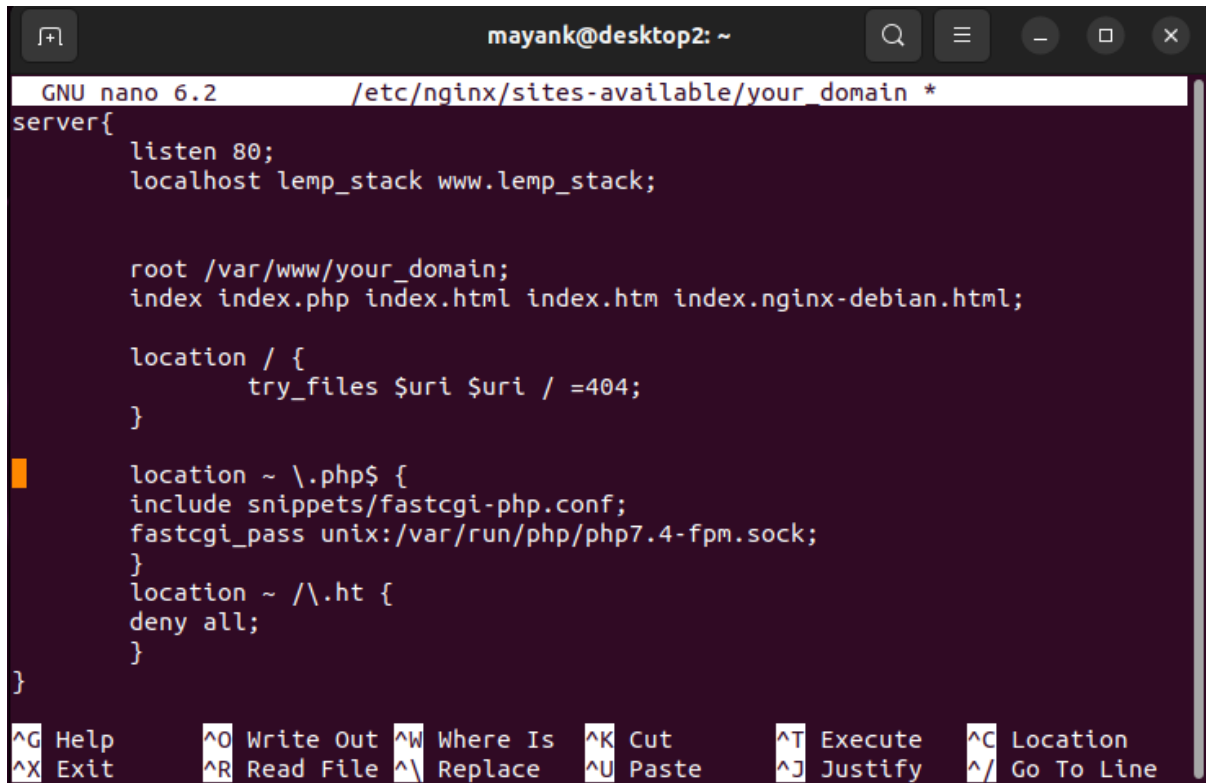
```
mayank@desktop2: ~  
Building dependency tree... Done  
Reading state information... Done  
nginx is already the newest version (1.18.0-6ubuntu14.4).  
0 upgraded, 0 newly installed, 0 to remove and 217 not upgraded.  
mayank@desktop2:~$ sudo apt install mysql-server  
Reading package lists... Done  
Building dependency tree... Done  
Reading state information... Done  
The following packages will be upgraded:  
  mysql-server  
1 upgraded, 0 newly installed, 0 to remove and 216 not upgraded.  
Need to get 9,468 B of archives.  
After this operation, 0 B of additional disk space will be used.  
Get:1 http://in.archive.ubuntu.com/ubuntu jammy-updates/main amd64 mysql-server  
all 8.0.39-0ubuntu0.22.04.1 [9,468 B]  
Fetched 9,468 B in 1s (7,437 B/s)  
(Reading database ... 212779 files and directories currently installed.)  
Preparing to unpack .../mysql-server_8.0.39-0ubuntu0.22.04.1_all.deb ...  
Unpacking mysql-server (8.0.39-0ubuntu0.22.04.1) over (8.0.36-0ubuntu0.22.04.1)  
...  
Setting up mysql-server (8.0.39-0ubuntu0.22.04.1) ...  
mayank@desktop2:~$
```

-
-
- **Setup the mysql similarly like we did in LAMP stack.**
- **Now we need to download php :**

```
mayank@desktop2:~$ sudo apt install php-fpm php-mysql  
Reading package lists... Done  
Building dependency tree... Done  
Reading state information... Done  
The following additional packages will be installed:  
  php8.3-bz2 php8.3-cli php8.3-common php8.3-curl php8.3-fpm php8.3-gd  
  php8.3-mbstring php8.3-mysql php8.3-openssl php8.3-opcache php8.3-phpdbg php8.3-readline  
  php8.3-xml php8.3-zip  
Suggested packages:  
  php-pear  
The following NEW packages will be installed:  
  php8.3-fpm php8.3-mysql  
The following packages will be upgraded:  
  php-fpm php-mysql php8.3-bz2 php8.3-cli php8.3-common php8.3-curl php8.3-gd  
  php8.3-mbstring php8.3-openssl php8.3-opcache php8.3-phpdbg php8.3-readline php8.3-xml  
  php8.3-zip  
13 upgraded, 2 newly installed, 0 to remove and 203 not upgraded.  
Need to get 7,858 kB/7,873 kB of archives.
```

-
- **Allow to take space while installing anything , by pressing “y”**
- **Configure Nginx to Use PHP Processor :**

- Create a new server block for your website. Open a new configuration file in Nginx's sites-available directory:



```

GNU nano 6.2 /etc/nginx/sites-available/your_domain *
server{
    listen 80;
    localhost lemp_stack www.lemp_stack;

    root /var/www/your_domain;
    index index.php index.html index.htm index.nginx-debian.html;


    location / {
        try_files $uri $uri / =404;
    }

    location ~ \.php$ {
        include snippets/fastcgi-php.conf;
        fastcgi_pass unix:/var/run/php/php7.4-fpm.sock;
    }
    location ~ /\.ht {
        deny all;
    }
}

^G Help      ^O Write Out ^W Where Is  ^K Cut       ^T Execute   ^C Location
^X Exit      ^R Read File ^\ Replace   ^U Paste     ^J Justify   ^_ Go To Line

```

- LEMP stack is now successfully installed.
- Add this code to the server block.
- Adjust PHP Processor Configuration
- Open the PHP configuration file for Nginx :sudo nano /etc/php/7.4/fpm/php.ini



The screenshot shows a terminal window with the title "mayank@desktop2: ~". The terminal is running the GNU nano 6.2 text editor, editing the file "/etc/php/7.4/fpm/php.ini". The current line of code is "cgi.fix_pathinfo=0". The bottom status bar displays various nano editor shortcuts: ^G Help, ^O Write Out, ^W Where Is, ^K Cut, ^T Execute, ^C Locati, ^X Exit, ^R Read File, ^\ Replace, ^U Paste, ^J Justify, and ^_ Go To.

