# STEP 1 – INSTALLING THE NGINX WEB SERVEr

* Since it is a new server I will update it by running the command

------------ sudo apt update

Text

Description automatically generated

* I will upgrade my server with the command;

---------- sudo apt upgrade

Text

Description automatically generated

* Next is to install nginx using the command:

---------- sudo apt install nginx

Text

Description automatically generated

* To verify that nginx was installed successfully I will run the command;

------ sudo systemctl status nginx

* To\*Text

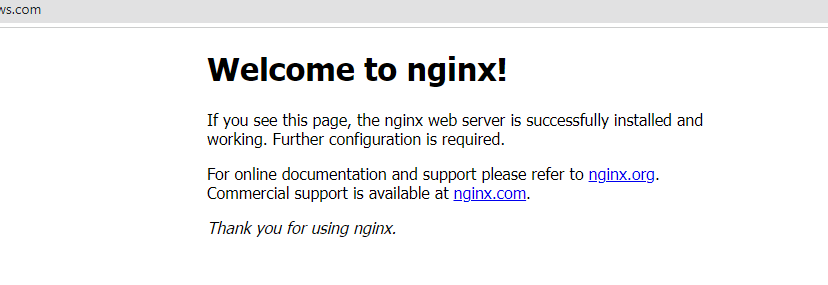
  Description automatically generated
* To access ngnix locally via DNS name I will run the command;

------------- curl <http://localhost:80>

Text

Description automatically generated

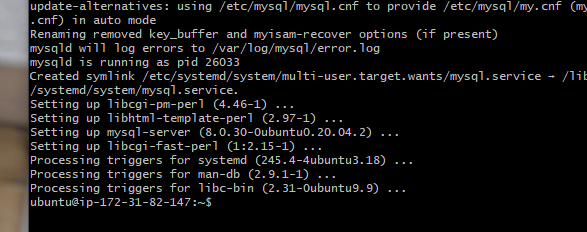
* I will go on my web browser to try to access nginx using the DNS compute information on AWS.
* It will not work because I need to edit the inbound rule on AWS by going to security, then edit it by allowing http from anywhere. Refresh the page and it will work.



# STEP 2 — INSTALLING MYSQL

* I will install MySQL by running the command.

---------- sudo apt install mysql-server



* Ones done I will log in to the MYSQL console by tying:

--------- sudo mysql

Text

Description automatically generated

* I will run a recommended security script that comes pre-installed with MYSQL.

------- ALTER USER 'root'@'localhost' IDENTIFIED WITH mysql\_native\_password BY 'PassWord.1';

Text

Description automatically generated

* I will start interactive script by running:

------ sudo mysql\_secure\_installation

Text

Description automatically generated

* I will test if I am able to log into MYSQL console by typing:

--------- sudo mysql -p

A screenshot of a computer

Description automatically generated with medium confidence

# STEP 3 – INSTALLING PHP

* I need to install PHP by running the command.

-------- sudo apt install php-fpm php-mysql

Text

Description automatically generated

* I will create the root web directory by tying:

---------- sudo mkdir /var/www/projectLEMP

Text

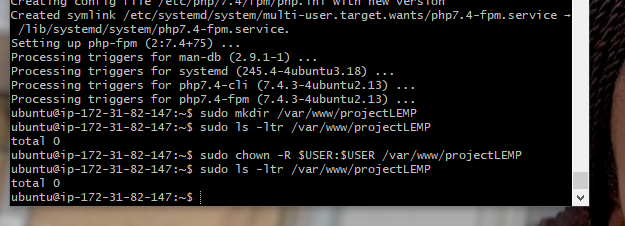
Description automatically generated

* I will assign and check if the directory was created and also assign ownership of the directory to the user:

----------- sudo ls – ltr /var/www/projectLEMP

------------ sudo chown -R $USER:$USER /var/www/projectLEMP

------------ sudo ls -ltr /var/www/projectLEMP



* I will open a new configuration:

------ sudo nano /etc/nginx/sites-available/projectLEMP

* I will paste the following in to it:

---------- server {

listen 80;

server\_name projectLEMP www.projectLEMP;

root /var/www/projectLEMP;

index index.html index.htm index.php;

location / {

try\_files $uri $uri/ =404;

}

location ~ \.php$ {

include snippets/fastcgi-php.conf;

fastcgi\_pass unix:/var/run/php/php8.1-fpm.sock;

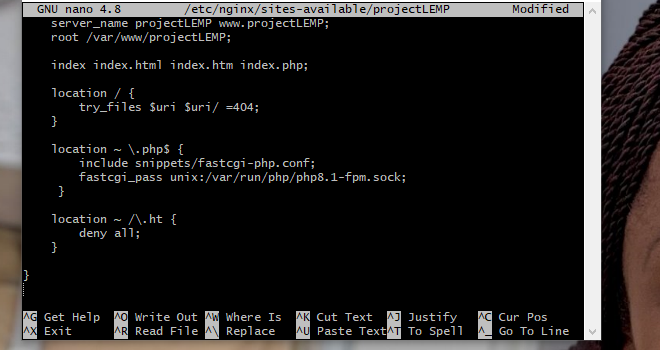
}

location ~ /\.ht {

deny all;

}

}



* I will activate my configuration by linking to the config file

---------- sudo ln -s /etc/nginx/sites-available/projectLEMP /etc/nginx/sites-enabled/

Text

Description automatically generated

* I will test if my configuration is successful by running the command:

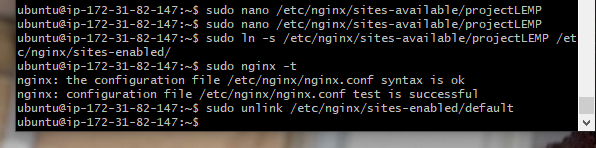
---------- sudo nginx -t

Text

Description automatically generated

* I need to disable nginx host that is currently configured to listen to ort 80 by running the command:

--------- sudo unlink /etc/nginx/sites-enabled/default



* I will reload nginx by running the command:

----------- sudo systemctl reload nginx

Text

Description automatically generated

* I need to create a index.html file in my /var/www/projectLEMP location because it is empty:

---------- sudo echo 'Hello LEMP from hostname' $(curl -s http://169.254.169.254/latest/meta-data/public-hostname) 'with public IP' $(curl -s http://169.254.169.254/latest/meta-data/public-ipv4) > /var/www/projectLEMP/index.html

Text

Description automatically generated

* I will go to my web browser and refresh it

Graphical user interface, text, application

Description automatically generated

# STEP 5 – TESTING PHP WITH NGINX

* To test PHP with nginx I will run the command:

----------- sudo nano /var/www/projectLEMP/info.php

* And paste the following into it:

----------- <?php

phpinfo();

Text

Description automatically generated

* I will reload my server running the following command:

Sudo systemctl reload nginx

Text

Description automatically generated

* I will refresh my browser and add /info.php to see the changes.

A picture containing table

Description automatically generated

* Leaving the file in my directory is not save so I will delete it by running the command:

---------- sudo rm /var/www/projectLEMP/info.php

Text

Description automatically generated

* I will refresh my browser and it will look like these:

Graphical user interface, text, application, email

Description automatically generated

# STEP 6 – RETRIEVING DATA FROM MYSQL DATABASE WITH PHP (CONTINUED)

* I will create a database by connecting to MYSQL running the command and creating my database, user and password.

---------- sudo mysql -p

Text

Description automatically generated

* I will test if the new user has the proper permission in MYSQL console by running the command:

--------- mysql -u example\_user -p

A picture containing text, monitor, indoor, screen

Description automatically generated

* I will run type the following to show database and create todo list:

Text

Description automatically generated

* I will create a PHP script that will connect to MYSQL database and queries for the content of the todo\_list table running the command:

---------- nano /var/www/projectLEMP/todo\_list.php

* I will then copy this content into it:

A screenshot of a computer

Description automatically generated with medium confidence

* Then I can access the page on my web browser using my ip address followed by the /todo\_list.php

Graphical user interface, text, application

Description automatically generated

* This means that my PHP environment is ready to connect and interact with my MYSQL server.