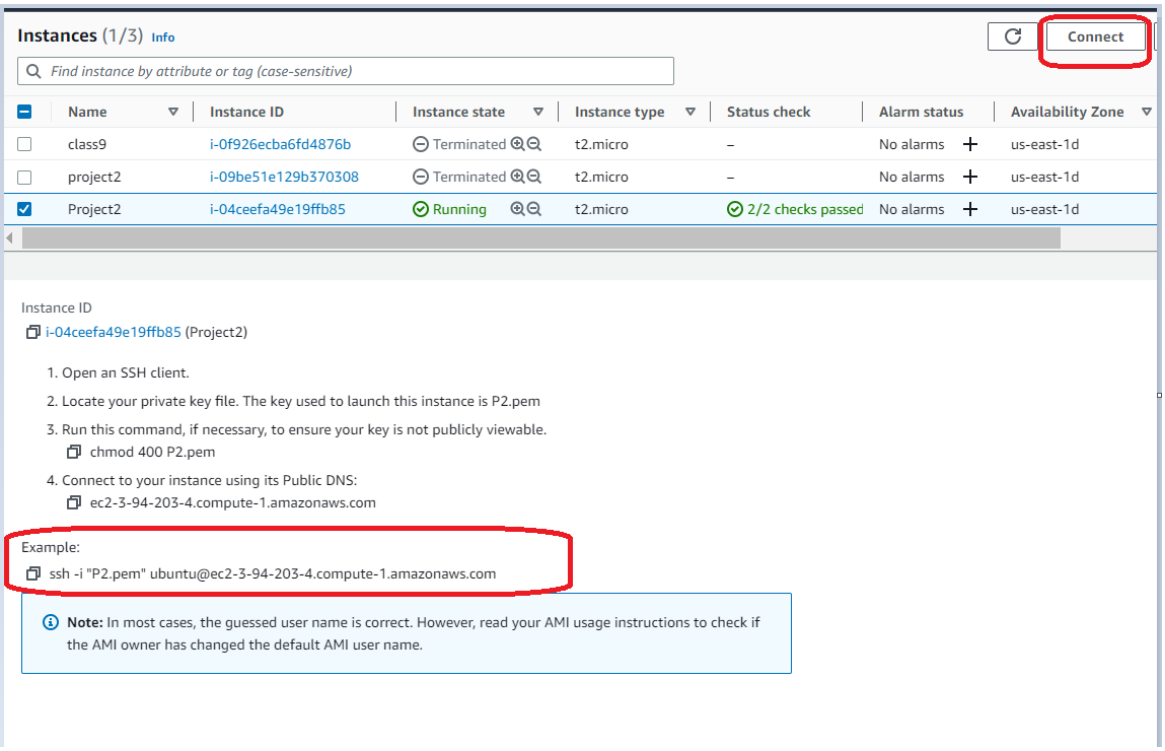
Project 2: LEMP STACK IMPLEMENTATION

Step 1 Creation of EC2 Instance on AWS using the free tier account.

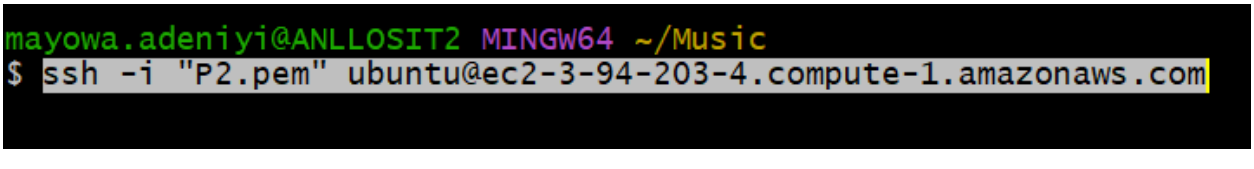
When the creation is done, you should see this on EC2 instance dashboard.

Step 2: Signing into the Ubuntu server using GIT BASH application.

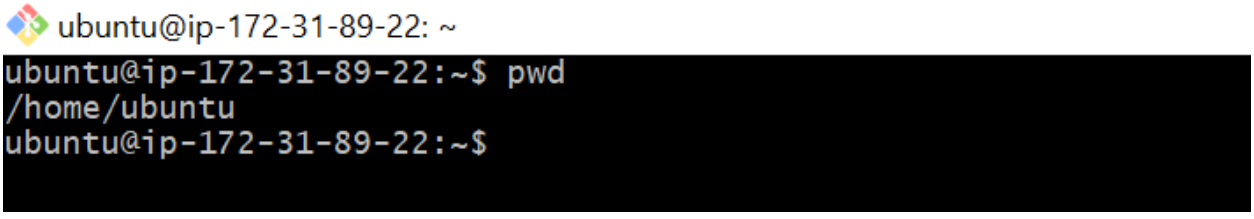
NB: you must change the directory on your GIT BASH application to the directory where the .pem file is saved. In my case I saved it in the music directory.



Connect to the EC2 instance using the GIT BASH application.



Connection is successful when you see this message below. Using the PWD command you can ensure you are on ubuntu home directory



LEMP: The L (Linux OS) is up and running

Installing NGINX – (LEMP)

Run the following commands

- sudo apt update
- sudo apt install nginx

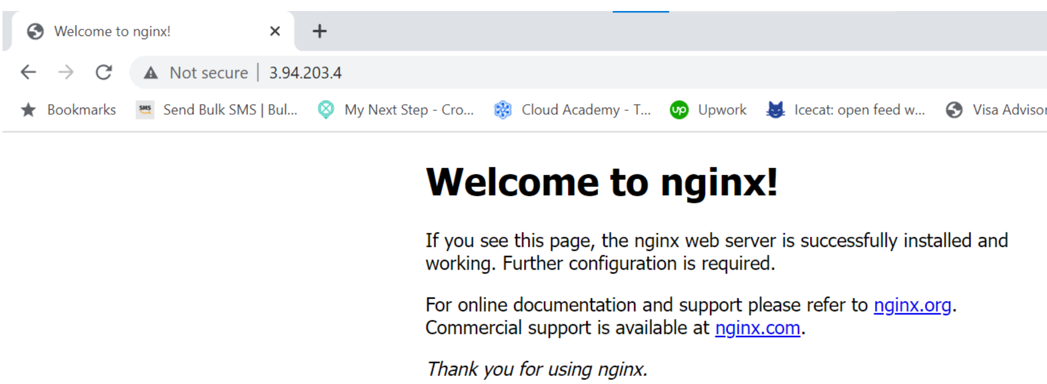
To check if nginx is successfully installed, run the command below

- sudo systemctl status nginx

```
● nginx.service - A high performance web server and a reverse proxy server
     Loaded: loaded (/lib/systemd/system/nginx.service; enabled; vendor preset: enabled)
     Active: active (running) since Sun 2023-06-18 13:02:40 UTC; 3min 7s ago
       Docs: man:nginx(8)
    Process: 2174 ExecStartPre=/usr/sbin/nginx -t -q -g daemon on; master_process on; (code=exited, status=0/SUCCESS)
    Process: 2175 ExecStart=/usr/sbin/nginx -g daemon on; master_process on; (code=exited, status=0/SUCCESS)
   Main PID: 2268 (nginx)
      Tasks: 2 (limit: 1141)
     Memory: 4.4M
        CPU: 23ms
     CGroup: /system.slice/nginx.service
             ├─2268 "nginx: master process /usr/sbin/nginx -g daemon on; master_process on;"
             └─2271 "nginx: worker process" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" ""

Jun 18 13:02:40 ip-172-31-89-22 systemd[1]: Starting A high performance web server and a reverse proxy server...
Jun 18 13:02:40 ip-172-31-89-22 systemd[1]: Started A high performance web server and a reverse proxy server.
```

To test that the Nginx server can be accessed from the Internet. Open a web browser of your choice and try to access the server using the public IP Address which is http://3.94.203.4/ in this case



Installing MYSQL (LEMP)

To install MYSQL, using the commands below.

- sudo apt install mysql-server

I use the command "which mysql" to confirm that mysql has been installed.

```
ubuntu@ip-172-31-89-22:~$
ubuntu@ip-172-31-89-22:~$ which mysql
/usr/bin/mysql
ubuntu@ip-172-31-89-22:~$
```

To go into the mysql console, use the command “sudo mysql”

```
ubuntu@ip-172-31-89-22:~$ sudo mysql
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 9
Server version: 8.0.33-0ubuntu0.22.04.2 (Ubuntu)

Copyright (c) 2000, 2023, 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>
```

It’s recommended that you run a security script that comes pre-installed with MySQL. This script will remove some insecure default settings and lock down access to your database system. Before running the script you will set a password for the root user, using mysql_native_password as default authentication method. We’re defining this user’s password as PassWord.1.

- ALTER USER 'root'@'localhost' IDENTIFIED WITH mysql_native_password BY 'PassWord.1';

Installing PHP (LEMP)

Run the command below to install PHP

- sudo apt install php-fpm php-mysql

Configuring NGINX to use PHP Processor

On Ubuntu 20.04, Nginx has one server block enabled by default and is configured to serve documents out of a directory at /var/www/html. While this works well for a single site, it can become difficult to manage if you are hosting multiple sites. Instead of modifying /var/www/html, we’ll create a directory structure within /var/www for the your_domain website, leaving /var/www/html in place as the default directory to be served if a client request does not match any other sites.

We will be creating the directory “projectLEMP” in the directory /var/www. You can do that using the command below

- sudo mkdir /var/www/projectLEMP

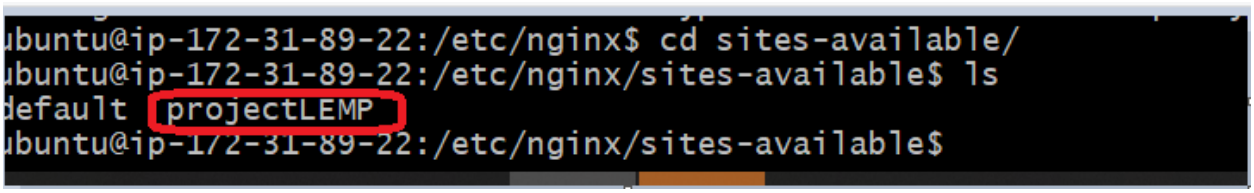
```
ubuntu@ip-172-31-89-22:~$ sudo mkdir /var/www/projectLEMP
ubuntu@ip-172-31-89-22:~$ ls
ubuntu@ip-172-31-89-22:~$ cd /var/www
ubuntu@ip-172-31-89-22:/var/www$ ls
html  projectLEMP
```

Next, assign ownership of the directory with the $USER environment variable, which will reference your current system user using the command below

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

We are then going to create and edit a file (projectLEMP) in Nginx **sites-available** (/etc/nginx/sites-available/) directory using your preferred editor. For this we shall use NANO. Below is the command.

You can use the ls command to list content in the directory



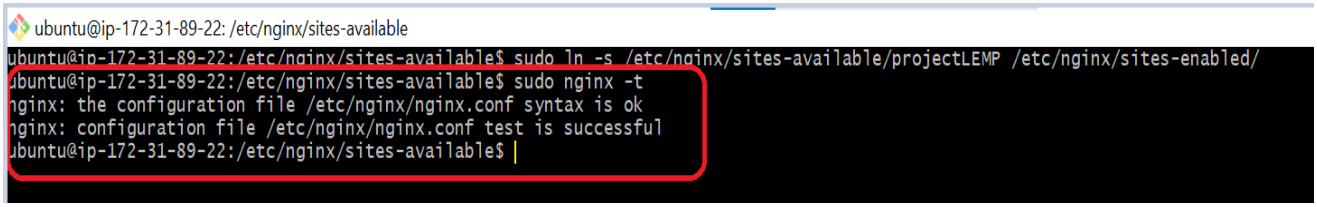
Activate your configuration by linking to the config file from Nginx's sites-enabled directory:

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

The command below will tell Nginx to use the configuration next time it is reloaded.

- sudo nginx -t

Below is the result.



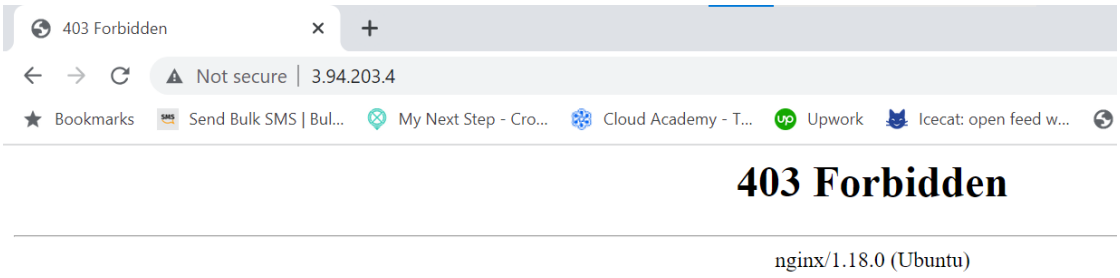
We need to disable the default Nginx host that is configured to listen on port 80 for this run the command below.

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

Now reload nginx to apply changes

- sudo systemctl reload nginx

Reloading the website gives this error message because there is not file in the /var/www/projectLEMP directory. We will now go ahead to create an index file there.

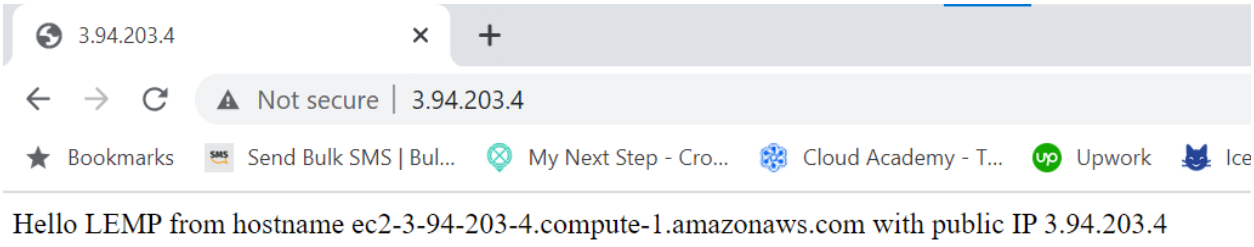


We can do this by running the command below.

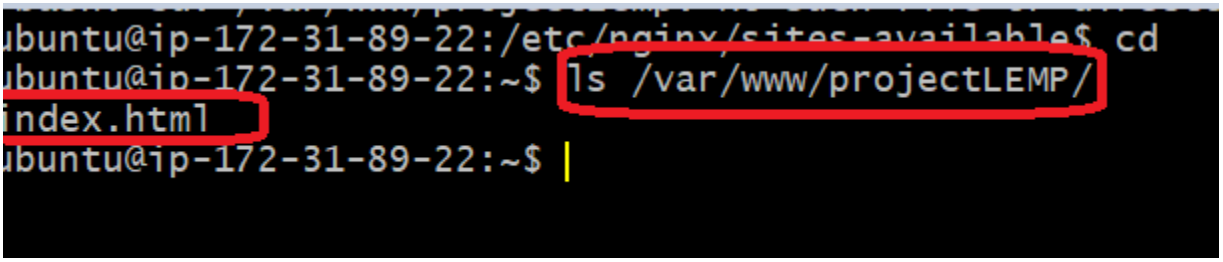
- sudo echo 'Hello LEMP from hostname' $(curl -s http://169.254.169.254/latest/meta-data/public-hostname) 'with public IP' $(curl -s http://169.254.169.254/latest/meta-data/public-ipv4) > /var/www/projectLEMP/index.html

This command is simply saying that we should display "'Hello LEMP from hostname' $(curl -s http://169.254.169.254/latest/meta-data/public-hostname)" and save it in a file named index.html and store in the directory /var/www/projectLEMP

If you refresh your browser now using the public IP Address, we get the result below.



We can also verify from the server that an index file now exists in the directory /var/www/projectLEMP



Testing PHP with NGINX

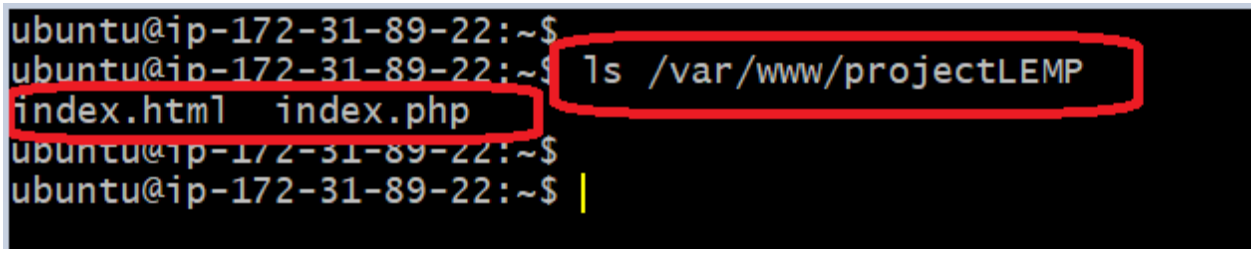
Now that LEMP has been completely set up, we can test to see if NGINX can correctly hand PHP files to your PHP Processor. To test this we will create a PHP file > info.php in the same directory as the index.html /var/www/projectLEMP

We can do that using the nano editor and using the command below.

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

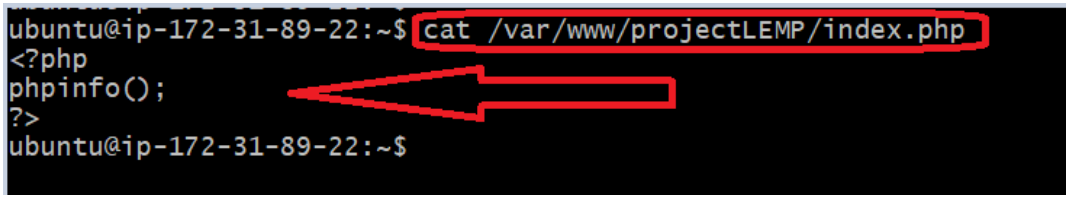
the nano editor comes up and we are to copy and paste the php code below

<?php
phpinfo();

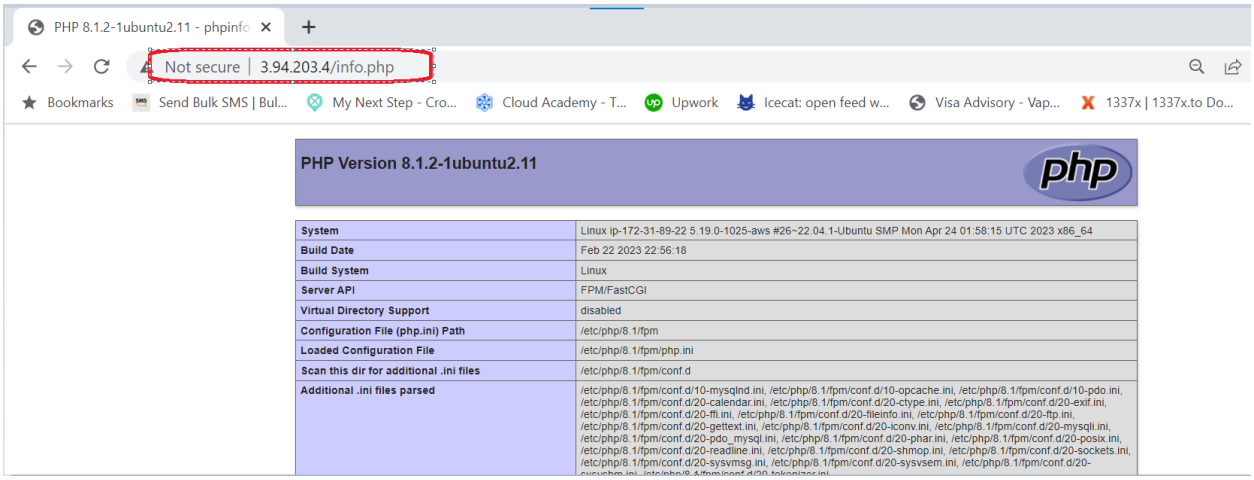


We can see both files now.

If i want to check the content of the info.php file, I can do that by running the command below



You can not access the page on your browser by going the public_ip _address /info.php in this case http://3.94.203.4/info.php

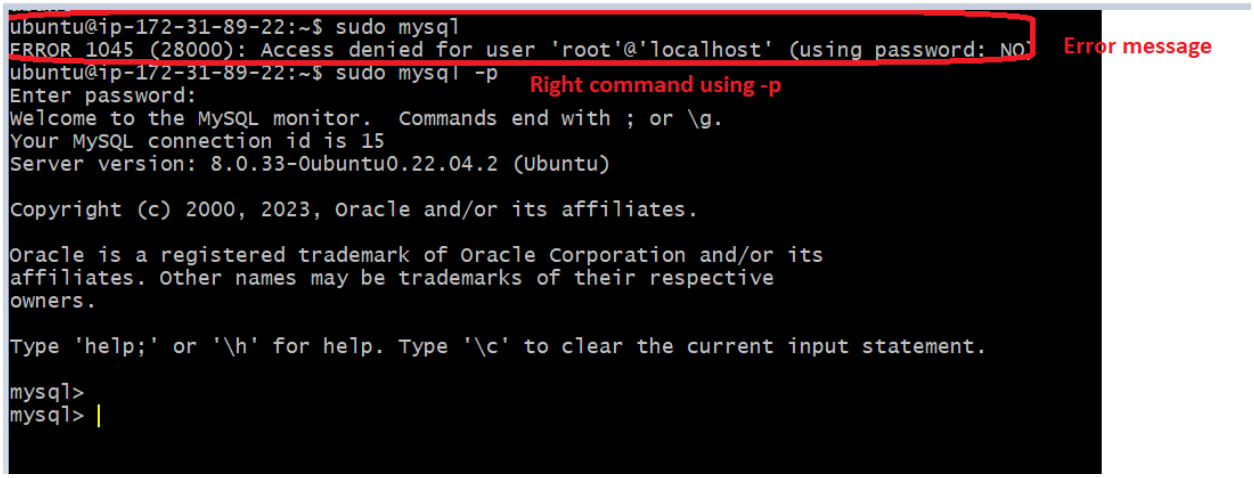


Retrieving data from mysql database using php

We will be creating a database using mysql which was installed some steps ago. To do this just type the command below

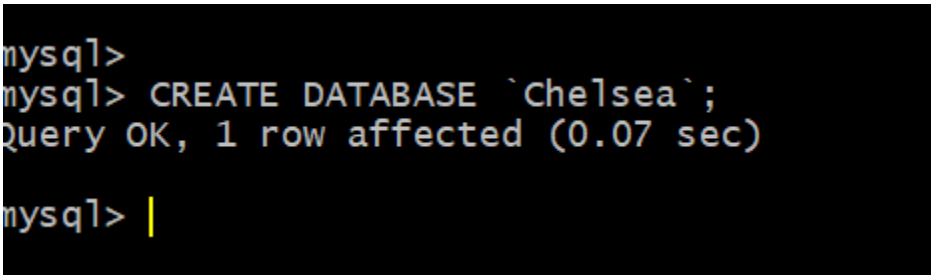
Sudo mysql -p

We are using -p because we have put a password on the mysql module. With out this you would get the error message you can see in the image below



To create a database use the command below

CREATE DATABASE `example_database`; in this case the name of our database is “Chelsea”



The next step is to create a user and assign a password to that user. In this case we will create the user Lampard and assign the password “password” to this user. We can do this using the command below

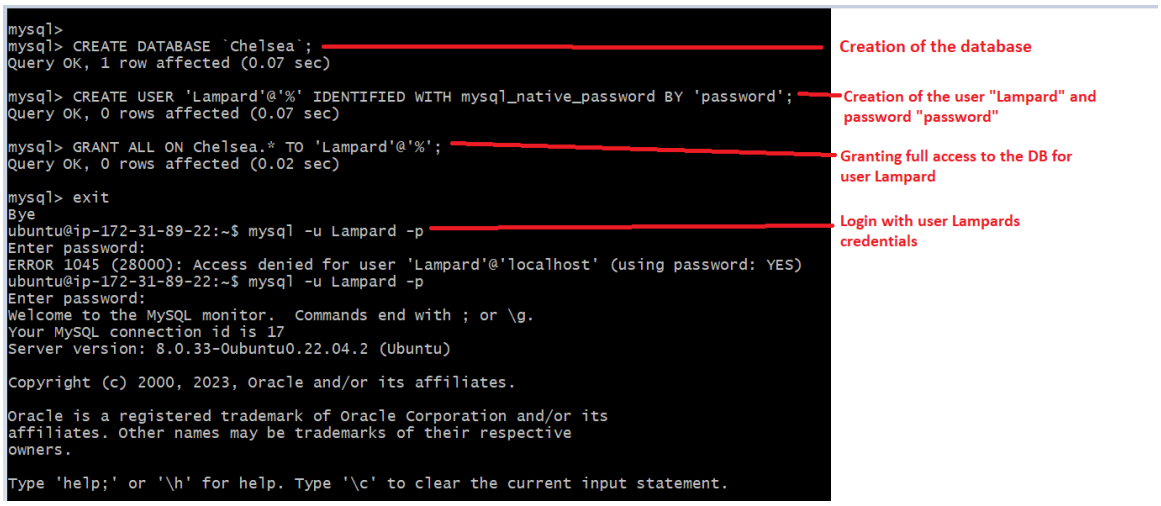
- CREATE USER 'Lampard'@'%' IDENTIFIED WITH mysql_native_password BY 'password';

Now we need to give this user permission over the Chelsea database: using the command below

- GRANT ALL ON Chelsea.* TO 'Lampard'@'%';

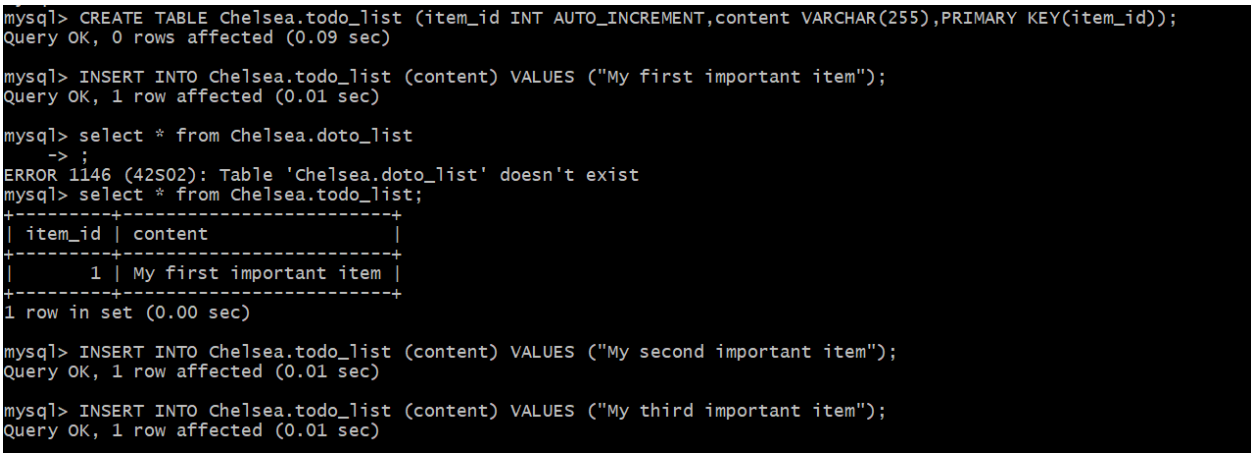
You can now exit from mysql and log in using the new users credentials. The command below does that

- mysql -u Lampard -p
    - A password prompt comes out and you can put the password



After logging with the new user credentials, we are going to create a table and insert some content into that table. We can do that by using the commands below

- `CREATE TABLE Chelsea.todo_list (item_id INT AUTO_INCREMENT,content VARCHAR(255),PRIMARY KEY(item_id));`
- `INSERT INTO Chelsea.todo_list (content) VALUES ("My first important item");`
- `INSERT INTO Chelsea.todo_list (content) VALUES ("My second important item");`
- `INSERT INTO Chelsea.todo_list (content) VALUES ("My third important item");`



Now you can create a PHP script that will connect to MySQL and query for your content. Create a new PHP file in your custom web root directory using your preferred editor. We will use nano for this using the command below.

- nano /var/www/projectLEMP/todo_list.php

I will then go ahead to copy and paste the code below and save the file todo_list.php.

```
<?php
$user = "Lampard";
$password = "password";
$database = "Chelsea";
$table = "Chelsea.todo_list";
```

```
try {
  $db = new PDO("mysql:host=localhost;dbname=$database", $user, $password);
  echo "<h2>TODO</h2><ol>";
  foreach($db->query("SELECT content FROM $table") as $row) {
    echo "<li>" . $row['content'] . "</li>";
  }
  echo "</ol>";
} catch (PDOException $e) {
    print "Error!: " . $e->getMessage() . "<br/>";
    die();
}
```

You can not access the database using the url below

Your_public_ip_address/todo_list.php which is http://3.94.203.4/todo_list.php in this case