INFOTRIXS INTERNSHIP

Task1:

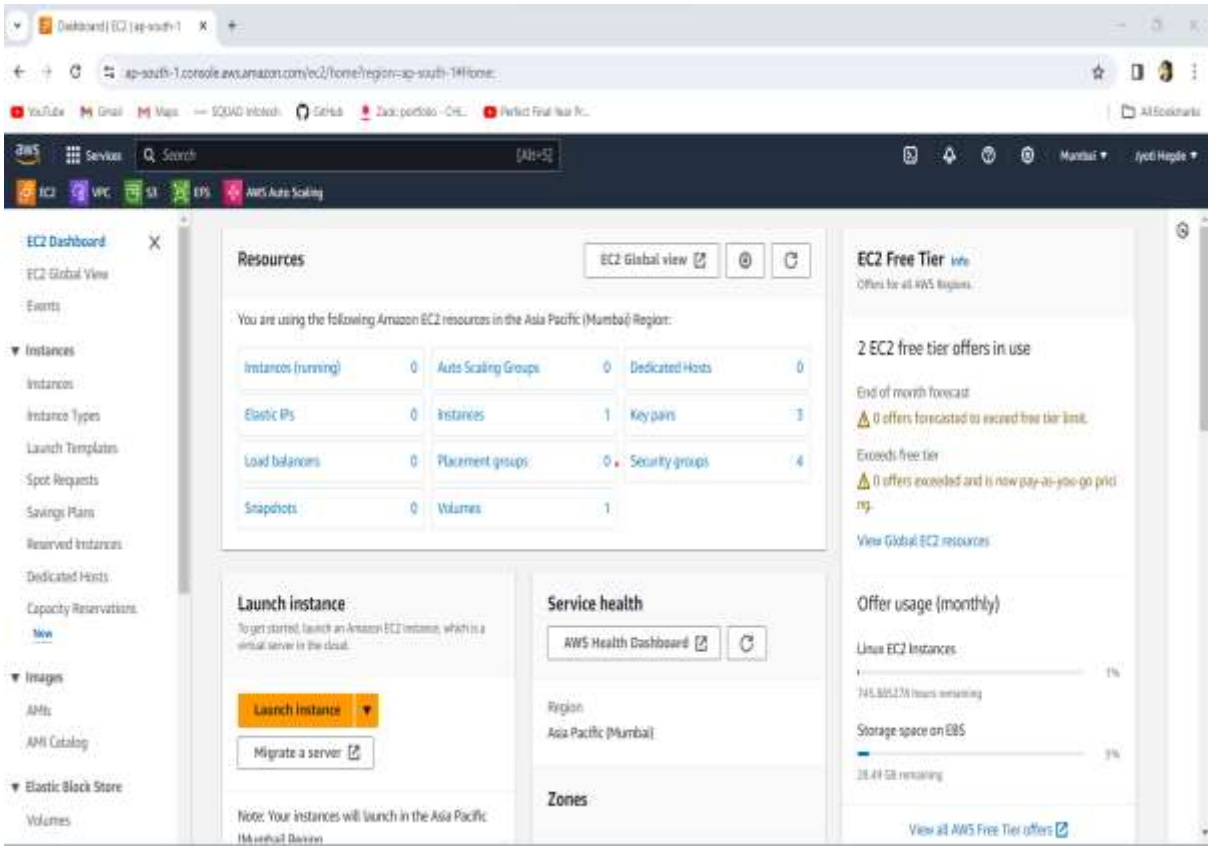
A. For monolith: 1 EC2 instance create, deploy WordPress and MYSQL on the same instance.
B. For microservice: 2 EC2 instance create, 1 for WordPress and 1 for MYSQL on the different instance.
C. Create a welcome page in WordPress that will be the homepage.

Pre-requisites:

- Configure the necessary security group for the instances.
- EC2 instance type: t2 micro, AMI: Ubuntu

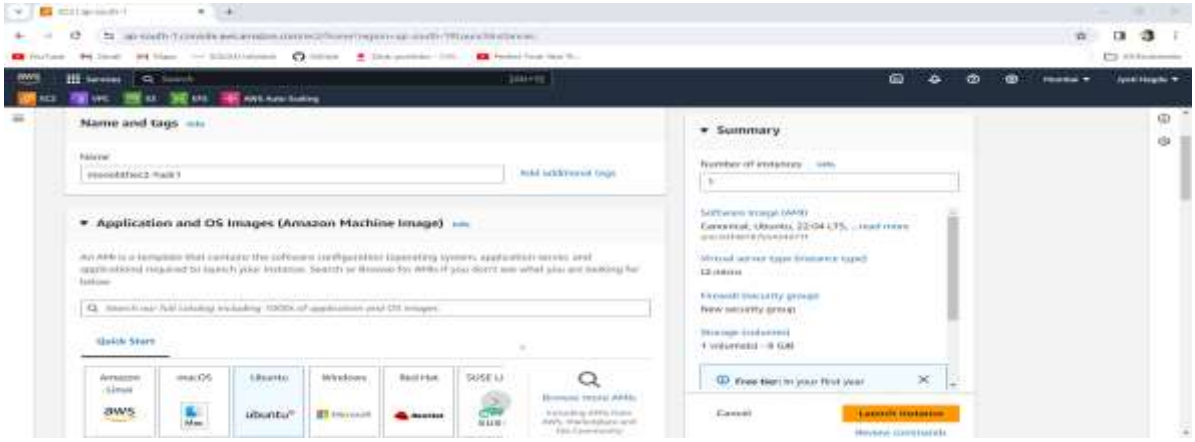
❖ Lab Performed on Monolith: [A]

Step1: Login to AWS free tier account. Search for EC2 instance in the dashboard.

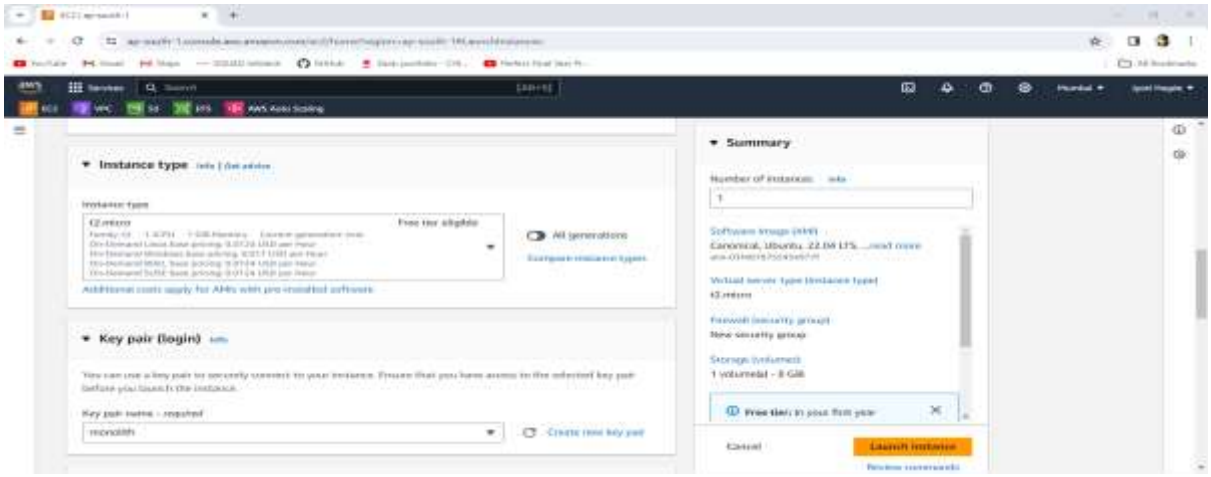


Step 2: Create one EC2 instance.

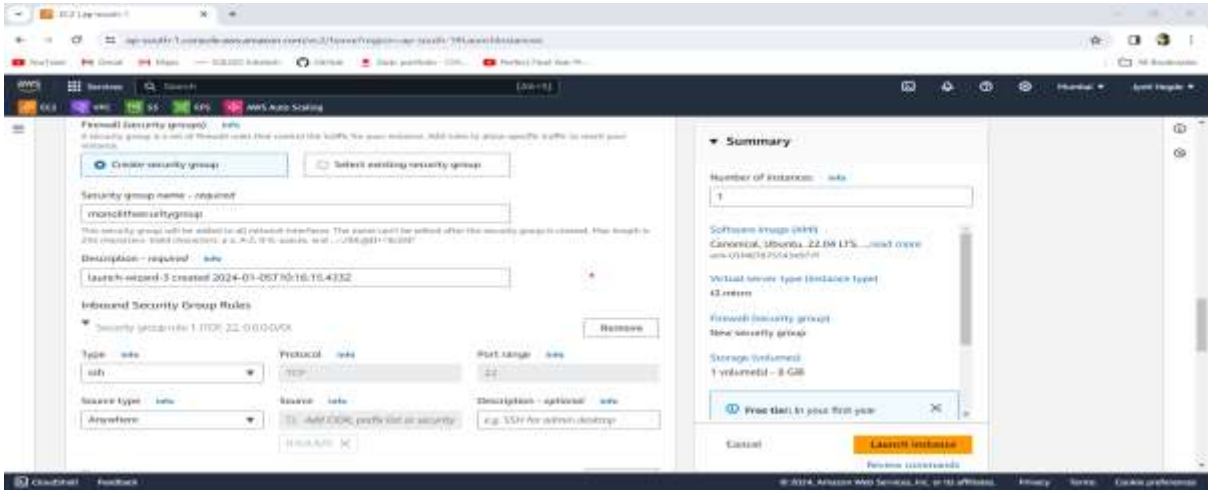
- Name: monolithec2-task1
- AMI: Ubuntu.

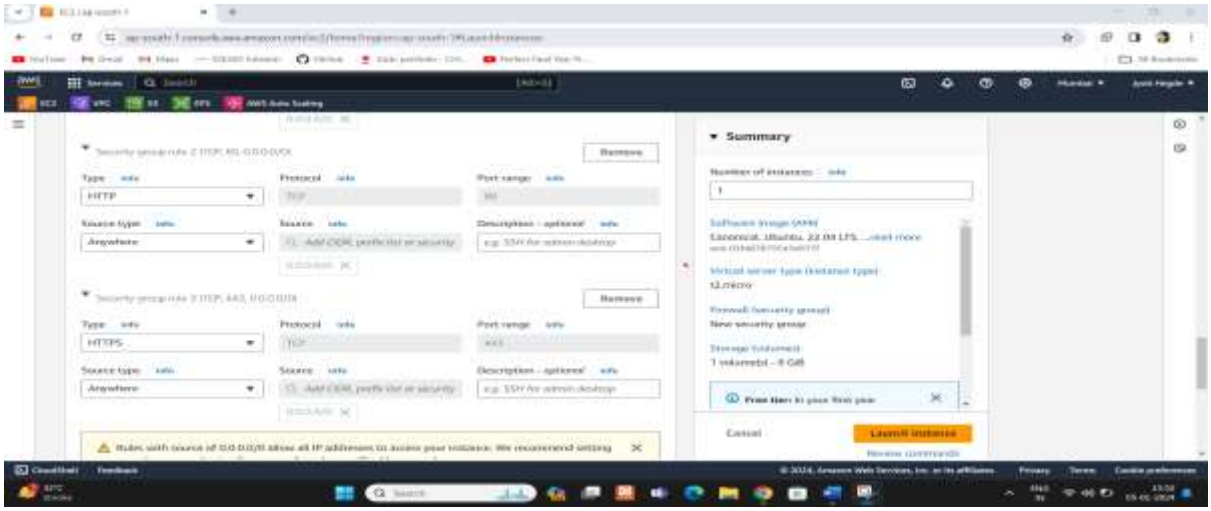


- Instance Type: t2.micro
- Key pair: monolith-key (new created)

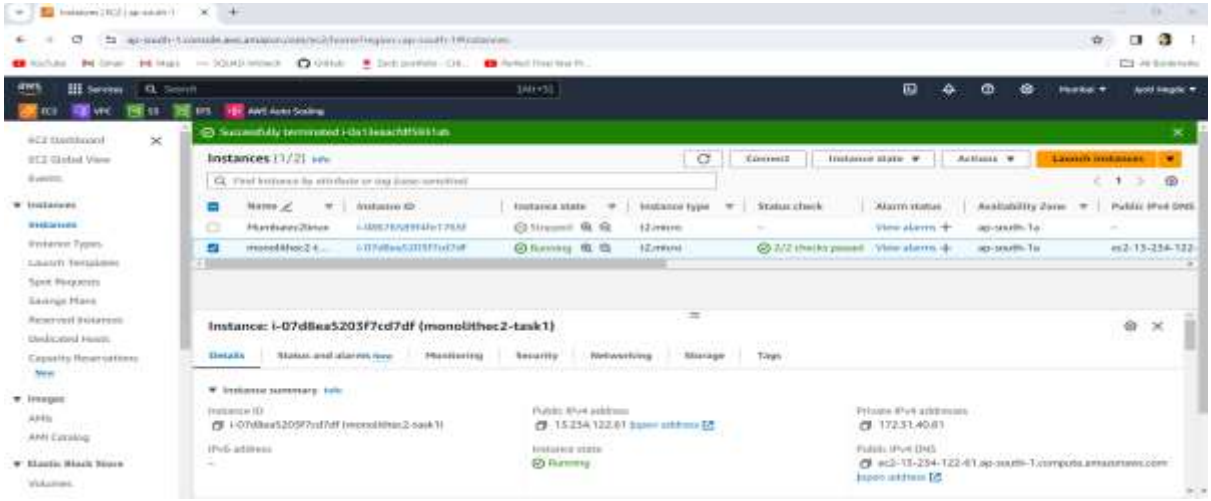


Step 3: allowed inbound traffic for SSH, HTTP/HTTPS (port 80 and 443) for WordPress access and port 3306 for MySQL access.

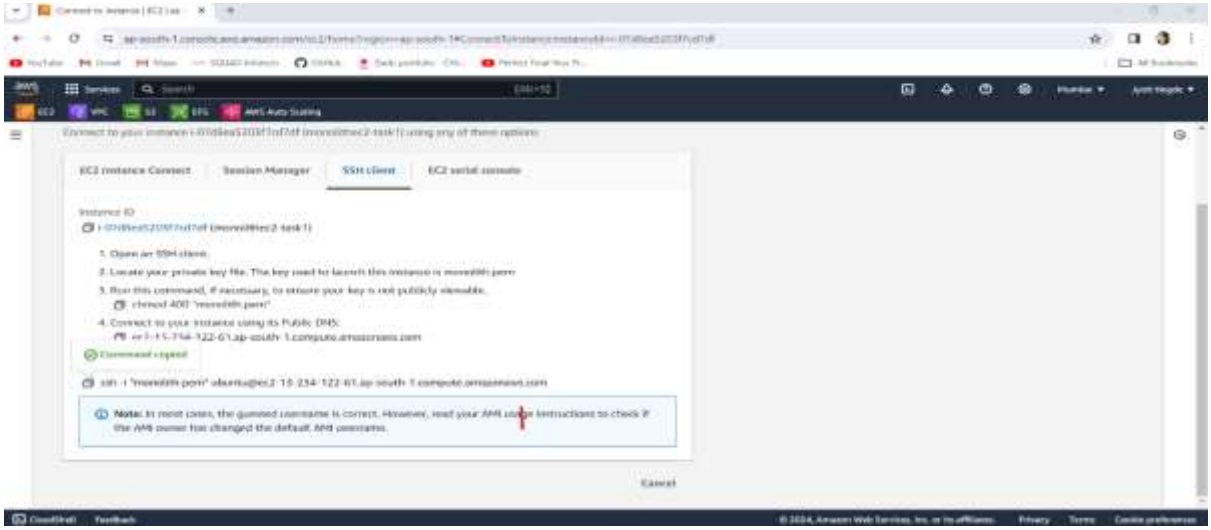




Step 4: The EC2 Instance launched Successfully->Click on connect to further process.



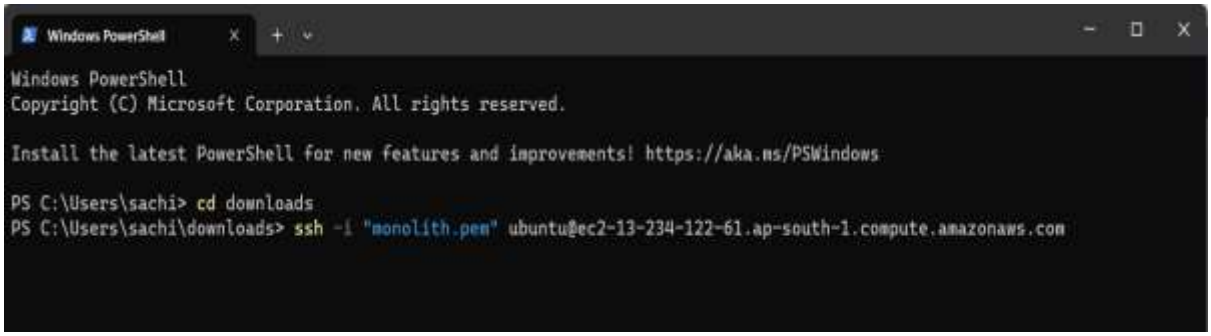
Step 5: To connect the Ubuntu, we need to copy the ssh.



Step 6: Open terminal, go to the download folder (where our security key is downloaded)

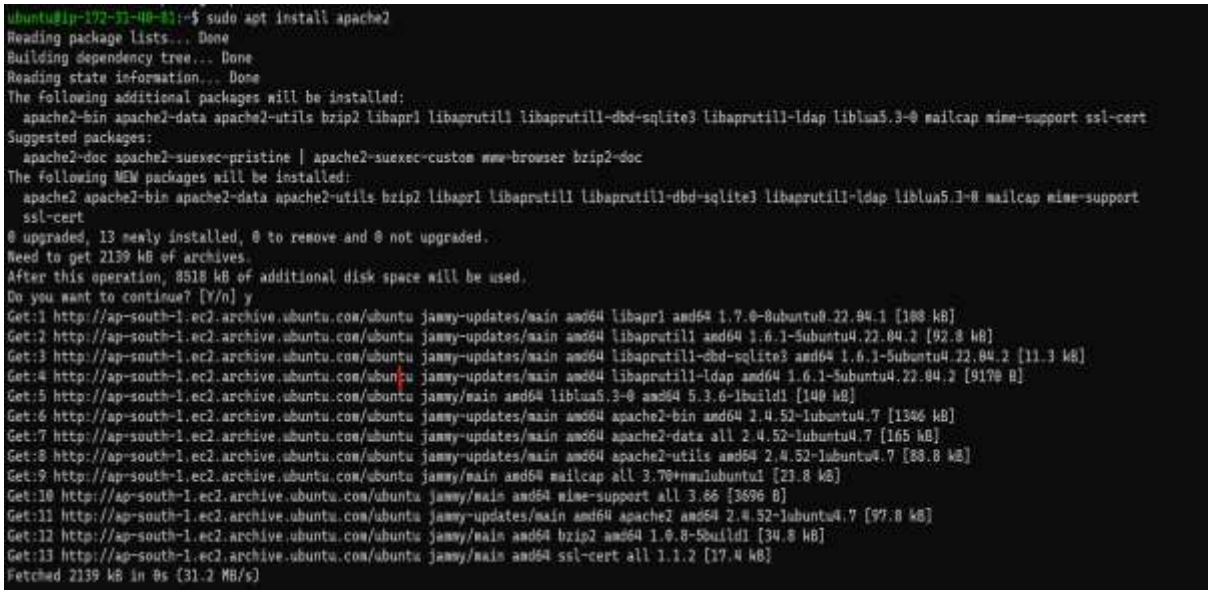
➔ **cd downloads**

paste the ssh path (copied above) on terminal to connect our Ubuntu server with our EC2 instance.



Step 7: Installed apache2 server on Ubuntu.

➔ **sudo apt install apache2**



Step 8: Now copy the public IP of the EC2 and paste on the new tab

➔ http://<public-ip>

The Apache2 Default page open means our wordpress is up and running successfully.

Step 9: Installed php runtime and php mysql connector.

➔ **sudo apt install php libapache2-mod-php php-mysql**

```
ubuntu@ip-172-31-40-81:~$ sudo apt install php libapache2-mod-php php-mysql
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  libapache2-mod-php8.1 php-common php8.1 php8.1-cli php8.1-common php8.1-mysql php8.1-opcache php8.1-readline
Suggested packages:
  php-pear
The following NEW packages will be installed:
  libapache2-mod-php libapache2-mod-php8.1 php php-common php-mysql php8.1 php8.1-cli php8.1-common php8.1-mysql php8.1-opcache php8.1-readline
0 upgraded, 11 newly installed, 0 to remove and 0 not upgraded.
Need to get 5265 kB of archives.
After this operation, 21.8 MB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy/main amd64 php-common all 2:92ubuntu1 [12.4 kB]
Get:2 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 php8.1-common amd64 8.1.2-1ubuntu2.14 [1127 kB]
Get:3 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 php8.1-opcache amd64 8.1.2-1ubuntu2.14 [365 kB]
Get:4 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 php8.1-readline amd64 8.1.2-1ubuntu2.14 [13.6 kB]
Get:5 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 php8.1-cli amd64 8.1.2-1ubuntu2.14 [1834 kB]
Get:6 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 libapache2-mod-php8.1 amd64 8.1.2-1ubuntu2.14 [1766 kB]
Get:7 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy/main amd64 libapache2-mod-php all 2:8.1+92ubuntu1 [2898 B]
Get:8 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 php8.1 all 8.1.2-1ubuntu2.14 [9158 B]
Get:9 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy/main amd64 php all 2:8.1+92ubuntu1 [2756 B]
Get:10 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 php8.1-mysql amd64 8.1.2-1ubuntu2.14 [130 kB]
```

Step 10: Intalled MySql Server.

➔ **Sudo apt install mysql-server**

```
ubuntu@ip-172-31-40-81:~$ sudo apt install mysql-server
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  libcgi-fast-perl libcgi-pm-perl libclone-perl libencode-locale-perl libevent-pthreads-2.1-7 libfcgi-bin libfcgi-perl libfcgi0ldbl libhtml-parser-perl
  libhtml-tagset-perl libhtml-template-perl libhttp-date-perl libhttp-message-perl libio-html-perl liblwp-mediatypes-perl libmecab2 libprotobuf-lite23
  libtimedate-perl liburi-perl mecab-ipadic mecab-ipadic-utf8 mecab-utils mysql-client-8.0 mysql-client-core-8.0 mysql-common mysql-server-8.0
  mysql-server-core-8.0
Suggested packages:
  libdata-dump-perl libipc-sharedcache-perl libbusiness-isbn-perl libwww-perl mailx tinyca
The following NEW packages will be installed:
  libcgi-fast-perl libcgi-pm-perl libclone-perl libencode-locale-perl libevent-pthreads-2.1-7 libfcgi-bin libfcgi-perl libfcgi0ldbl libhtml-parser-perl
  libhtml-tagset-perl libhtml-template-perl libhttp-date-perl libhttp-message-perl libio-html-perl liblwp-mediatypes-perl libmecab2 libprotobuf-lite23
  libtimedate-perl liburi-perl mecab-ipadic mecab-ipadic-utf8 mecab-utils mysql-client-8.0 mysql-client-core-8.0 mysql-common mysql-server
  mysql-server-8.0 mysql-server-core-8.0
0 upgraded, 28 newly installed, 0 to remove and 0 not upgraded.
Need to get 29.6 MB of archives.
After this operation, 243 MB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy/main amd64 mysql-common all 5.8+1.0.8 [7212 B]
Get:2 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 mysql-client-core-8.0 amd64 8.0.35-0ubuntu0.22.04.1 [2682 kB]
Get:3 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 mysql-client-8.0 amd64 8.0.35-0ubuntu0.22.04.1 [22.7 kB]
Get:4 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy/main amd64 libevent-pthreads-2.1-7 amd64 2.1.12-stable-1build3 [7642 B]
Get:5 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy/main amd64 libmecab2 amd64 0.996-14build9 [199 kB]
```

Step 11: Login to MySql Server

➔ **sudo mysql -u root**

```
ubuntu@ip-172-31-40-81:~$ sudo mysql -u root
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 8
Server version: 8.0.35-0ubuntu0.22.04.1 (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>
```

Step 12: Change authentication plugin to mysql_native_password (Choose strong Password)

```
mysql> ALTER USER 'root'@localhost IDENTIFIED WITH mysql_native_password BY 'jyoti123';
Query OK, 0 rows affected (0.01 sec)
```

Step 13: Create new database user for wordpress

```
mysql> CREATE USER 'jyoti'@localhost IDENTIFIED BY 'jyoti123';
Query OK, 0 rows affected (0.02 sec)
```

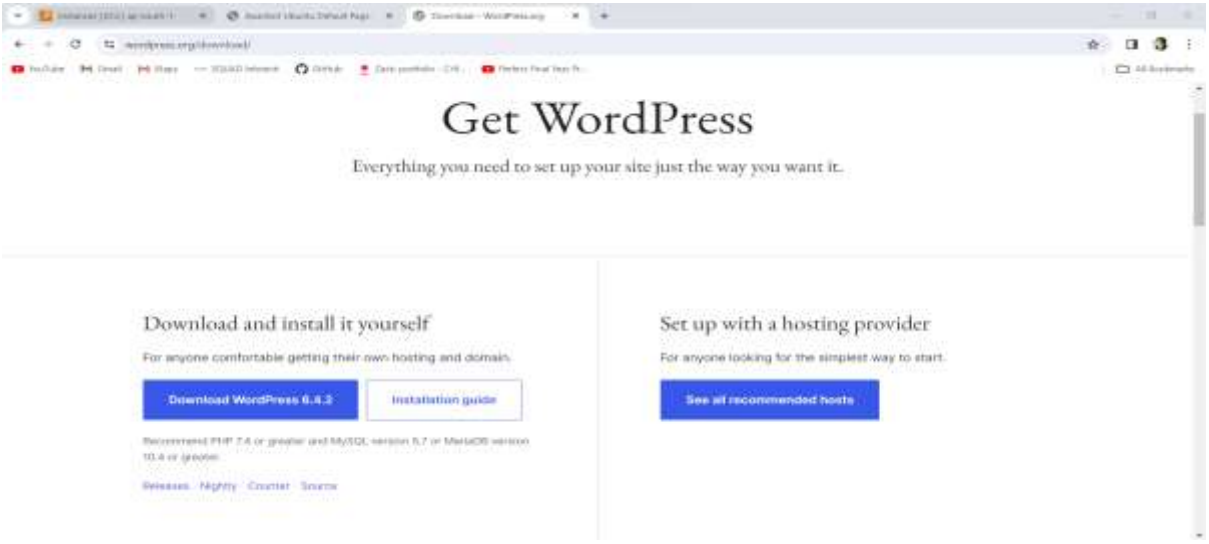
Step 14: Create a database for wordpress

```
mysql> CREATE DATABASE jyotiwordpress;
Query OK, 1 row affected (0.01 sec)
```

Step 15: Grant all Privileges on the database ‘wordpress’ to the newly created user.

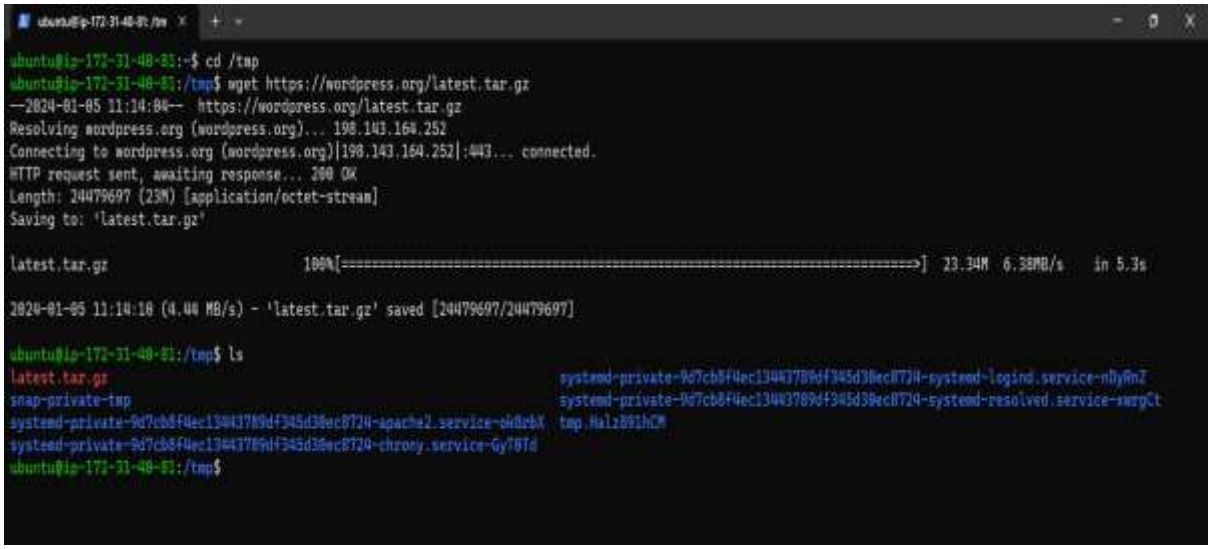
```
mysql> GRANT ALL ON *.* TO 'jyoti'@localhost;
Query OK, 0 rows affected (0.01 sec)
```

Step 16: For download wordpress ->go to the wordpress.org website on Google, copy the download link.



Step 17: on terminal type command as:

- ➔ cd /tmp
- ➔ wget https://wordpress.org/latest.tar.gz

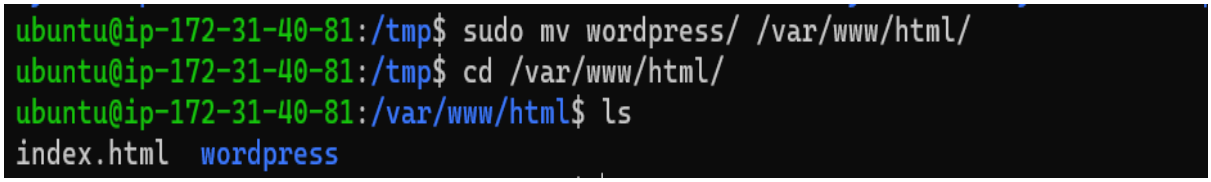


Step 18: Extract the zip file.

Step 19: Now WordPress is successfully installed.

➔ ls

Step 20: wordpress file contain index.html



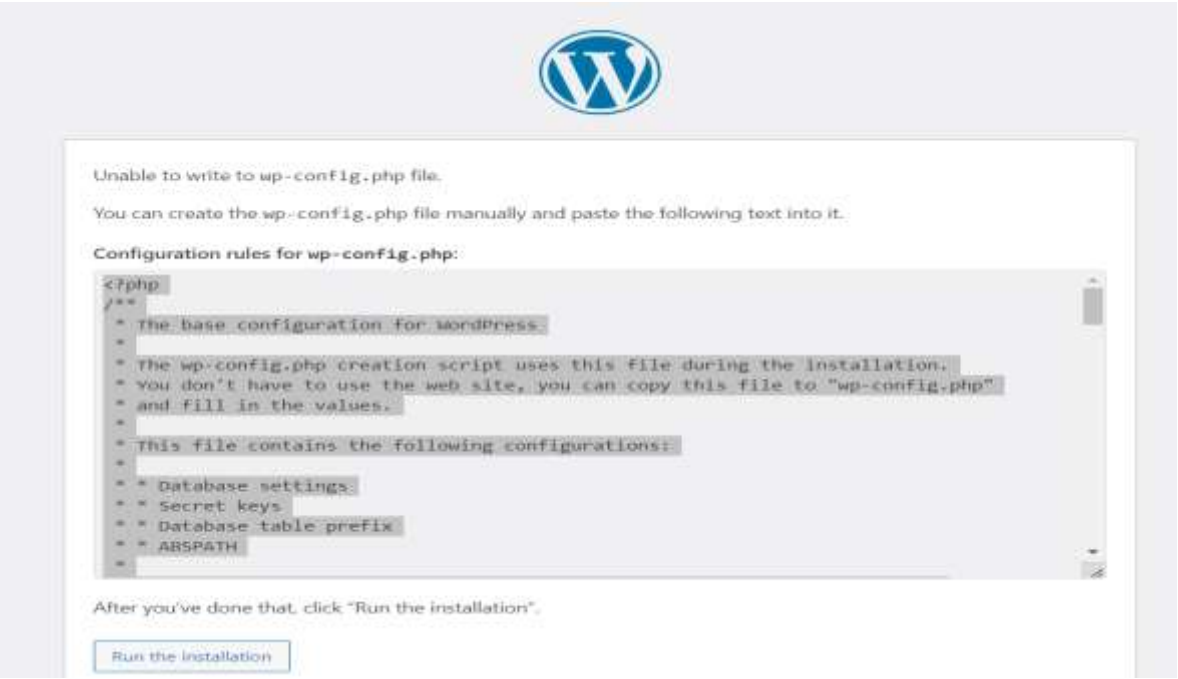
Step 21: Now copy the public IP of the EC2 and paste on the new tab as below,

➔ http://<public-ip-of-ec2>/wordpress

Step 22: go to next step and fill all details – database name, username & password.

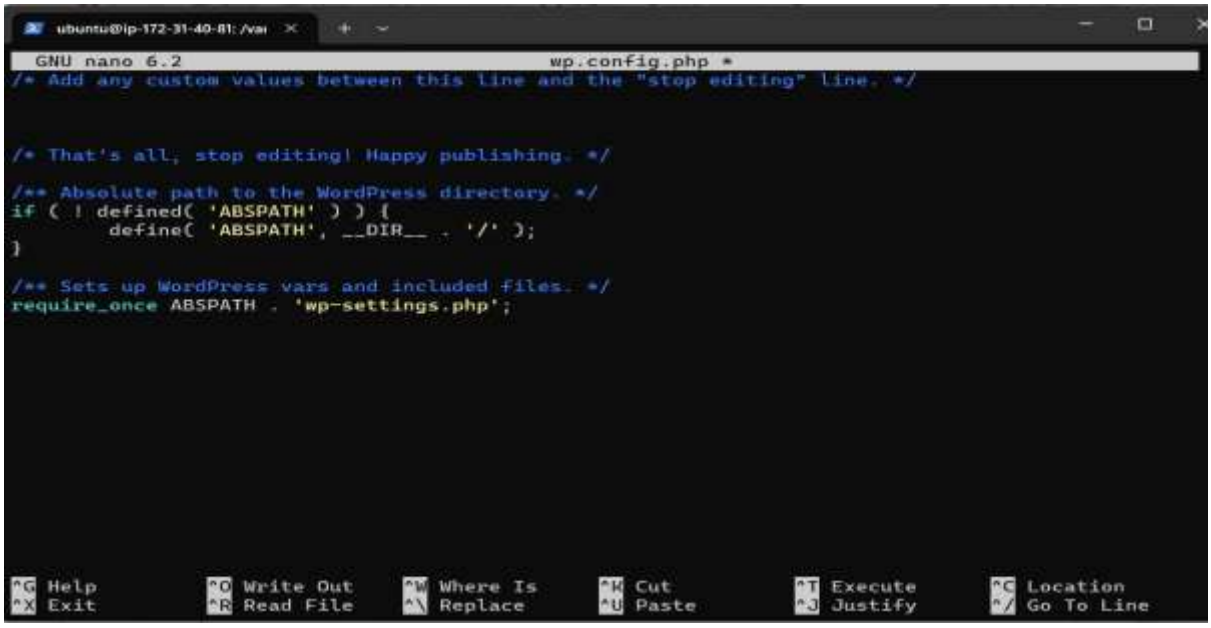
Click on submit.

Step 23: wp-config.php error occur to fix it copy the code below.

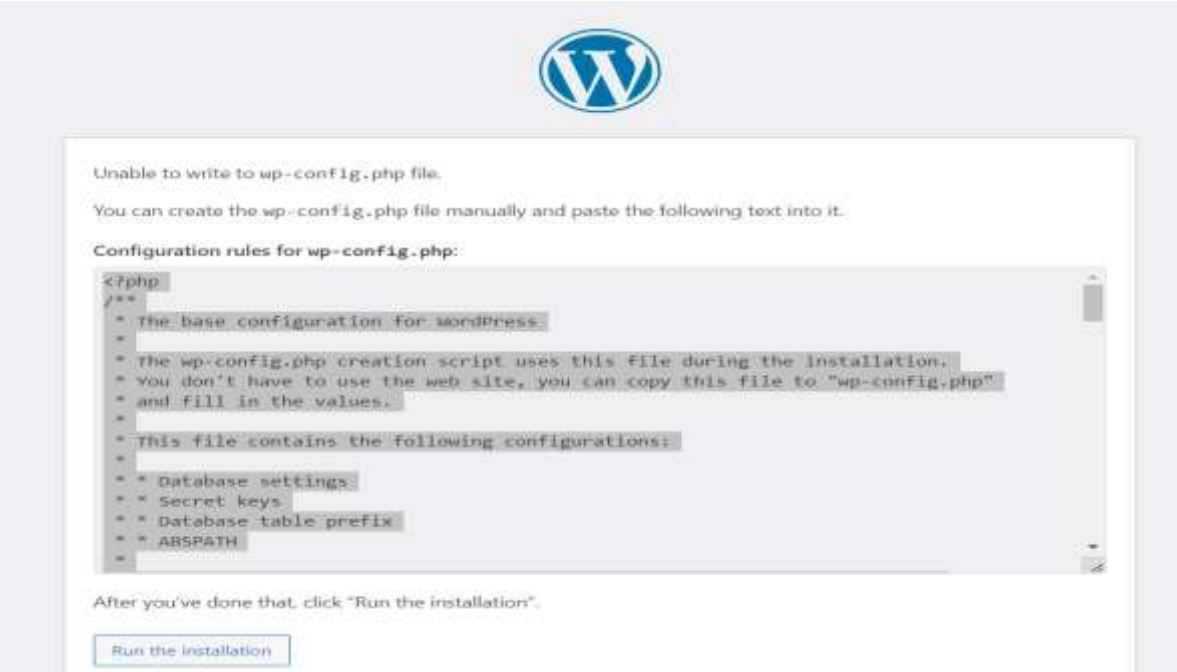


Step 24: on the terminal, inside wordpress folder-> create nano wp.config.php file and paste above copied code.

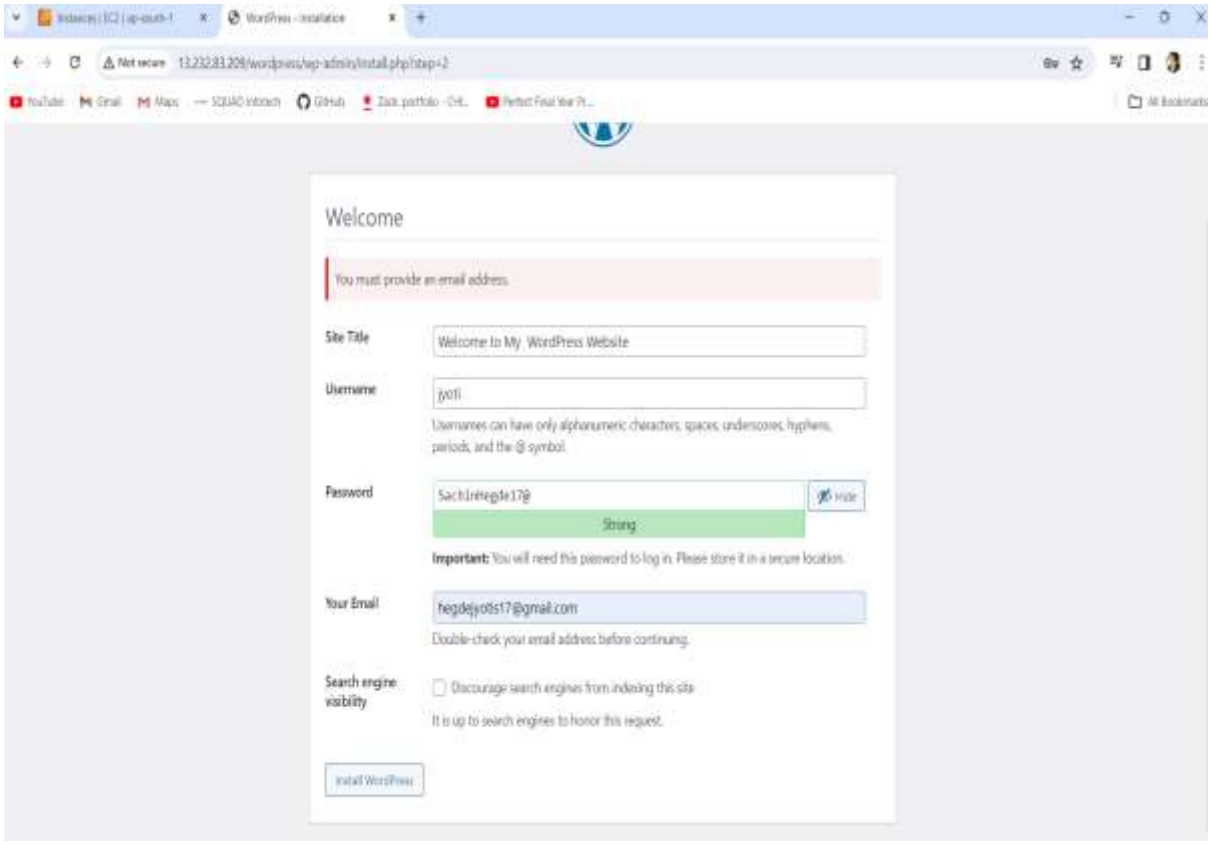
```
ubuntu@ip-172-31-40-81:/var/www/html$ cd wordpress
ubuntu@ip-172-31-40-81:/var/www/html/wordpress$ nano wp.config.php
```



Step 25: Click on Run the installation button.

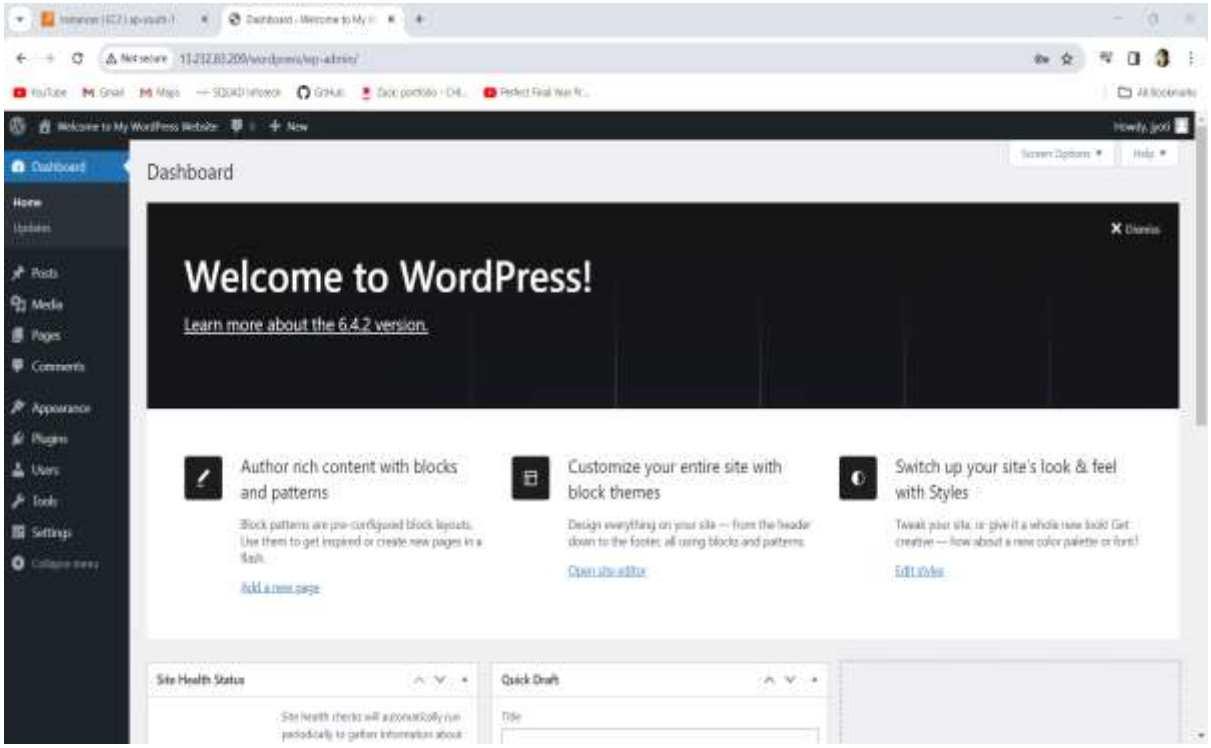


Step 26: Fill all the details to host/deploy your webpage.

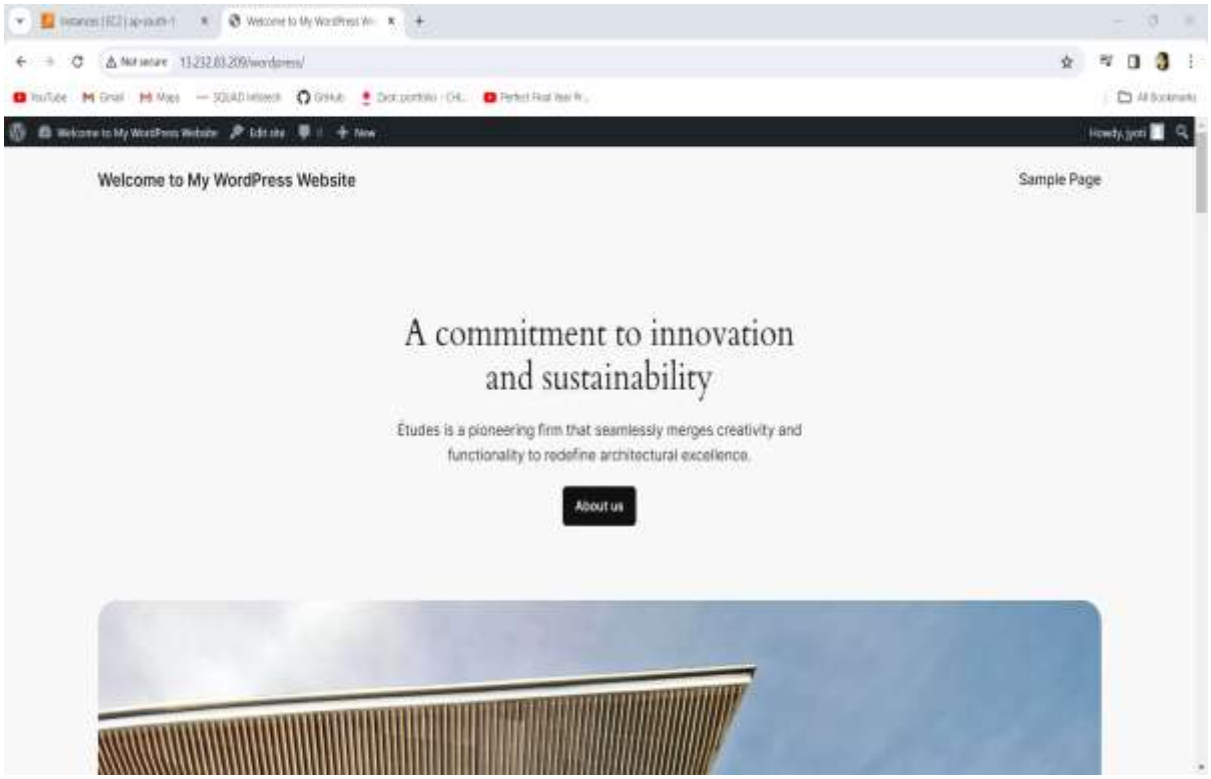


Step 27: Now install wordpress and Login with username & password.

Step 28: See your first webpage is open successfully on the wordpress.



Welcome to My WordPress Website



Conclusion: successfully performed my 1st task for monolith. In that 1 EC2 instance created, deployed WordPress and MYSQL on the same instance and created welcome page as “Welcome to My WordPress Website” as homepage.