**Containerized Web Application Deployment with Docker and Docker Compose**

**Project Overview**

🚀 Developed a containerized web application environment using Docker and Docker Compose, featuring WordPress, MySQL, and phpMyAdmin.

🚀 The project focused on creating Docker images and orchestrating them using Docker Compose to provide a scalable, consistent, and easily deployable setup for web development and database management.

➡️ **We are going to create :-**

* Ubuntu based **EC2 Instance ( ubuntu 22.04 LTS )**.
* 3 Docker container using **docker-compose**.
* **MySQL** with pre-configured environment variables for database name, user, and password to ensure smooth integration with WordPress.
* **WordPress** incorporating necessary PHP extensions and configurations tailored to the project’s needs.
* **phpMyAdmin** allowing for easy web-based management of the MySQL database.

**Getting Started**

✅ Steps: -

* Launch EC2 Instance with Ubuntu 22.04 LTS OS. Add the script from enable\_ssh\_ubuntu22.0 to enable SSH password authentication under User data block while launching the instance.
* Connect to the instance with SSH.
* Now use below commands to Install docker and docker-compose using script ubuntu22.0\_docker\_and\_docker\_compose\_install.sh

$sudo chmod +x ubuntu22.0\_docker\_and\_docker\_compose\_install.sh

$sh ubuntu22.0\_docker\_and\_docker\_compose\_install.sh

* Use the below command to check the docker-compose version.

$ docker-compose –version

* Now create docker compose file to create 3 container using docker-compose.yaml

$docker-compose up -d

You will get following Output



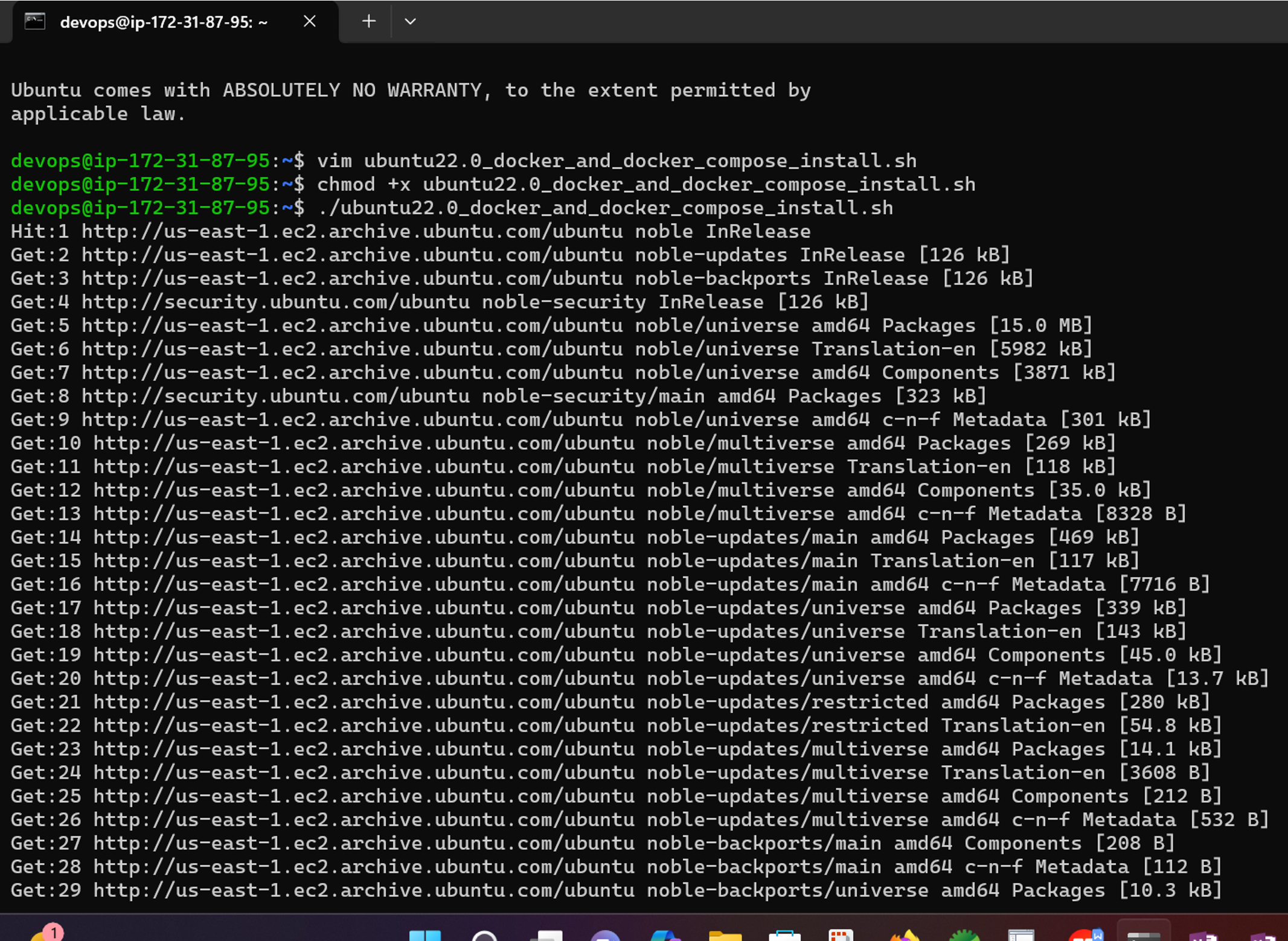
* When you hit this command, it actually searches for docker-compose.yaml file in the running directory and it executes all the instructions given in YAML file one by one, it will create container application, network, and volume, based on given instruction in YAML file.
* Verify containers are created or not using following command

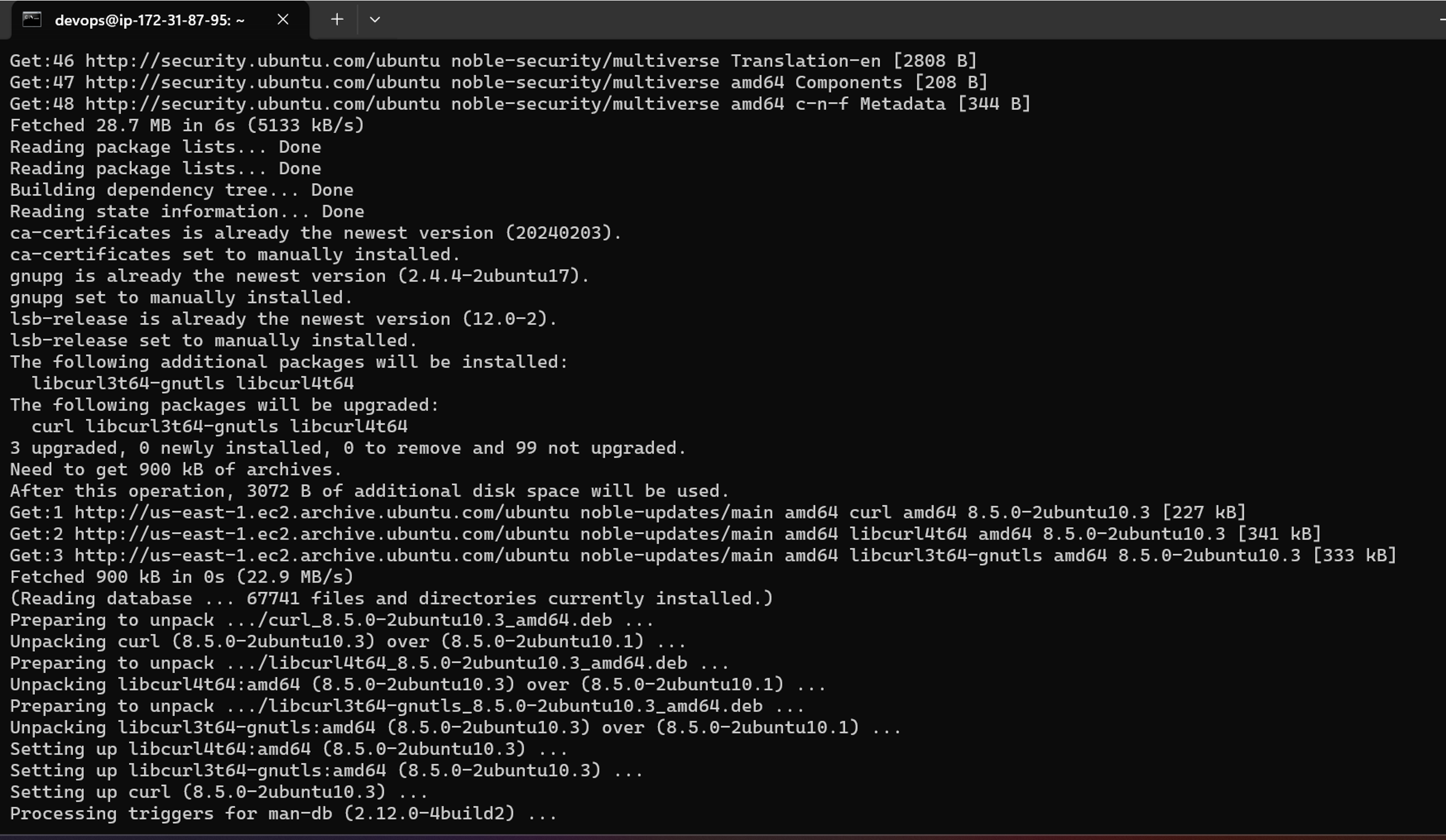
$docker container ls

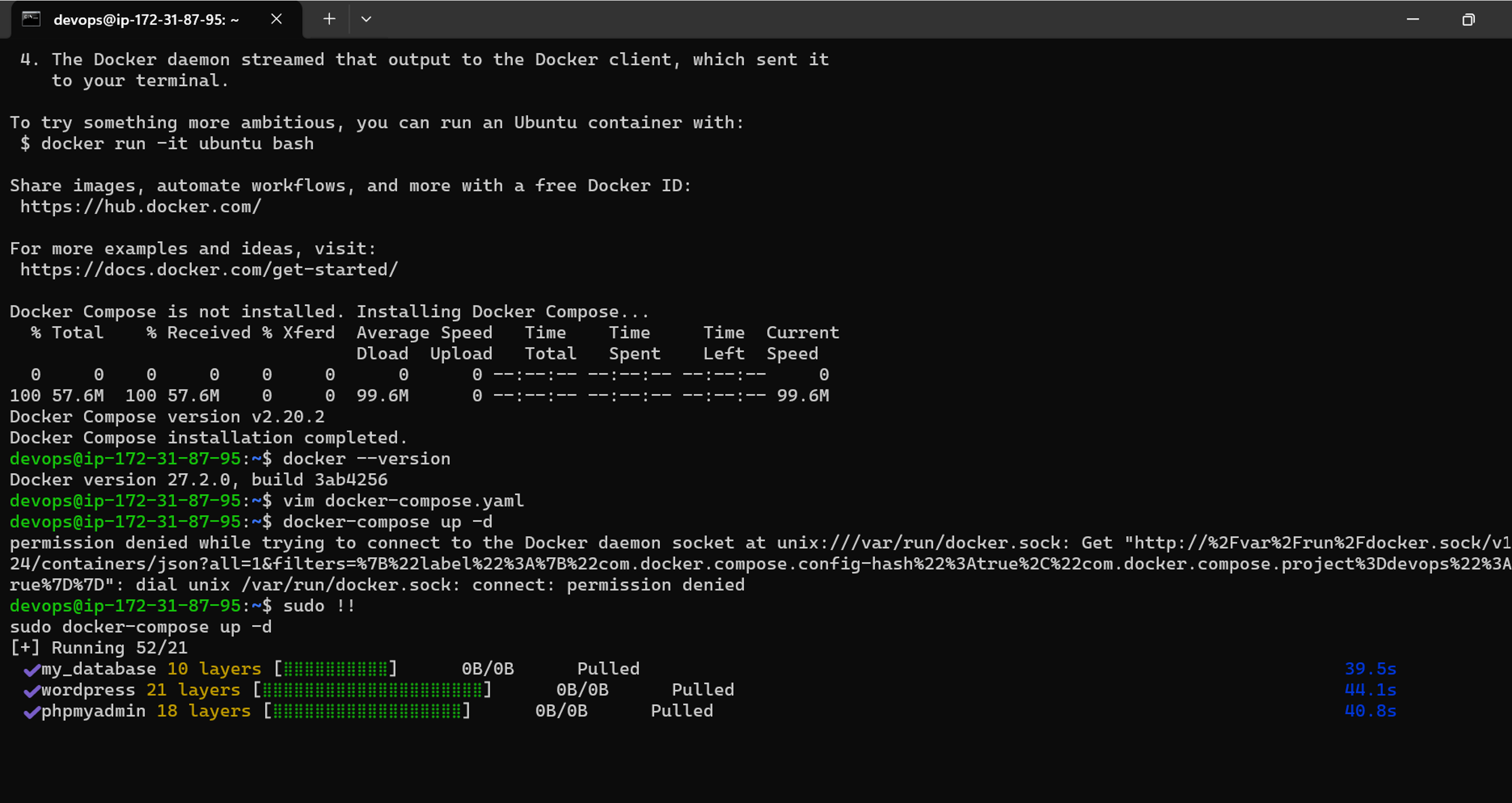
* Now try to access WordPress URL 🡪 Public IP of EC2 :8000

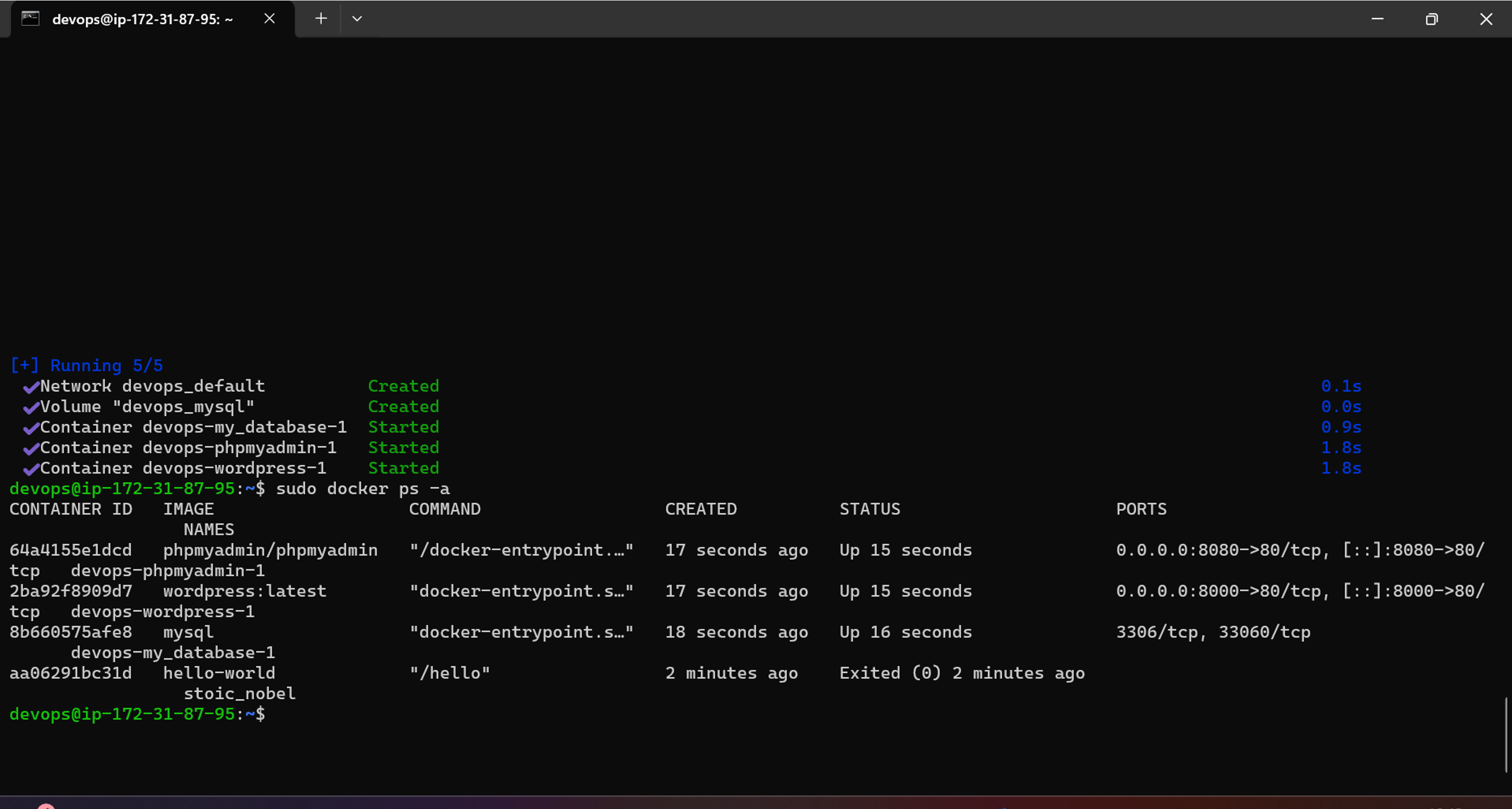
Setup WordPress username password and you will able to access your WordPress application to design your website.









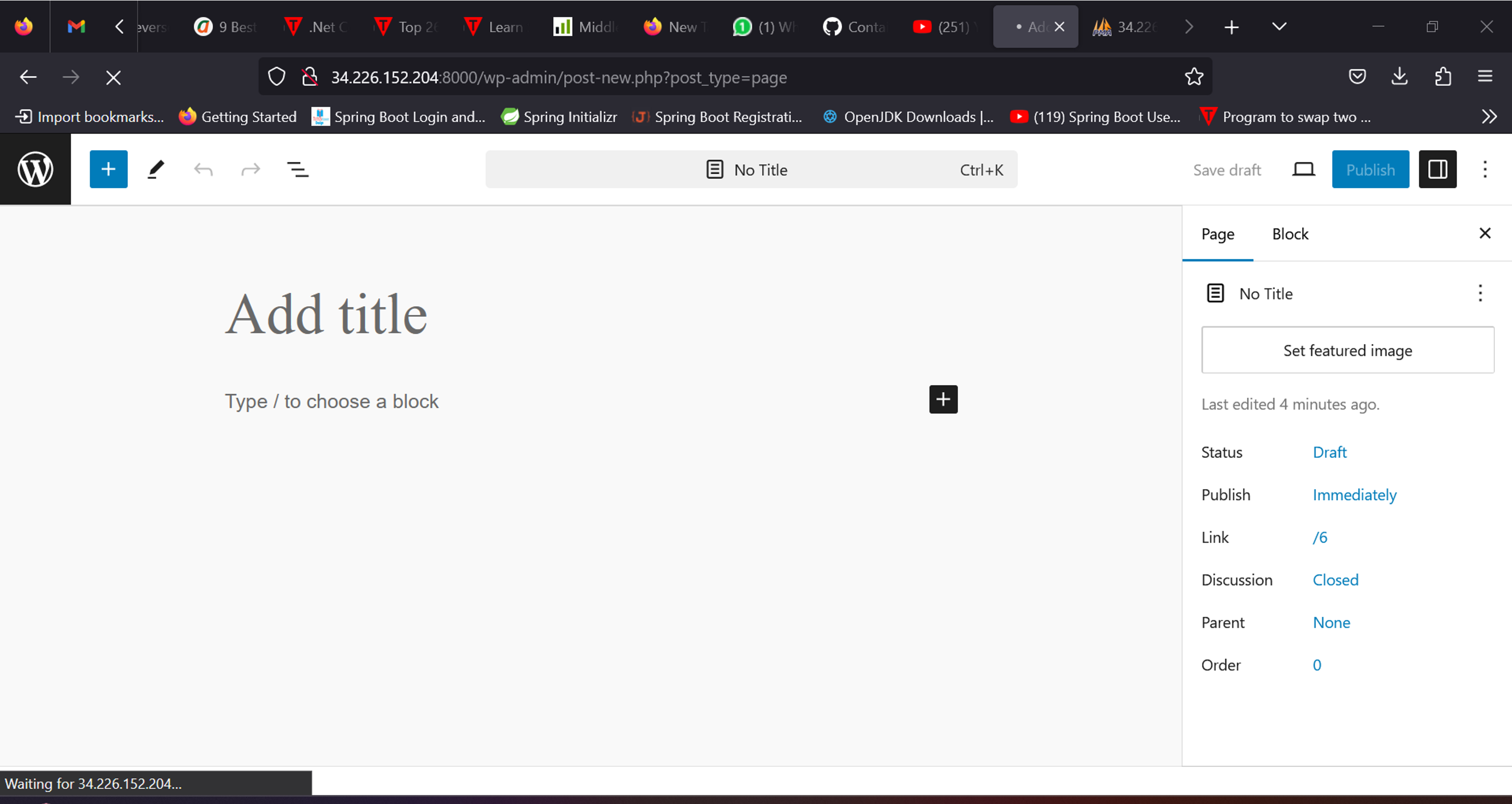


Wordpress :

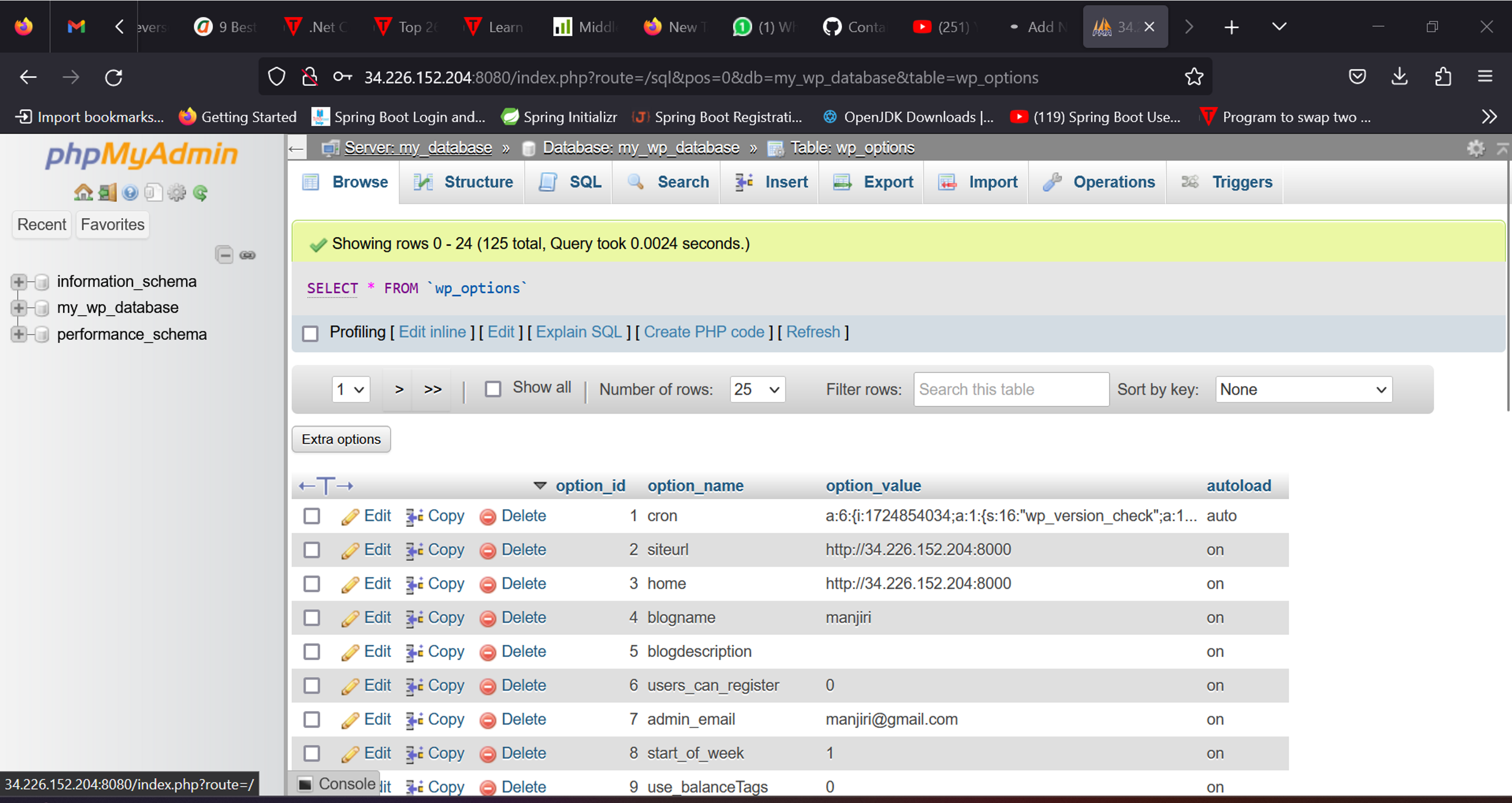
WordPress URL 🡪 Public IP of EC2 :8000

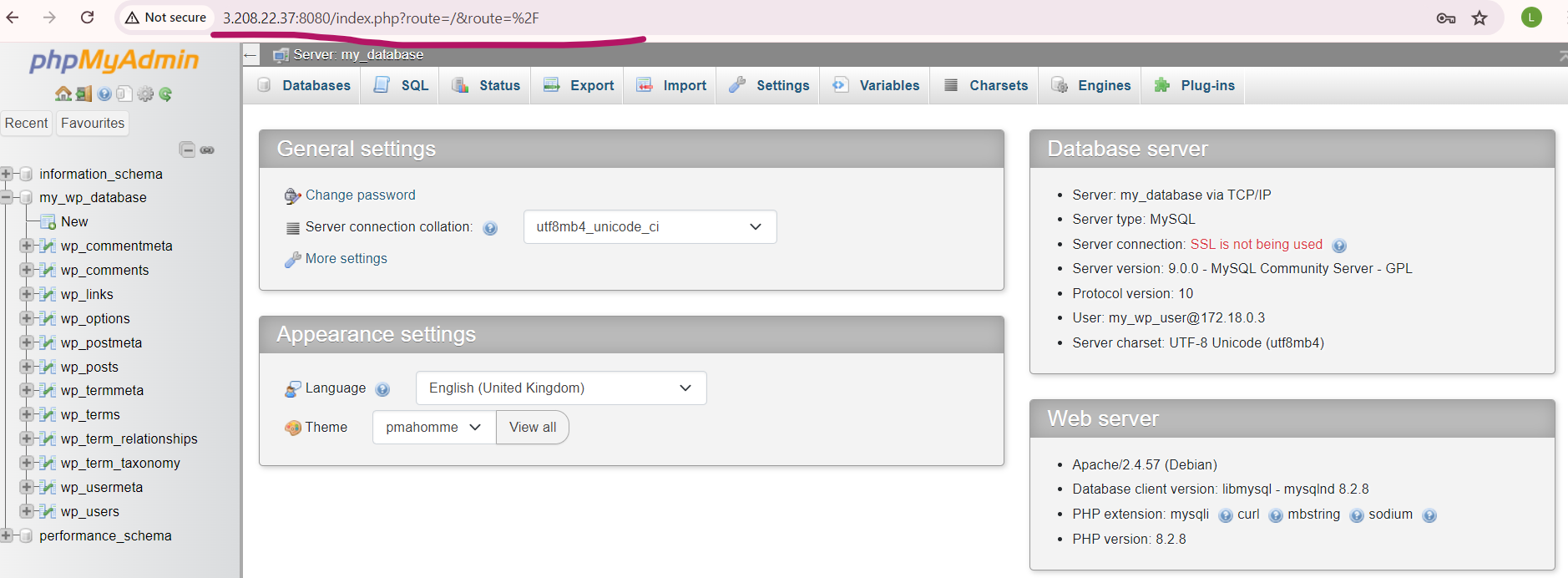
WordPress username : manjiri

WordPress password : manjiri123



PhpMyAdmin:





* Access phpMyAdmin 🡪 public IP of EC2 :8080

MYSQL\_USER: my\_wp\_user

MYSQL\_PASSWORD: my\_wp\_user\_password

