**WRITE DOCKERFILE INSTALL AND RUN PHPMYADMIN AND WITH THE MY ADMIN NEED TO ACCESS DB**

**Step:1:-**

*Create a Dockerfile:*

mkdir one

cd one

sudo vi Dockerfile

*Inside the Dockerfile:*

FROM phpmyadmin

ENV PMA\_HOST=db

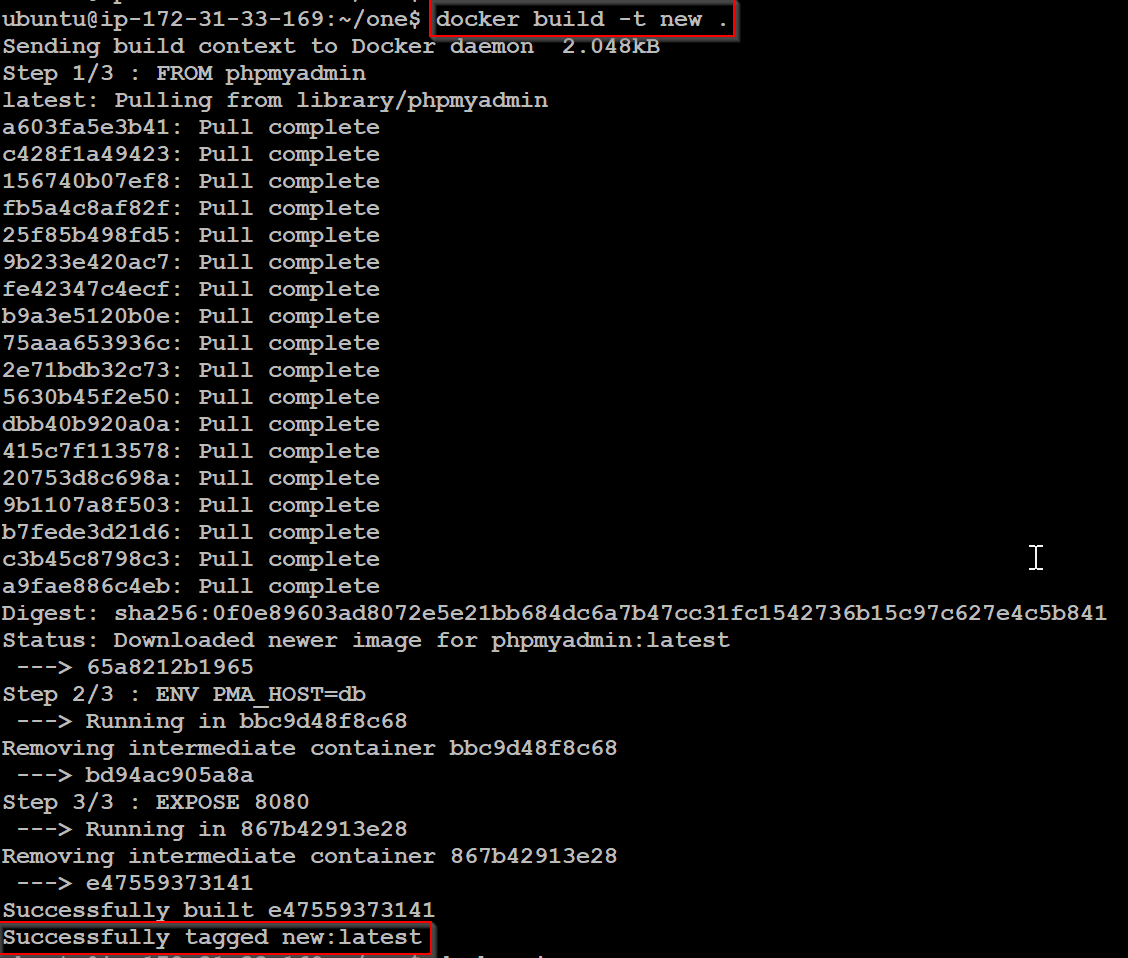
EXPOSE 8080

🡺 save the file.

**Step:2:-**

*Building the Dockerfile by using Docker Build command:*

docker build -t new .

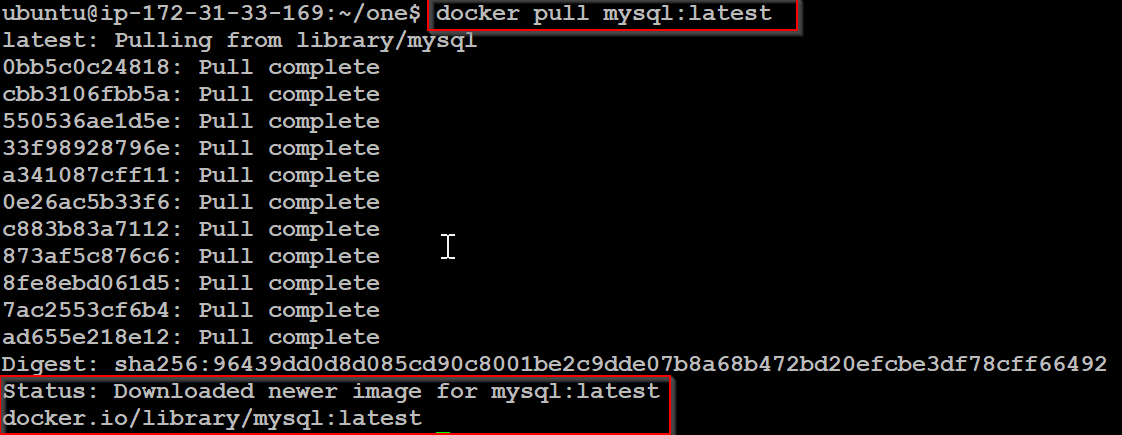


Successfully build the Dockerfile.

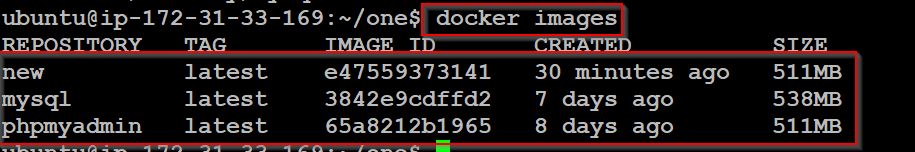
**Step:3:-**

*Pull the* ***mySQL*** *image from Docker Hub:*

docker pull mysql:latest



*Now we can see the build images like this:*



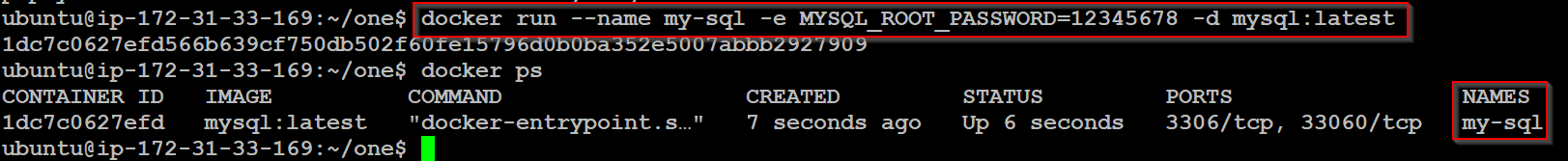
**Step:4:-**

*Creating the containers by using the build images that we created before:*

*First one Creating the Container of MySQL:*

**docker run --name <container\_name> -e <environment\_variable> -d <image\_name>**

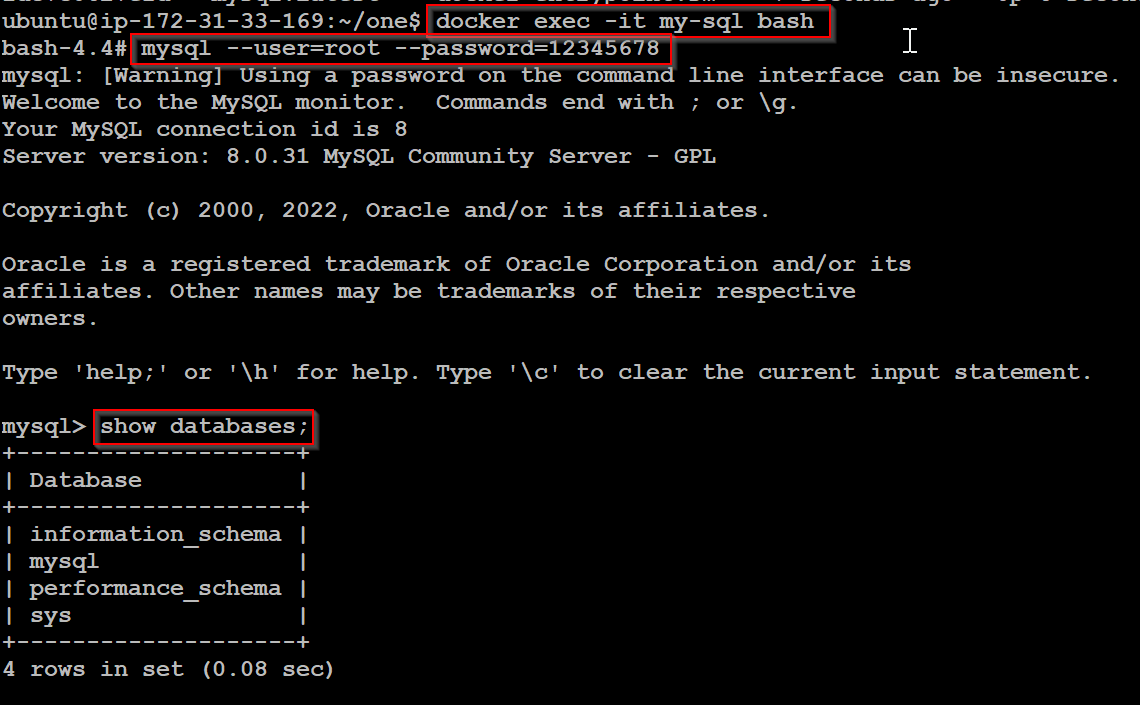
docker run --name my-sql -e MYSQL\_ROOT\_PASSWORD=12345678 -d mysql:latest



Here MySQL Container is up and running.

*Go inside the Container of my-sql and check with the username and password is working or not:*

docker exec -it my-sql bash

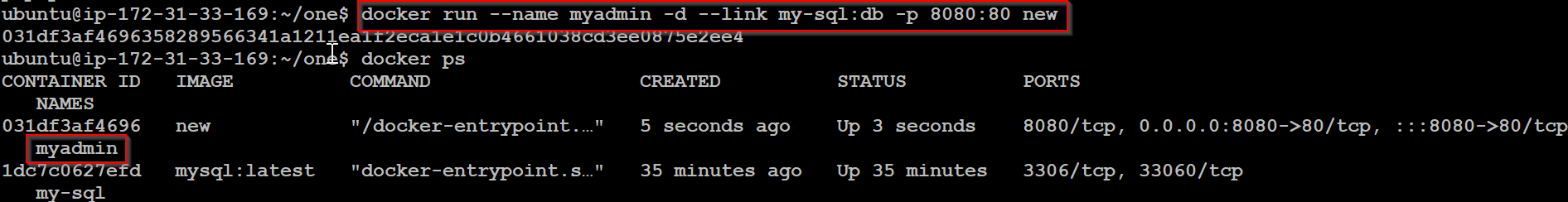


And then exit of this container and after create another container.

*Second one is Create another container of* ***phpmyadmin*** *and this container to be linked to* ***MySQL container*** *by the new name image:*

**docker run –name <container\_name> -d <link with another container>:db -p 8080:80 <image\_name>**

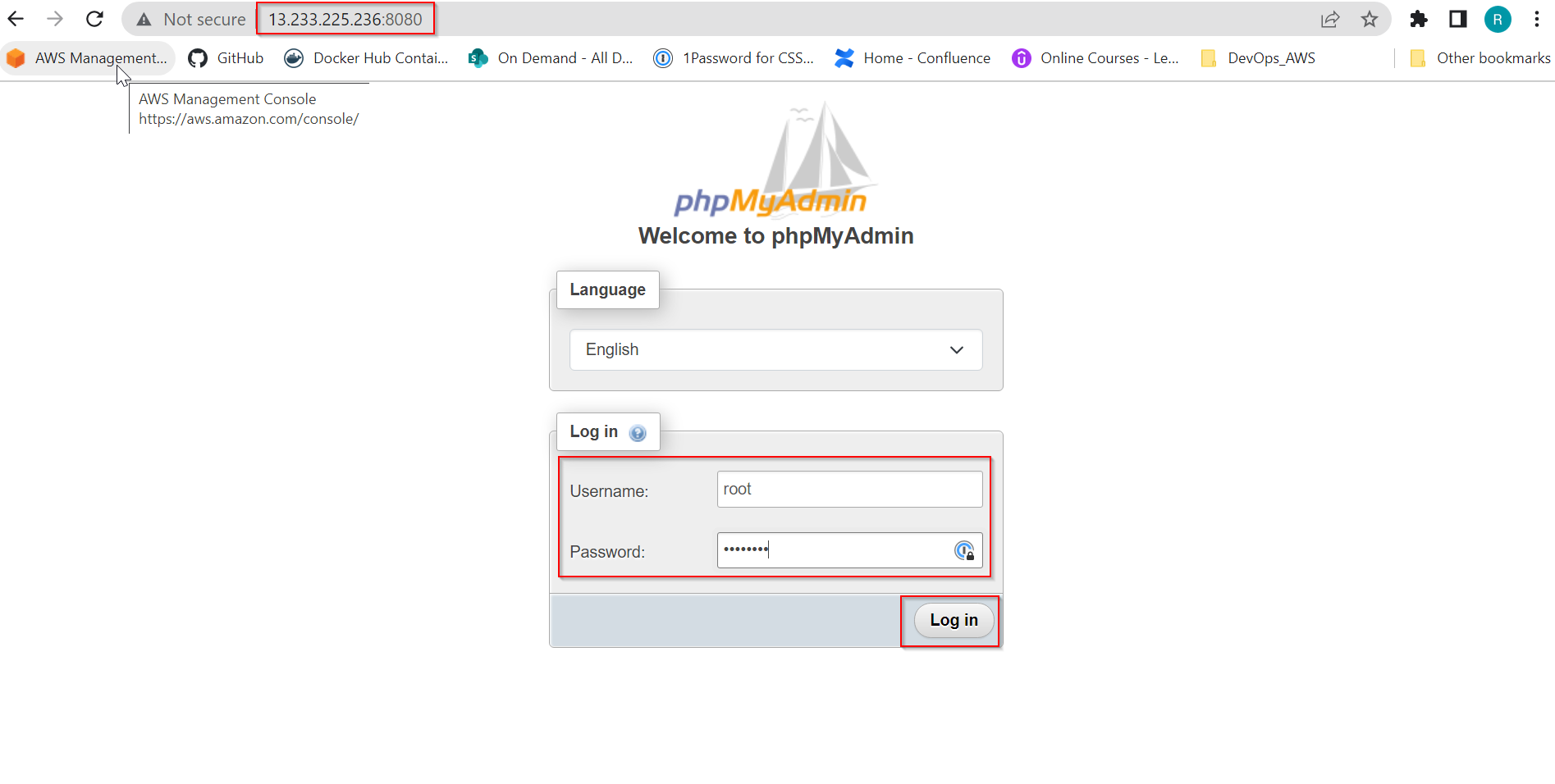
docker run --name myadmin -d --link my-sql:db -p 8080:80 new



**Step:5:-**

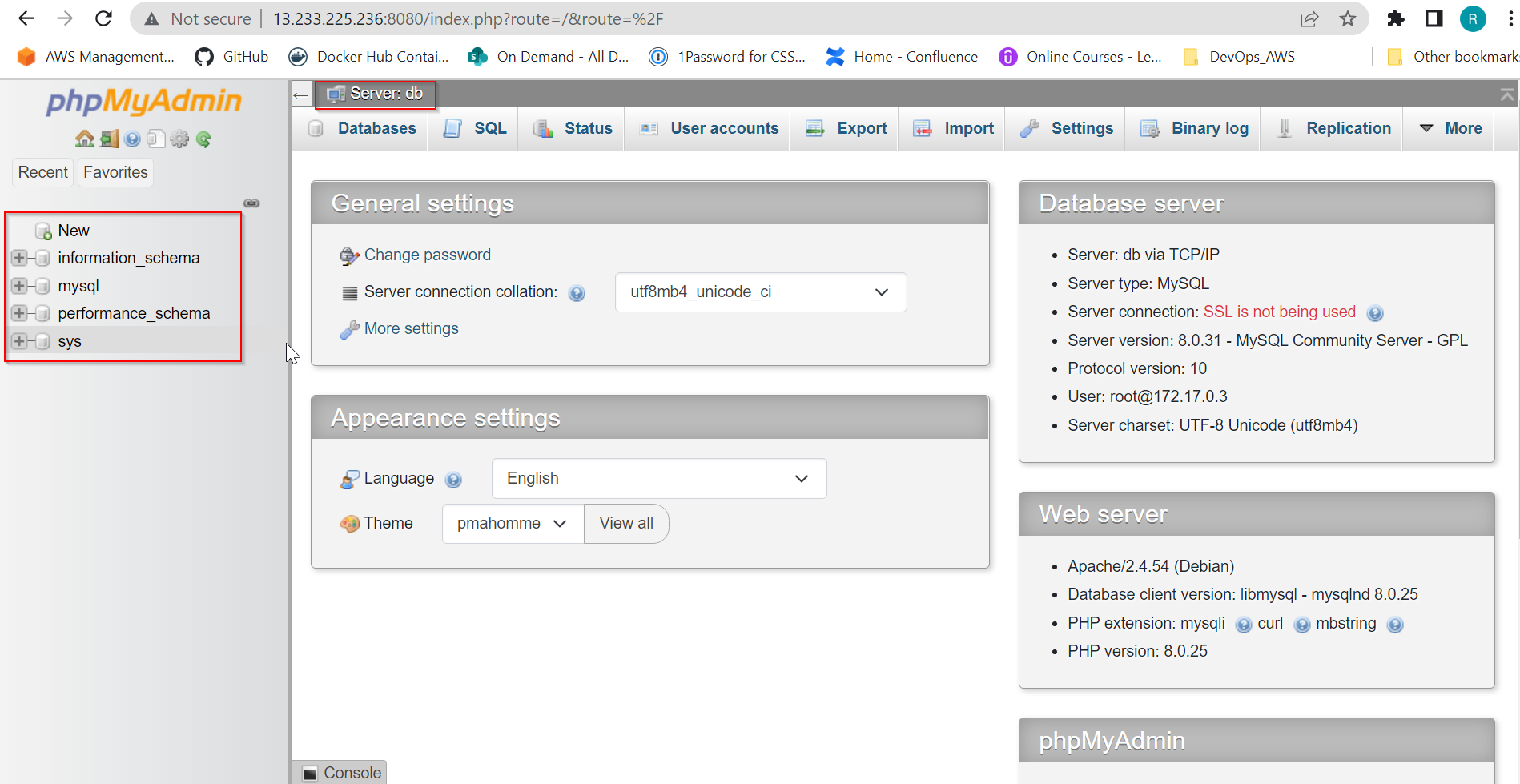
*After link the containers go to the browser by opening with IP and with port and login with the username and password that we gave in my-sql container:*

**Note: For that first we need to give the port 8080 in AWS Security-Group InboundRules.**



In here we opened with the IP with Port of 8080 and given the login credentials to open.

*After login the page will look like this:*



In the selected area we have our databases to access.