**Docker Compose**

In docker if we want to create an infrastructure which we could manage by 2 ways

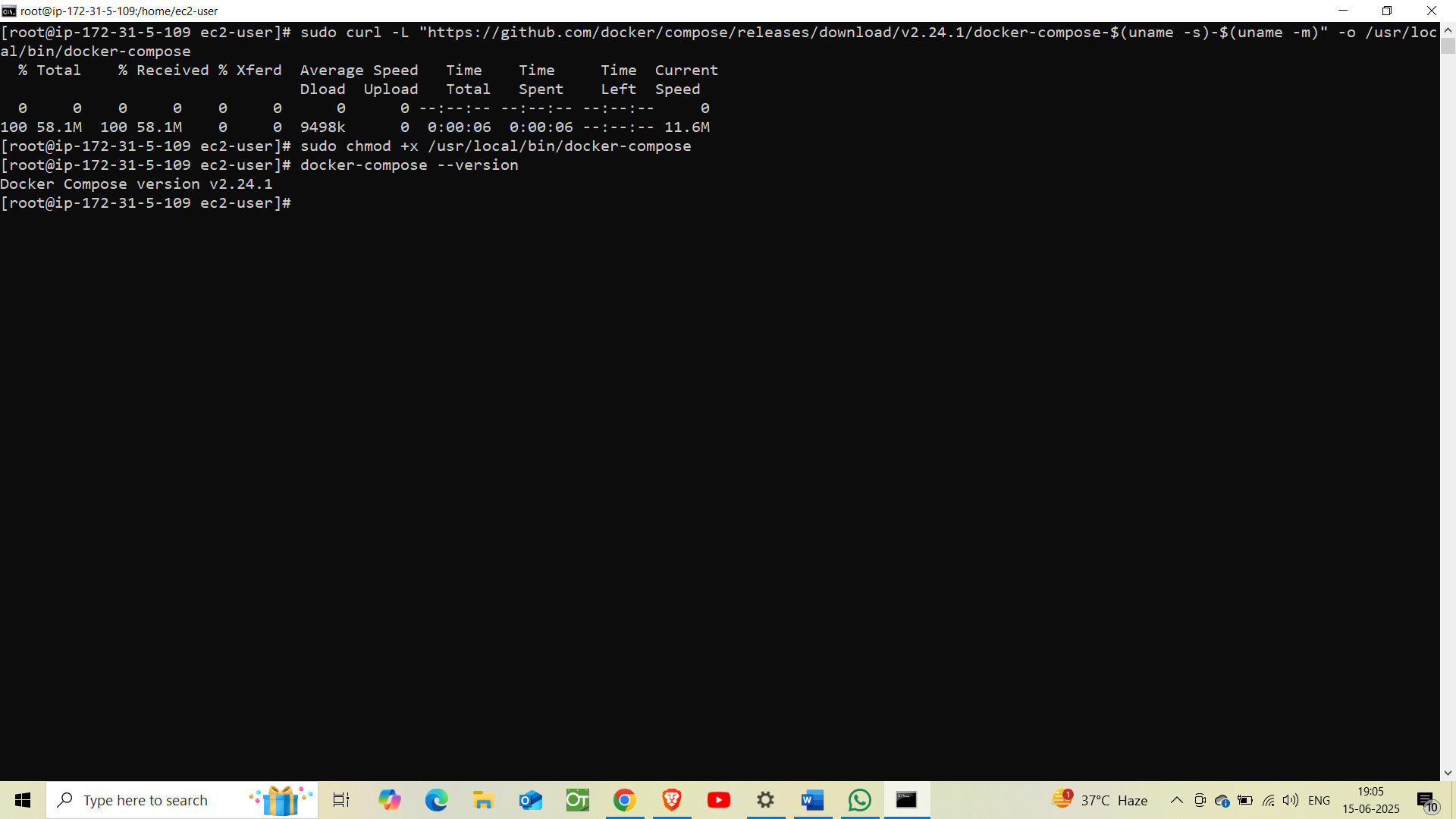
* Manually (It means manually pulling images and running and all.)
* Coding (It means whatever doing manually the same would be done by coding which is known as **Docker Compose**)
* **Docker Compose** we have to externally install to use for which we have to search for **docker compose rpm** on browser.

**(OR)**

* Directly run the command
* **sudo curl -L "https://github.com/docker/compose/releases/download/v2.24.1/docker-compose-$(uname -s)-$(uname -m)" -o /usr/local/bin/docker-composecompose-linux-x86\_64 -o $Docker/docker-compose**

This command downloads and installs the version **v2.24.1** of Docker Compose which we can change as per the latest version.

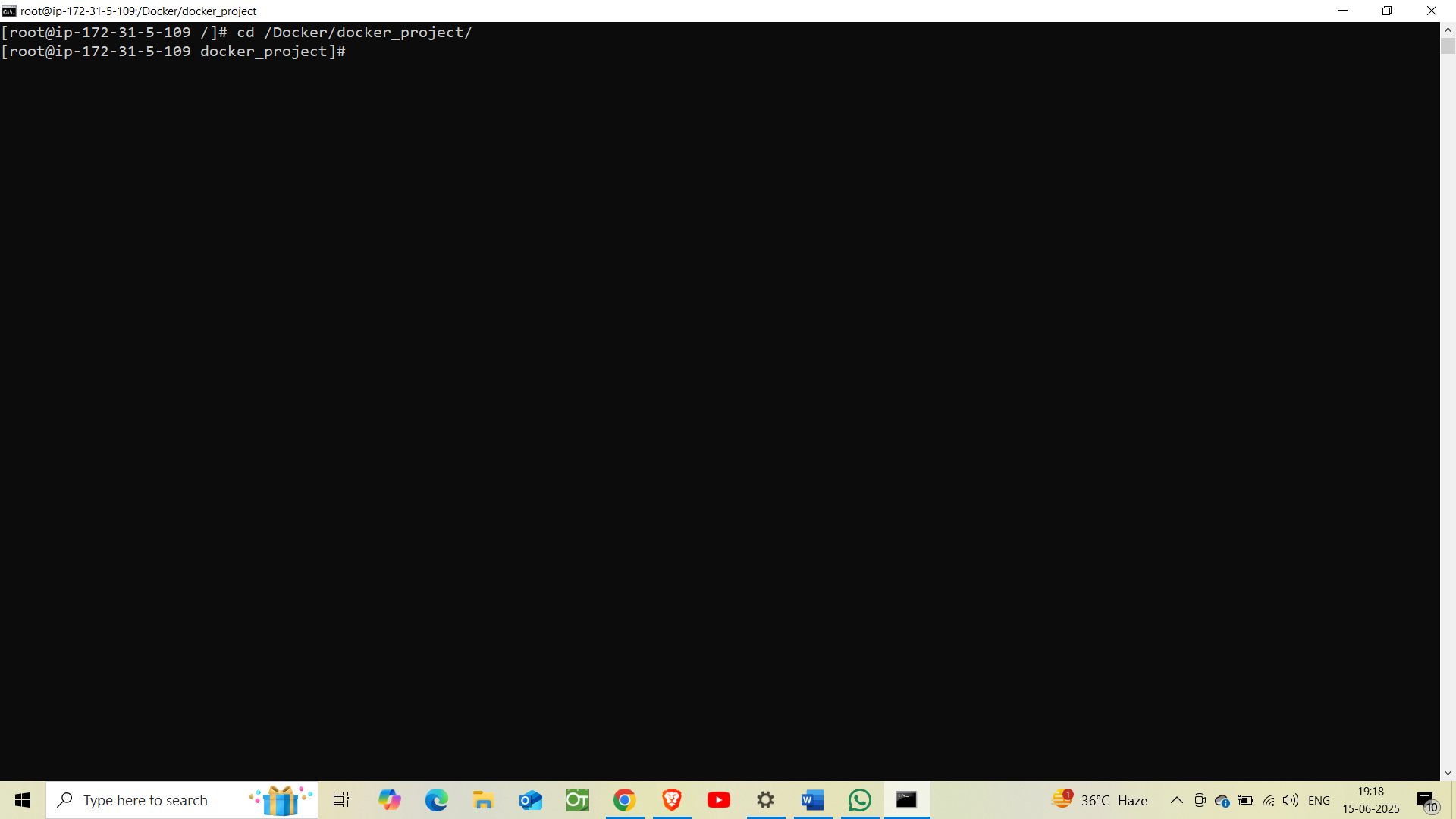
* **We can check latest version**
* [**https://github.com/docker/compose/releases**](https://github.com/docker/compose/releases)
* **After installing to make program executable**
* **sudo chmod +x /usr/local/bin/docker-compose**
* **To check docker compose version run command**
* **docker-compose --version**



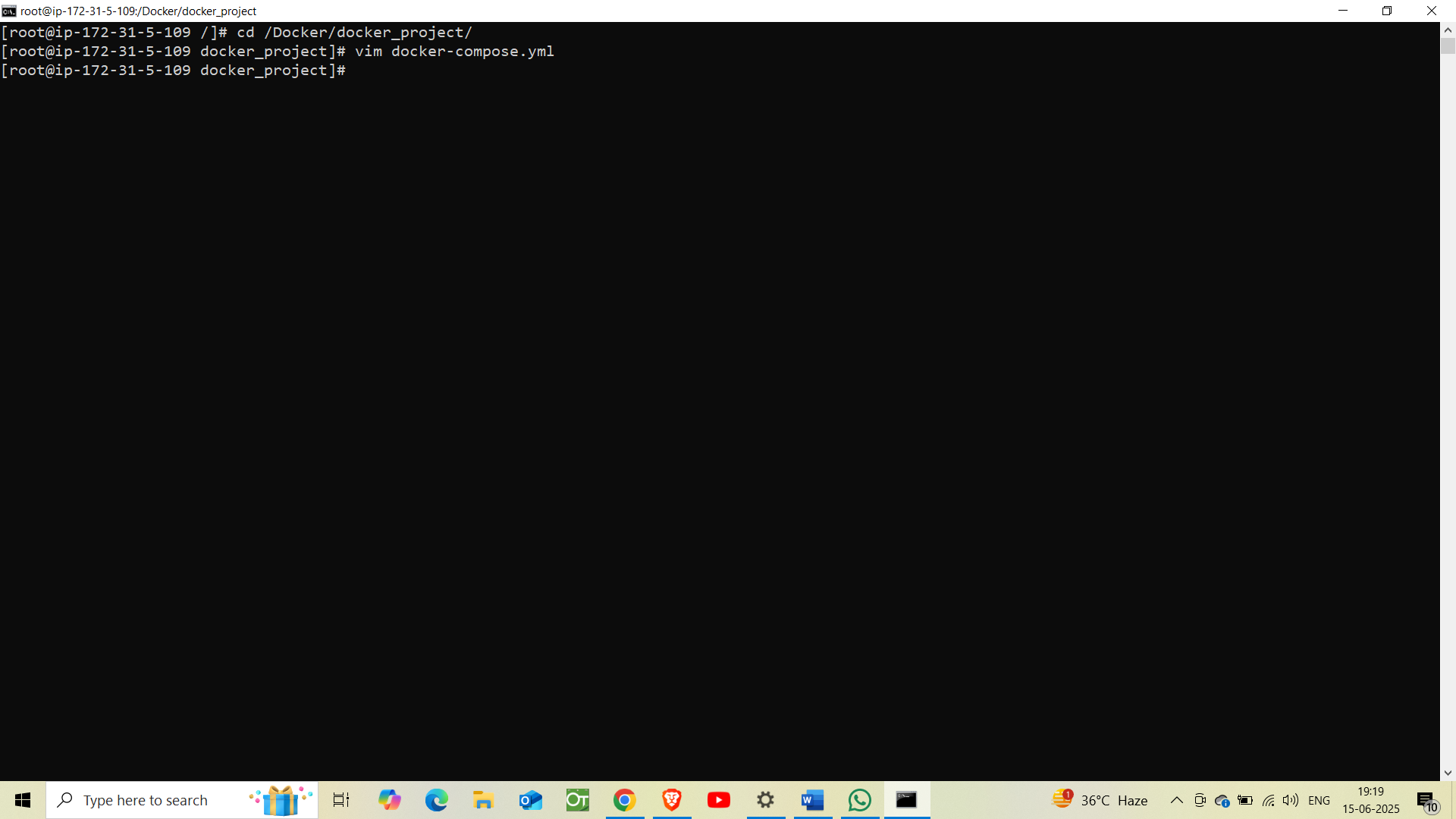
Now we can use Docker Compose with your **docker-compose.yml** files to manage multi-container apps.

**Now I am creating a separate directory to save docker-compose.yml.**

* **mkdir /Docker/docker\_project**



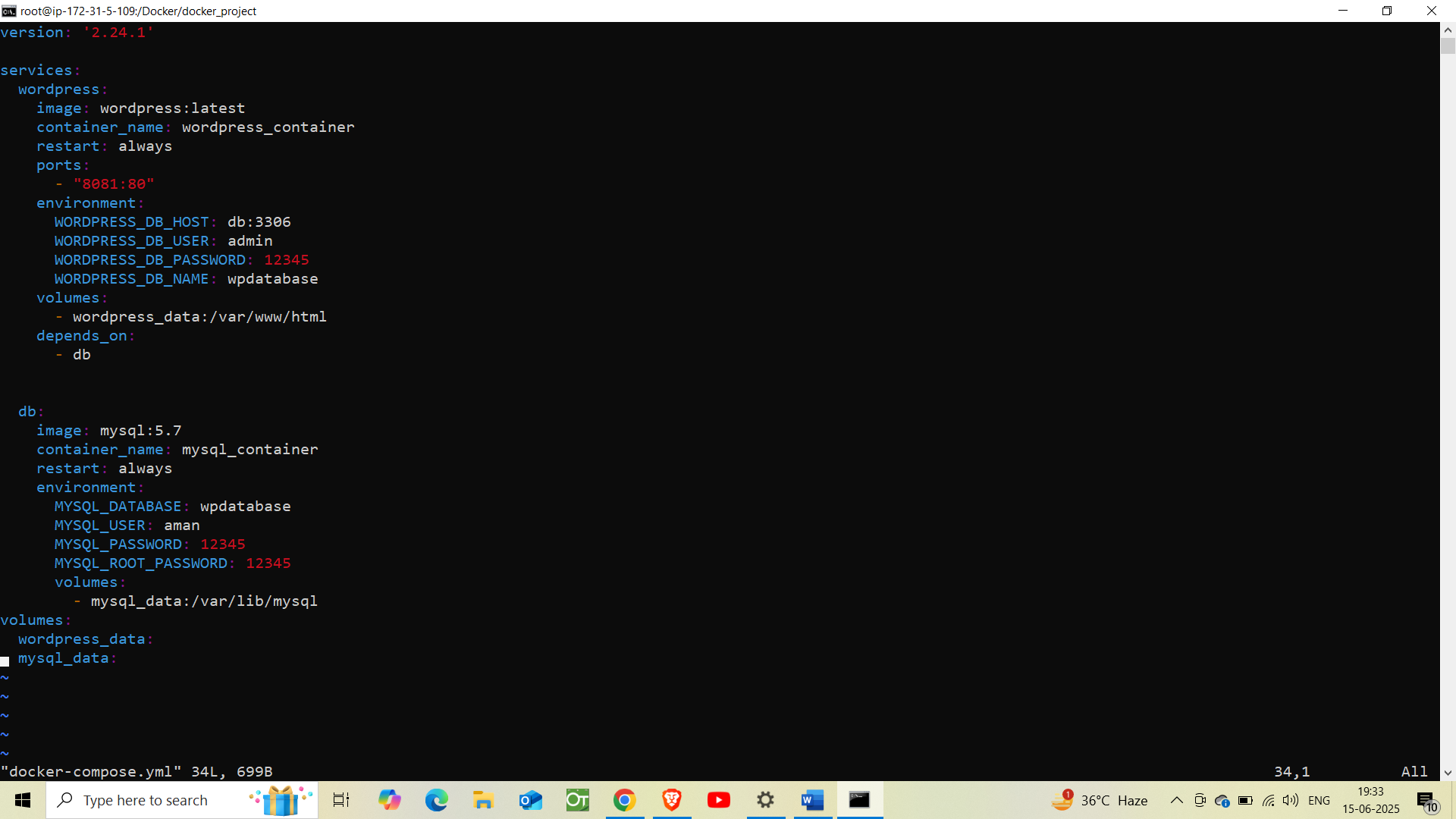
Now creating a file name **docker-compose.yml**



Now, Let’s complete a task with docker-compose.yml file to set up **WordPress and MySQL**, linked together with:

* WordPress on port 8081
* MySQL with persistent volume
* WordPress files also persisted via volume

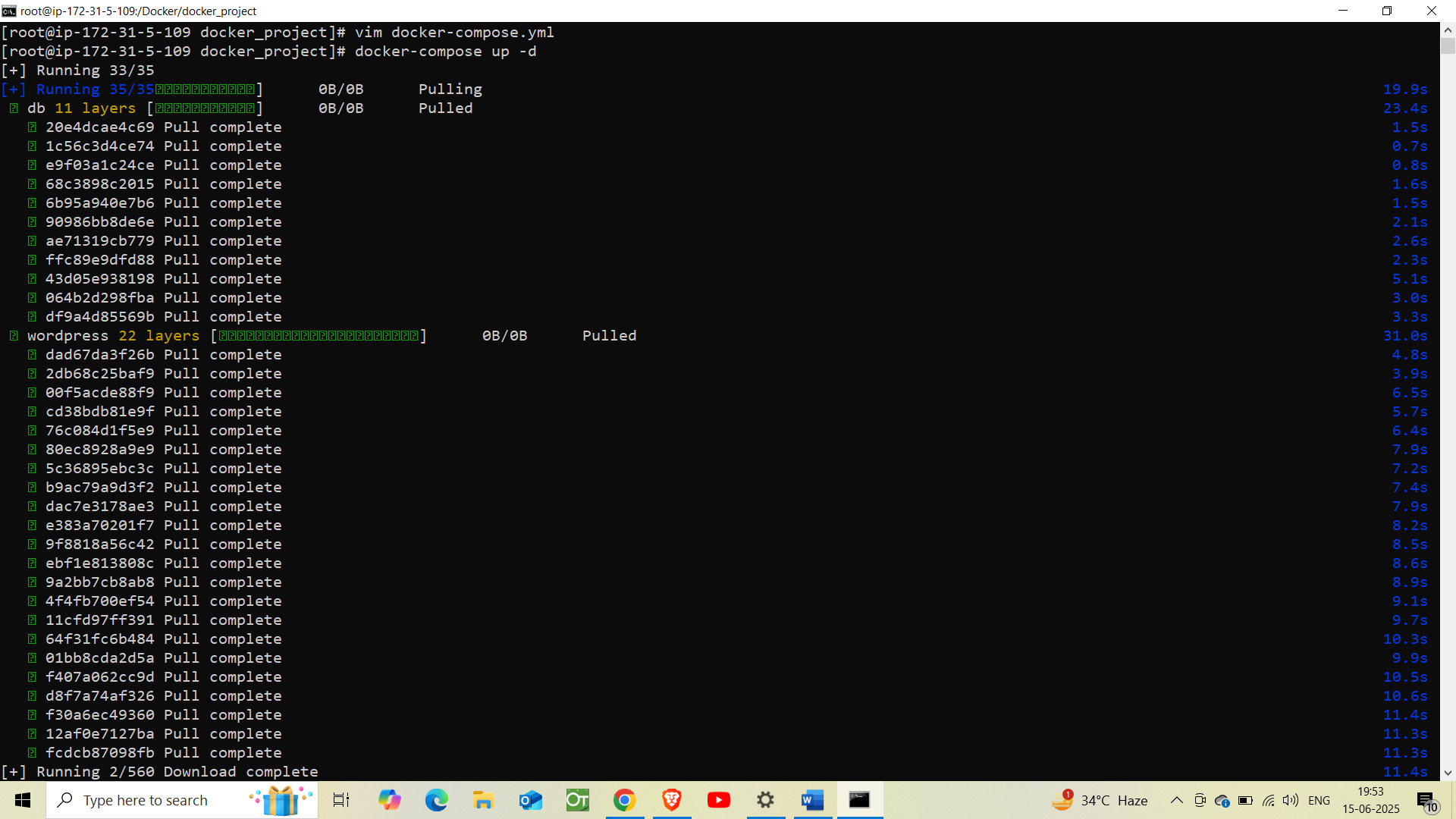
For this we have to write some codes in YAML formal which we were doing manually earlier.



Insert commands like this and then run the compose command.

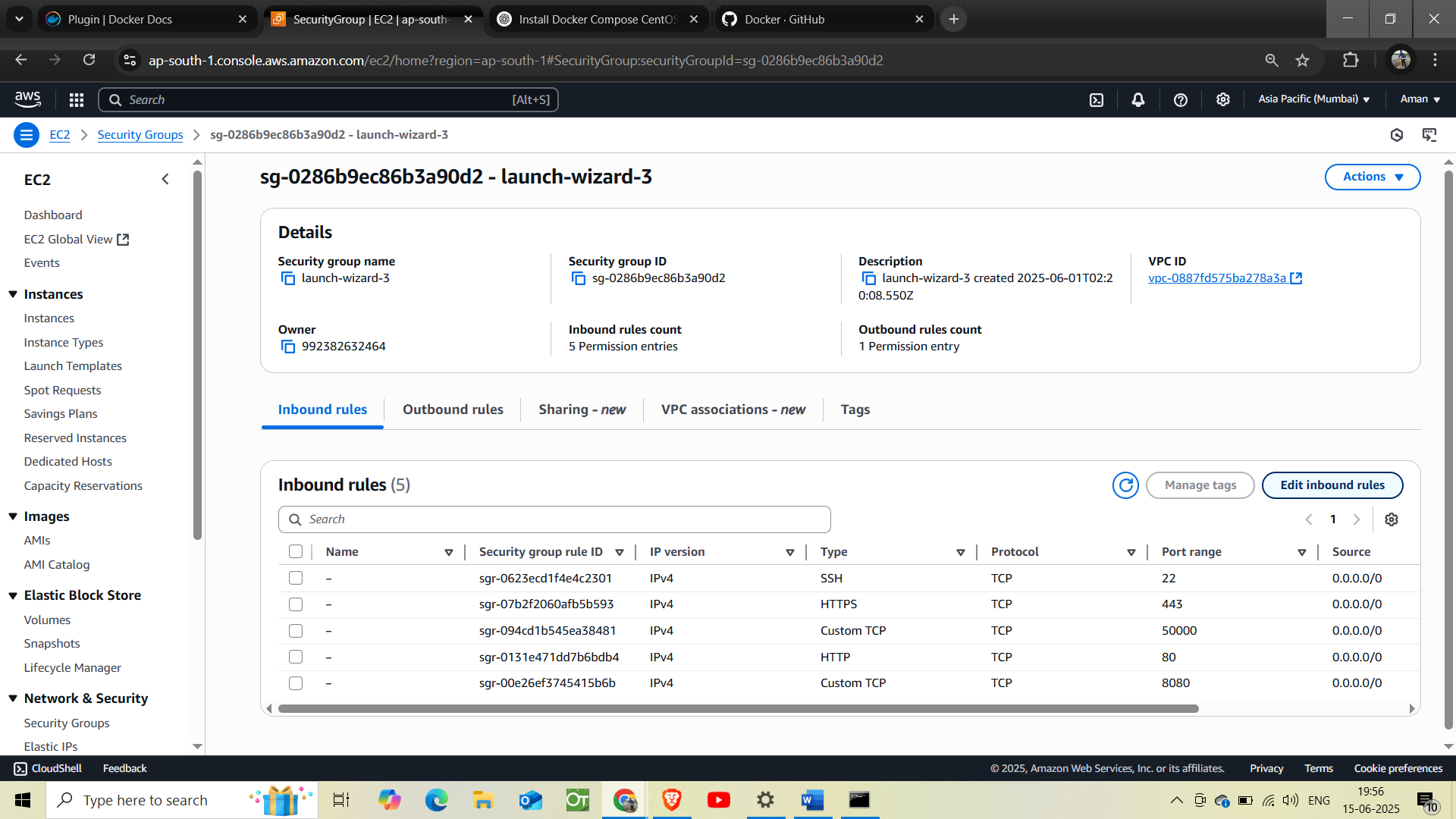
**Now,** to execute the code run the command

* **Docker-compose up -d**

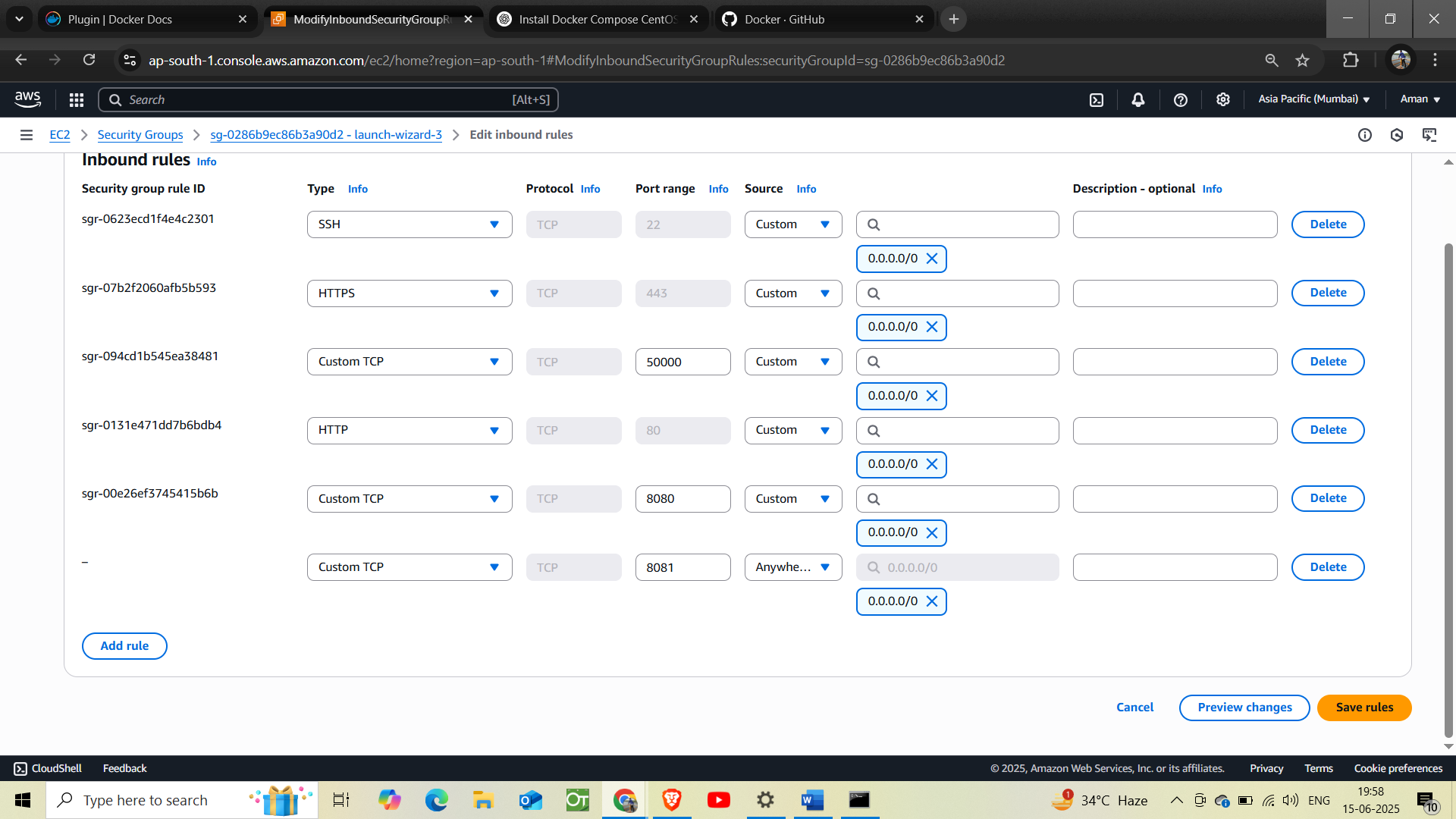


**But still we haven’t given the permission through the AWS security group to the port 8081. So, now we have to give permission**

**First go to the security group of EC2-instance**



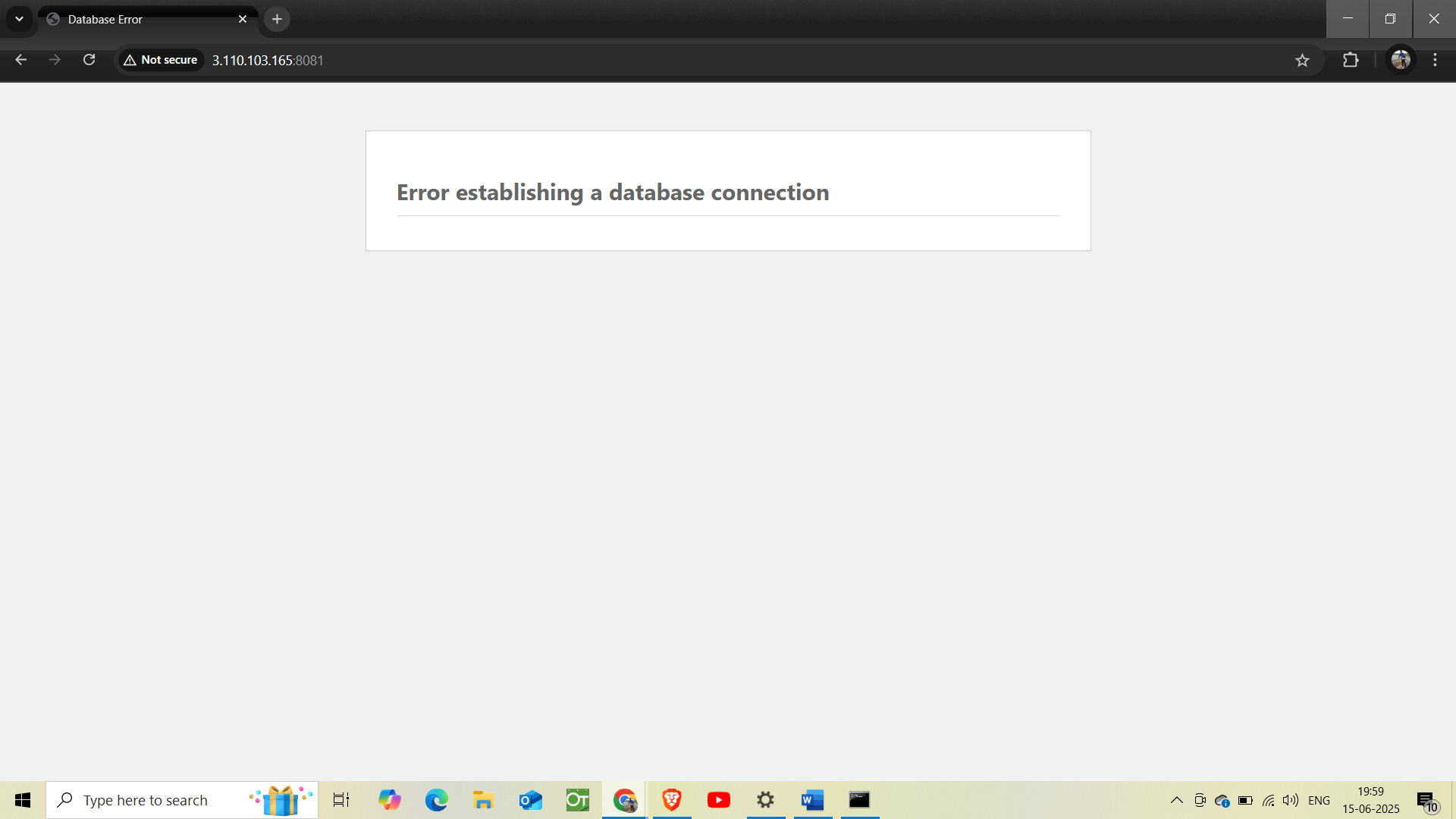
**Click on the edit inbound rules and add a new rule for port 8081.**



**like this and save.**

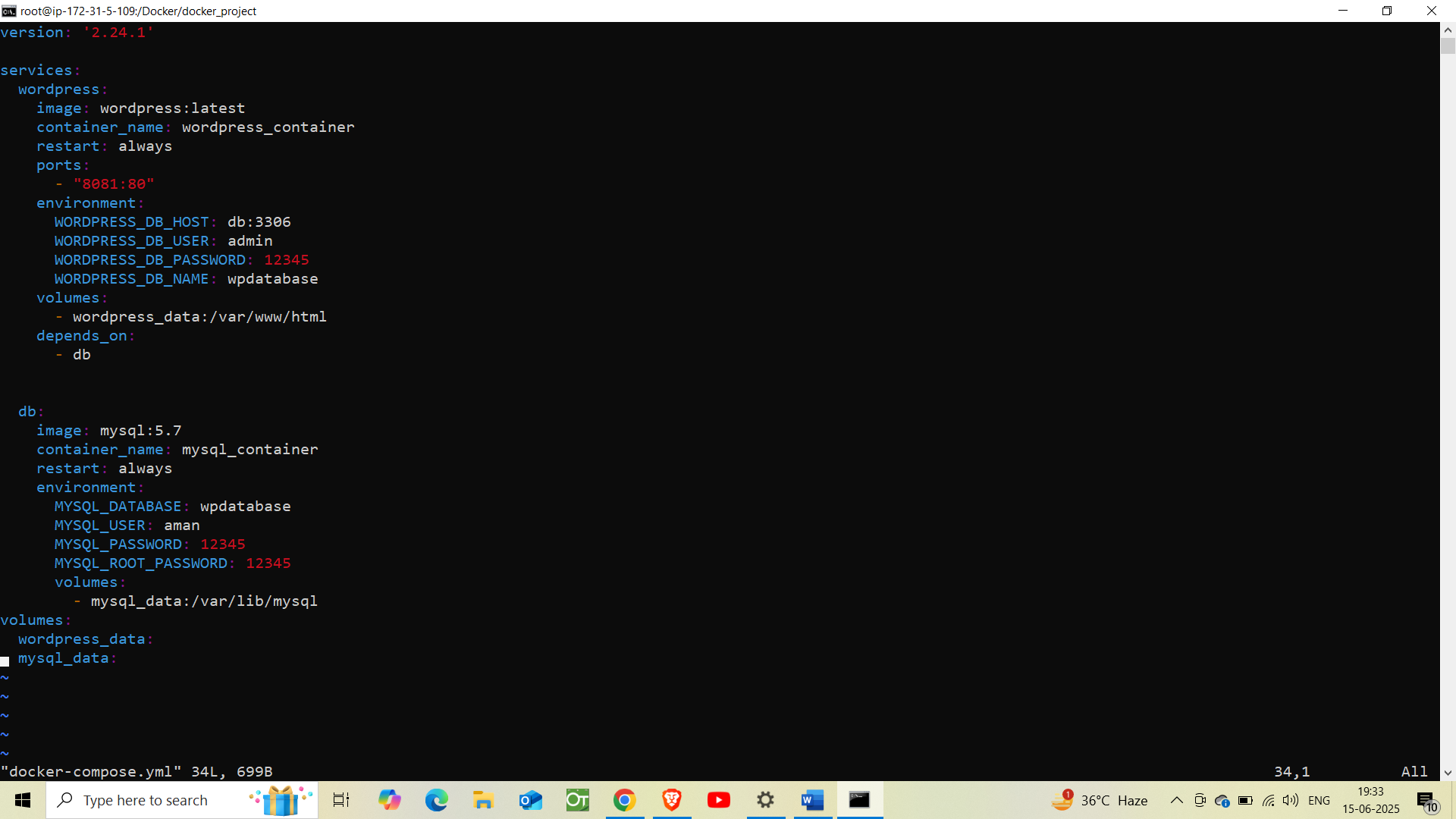
**Now, from the browser we have to search**

* [**http://public\_IP:8081**](http://public_IP:8081)

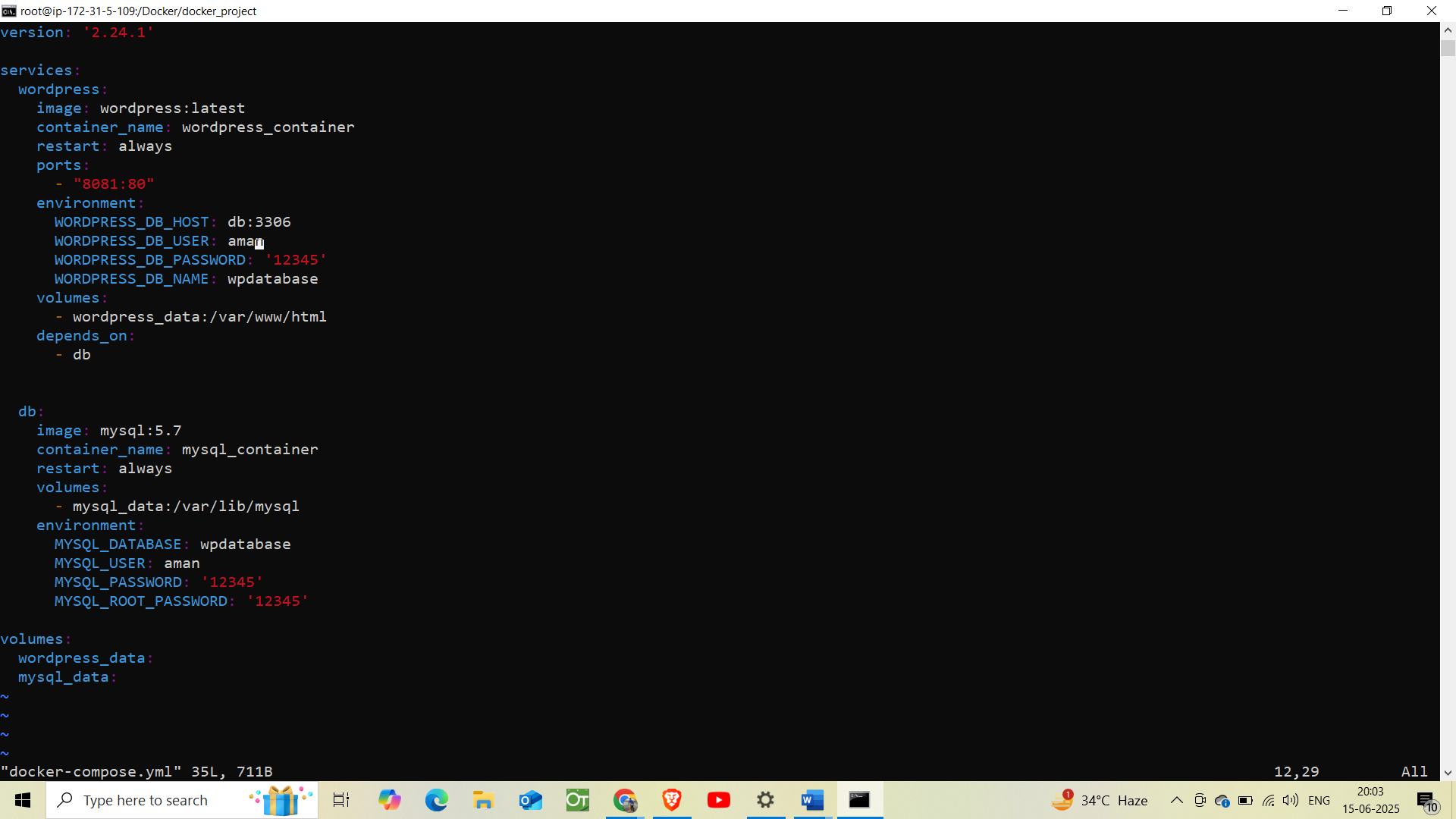


**This shows I made some mistake in the code so in code**

**Error: As, both the databases are linked so the user and password should be the same and that’s the mistake.**

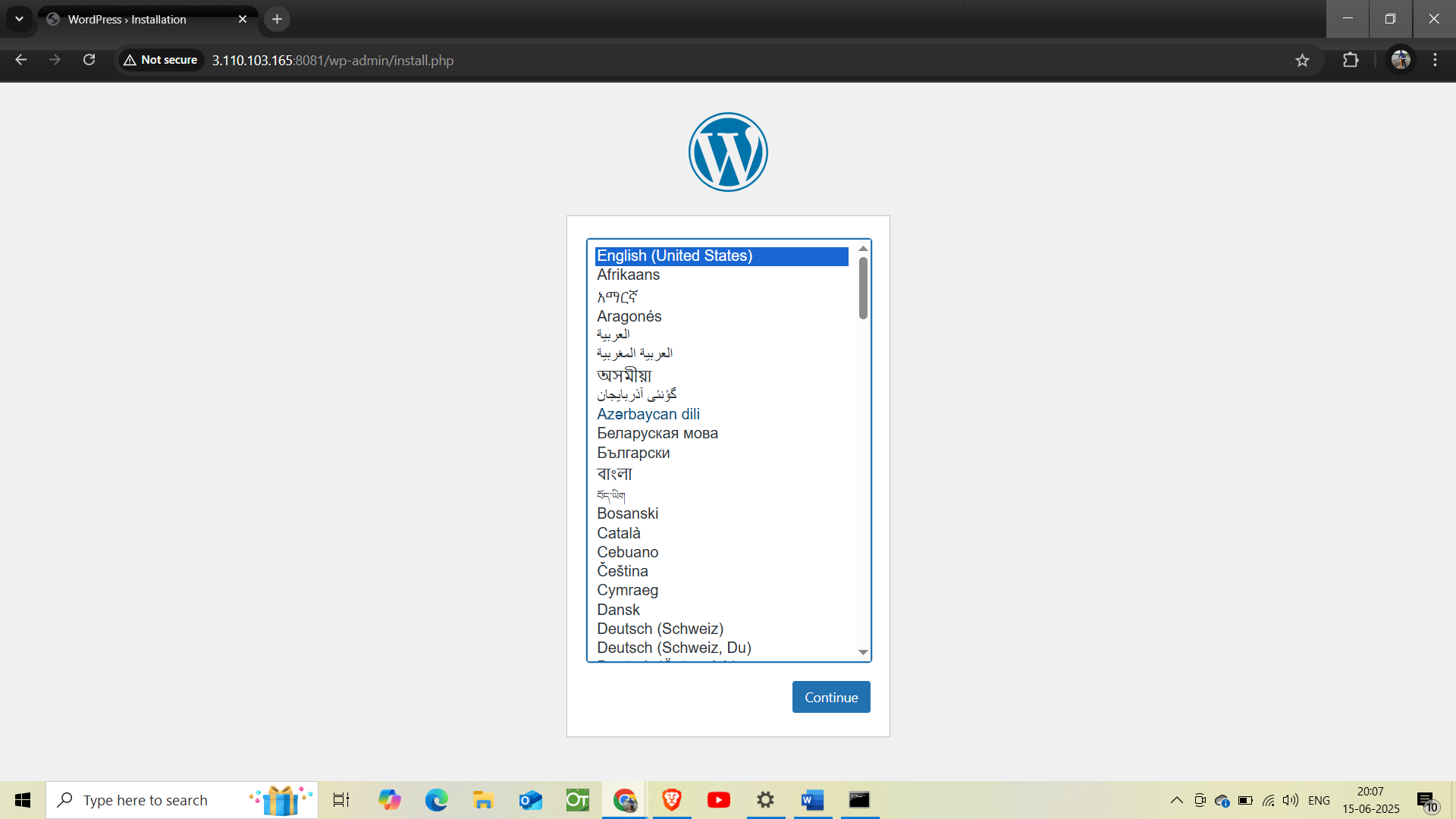


**Here I have corrected the things**



**Now, the output on browser:**

* **http://public\_IP:8081**



**Here we can do the setup and use it.**

**Username: aman**

**Password: aman**

