DEPLOYING   
 WORDPRESS  
 WEB APPLICATION USING   
 DOCKER  
 IN  
 AMAZON WEB SERVICES

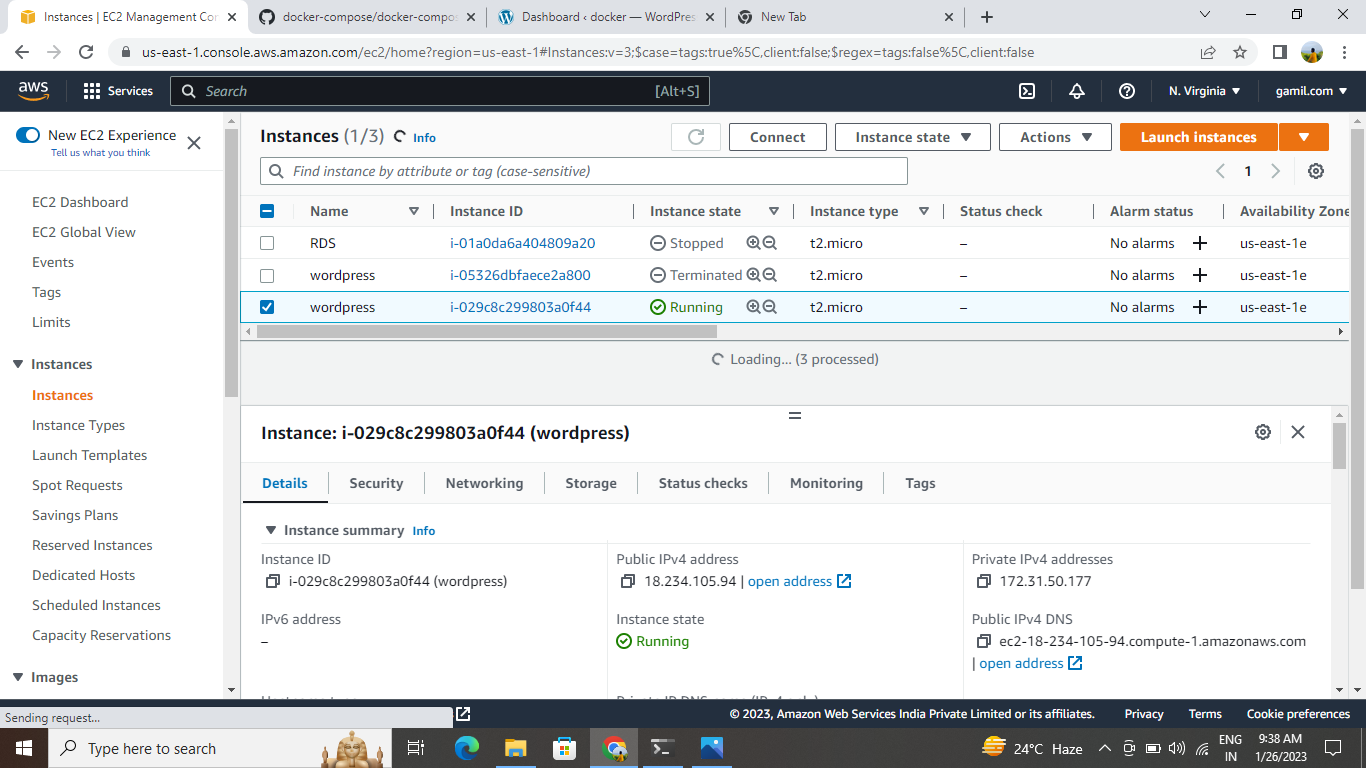
**CREATING AND LAUNCHING AN AMAZON LINUX EC2 INSTANCE**

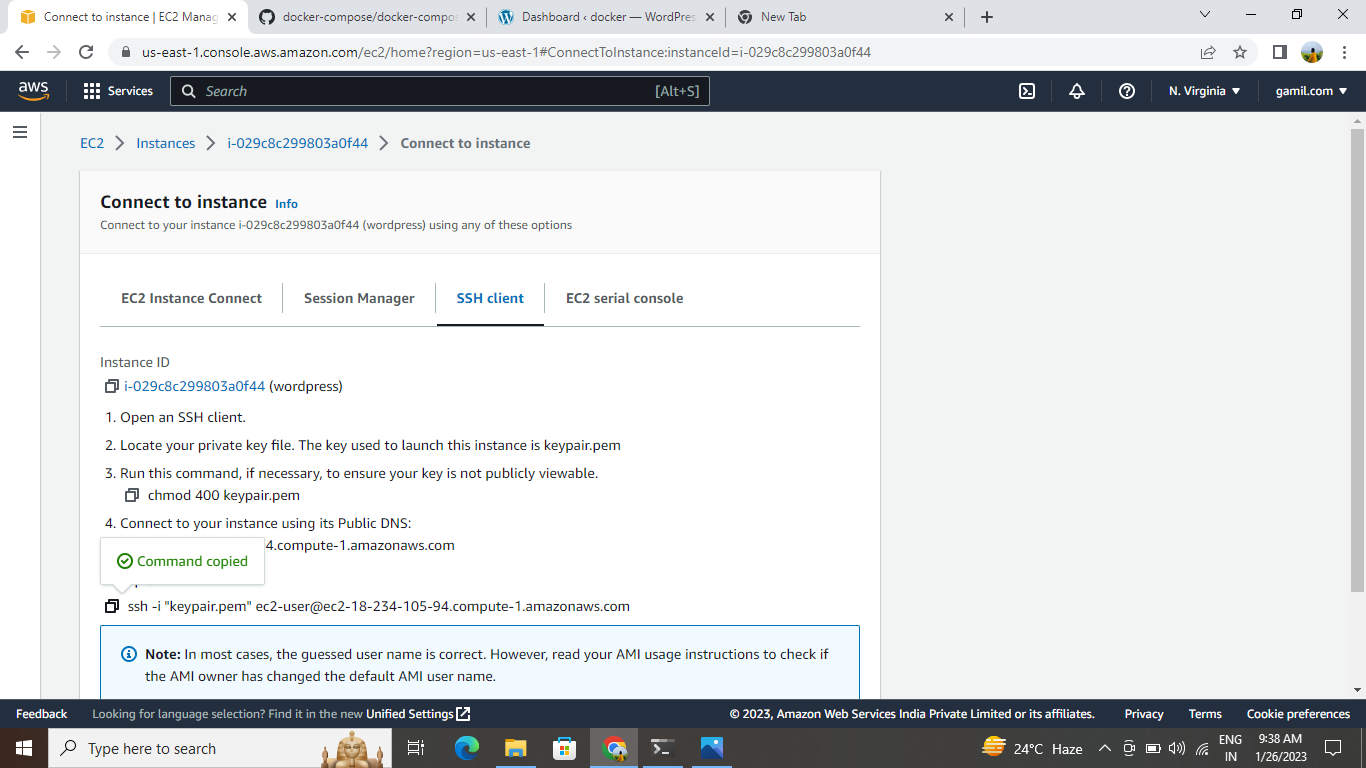
Create an EC2 instance with a Linux machine.

The security group for

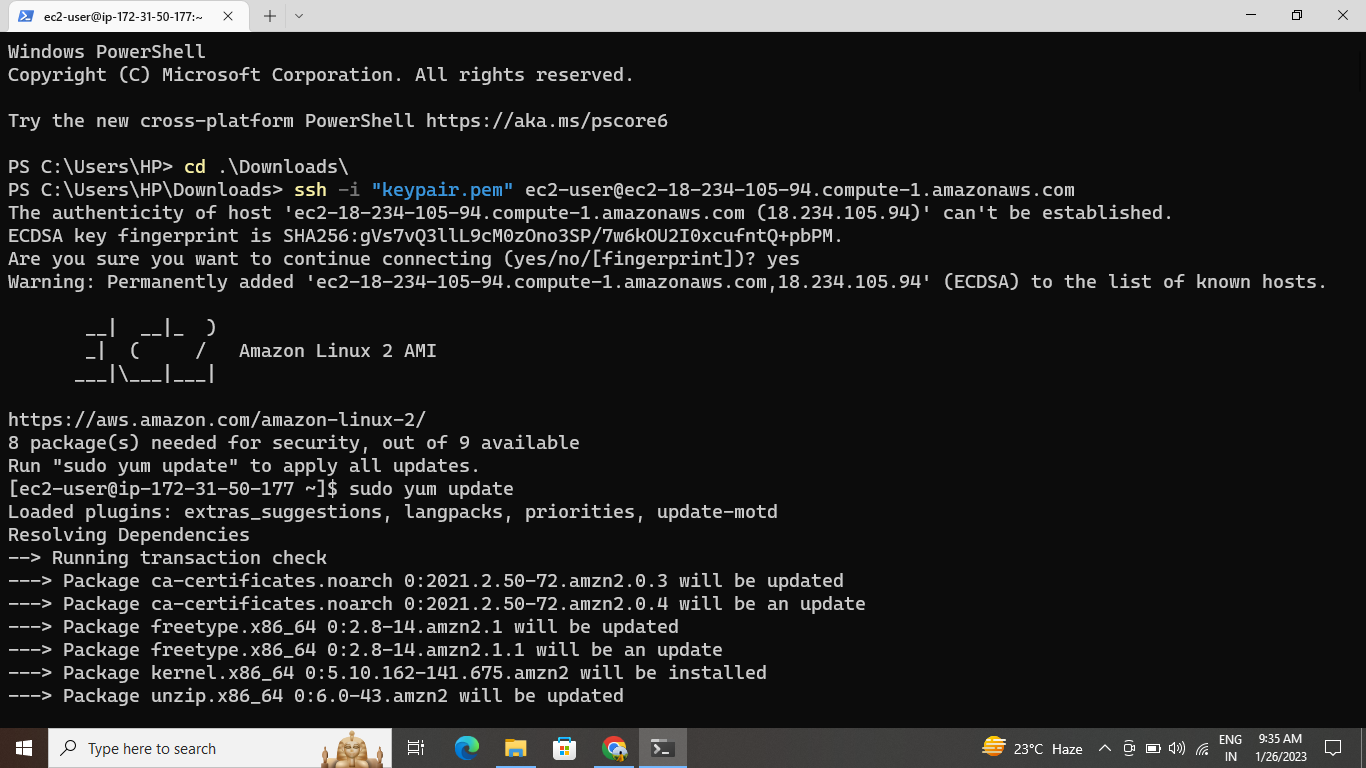
🡪Inbound rules as SSH-22, HTTP-80, HTTPS-443

🡪Outbound rules as ALL TRAFFIC.





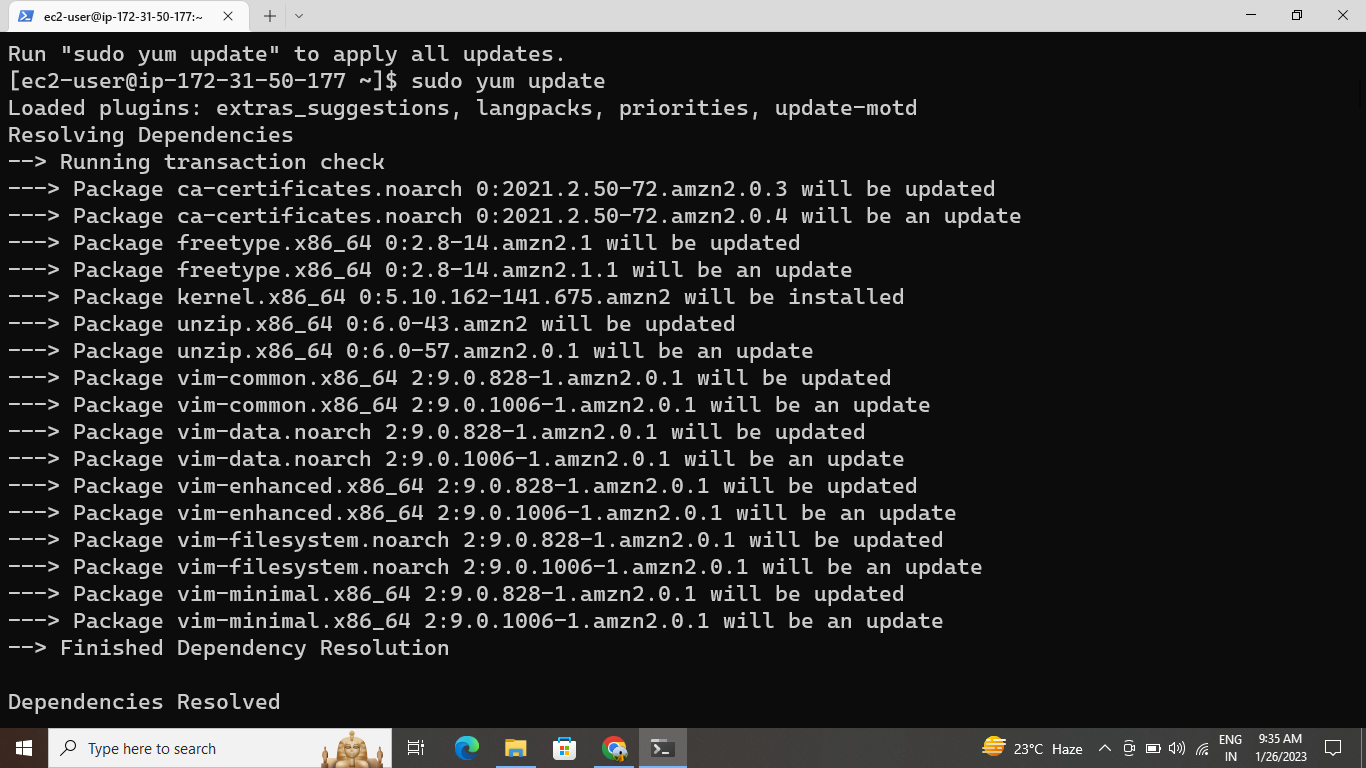
Now connect the instance to the terminal.



**INSTALLING GIT, DOCKER, AND RELATED REPOS**

1. Update the launched instance by using command “sudo yum -y update”
2. Now install git by using command “sudo yum -y

install git”



1. Now install docker by using command “sudo yum -y install docker”

Give the permission to add a limited linux user account to the “docker” group by using the command “sudo usermod -a -G docker ec2-user”.

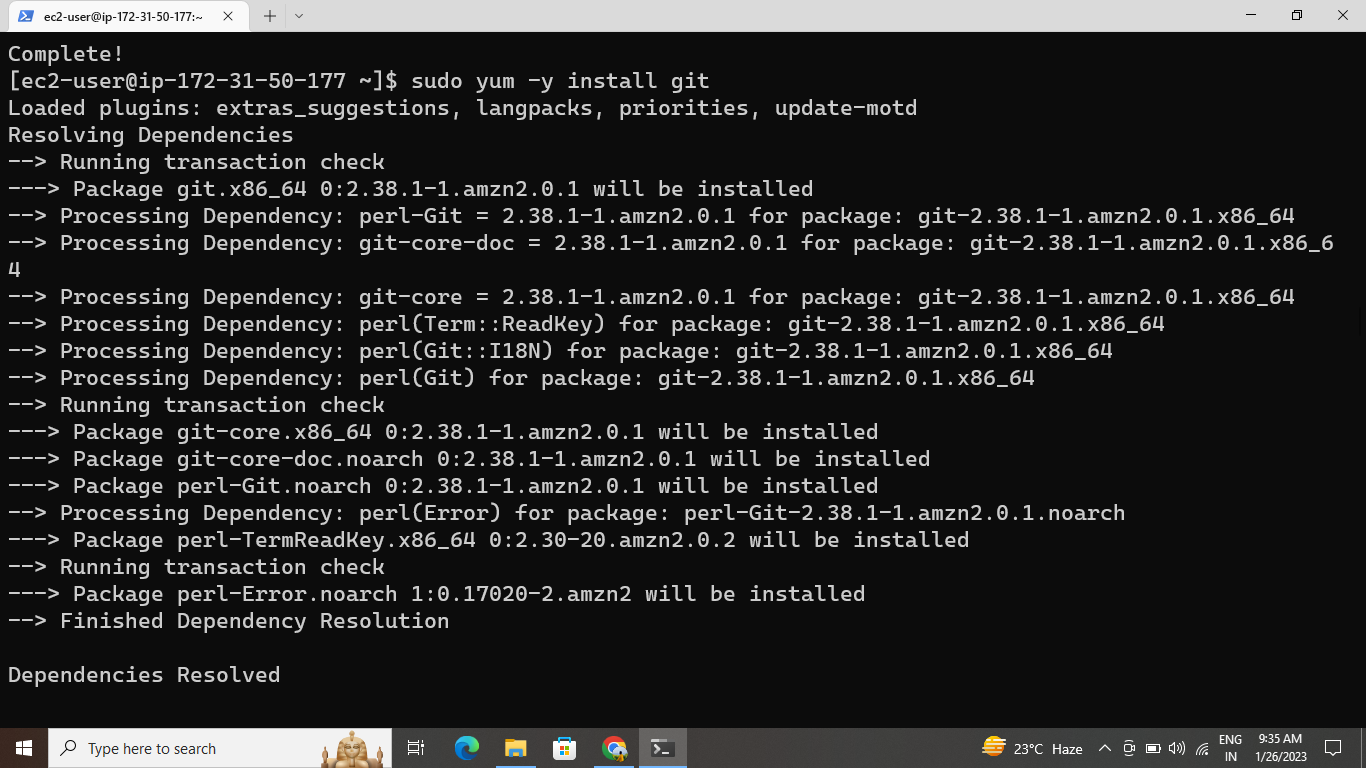
1. Now start the docker service by using command “sudo service docker start”
2. Now start the docker service by using command “sudo chkconfig docker on”

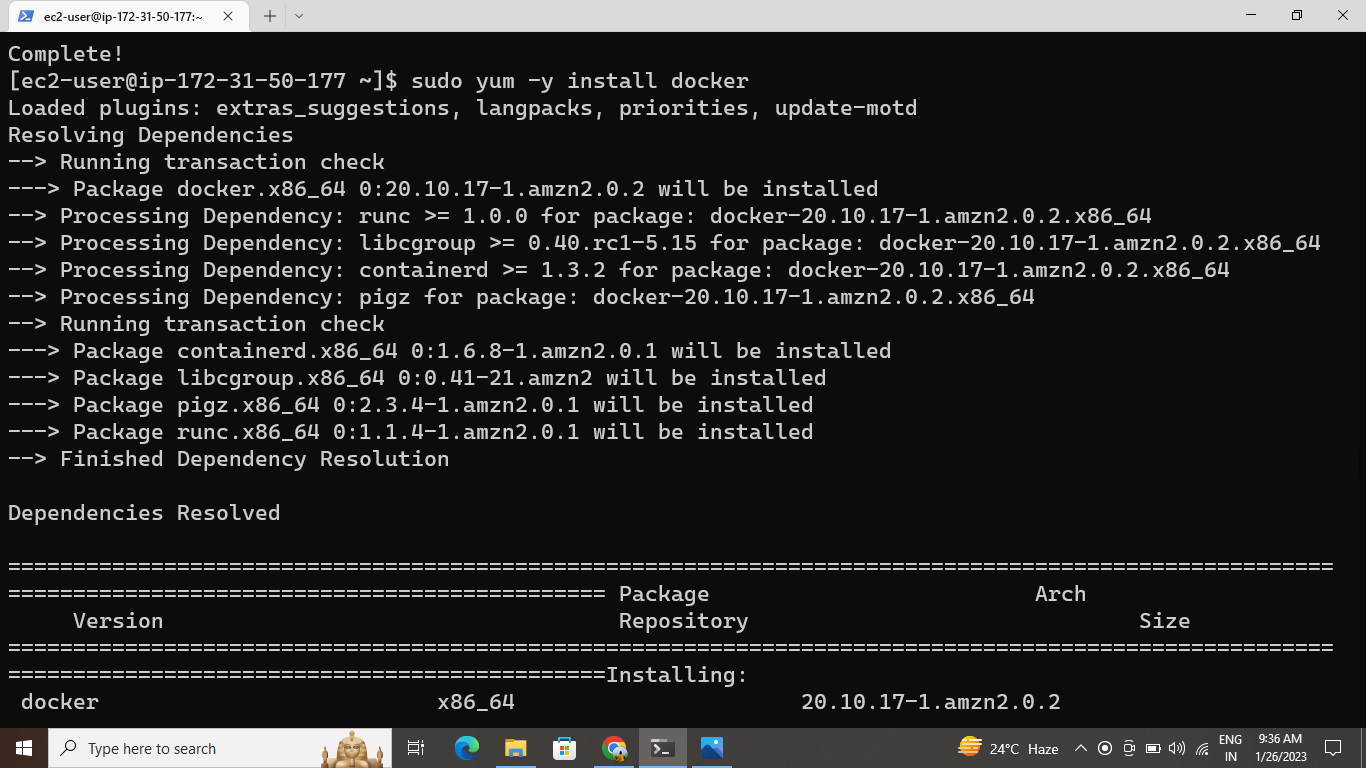
7.Download docker-compose latest version from the browser by clicking “install docker-compose on amazon linux -2 ”.

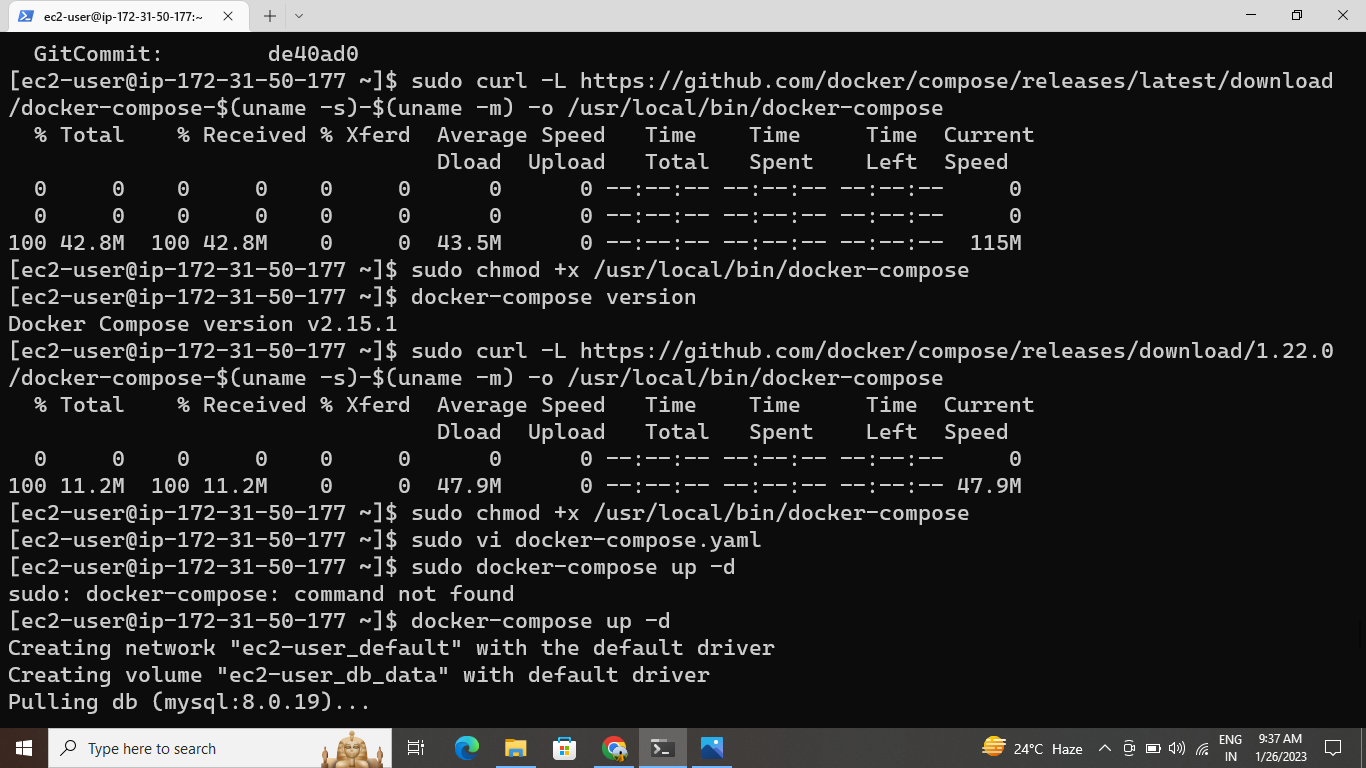
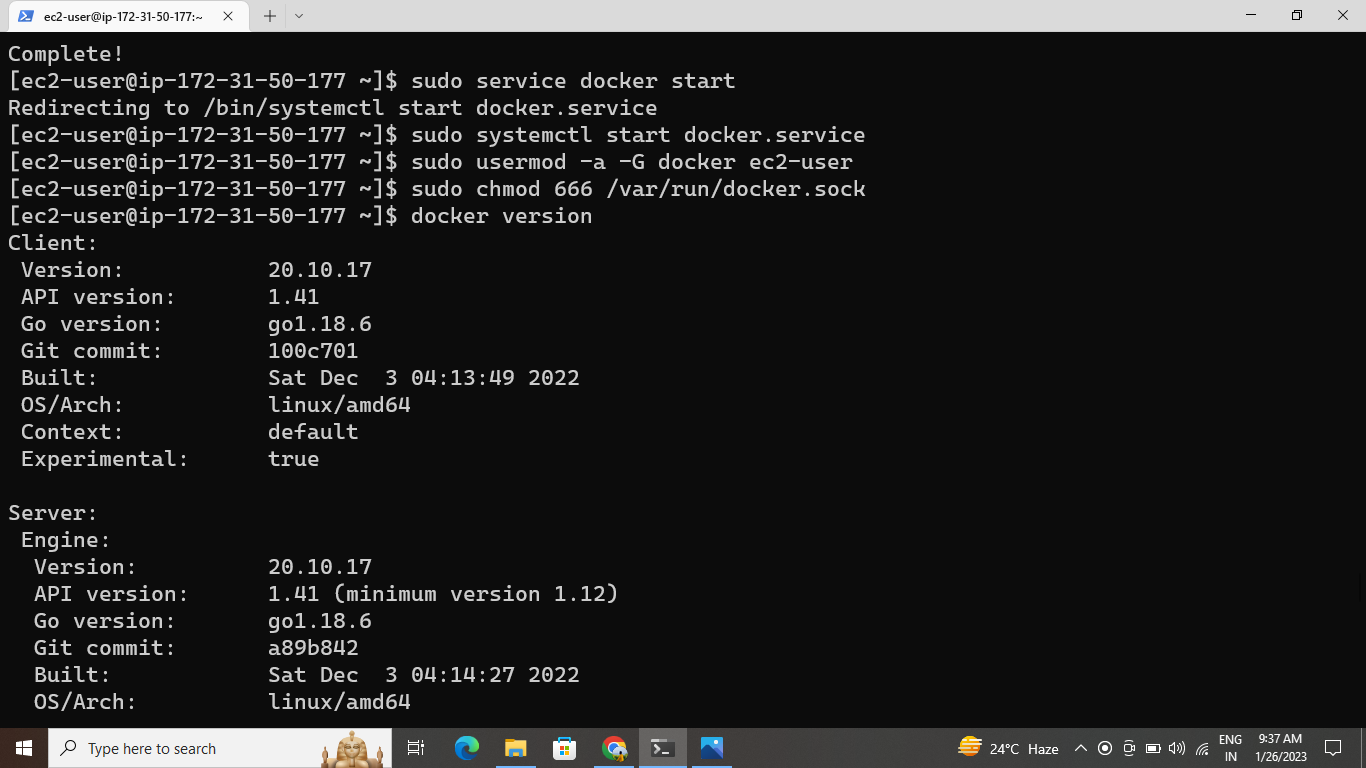
8.Now give the executable permissions to the binary by using command “sudo chmod +x /usr/local/bin/docker-compose”.

9. Create a symbolic link by using command “sudo ln -s /usr/local/bin/docker-compose /usr/bin/docker-compose”.

10. Now check installed docker-compose version by using command “docker-compose -version”.





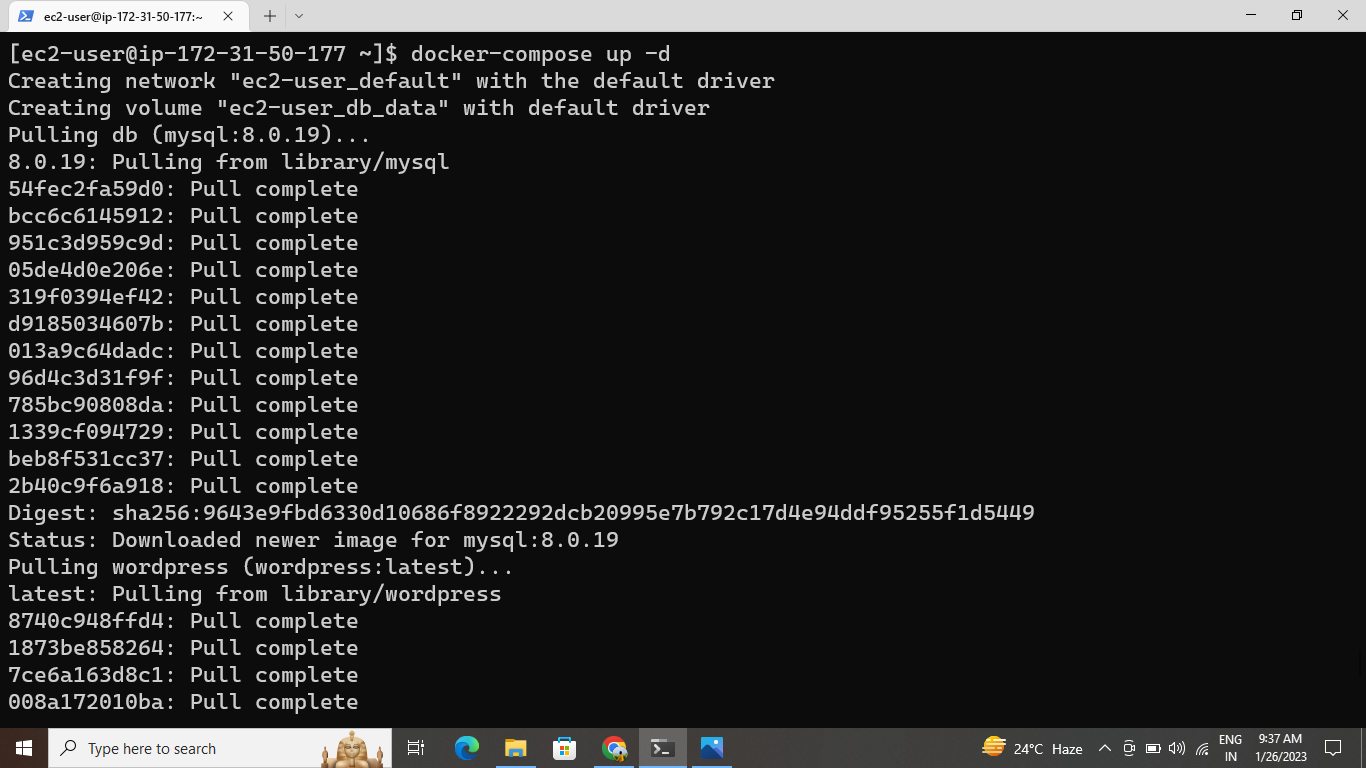


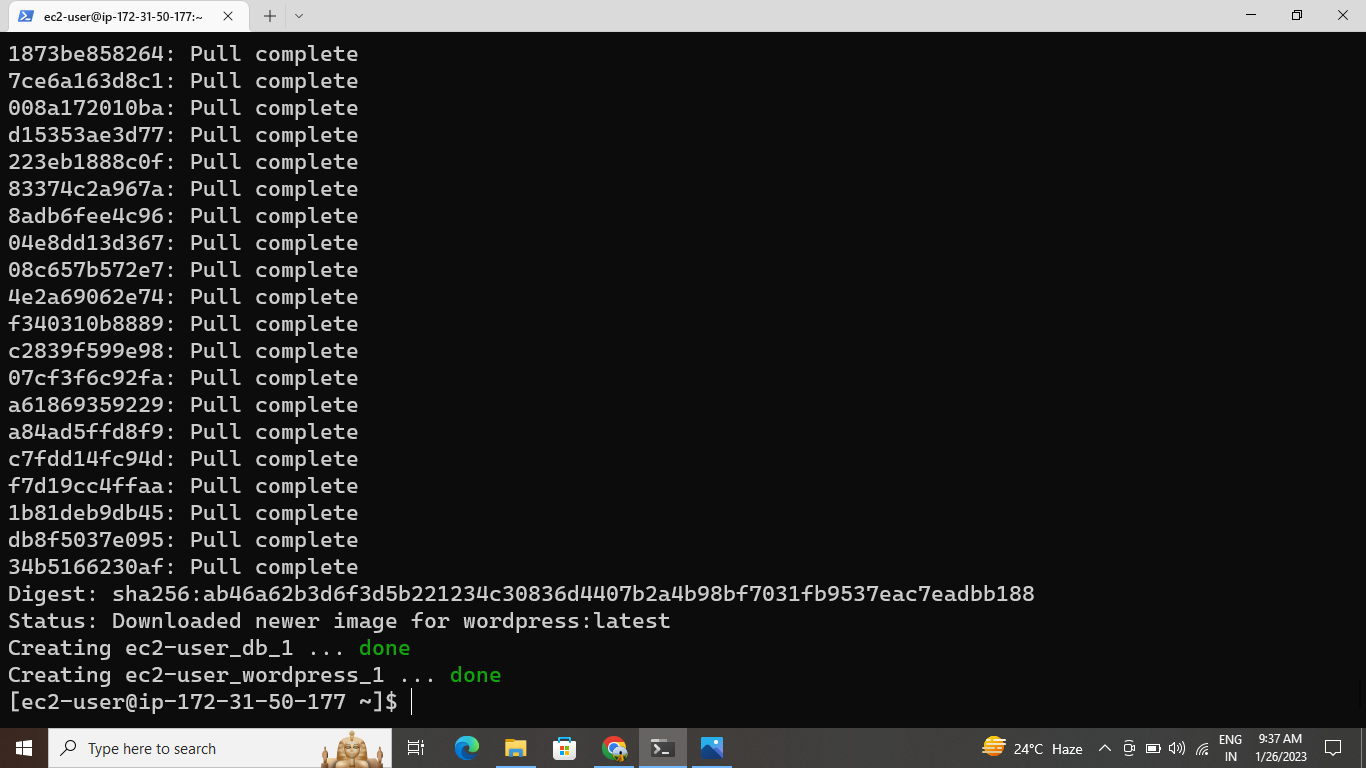
**CREATING WORDPRESS SETUP FOR DOCKER IMAGES WITH THE HELP OF YAML SCRIPTING**

1. We have to create docker-compose.yaml file to pull the image from docker hub by using command “sudo vi docker-compose.yaml”.

Here is our docker-compose.yaml file:

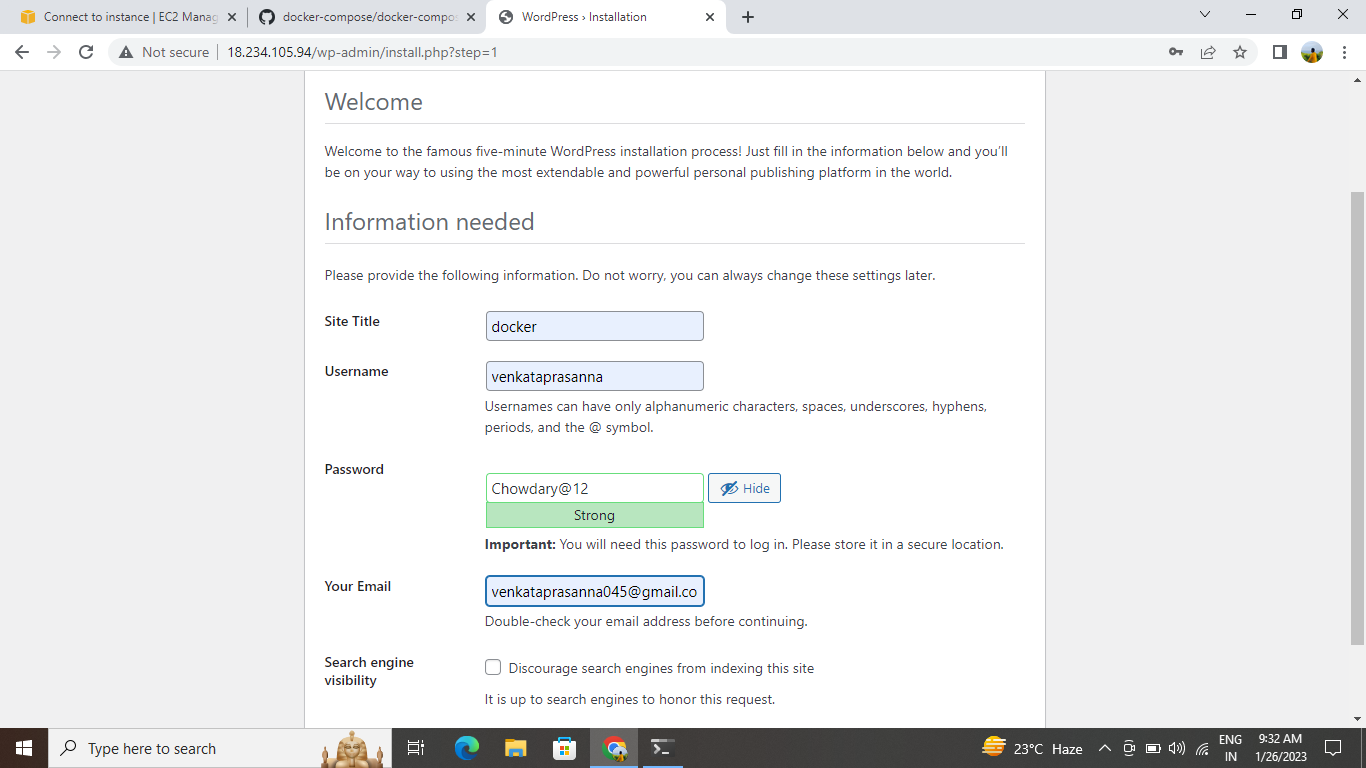
1. Initially I had a go at running this locally on with docker compose by using command “sudo docker-compose up -d”

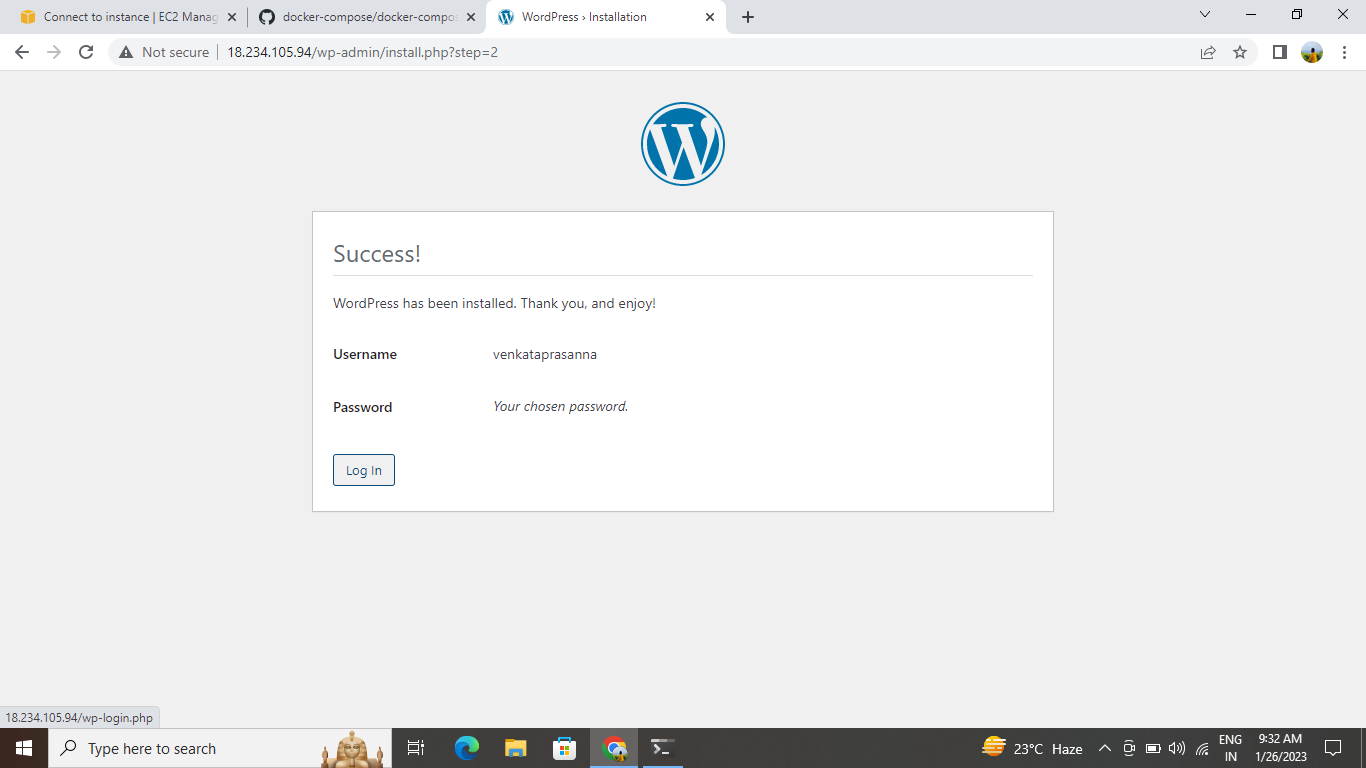


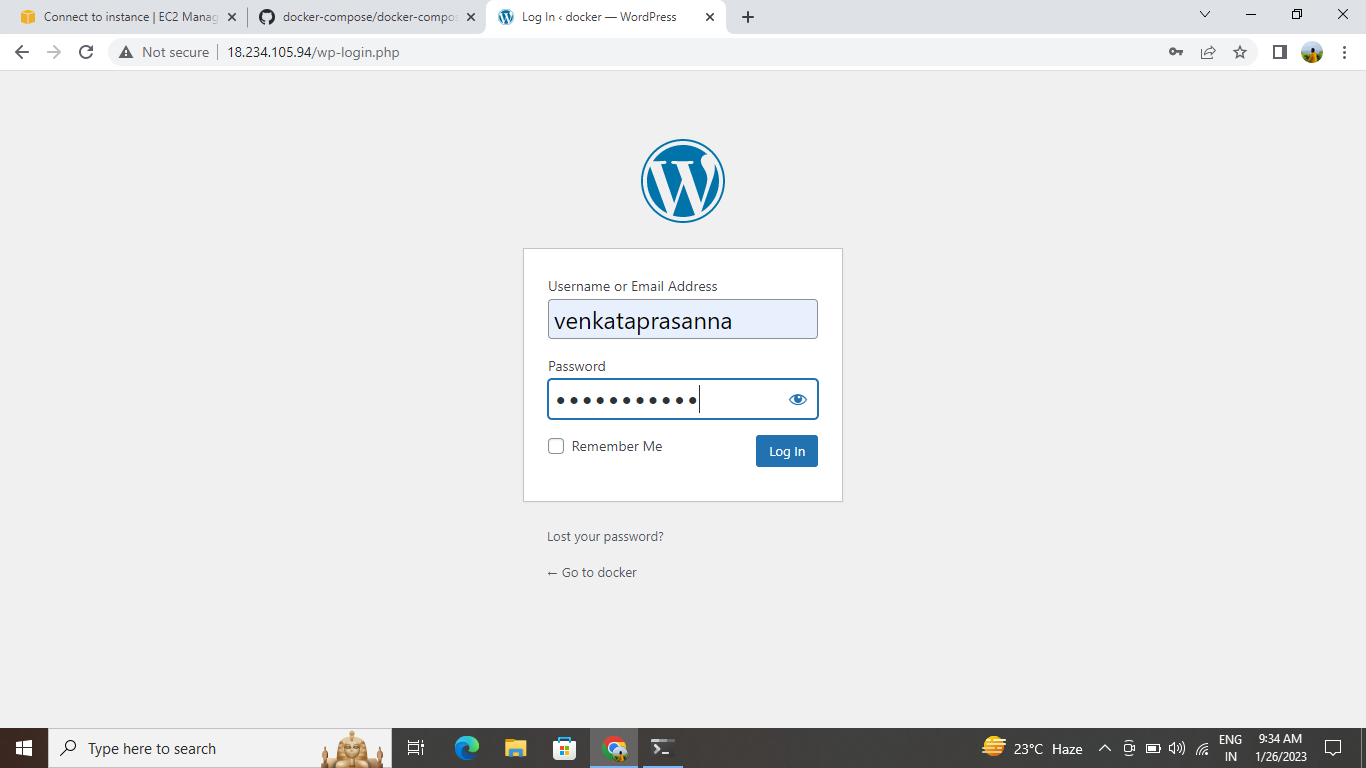


1. Now I can get the list of running containers by using command “sudo docker ps”.
2. And then lastly, I had look to see this is running correctly.

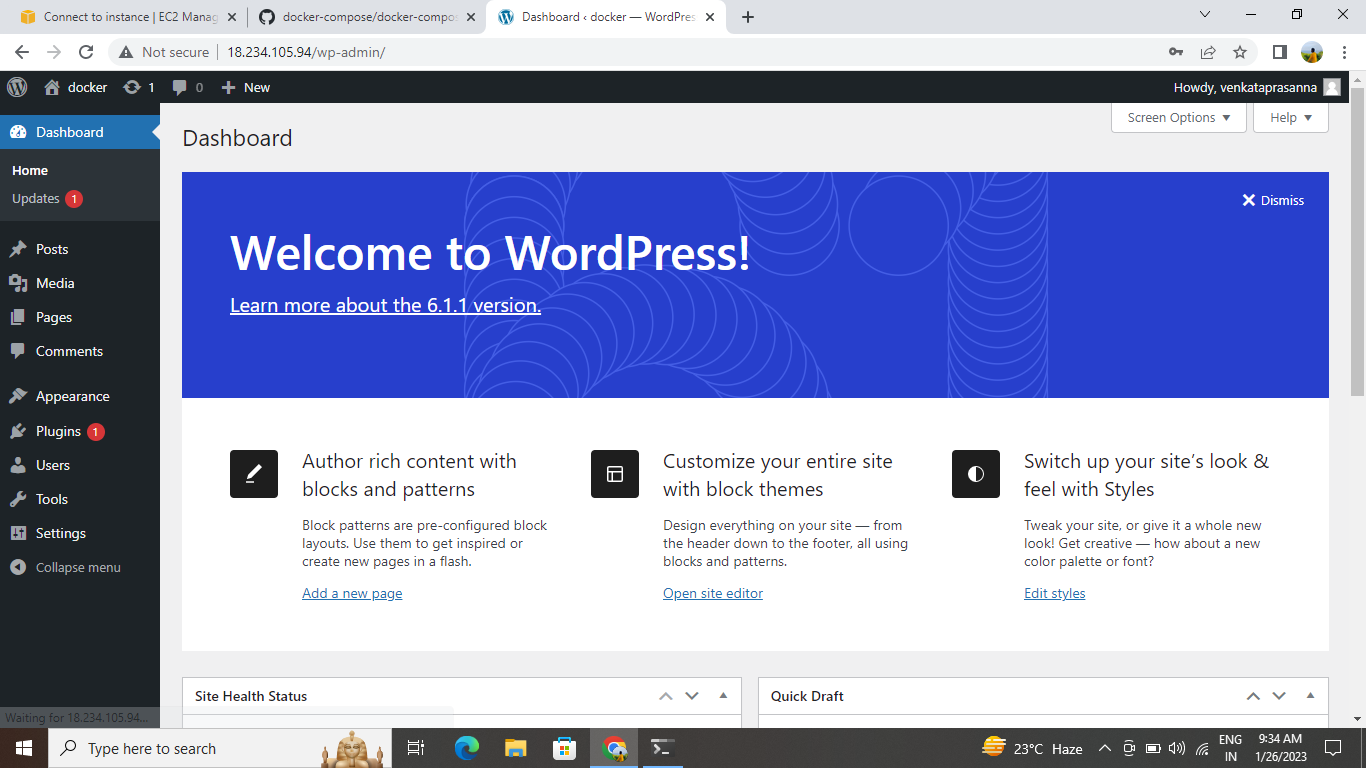








This is the page wordpress



1. Create a repo in the github account, and clone that code .
2. Now move the file local repository to remote repository.
3. Now add the file by using command “git add <filename>”
4. Now commit the file by using “git commit -m “add””

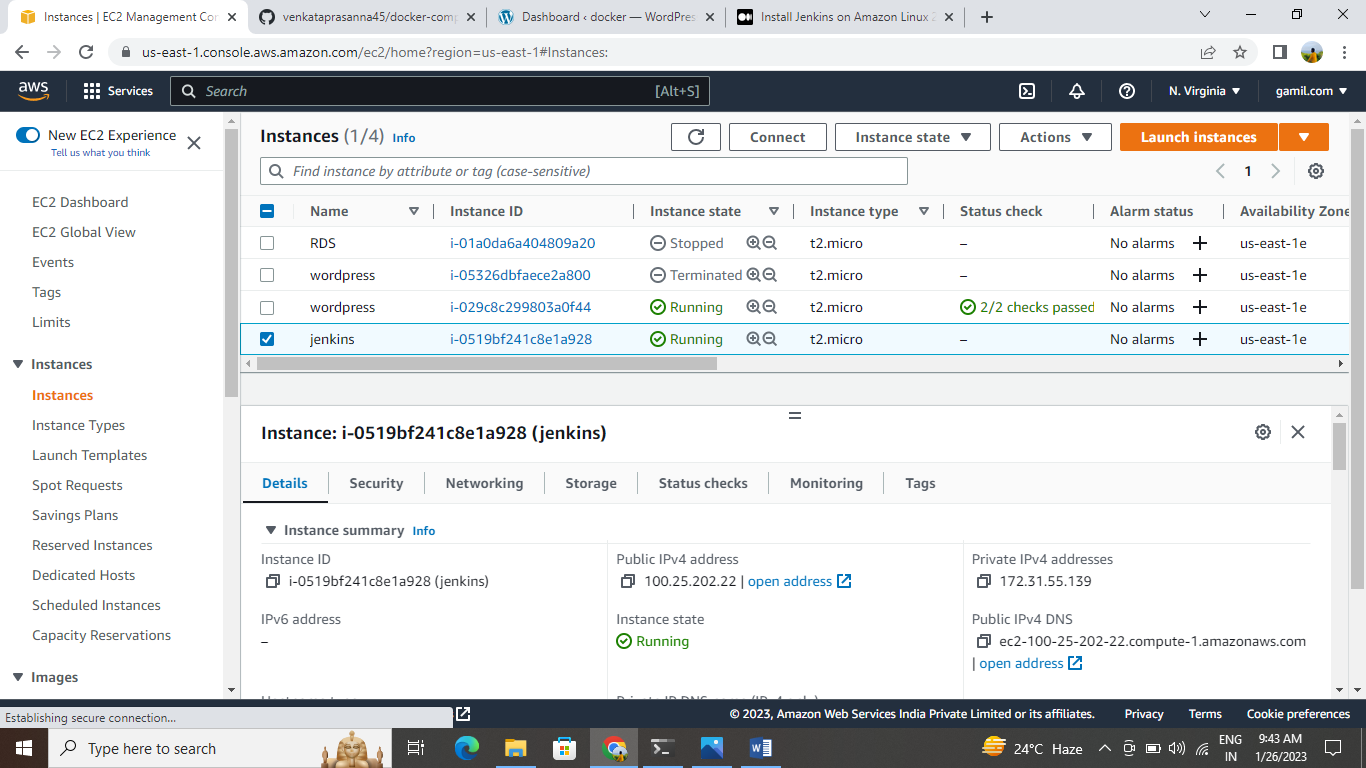
10. Now push the file by using the command “git push”

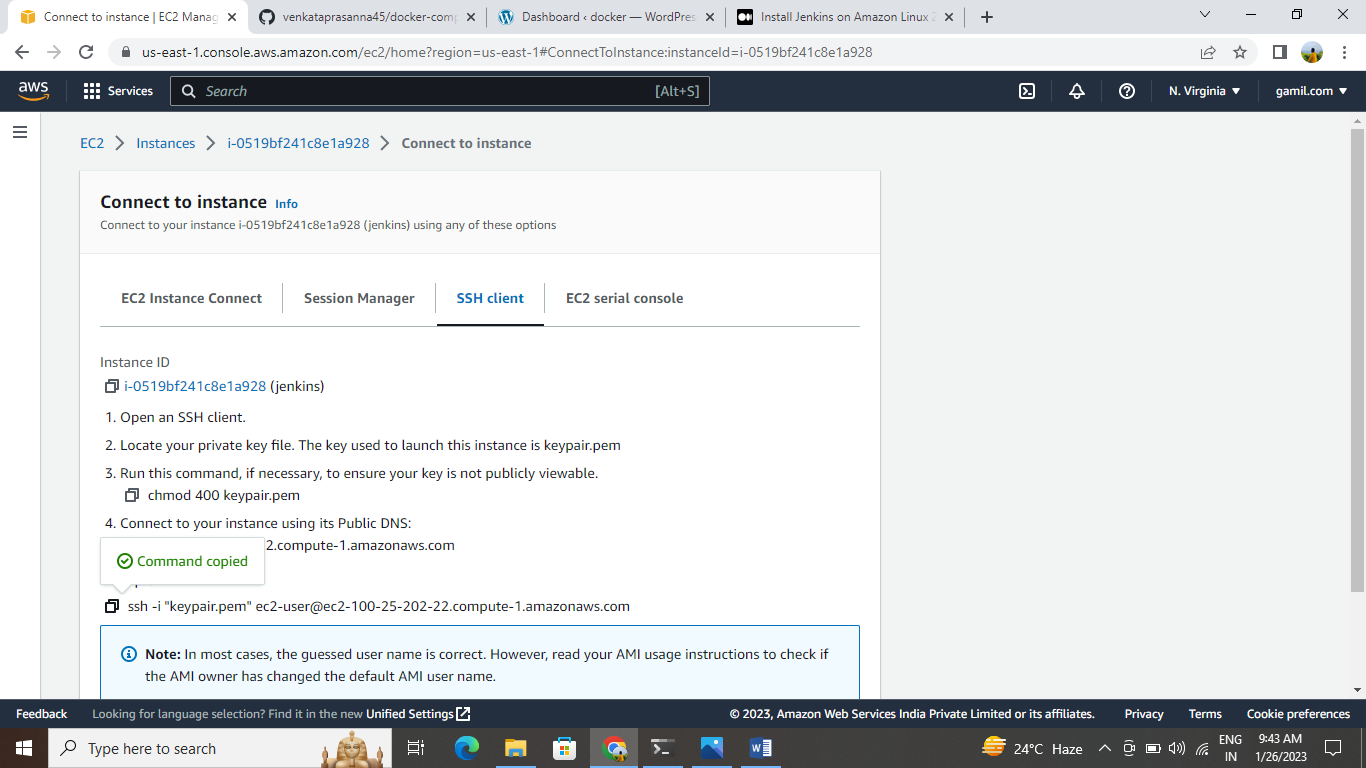
11. Now create an EC2 instance on the centos machine and connect to the terminal.

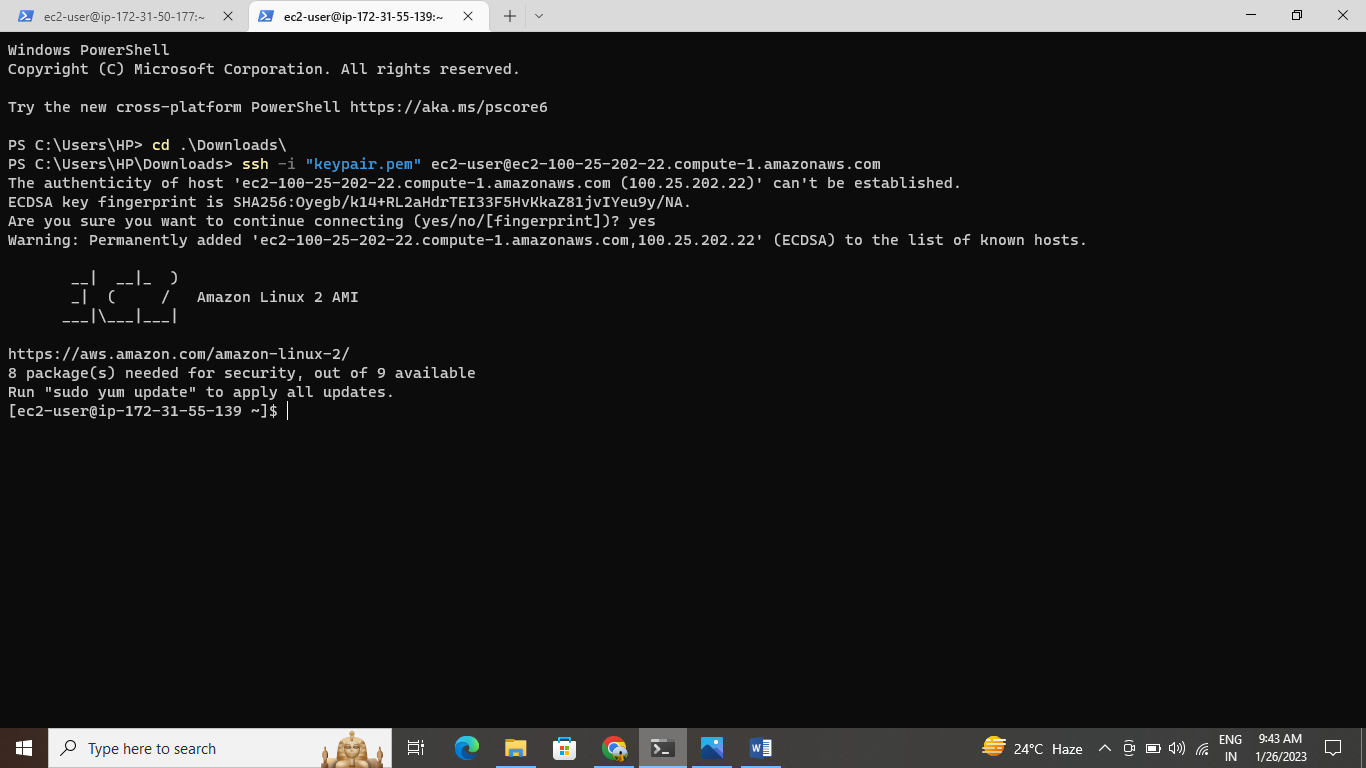
12. Now install git by using command “sudo yum -y install git”.

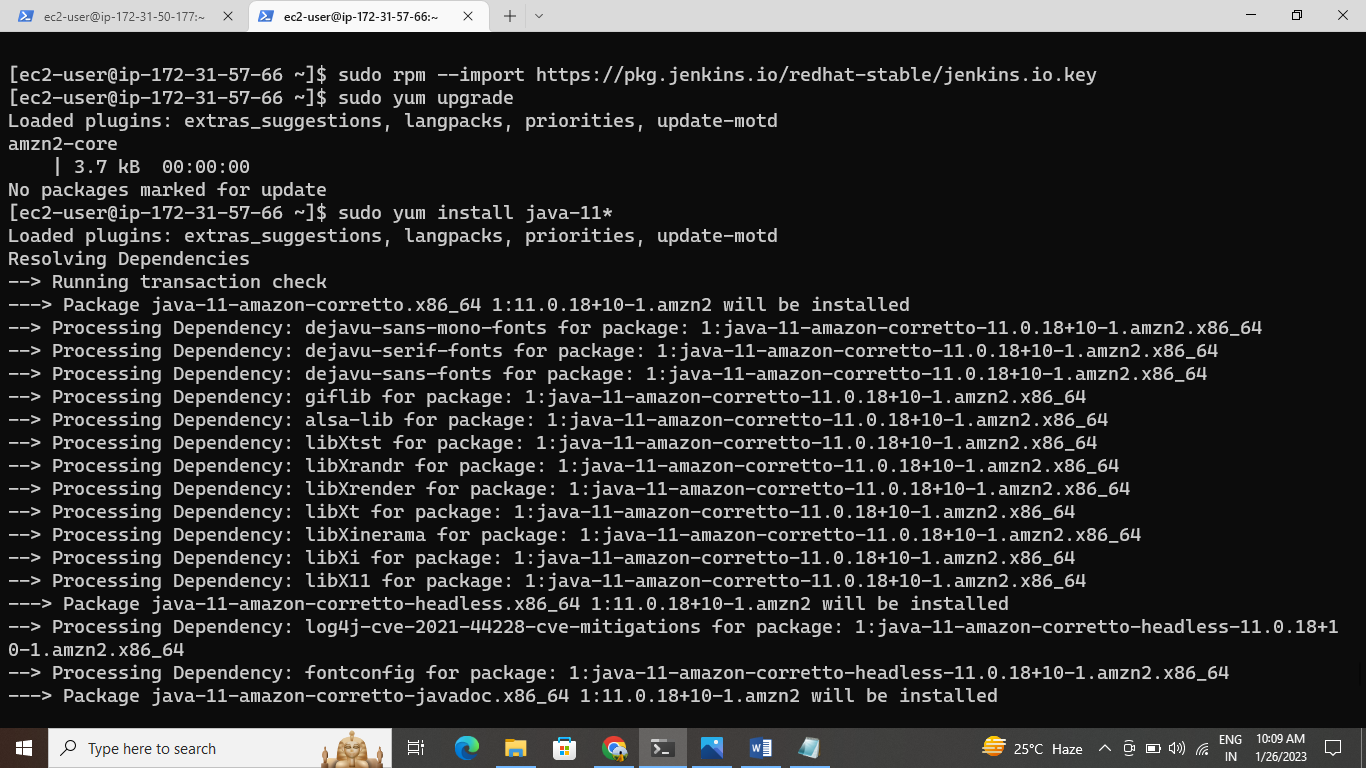
13.Install java by using command “sudo yum -y install java-11\*”.

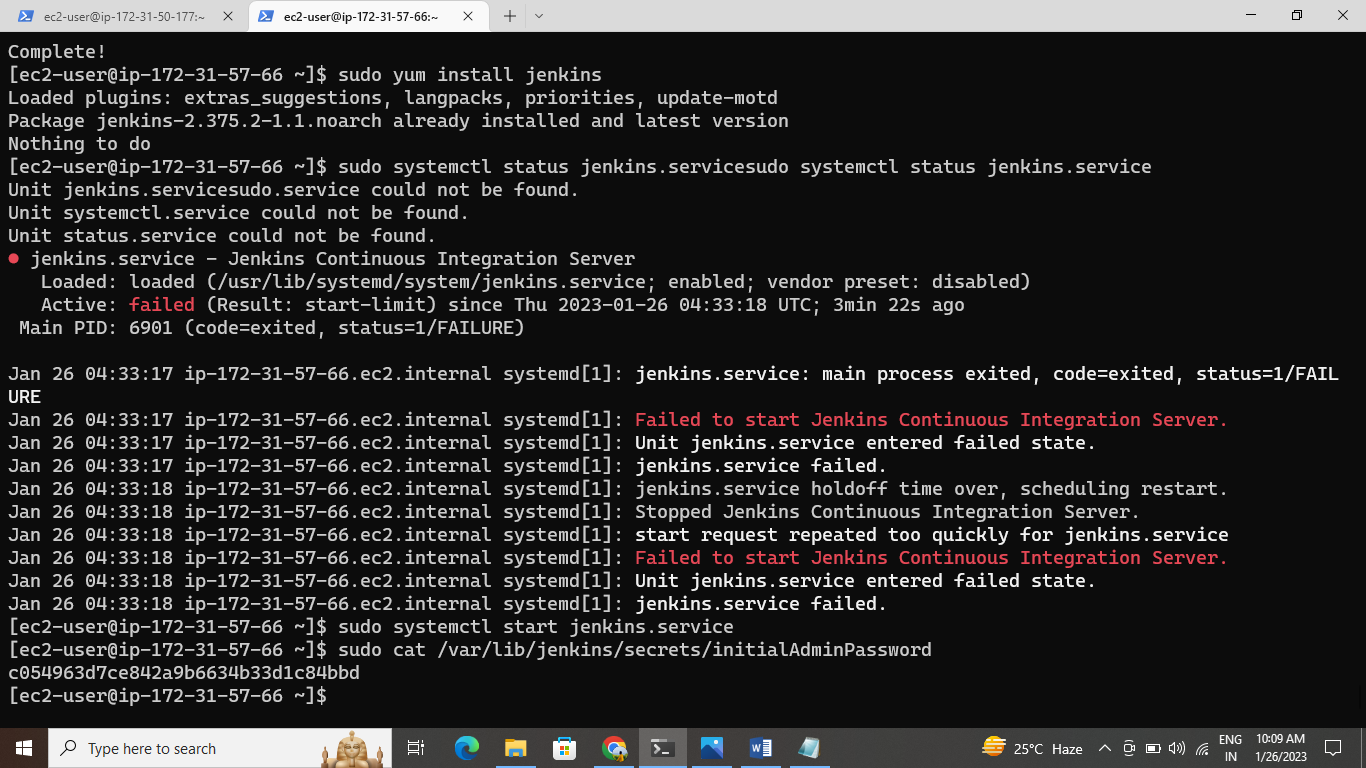
14. Now install wget by using command “sudo yum -y install git”.

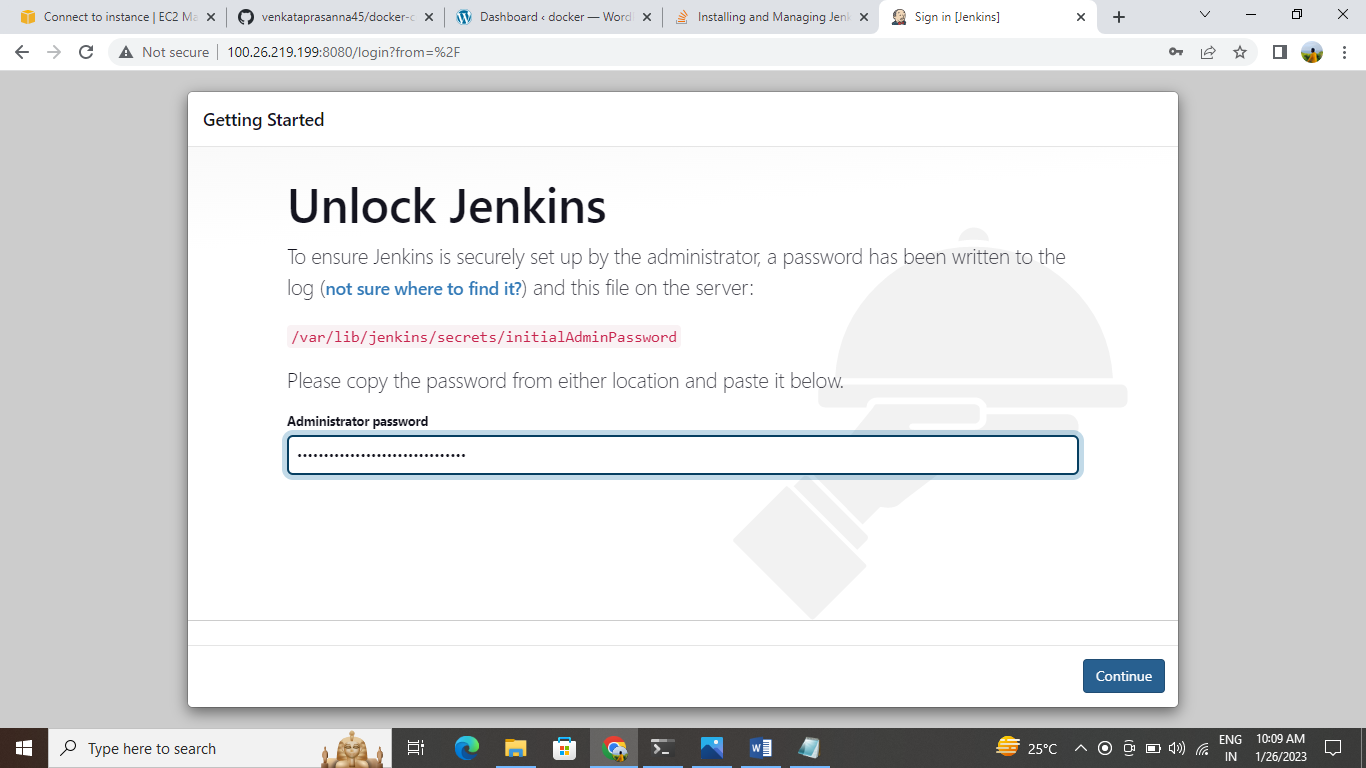


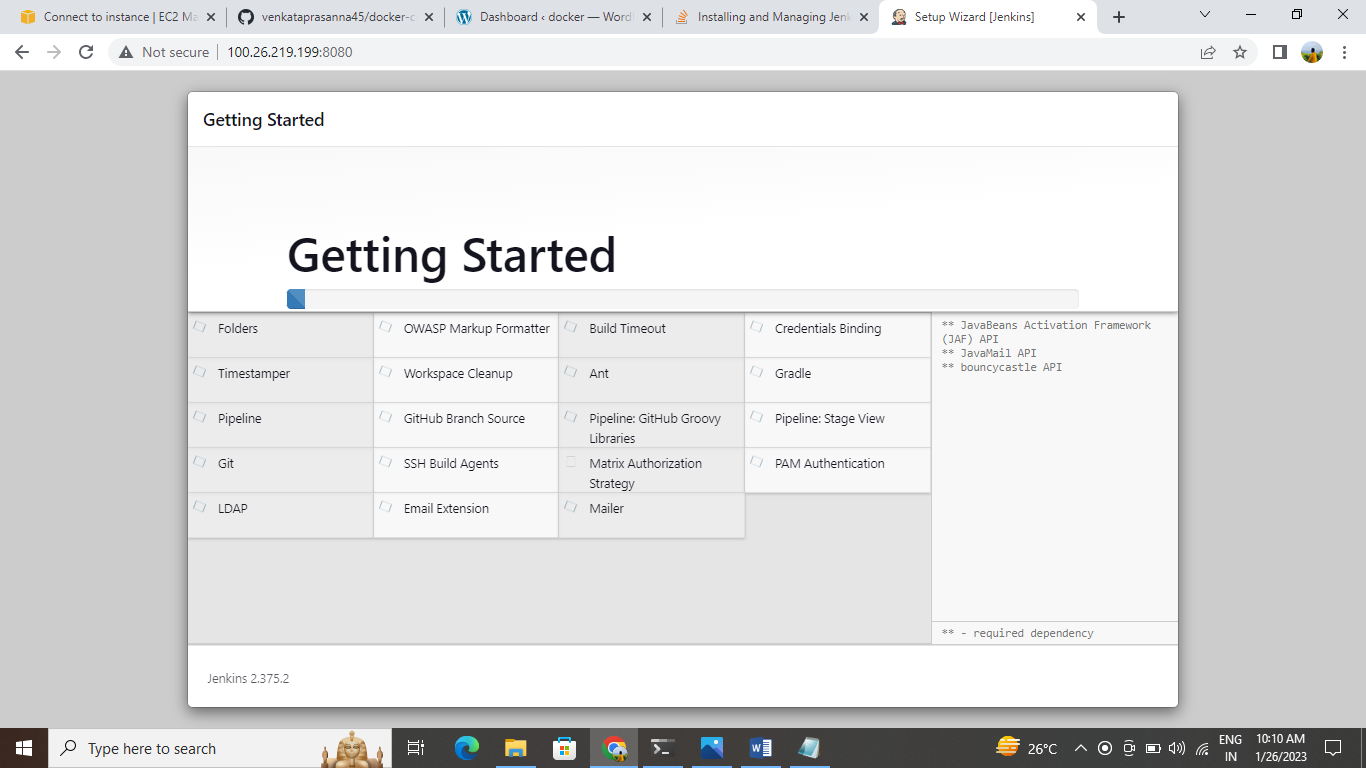


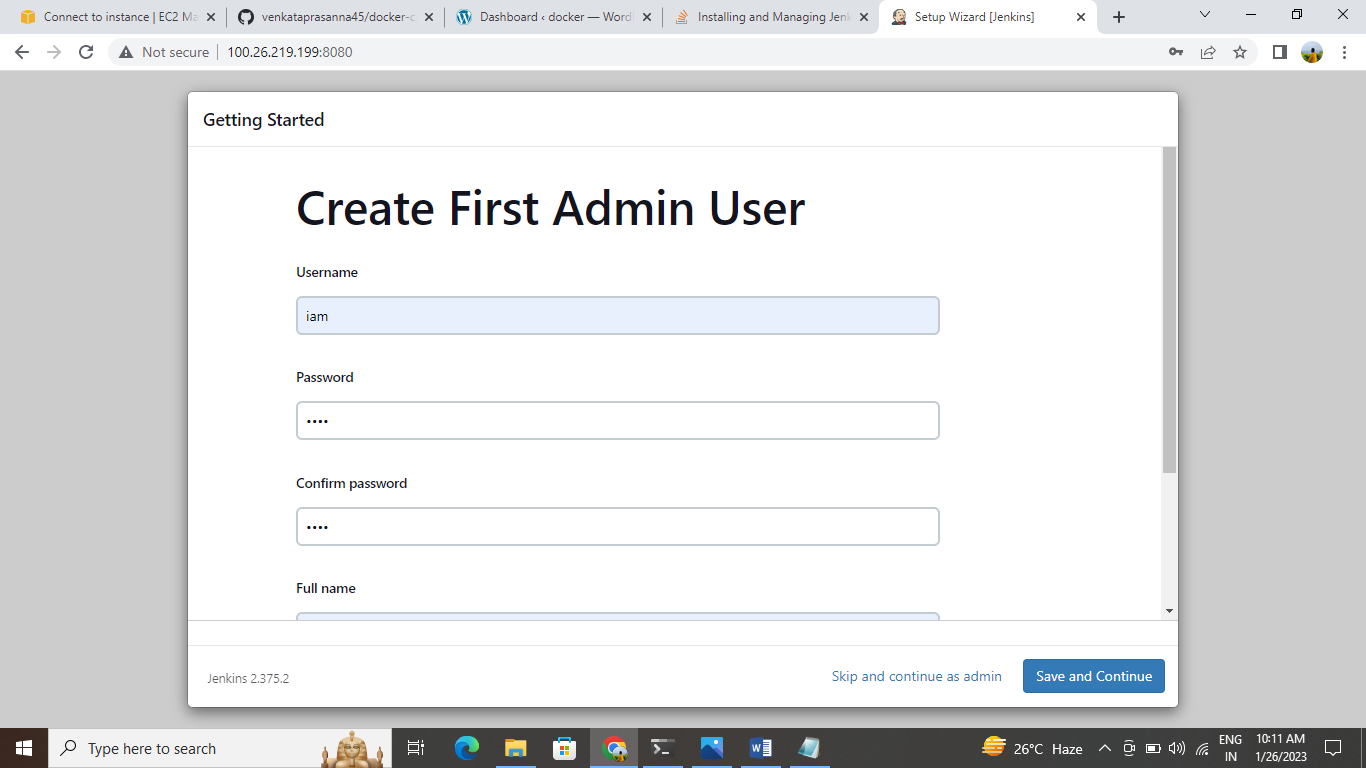


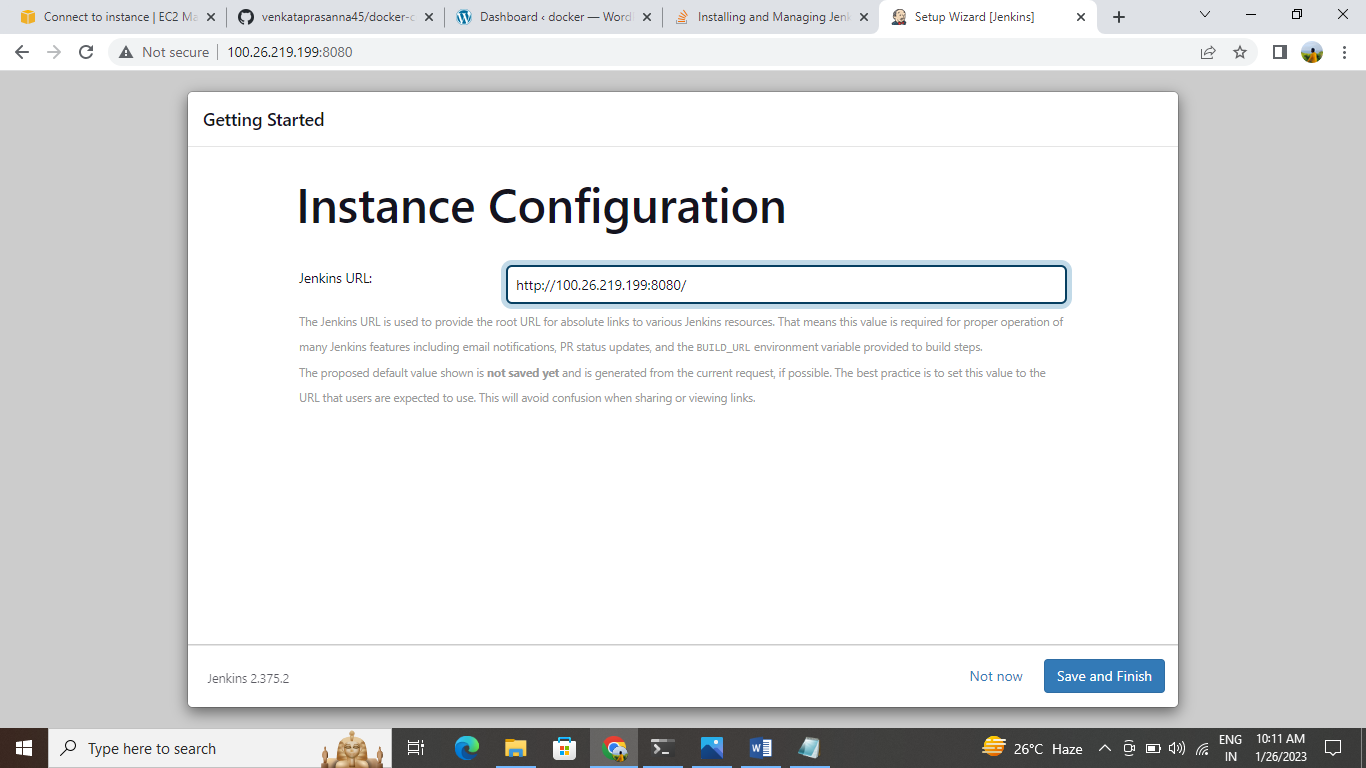












15. Now install Jenkins latest version by using command from the browser.

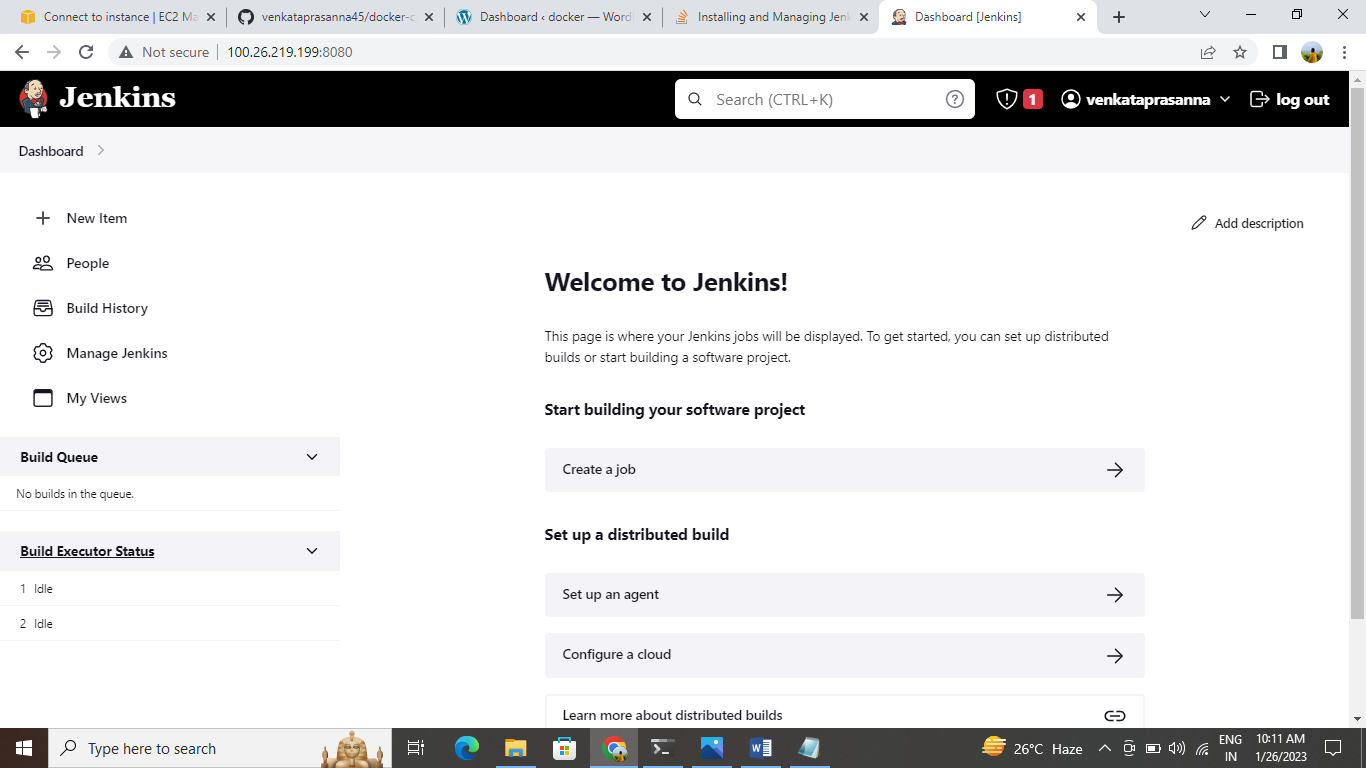
16. And then lastly, I had to look to see this is running correctly.

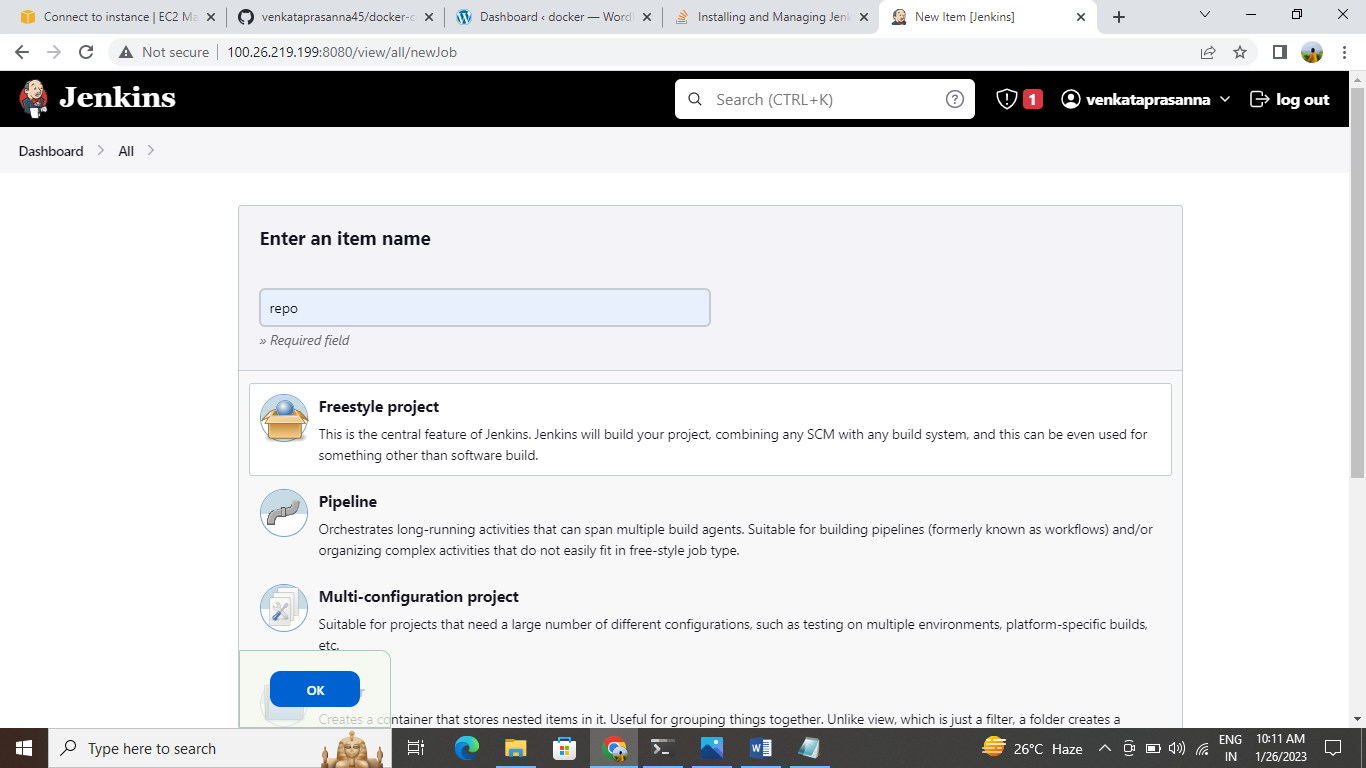
17. Now start the Jenkins by using command “sudo systemctl start jenkins”.

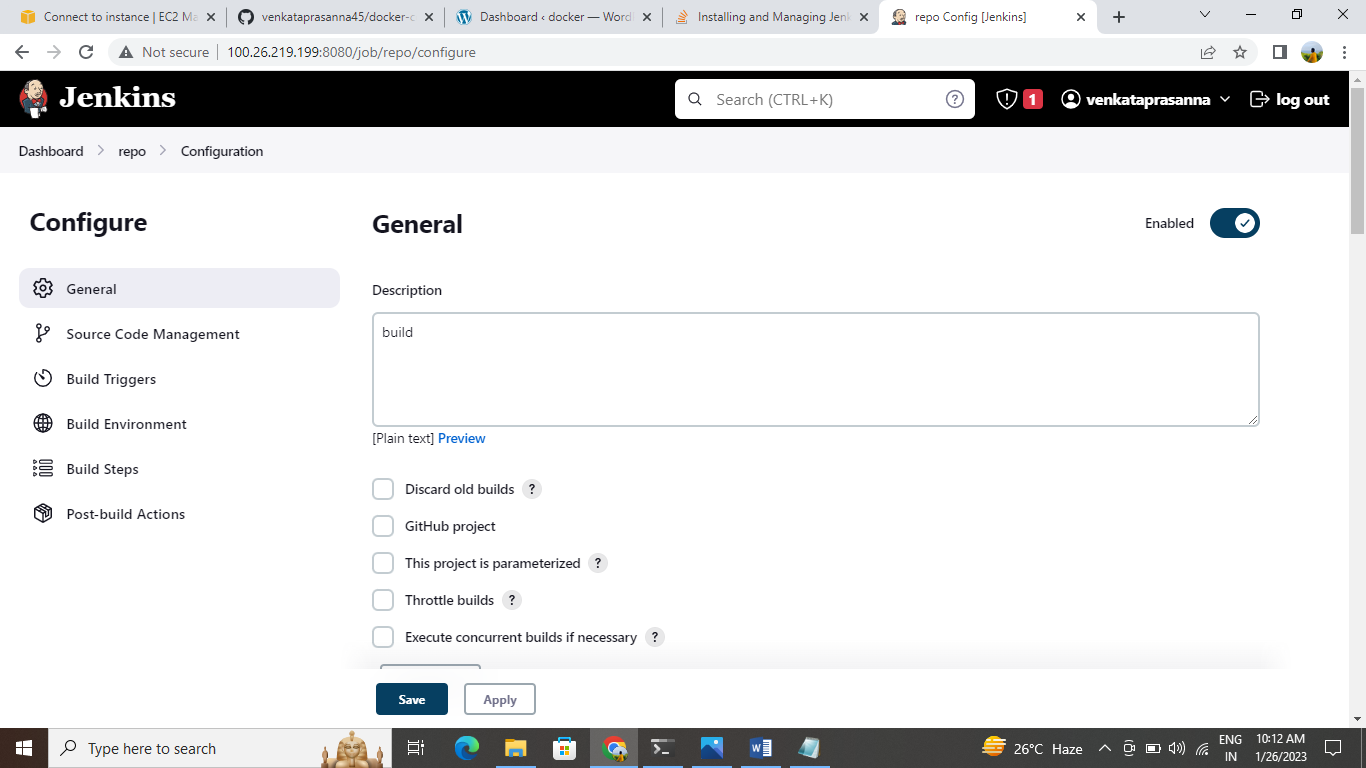
18. Now enable the Jenkins by using command “sudo systemctl enable jenkins”.

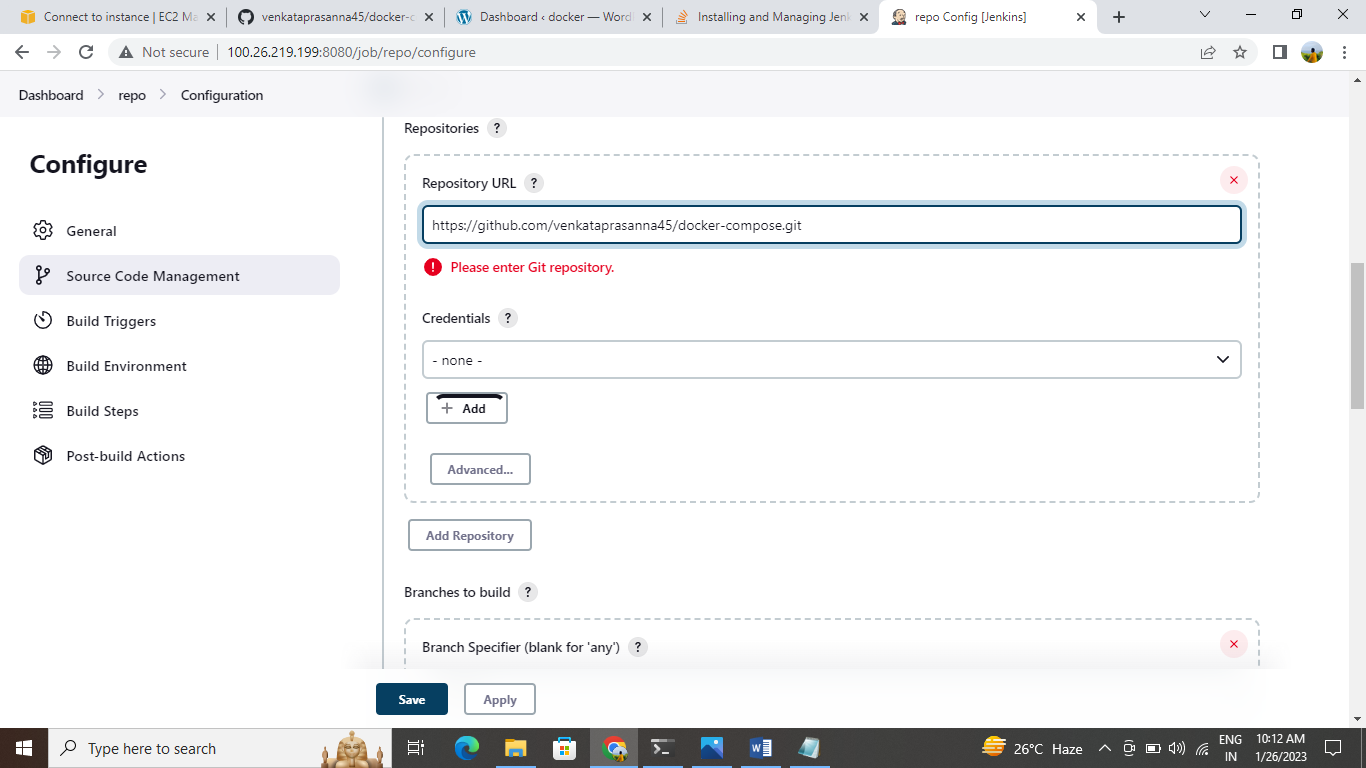
19. Now do git clone in Jenkins.

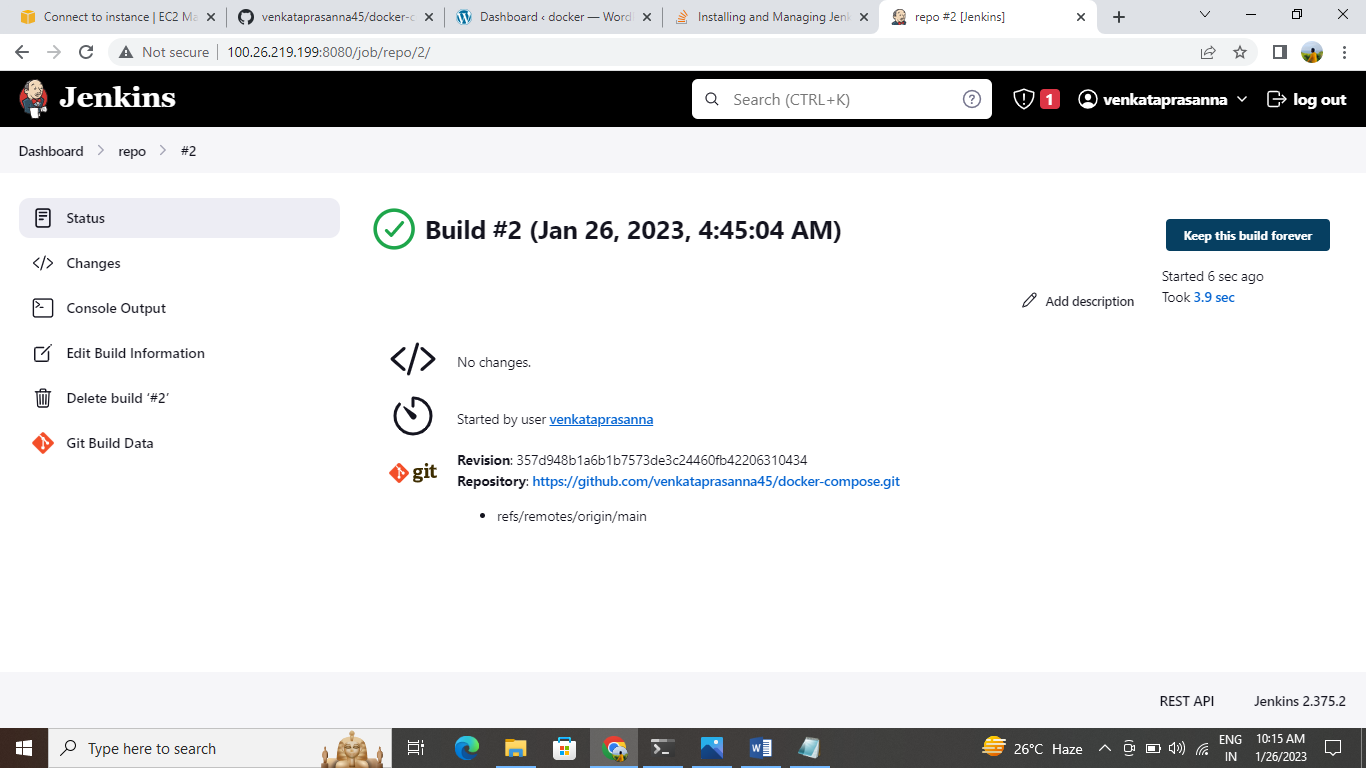
Copy the Public IPv4 address and add port no 80 in the web browser

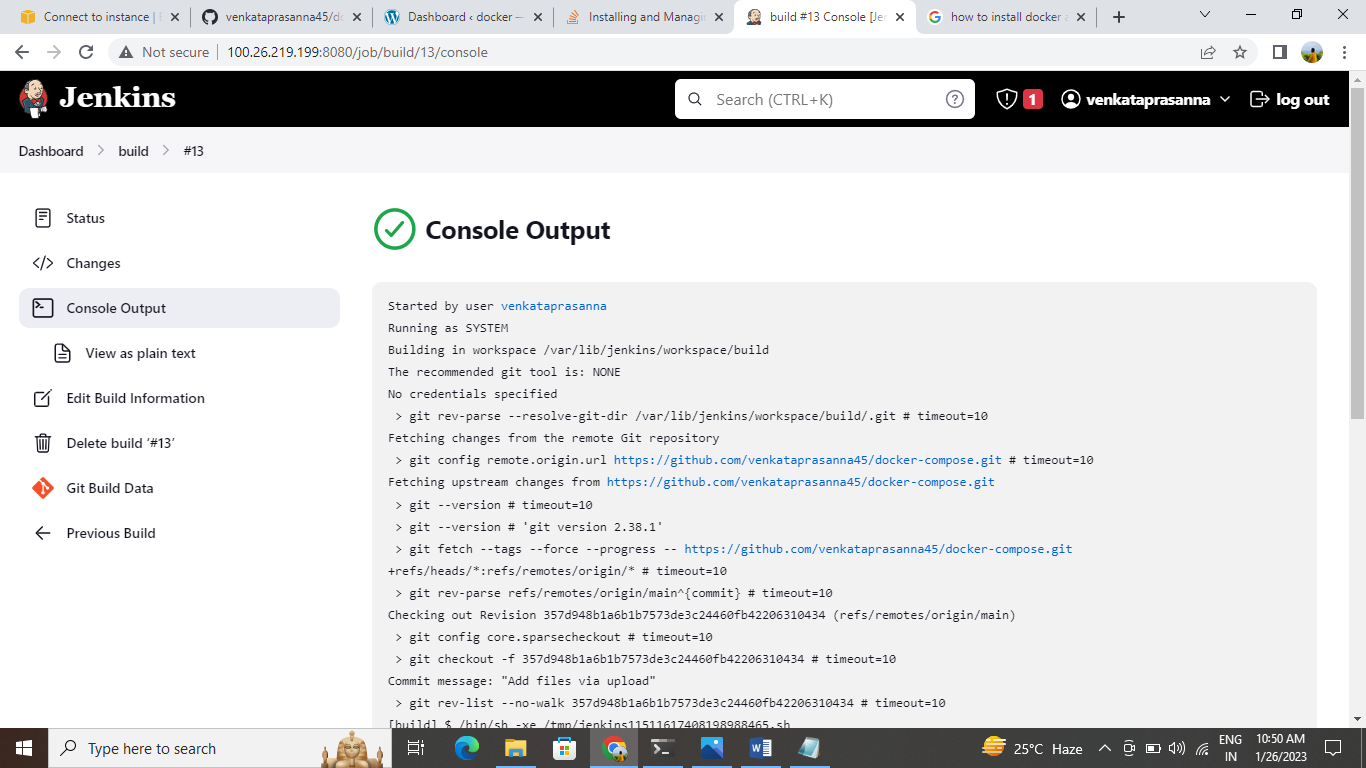












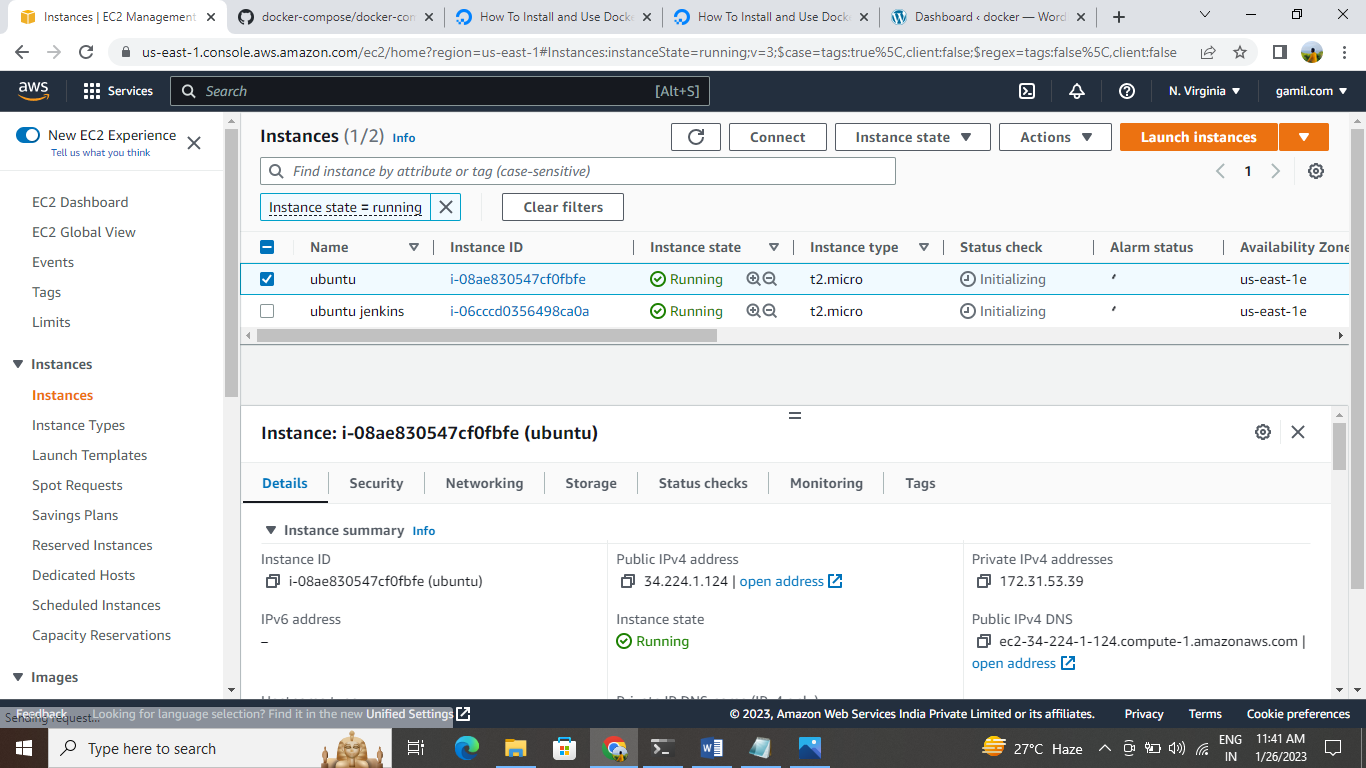
**CREATING AND LAUNCHING AN AMAZON LINUX EC2 INSTANCE USING UBUNTU**

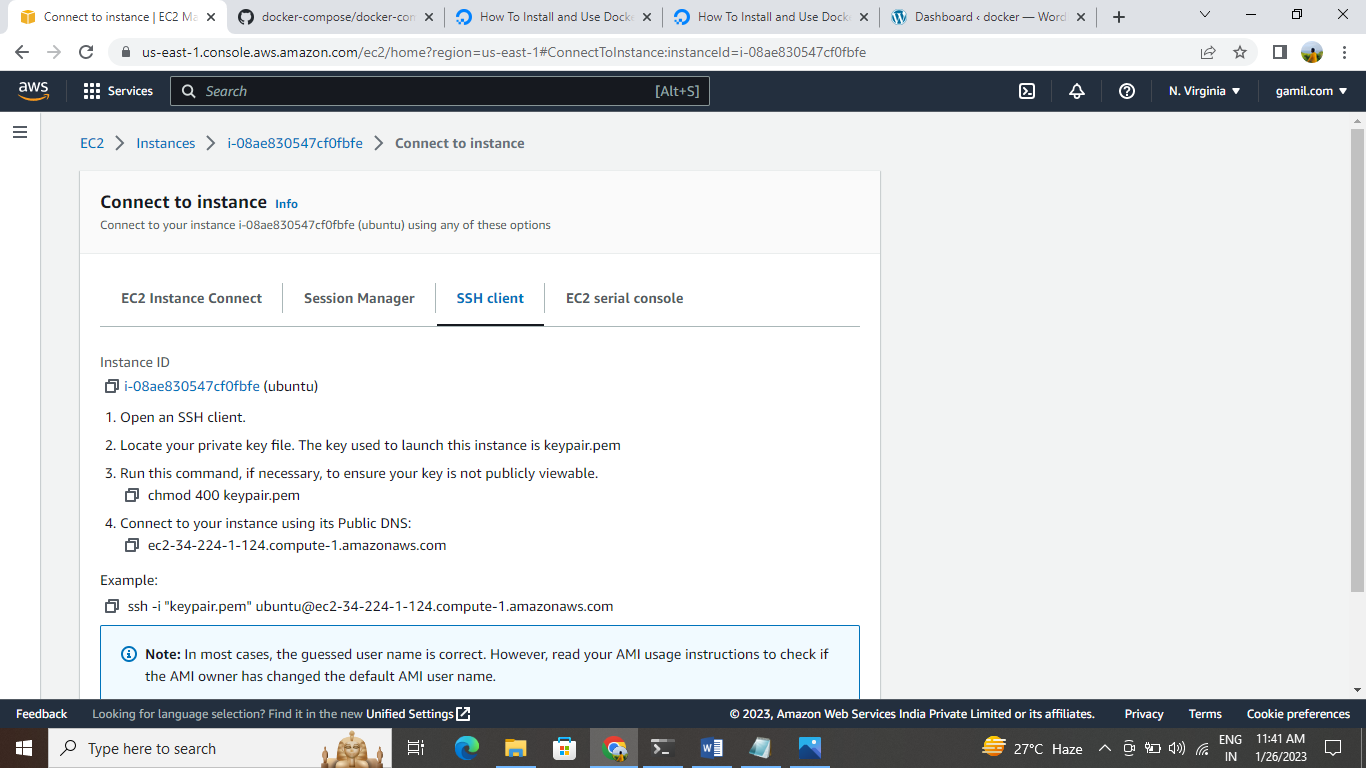
Create an EC2 instance with a Linux machine.

The security group for

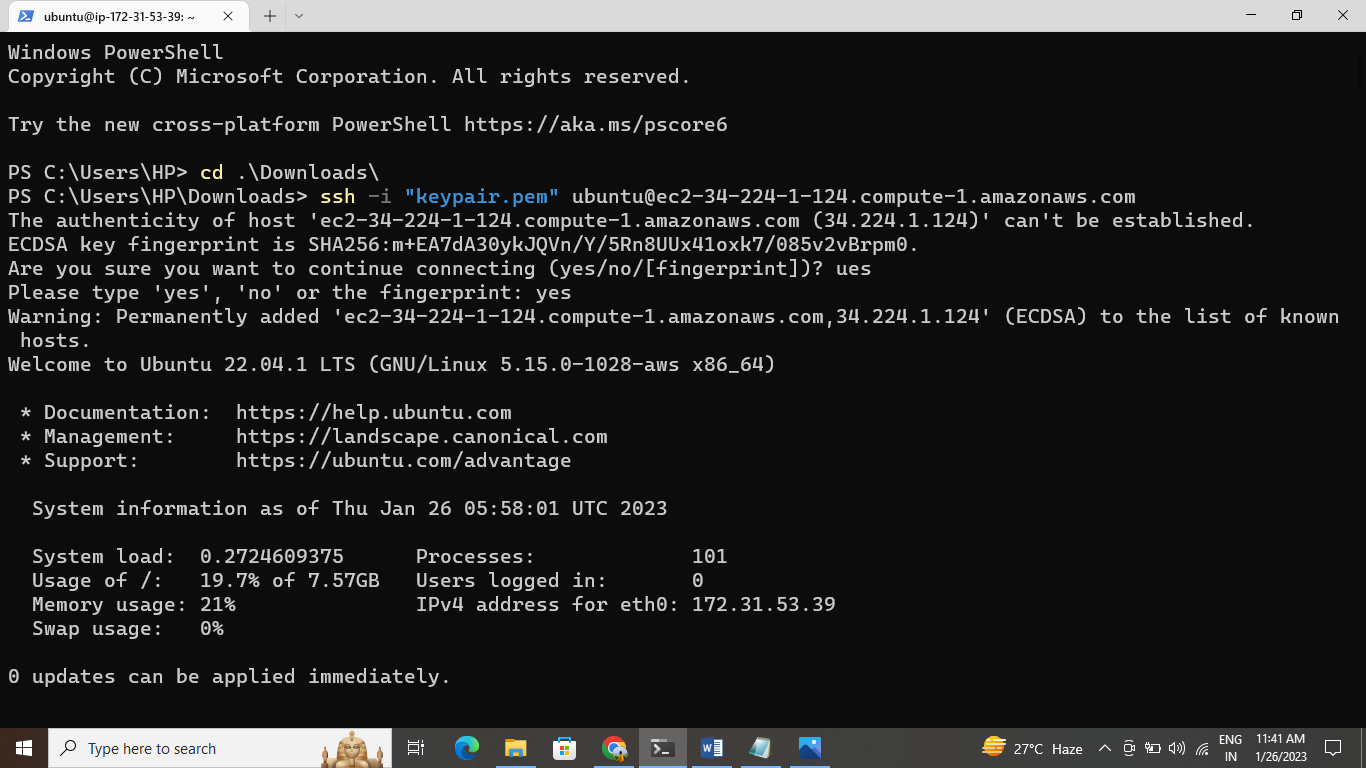
🡪Inbound rules as SSH-22, HTTP-80, HTTPS-443

🡪Outbound rules as ALL TRAFFIC.



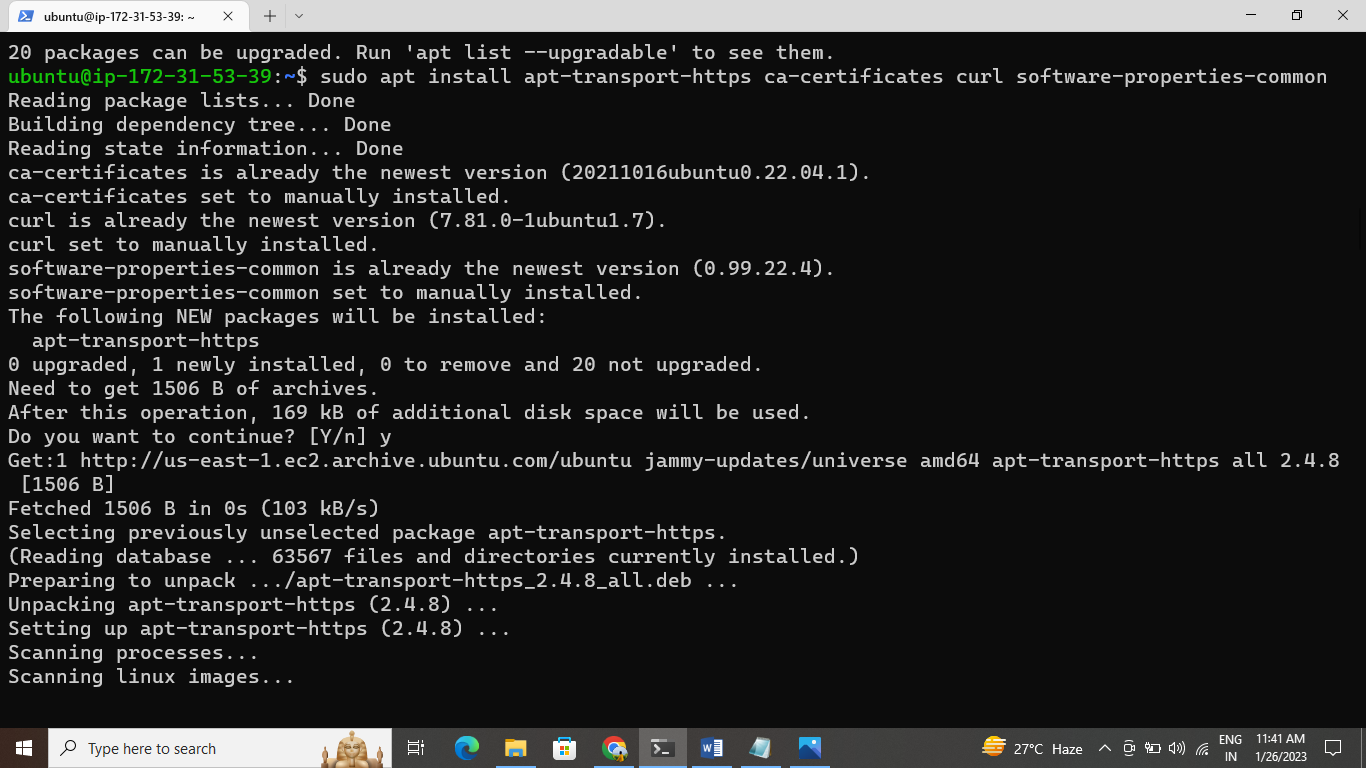


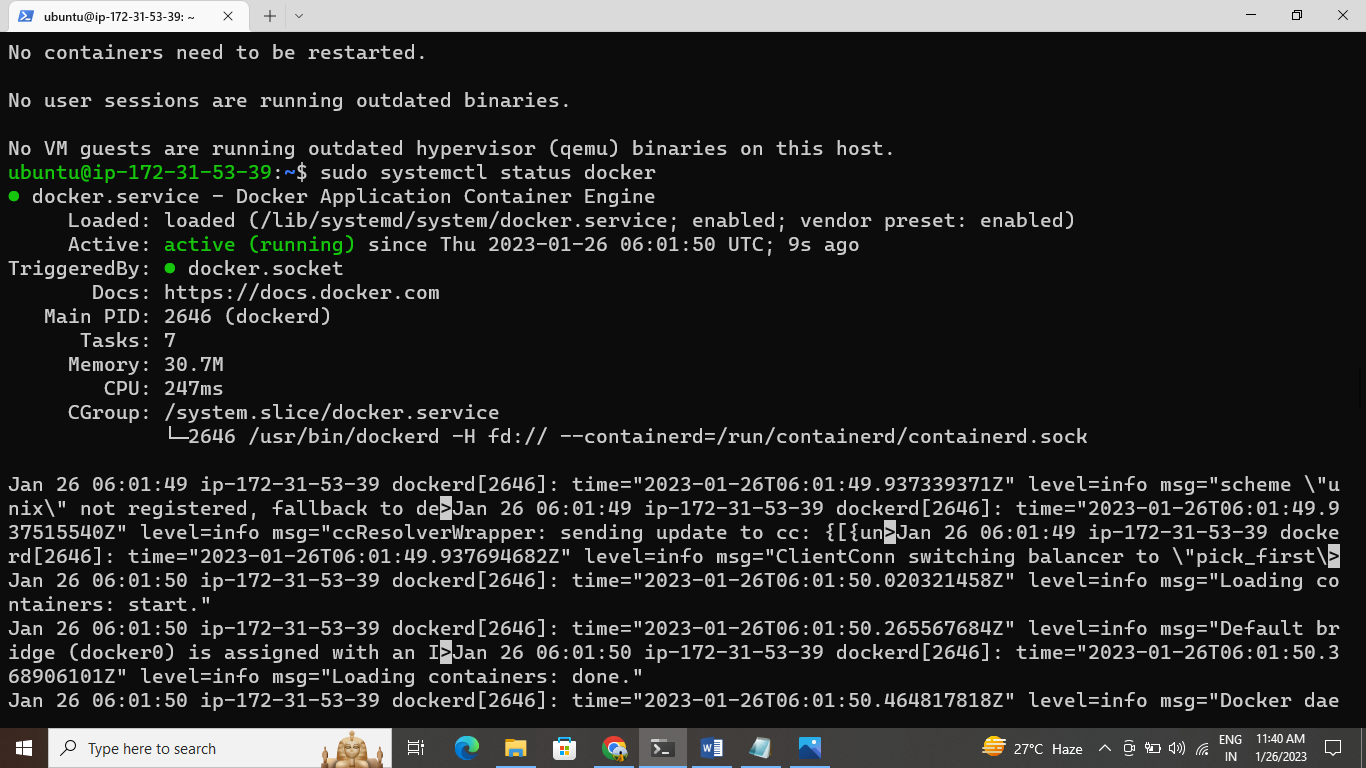
Connect to terminal

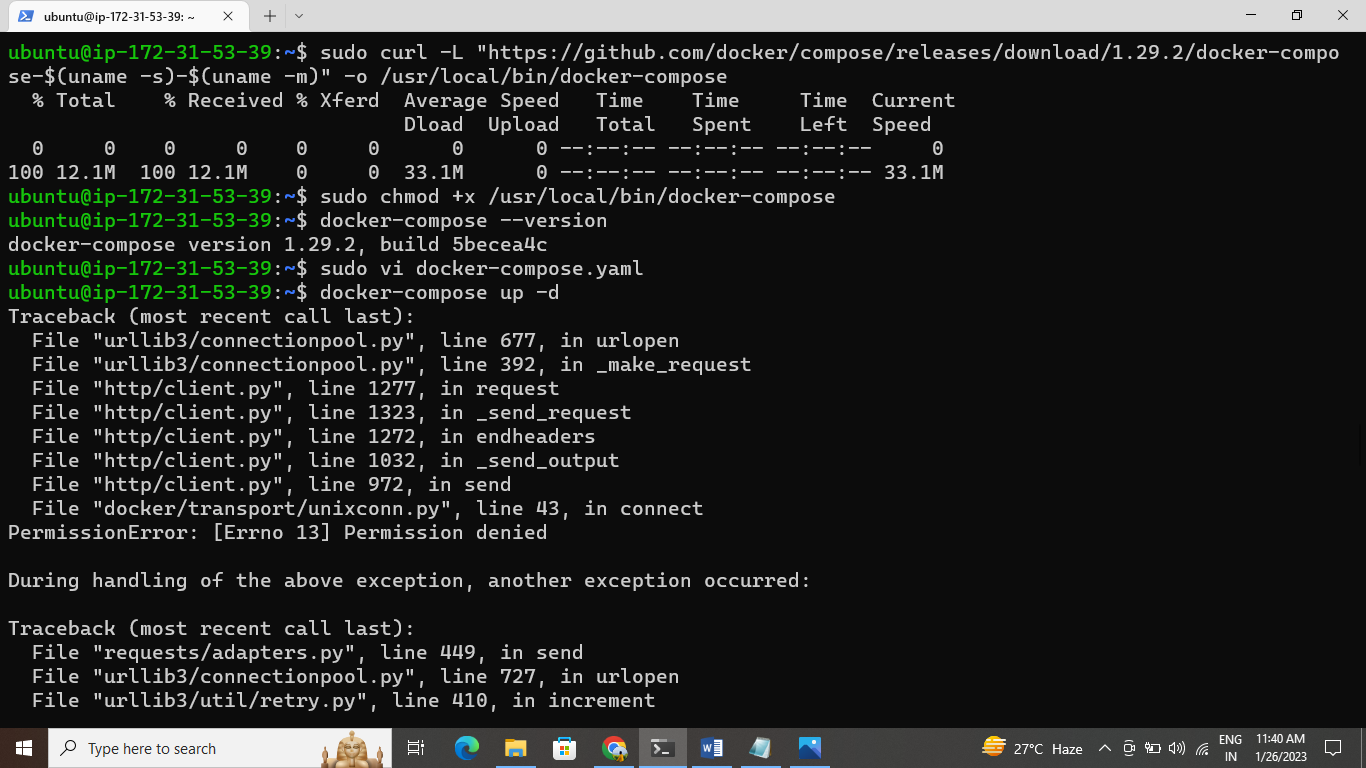


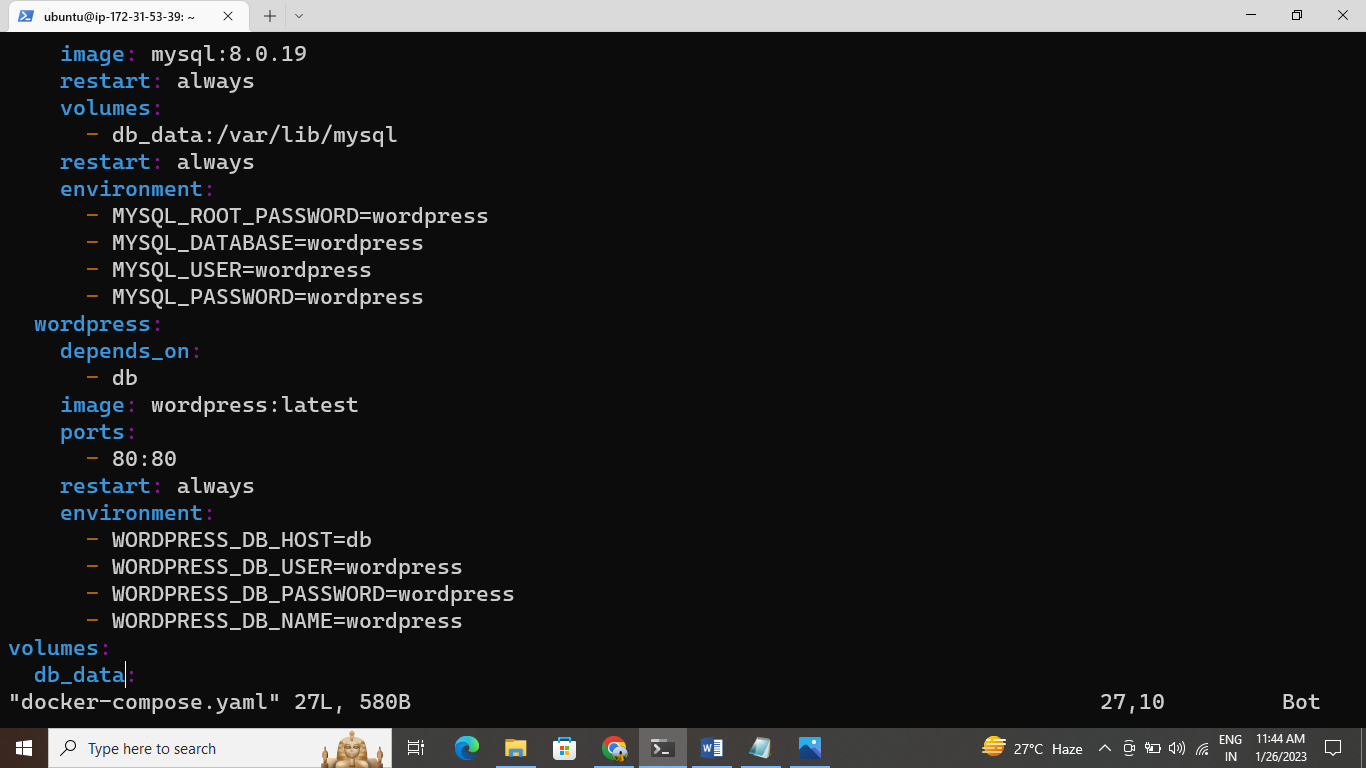
Install git in terminal (Sudo apt install git)

Install docker and docker-compose

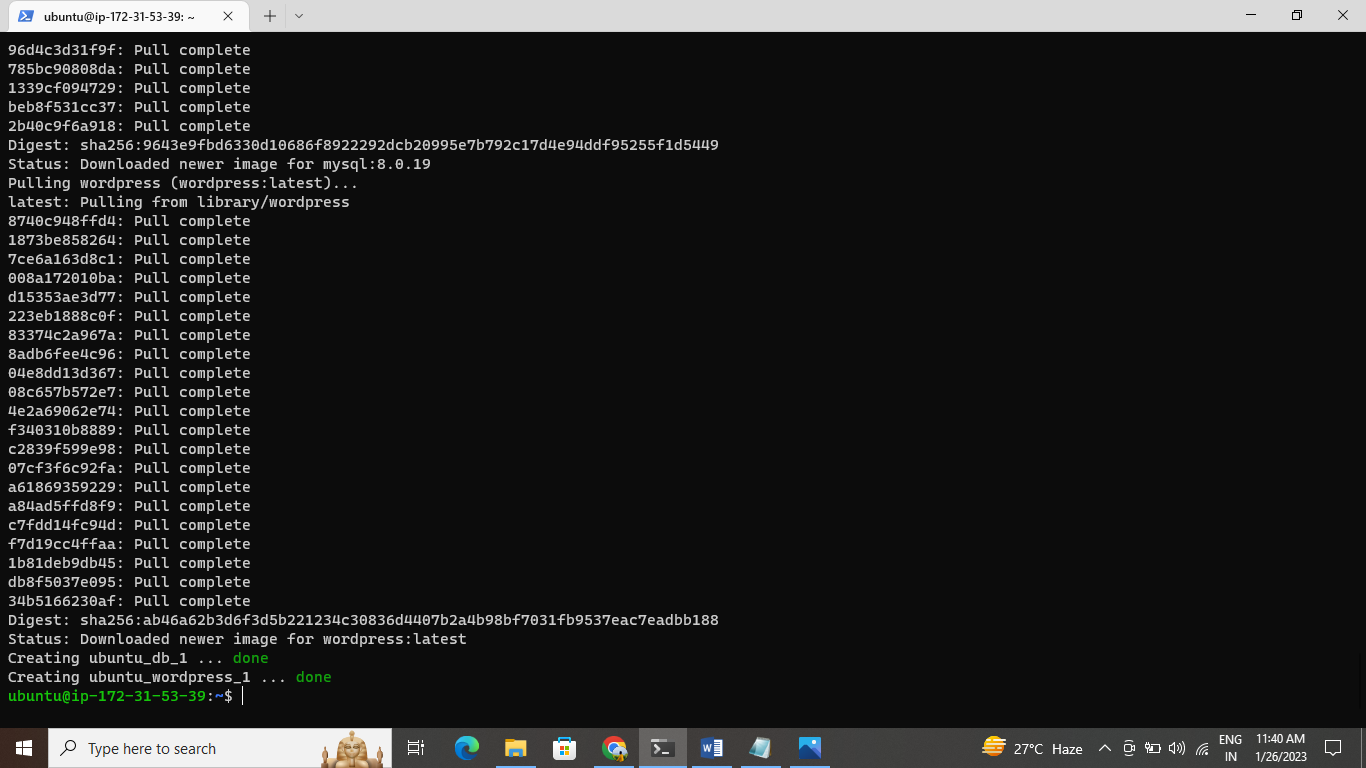












Copy the Public IPv4 address add an 8080 port browse and Jenkins open

