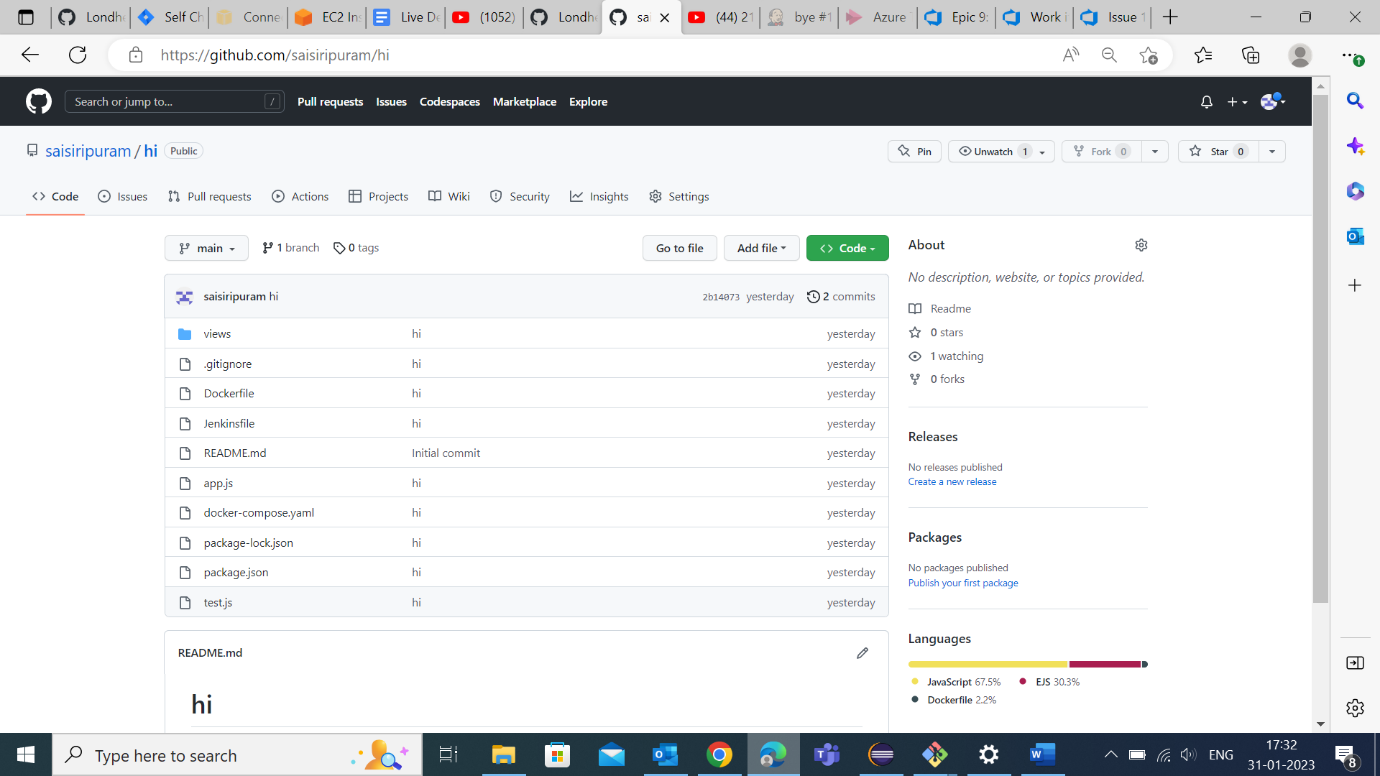
**Continuous Integration with Jenkins (**Pipeline)**:**

**1.**we need to take the demo project from google by using git clone into our local repository.

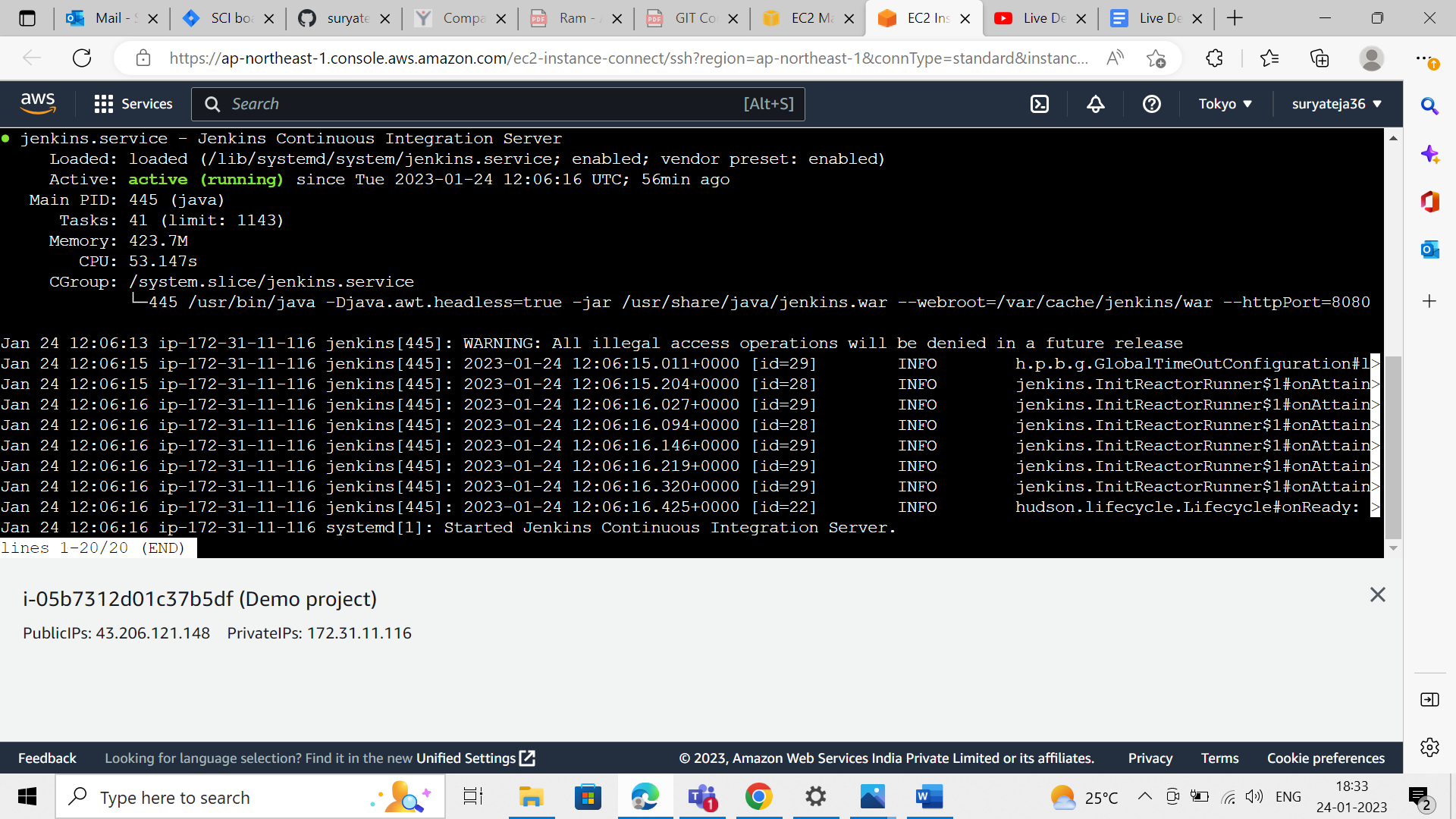
**2.** we need to push our code from local machine to git hub repository.



**Jenkins and java installation:**

1. Create and launch the AWS EC2 instance.
2. sudo apt update
3. sudo apt install openjdk-11-jre
4. java -version
5. curl -fsSL https://pkg.jenkins.io/debian/jenkins.io.key | sudo tee \   /usr/share/keyrings/jenkins-keyring.asc > /dev/null
6. echo deb [signed-by=/usr/share/keyrings/jenkins-keyring.asc] \   https://pkg.jenkins.io/debian binary/ | sudo tee \   /etc/apt/sources.list.d/jenkins.list > /dev/null
7. sudo apt-get update.
8. sudo apt-get install jenkins.
9. sudo systemctl enable jenkins
10. sudo systemctl start jenkins
11. sudo systemctl status jenkins
12. sudo cat /var/lib/jenkins/secrets/initialAdminPassword
13. History

* Once the Jenkins are up and running, in order to access we need to create the inbound security rule on 8080 port.

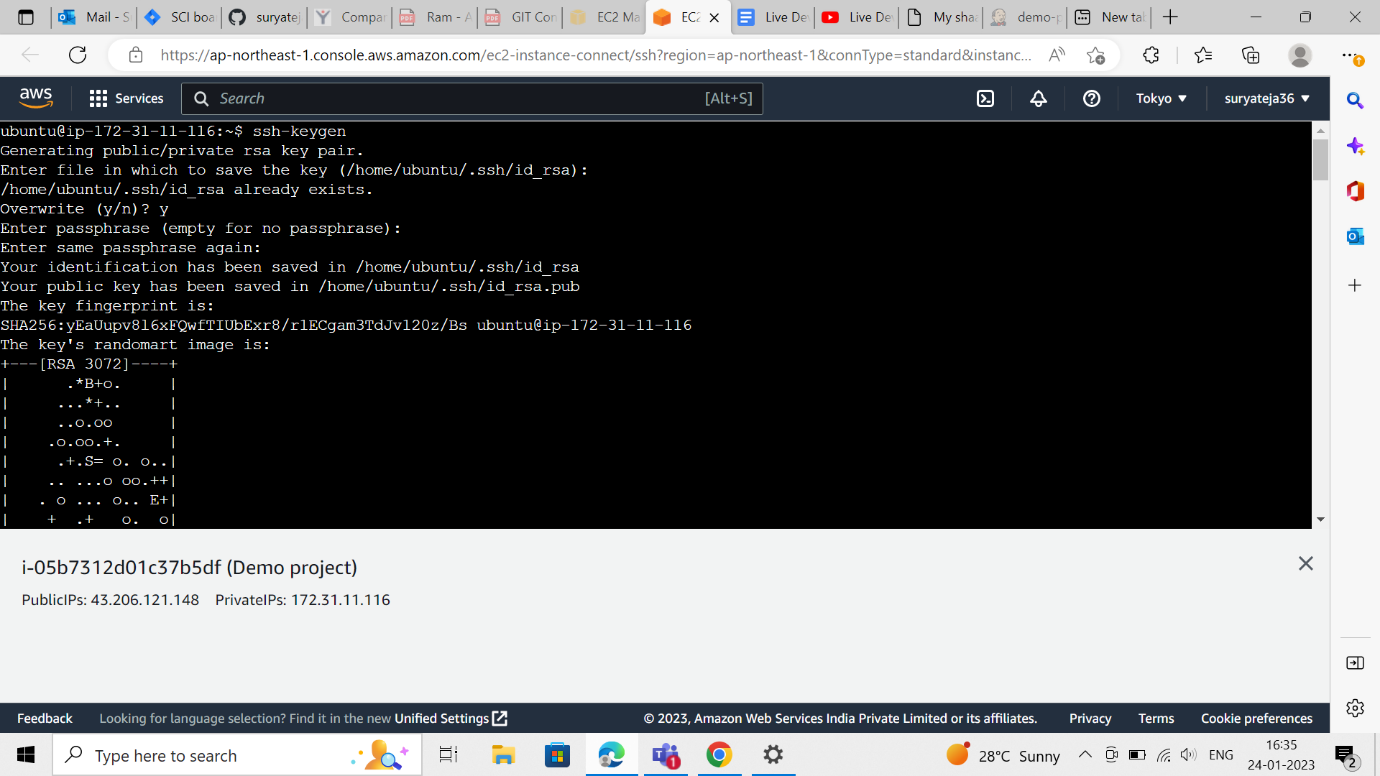


* Public Ip (:8080)

Graphical user interface, text, application

Description automatically generated

* Create the admin user with name and password.
* Now ready to start the jenkins.
* Create a job in Jenkins enter an item name by using freestyle project.
* Integrate the GitHub and Jenkins by providing the credentials.
* Paste our GitHub URL in GitHub project option and source code management repositories.
* **Follow the below steps are execute in to EC2 instance for generate the public and private key for GitHub integration.**
* SSH-keygen and 3 enters.



* Cd .SSh and “ls”
* Cat id \_rsa (for private key)
* Cat id \_rsa.pub (for public key)

A screenshot of a computer

Description automatically generated

* go to settings and select (ssh and GPG) keys in dashboard.
* now we have connected to GitHub with ssh public key.

* **Source code management (git):**
* paste the repository URL in git and add the credentials fill the private key.
* Now run the Build and Will get the console output (success/failed.)

Graphical user interface, text, application, email

Description automatically generated

* We need to check whether the Jenkins integrated with GitHub or not.
* After integration we need to execute the commands like:
* sudo cat /var/lib/jenkins/secrets/initialAdminPassword.
* To run the applications, we need to install these.

Sudo apt install node.js

Sudo apt install npm

Sudo npm install

Node app.js

* We need to create the inbound rule on 8000 ports to give the public access.
* Then the application will be executed , If we exit from instance then the application will be disappeared .
* So, to resolve that we need run or integrate our project in docker.
* We need to install docker and create the docker file.
* So, we need to execute the below commands.
* sudo apt install docker.io
* FROM node:12.2.0-alpine
* WORKDIR app
* COPY.
* RUN npm install.
* EXPOSE 8000
* CMD ["node","app.js"]
* docker build. -t node-app
* sudo usermod -a -G docker $USER
* docker run -d --name node-todo-app -p 8000:8000 todo-node-app.
* After that we need do,

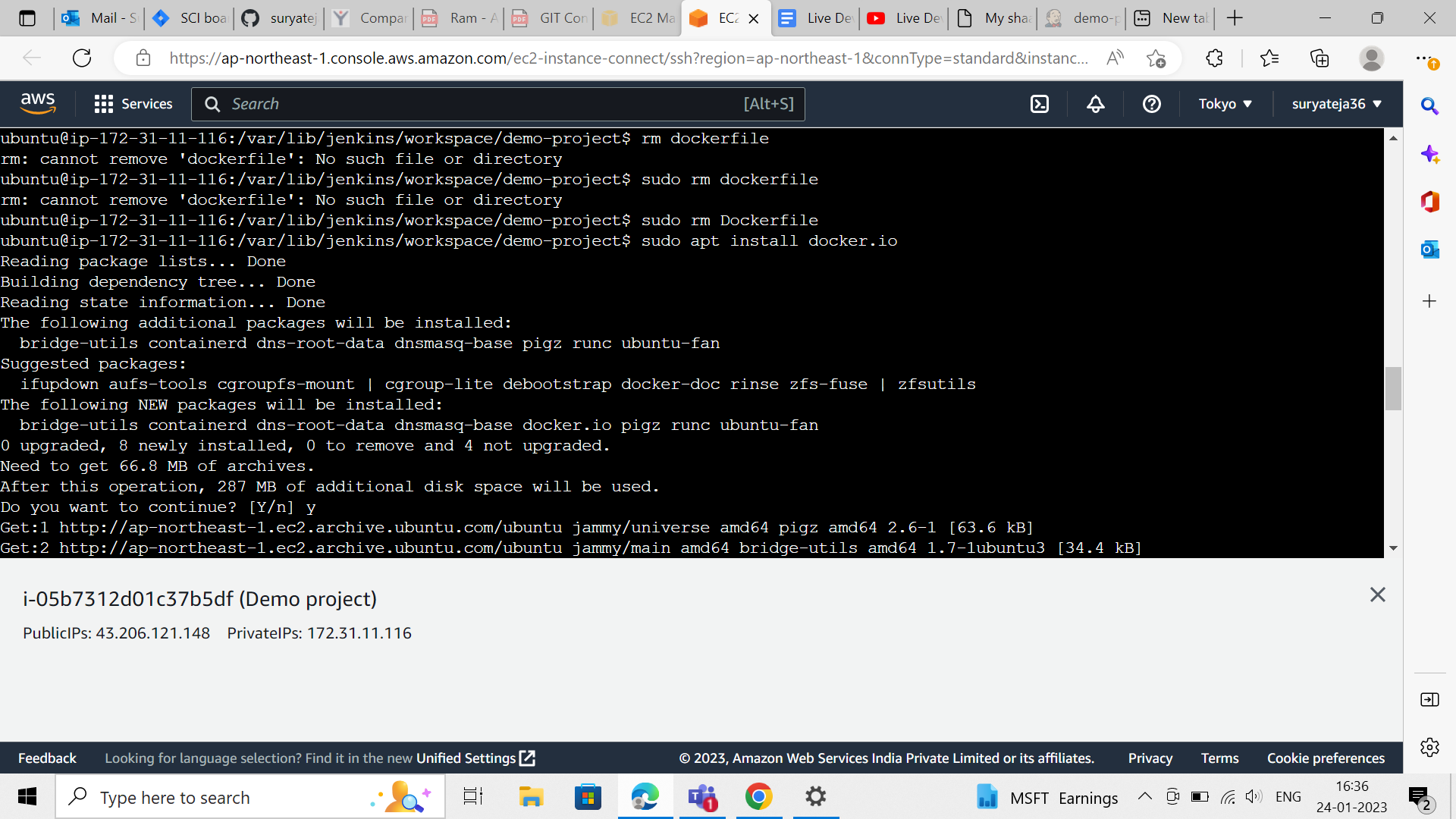
Go to jenkins job.

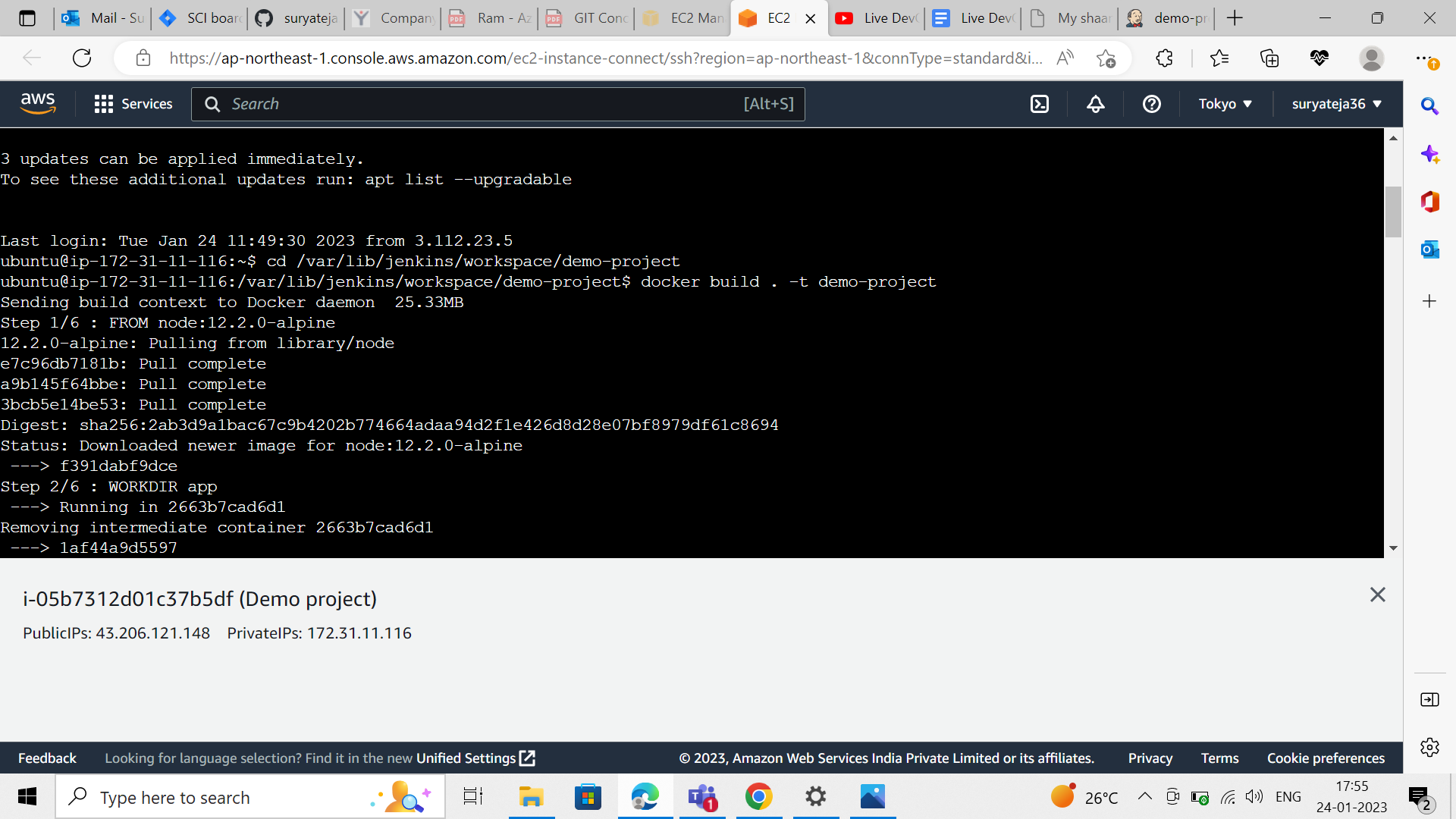
Execute shell.

docker build -t node-app-todo.

docker run -d --name node-app-container -p 8000:8000 node-app-todo.

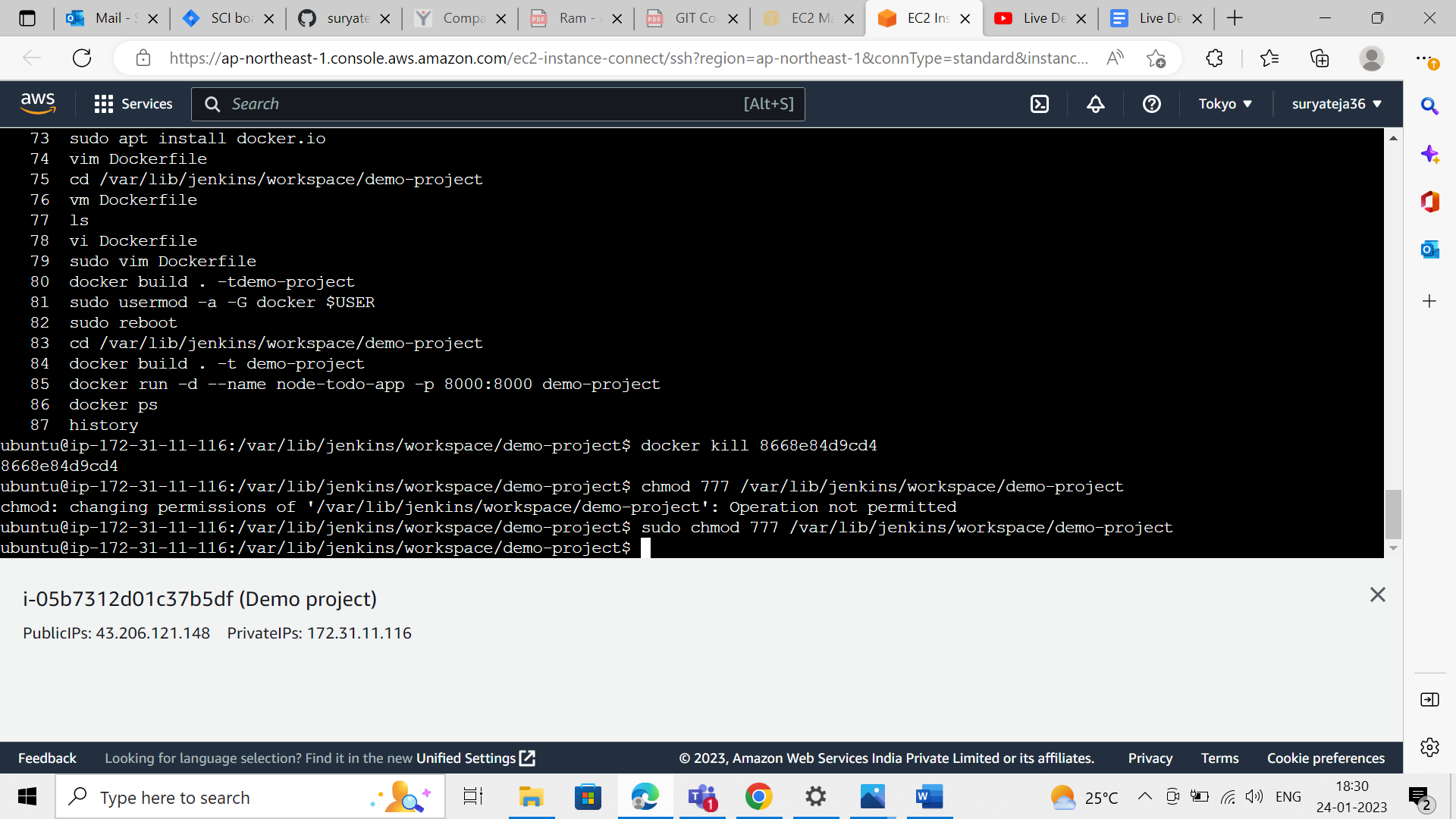
* Docker commands execution





A screenshot of a computer

Description automatically generated



A screenshot of a computer

Description automatically generated