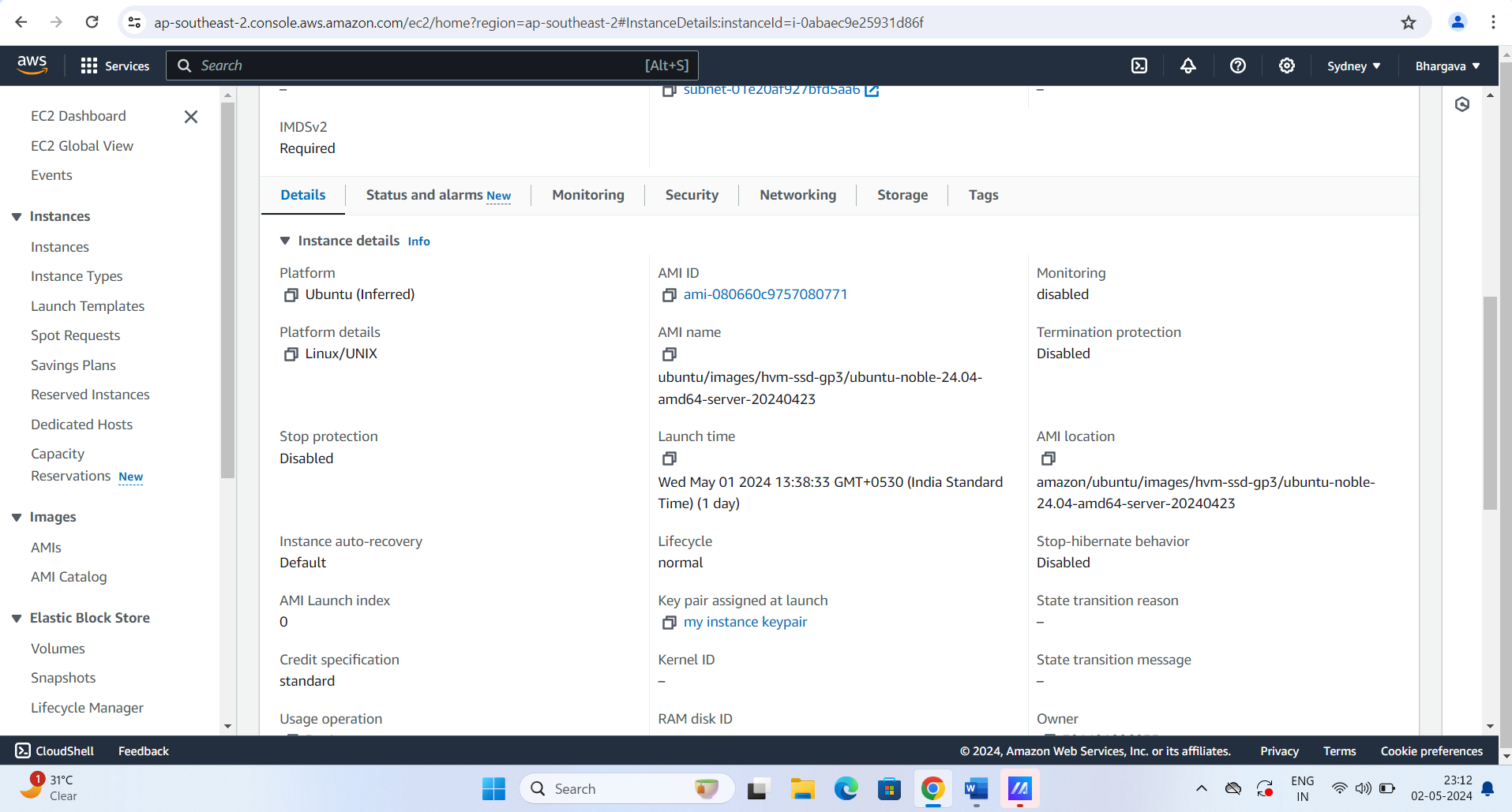
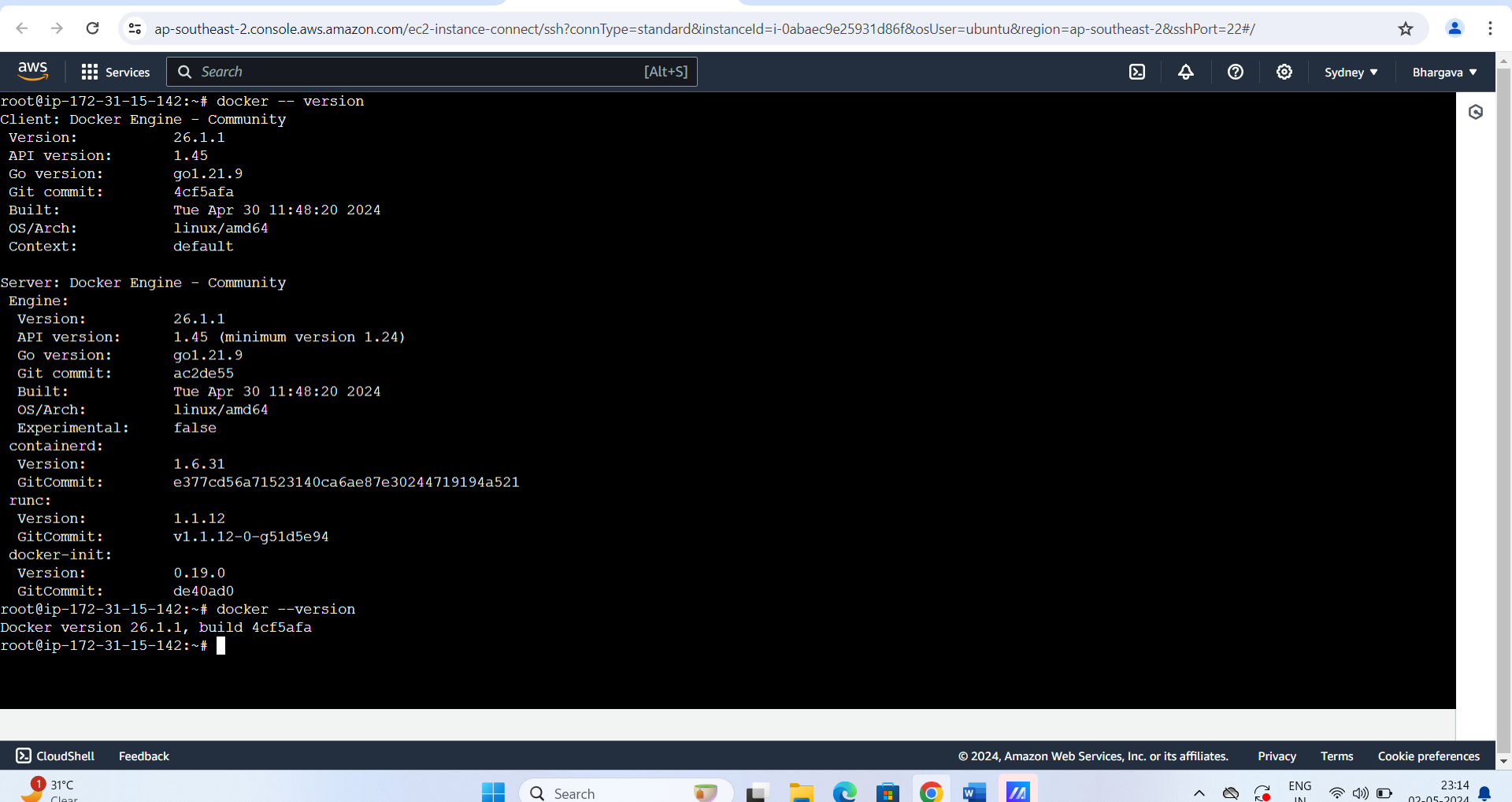
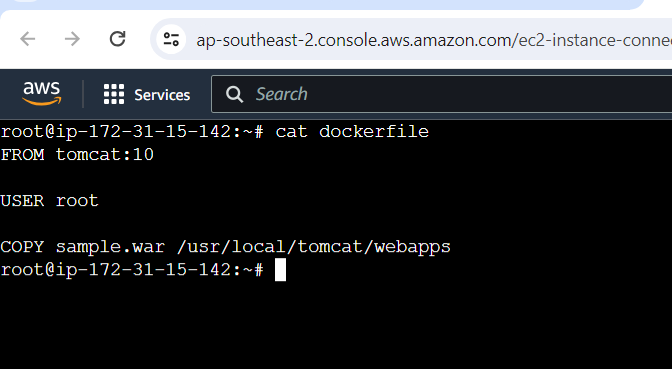
My Assignment

Provisioned an Ubuntu 24.04 instance in AWS



Install the docker on the above ubuntu machine , please refer the docker version

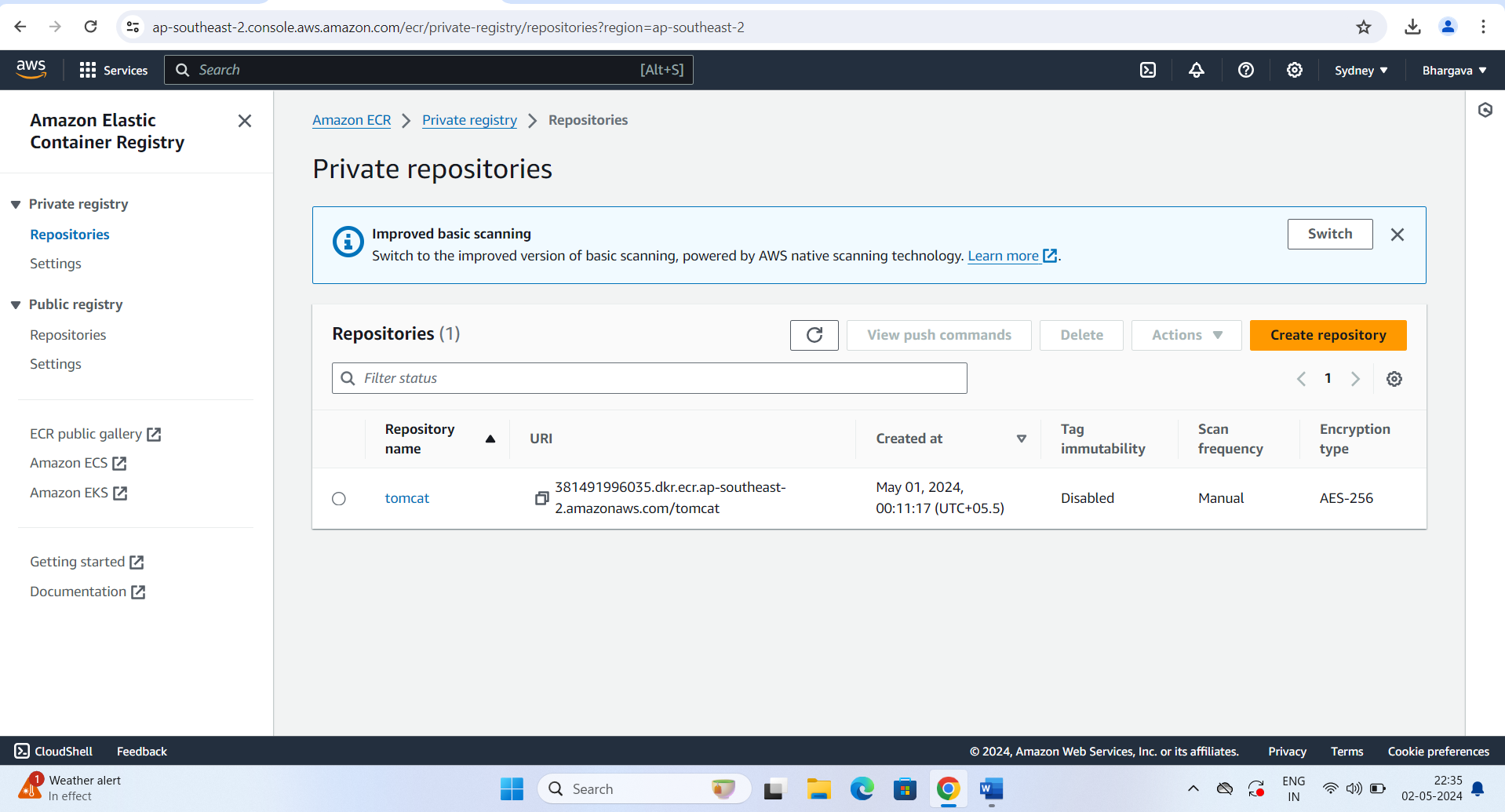
Create a docker file with tomcat base 10 image and copied a sample.war app file



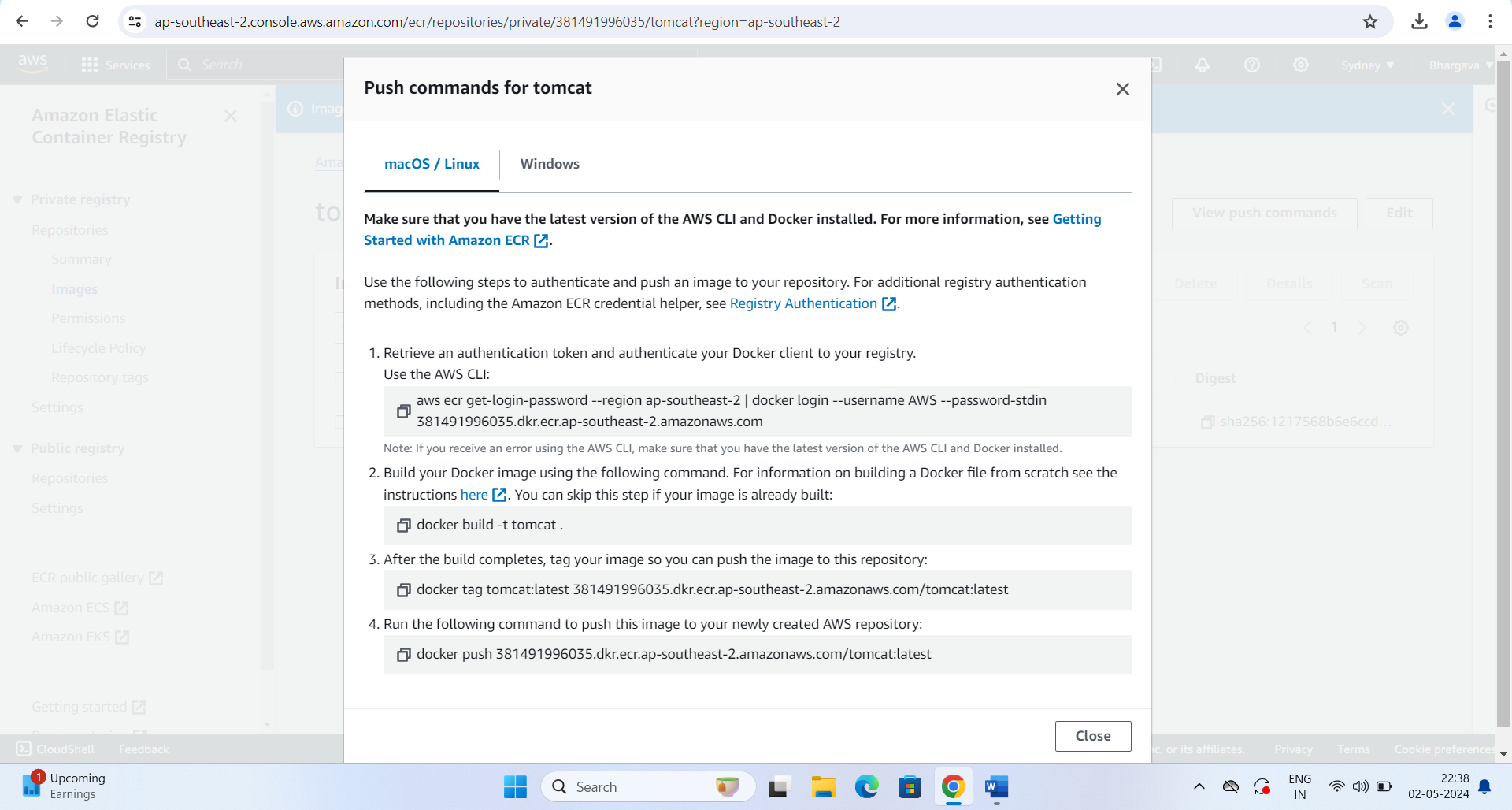
By using docker file we build a docker image by using the command **docker build -t tomcat10:latest**

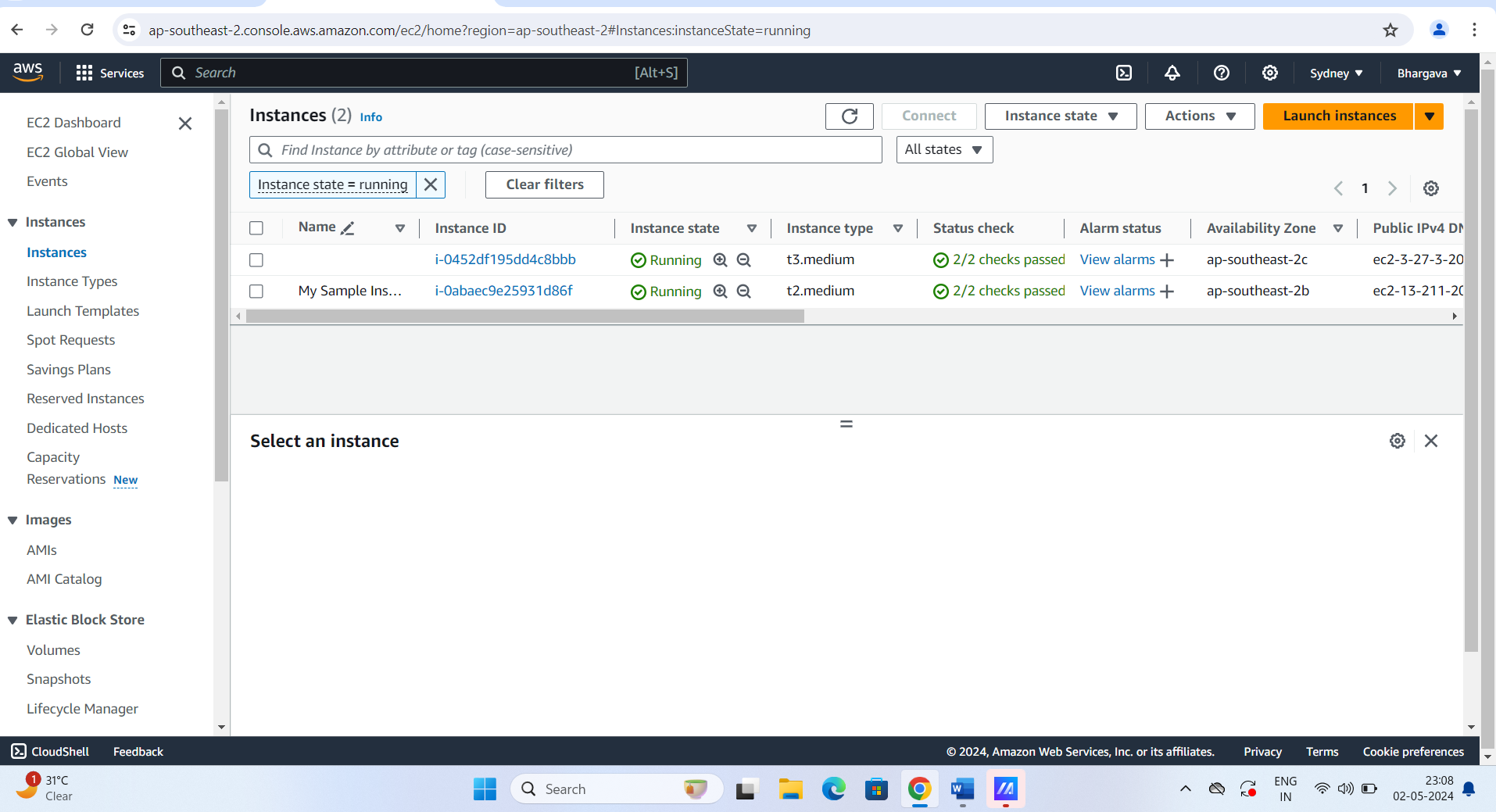
Please refer the below for the image details

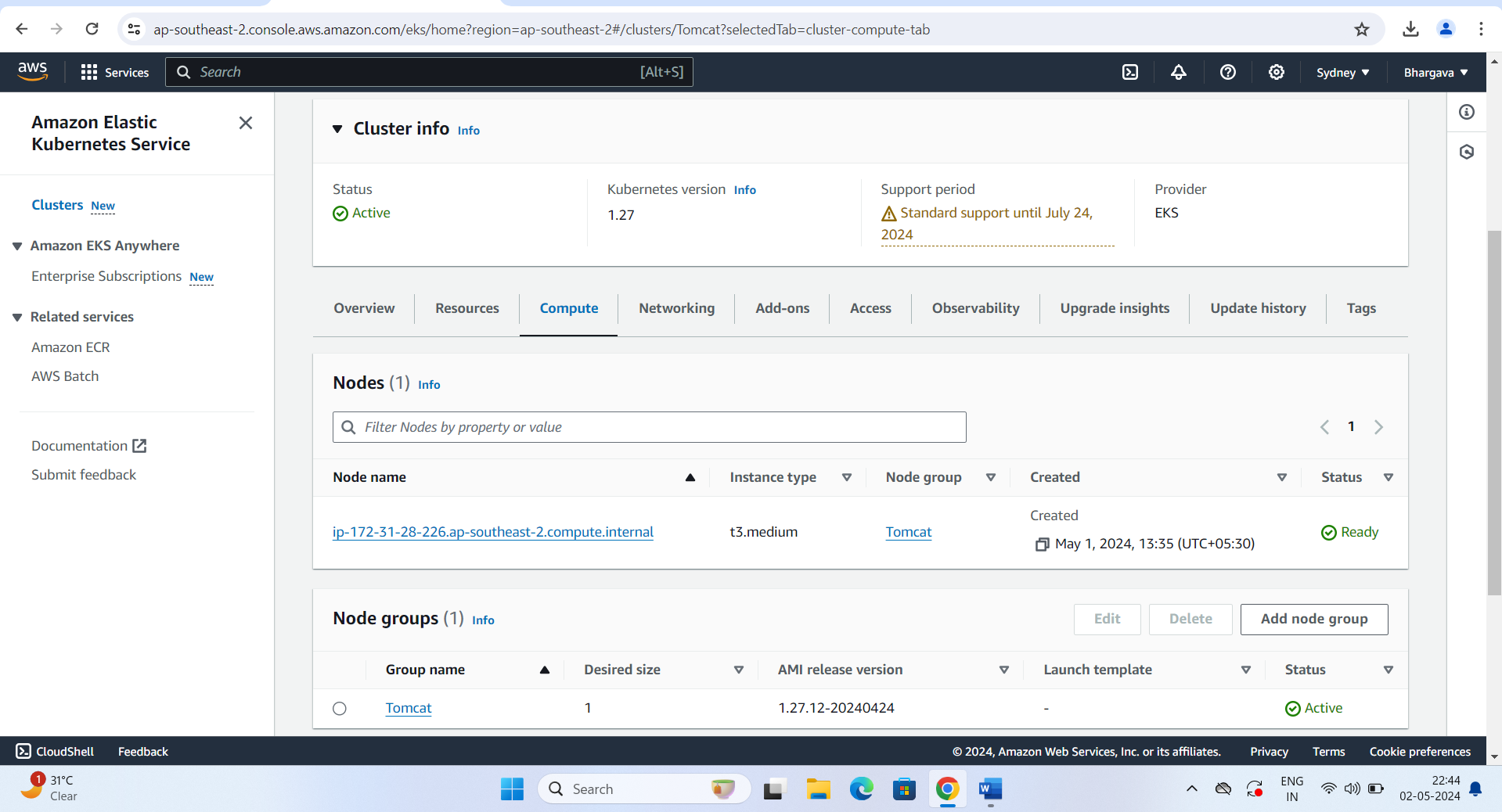
Navigate to ECR and create a repository (tomcat name) in ECR registry



Upon successfully repository creation, push the docker image to repository that we have created by using the below push commands



Navigate to EKS and create a EKS cluster (Tomcat) with role EKS cluster policy and then add the node groups with EC2 container register ReadOnly policy, Amazon EKS\_CNI\_Policy and Amazon EKS Workernode policy. Please refer below for worker node and cluster details 

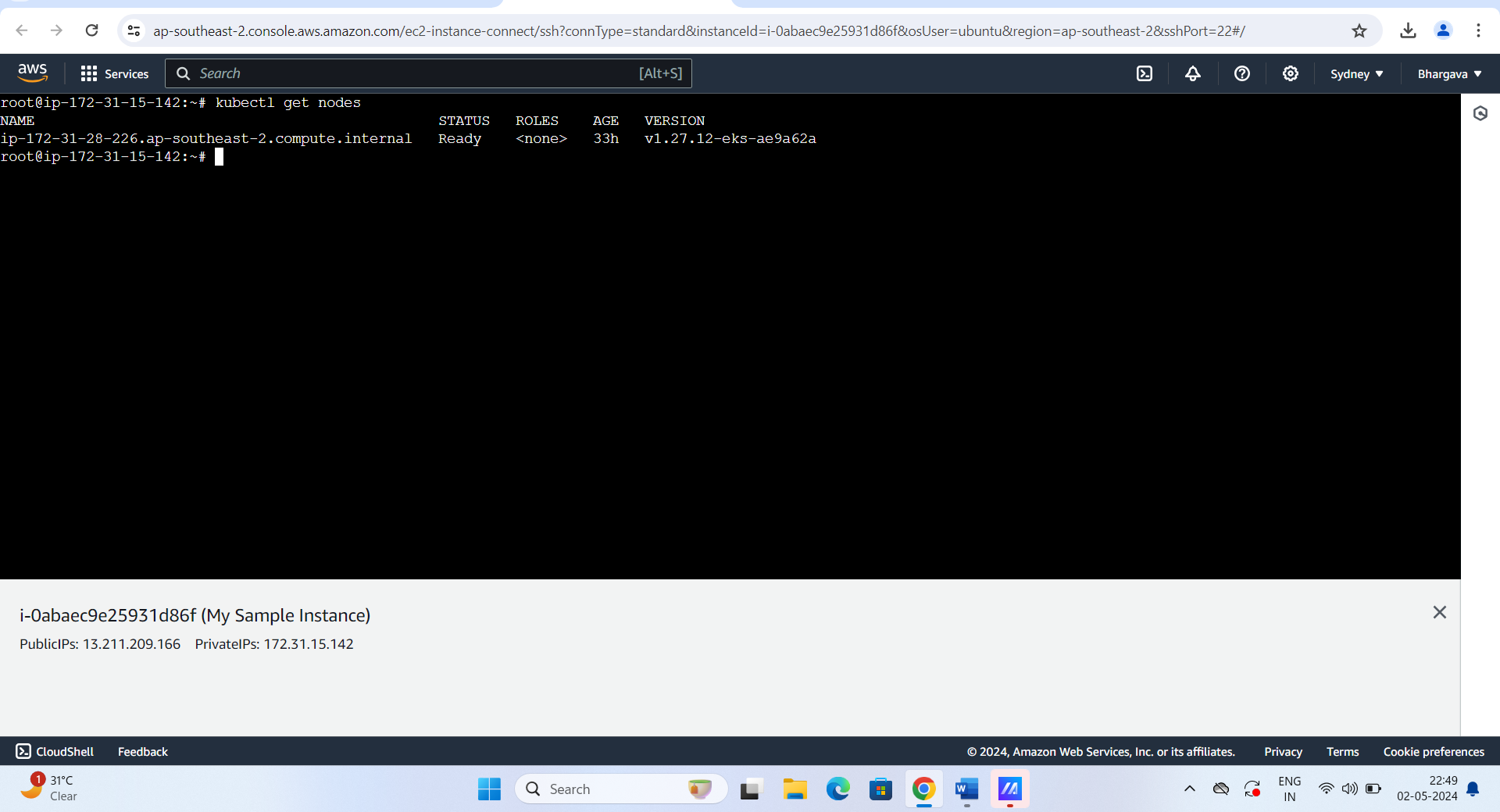


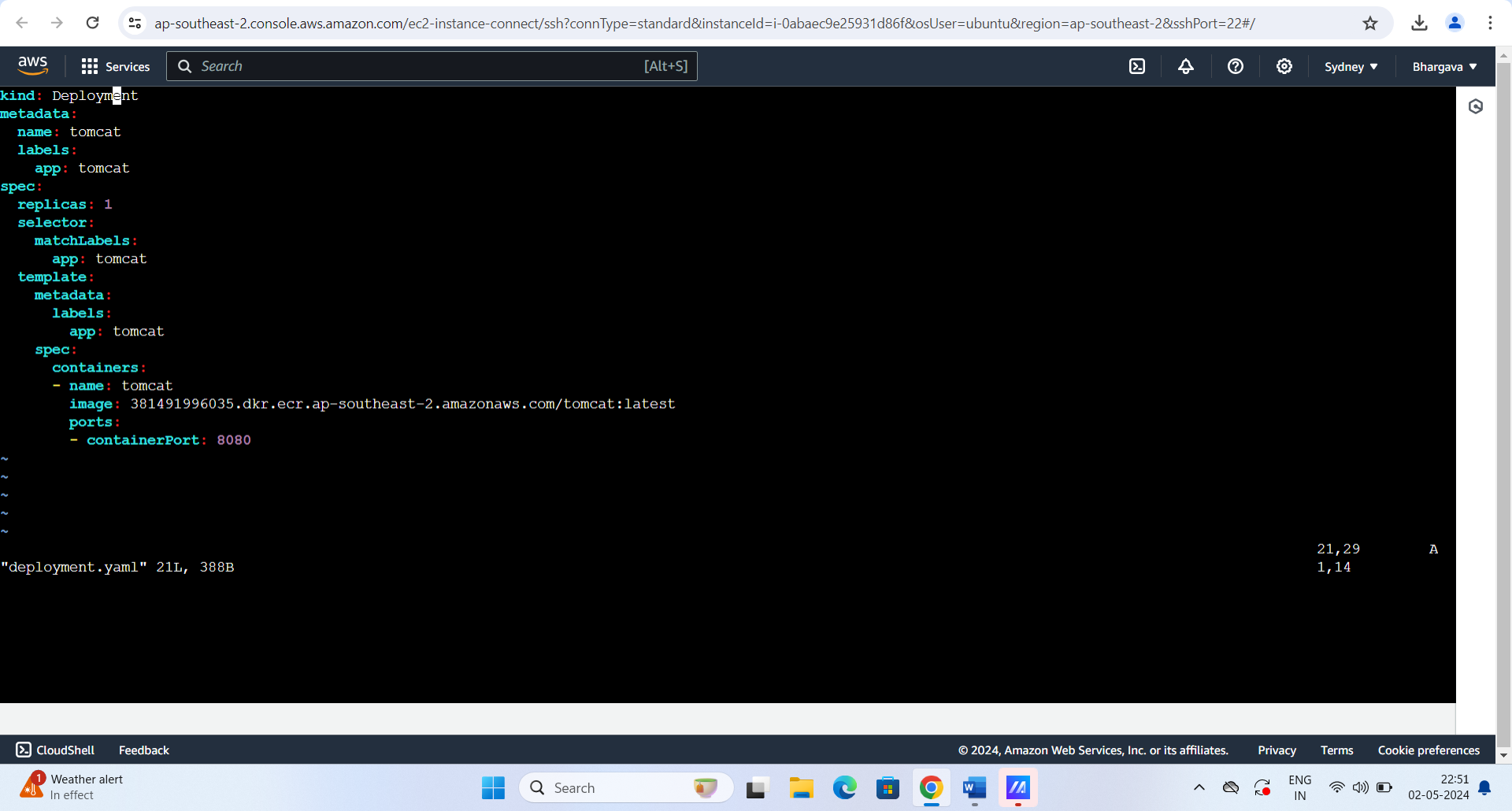
Connect to cluster using AWS CLI

AWS configure

Update the cluster using the kubeconfig by using the command

**aws eks update-kubeconfig - -region ap-southeast-2 - -name tomcat**

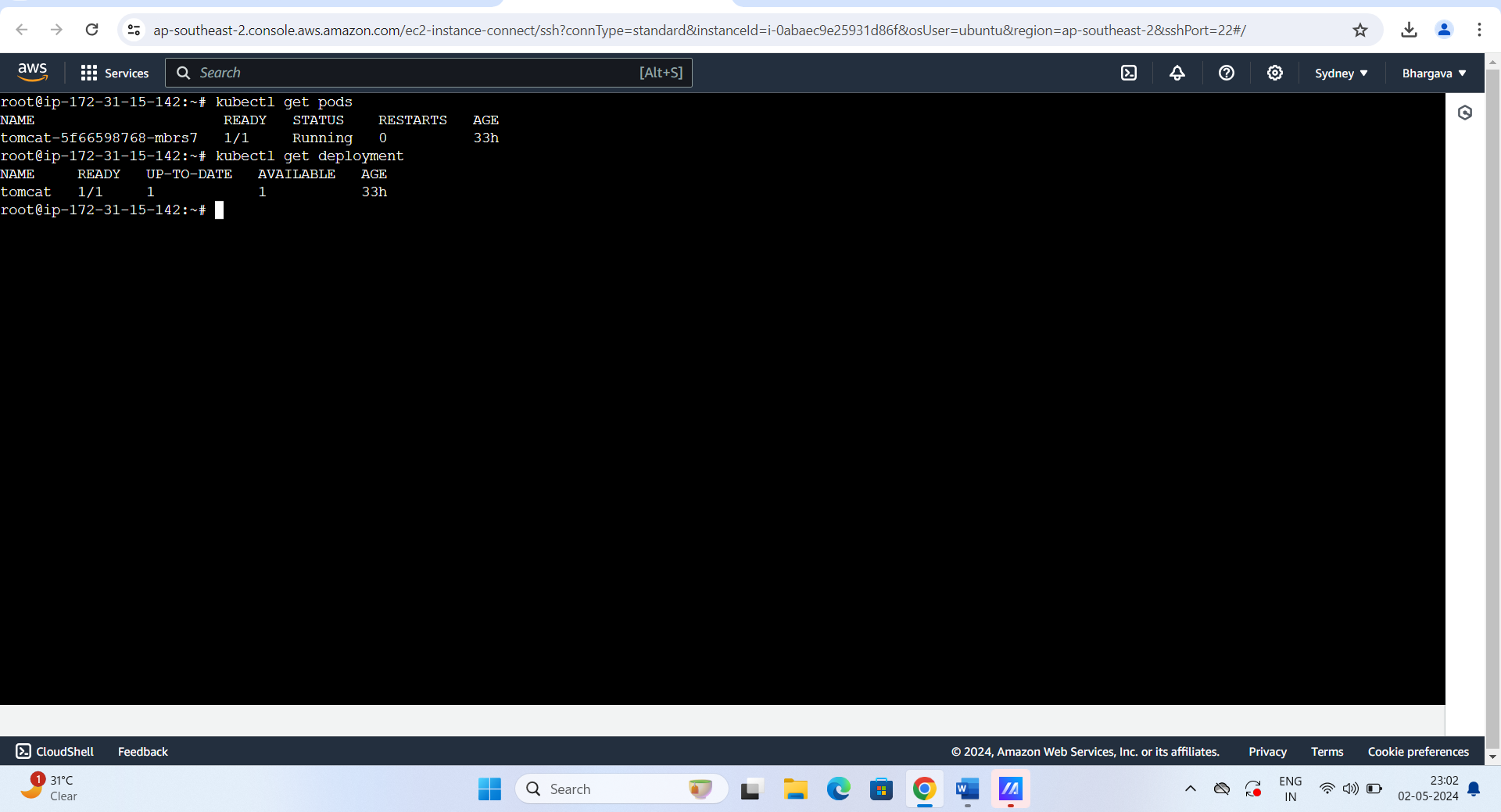
now we connect to cluster through CLI 

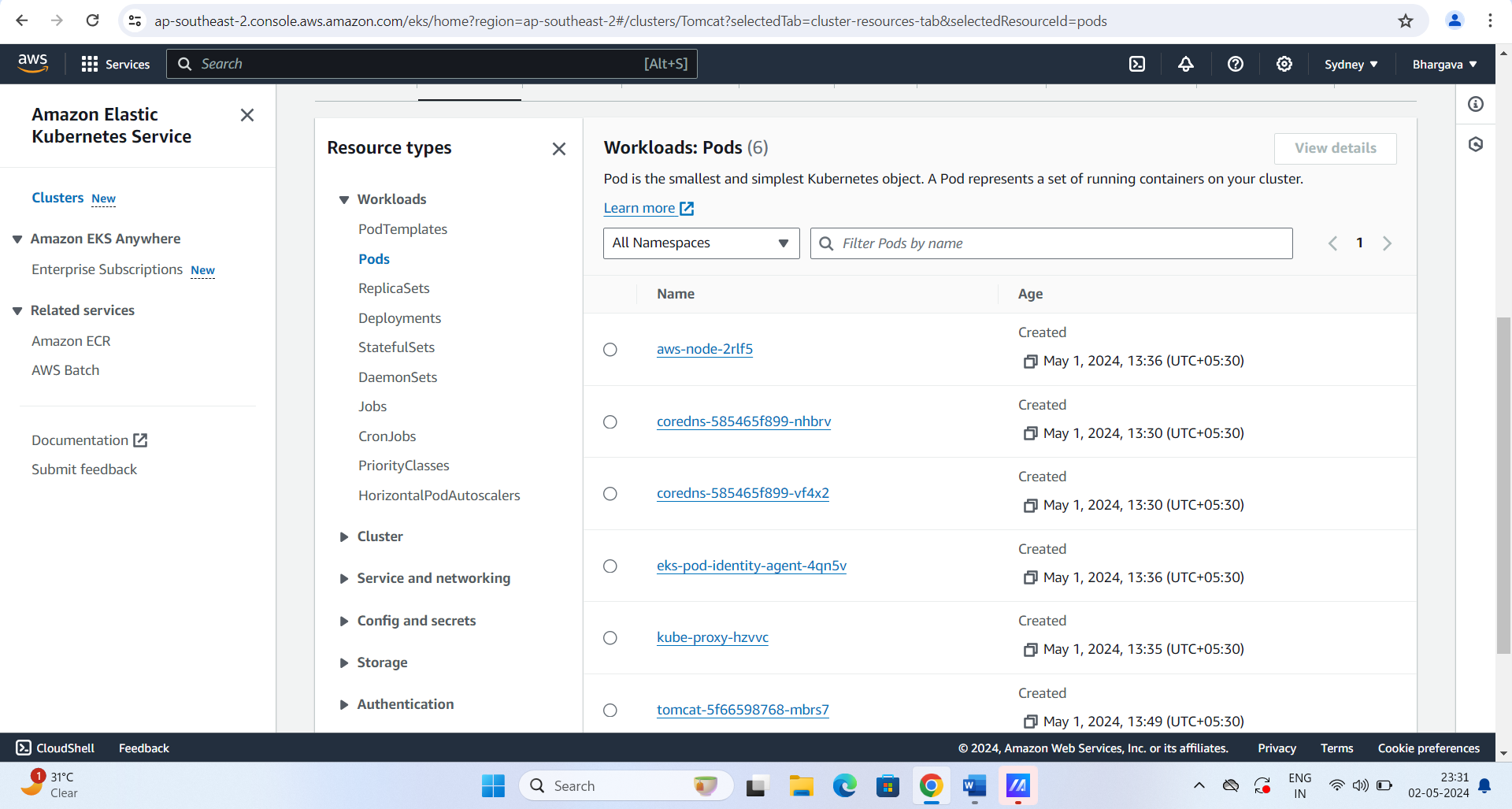
for deployment I have created a deployment.yaml manifest file

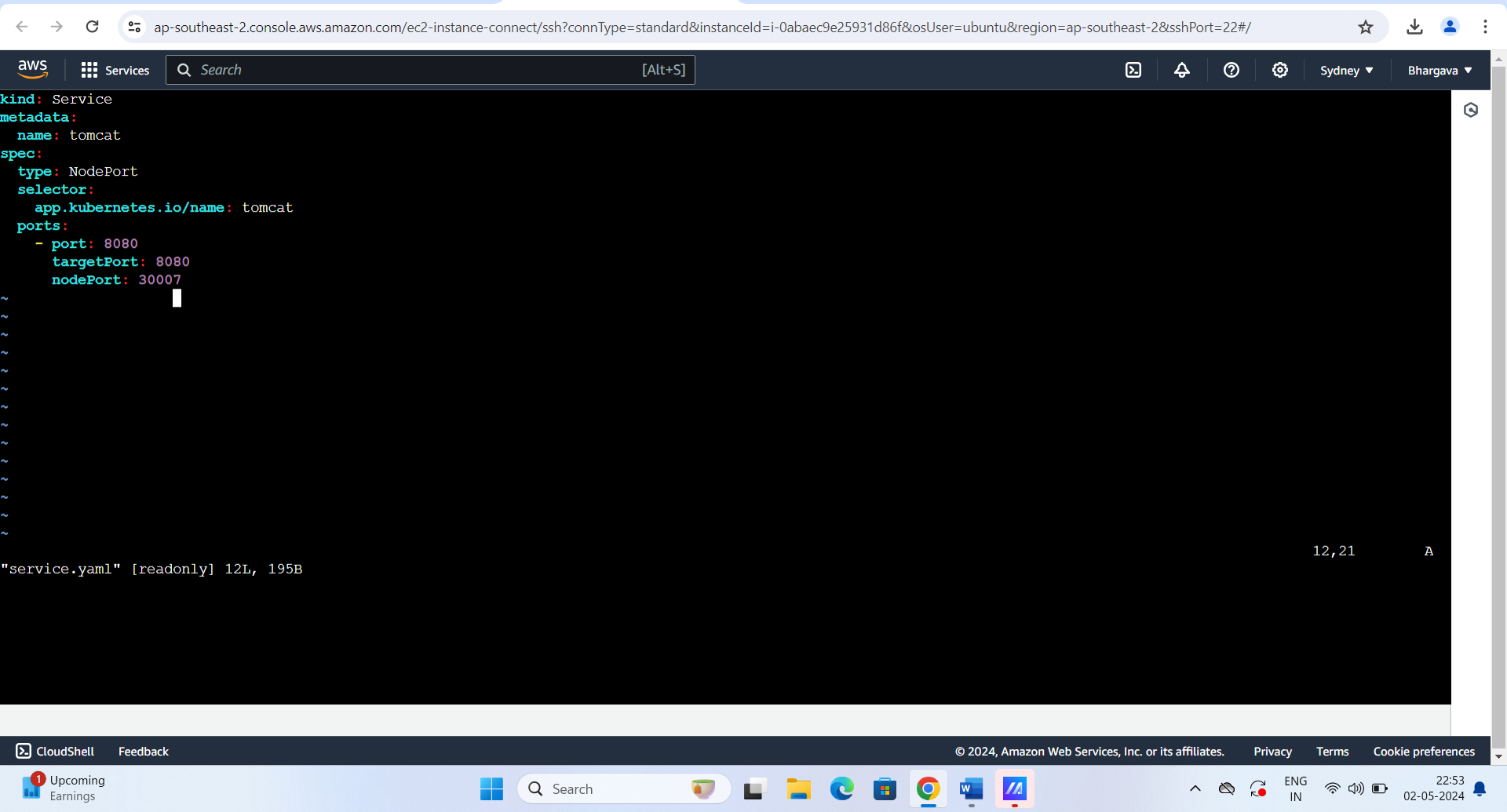
To deploy above manifest file below is the command used

Kubectl apply -f deployment.yaml -n default

Respective pod and deployment will got created , please refer below



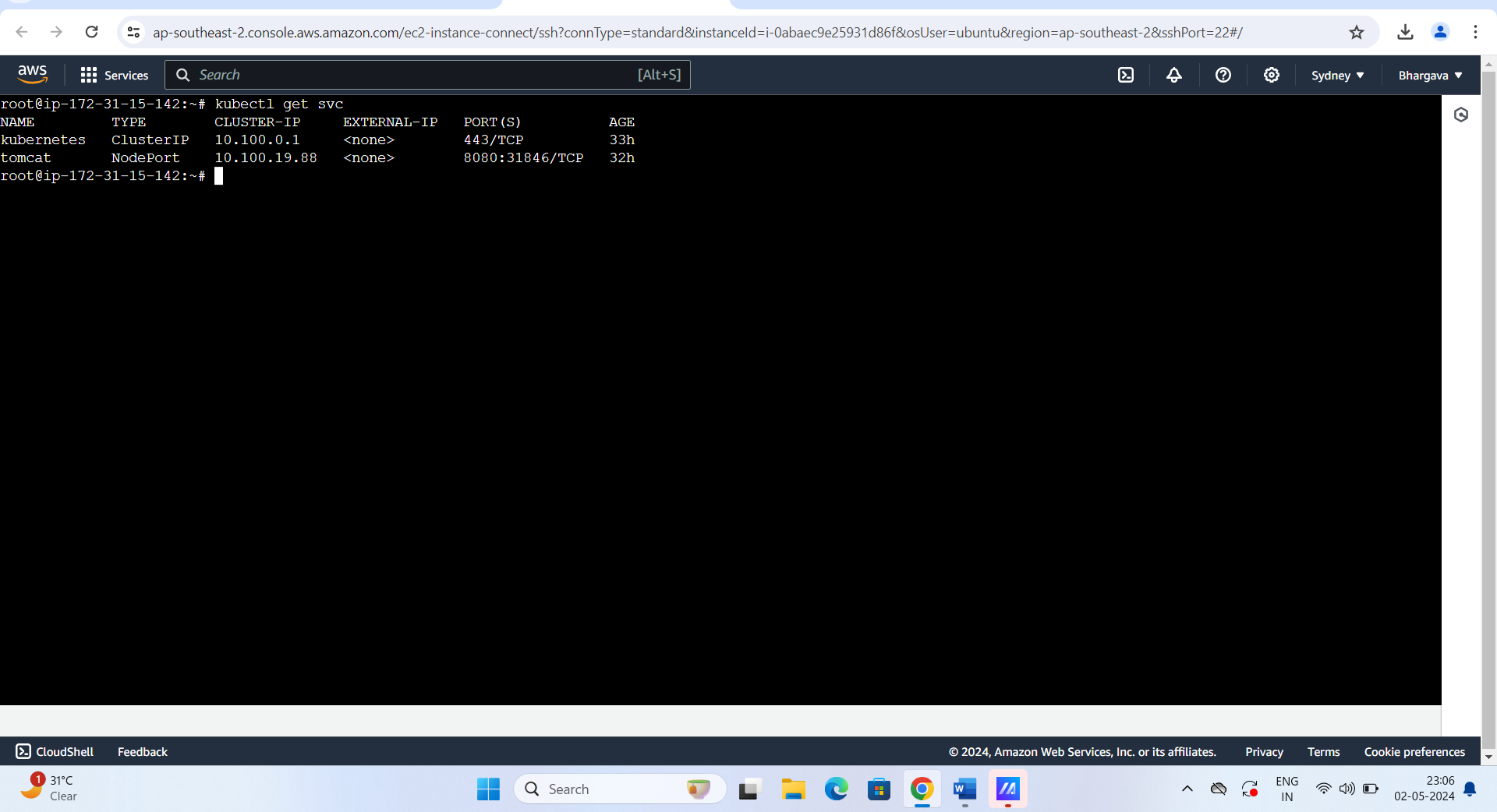


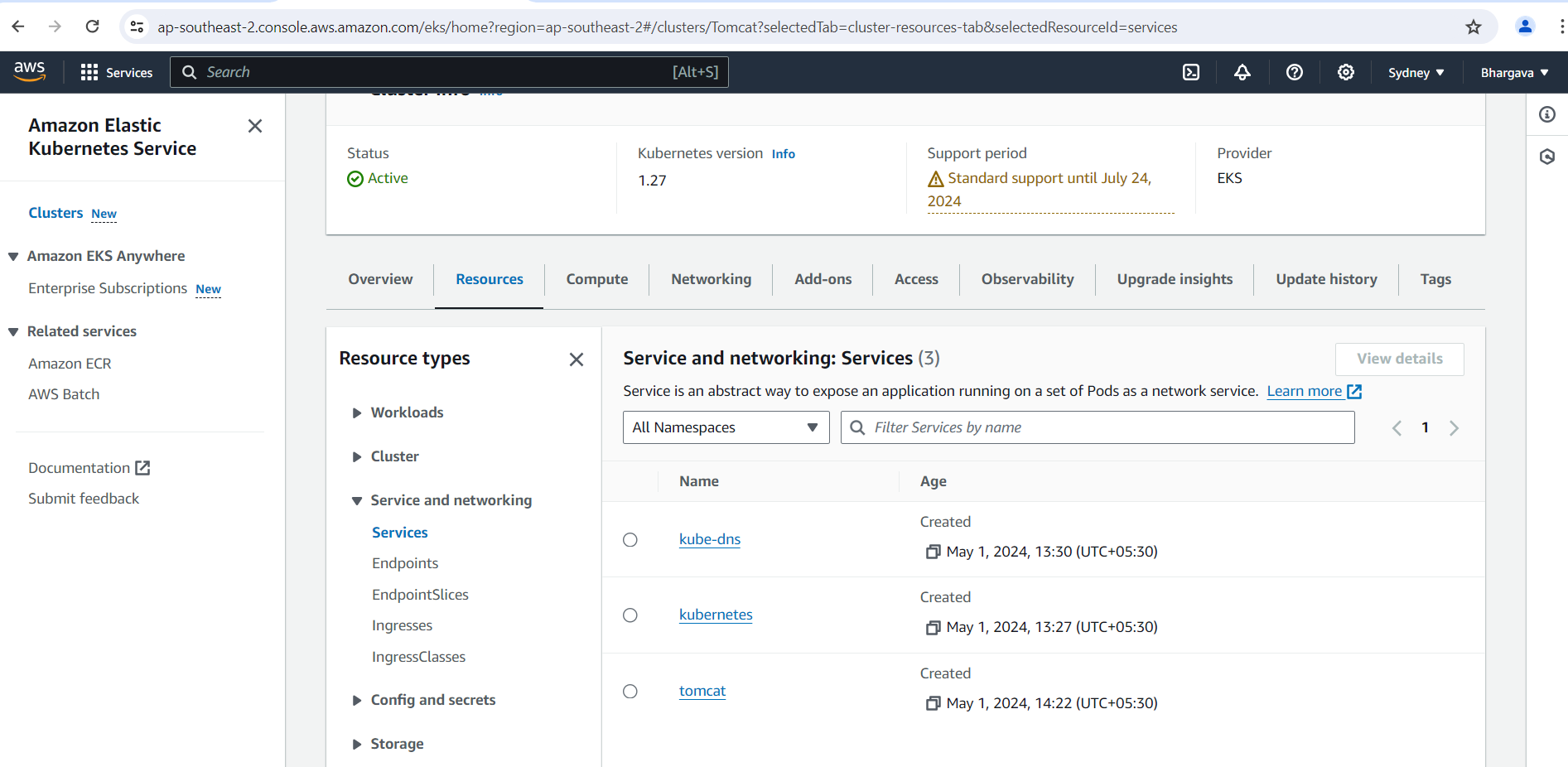
for accessing the application created a service(nodeport) manifest file

Kubectl apply -f service.yaml -n default

kubectl expose deployment tomcat --type=NodePort --name=tomcat

Please refer the below for node port details





Finally, we can access the application by using worker node public ip:nodeport/sample please refer below

