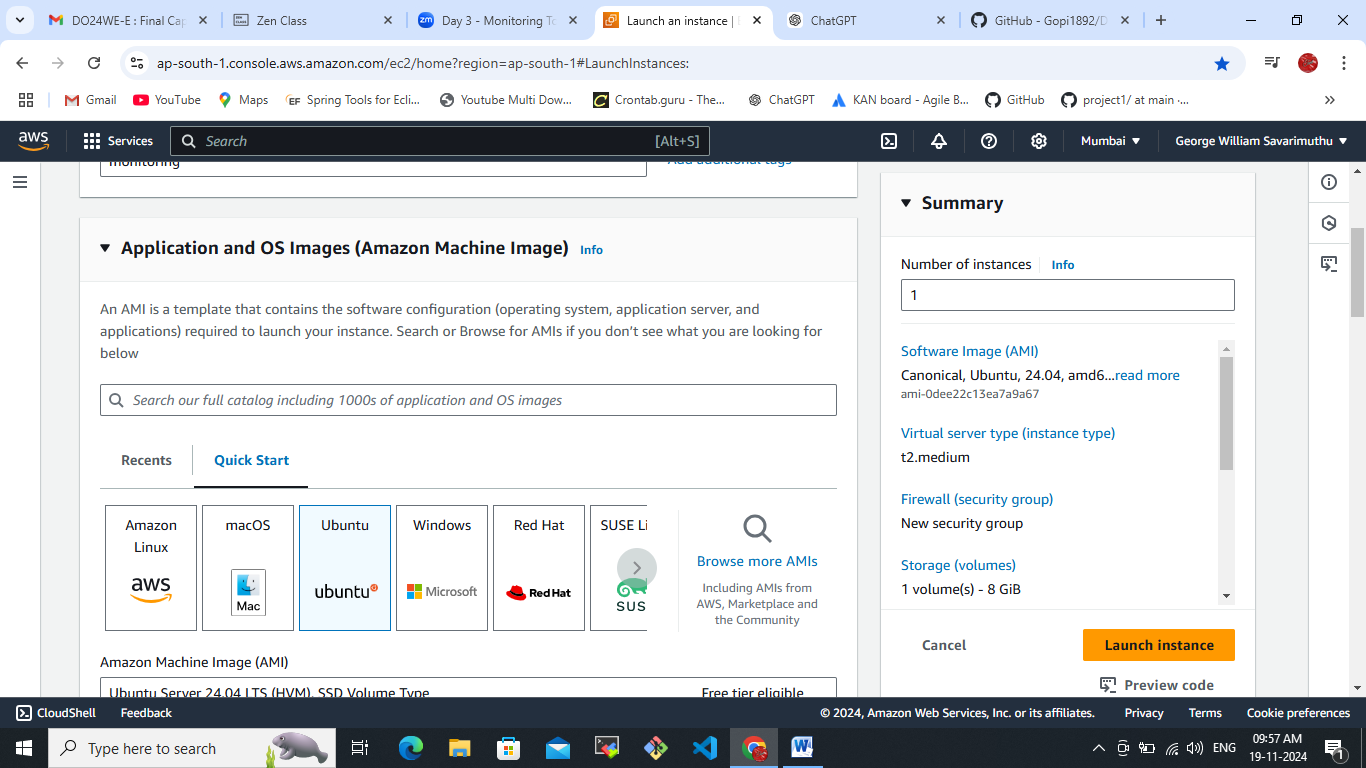
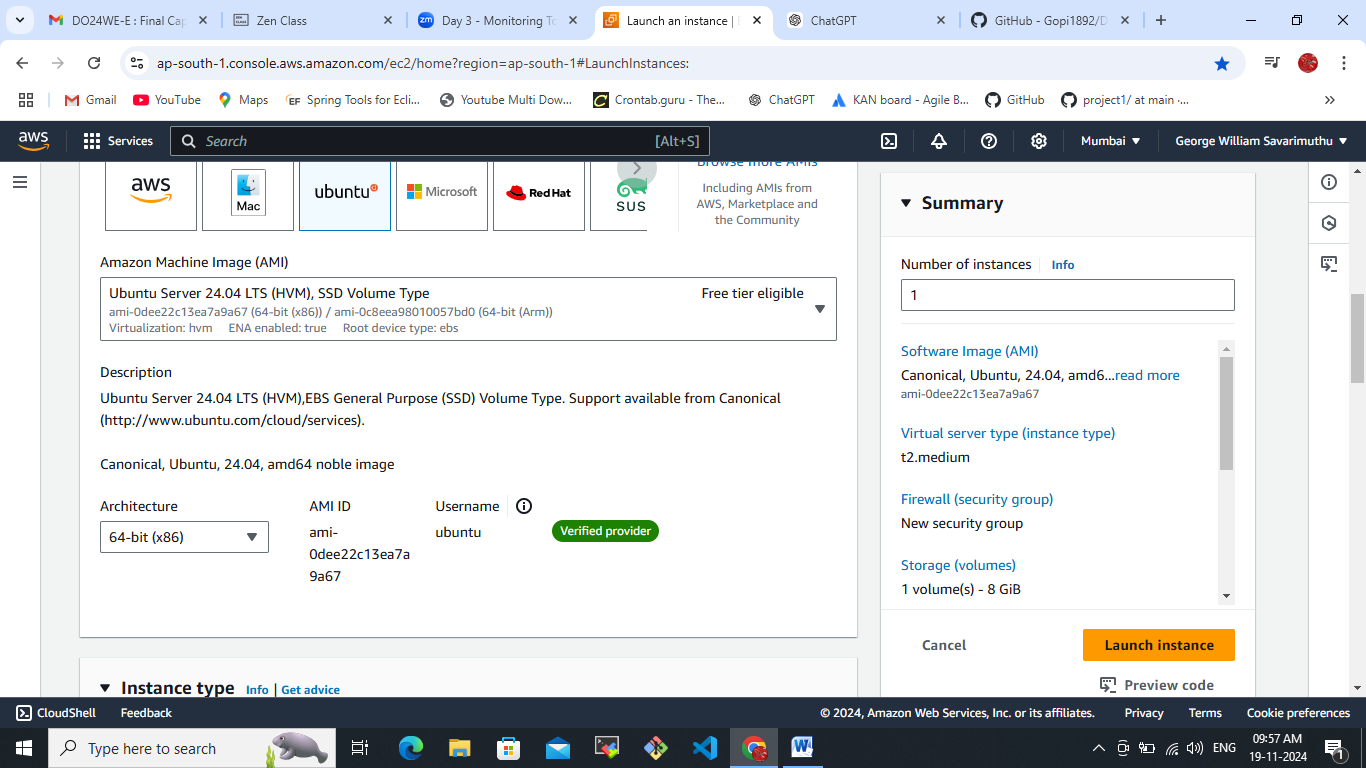
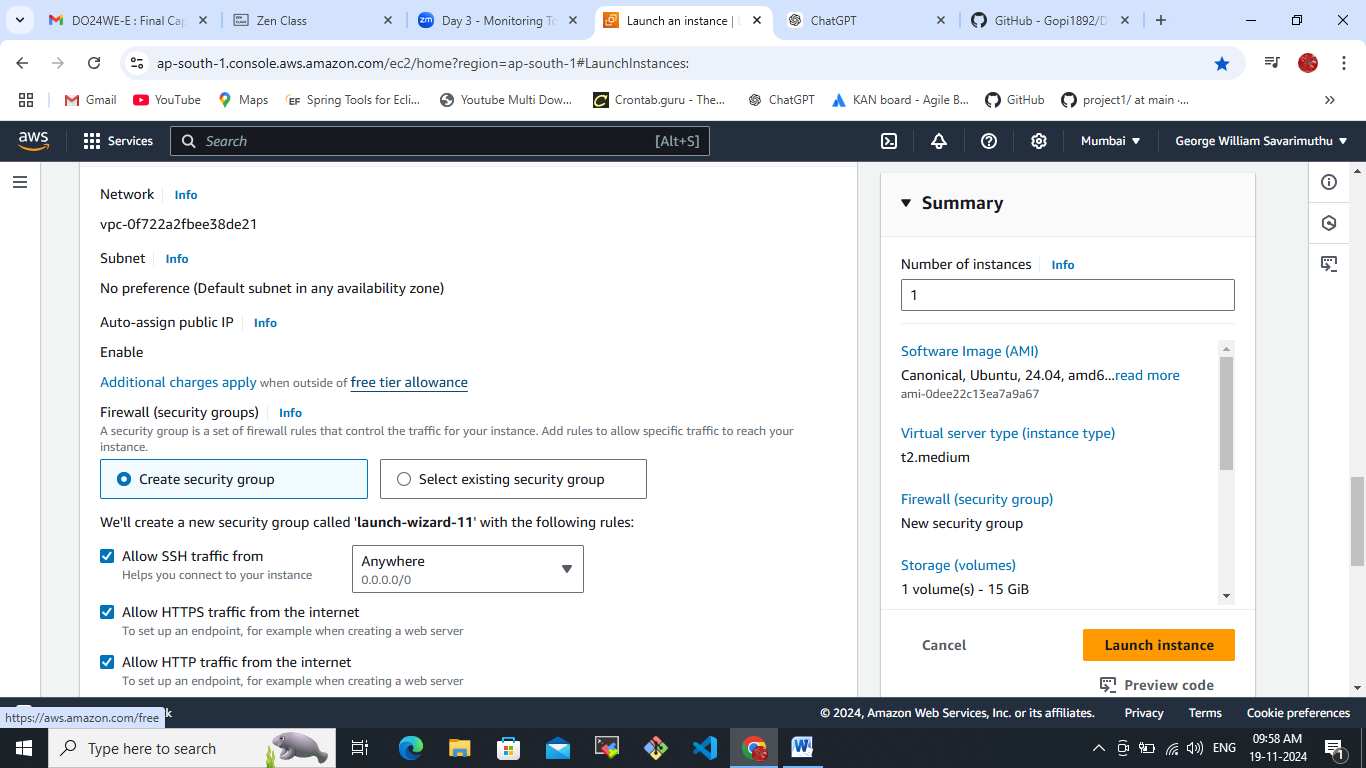
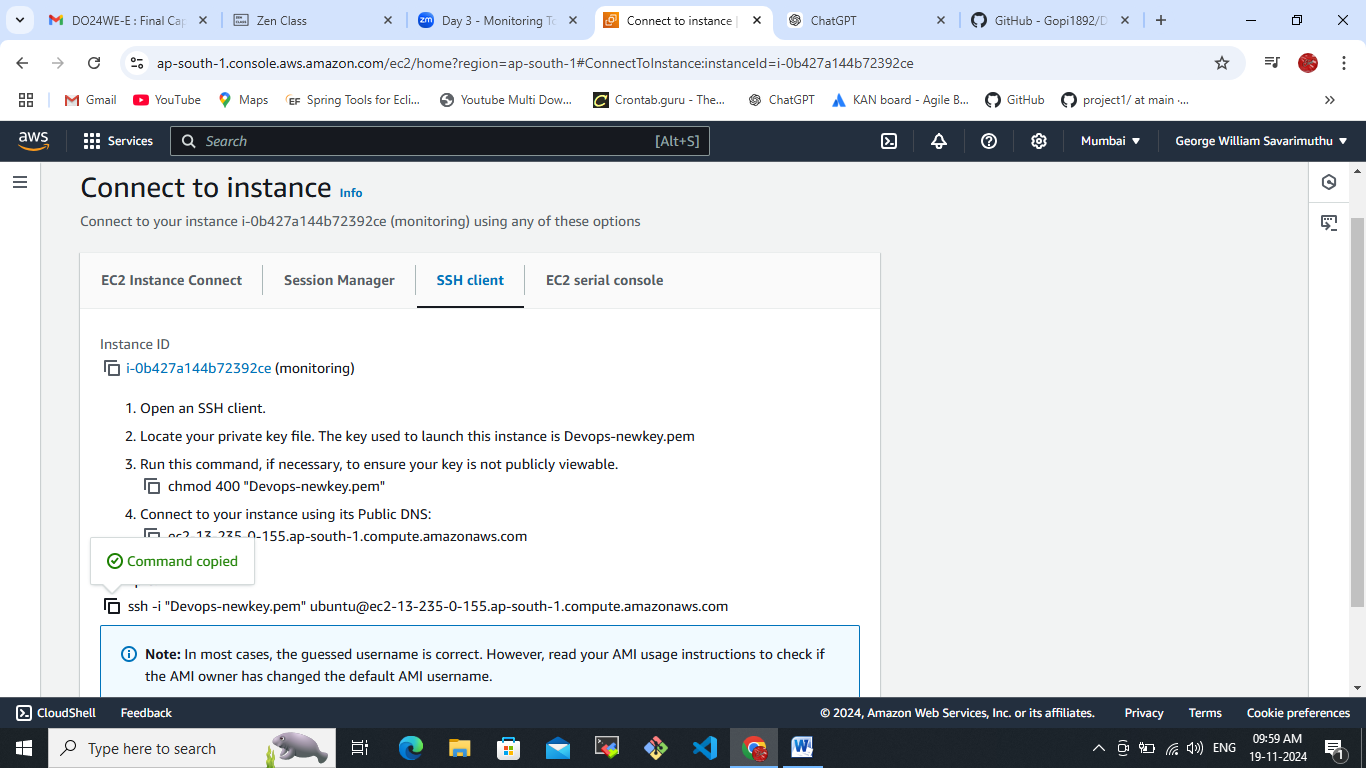
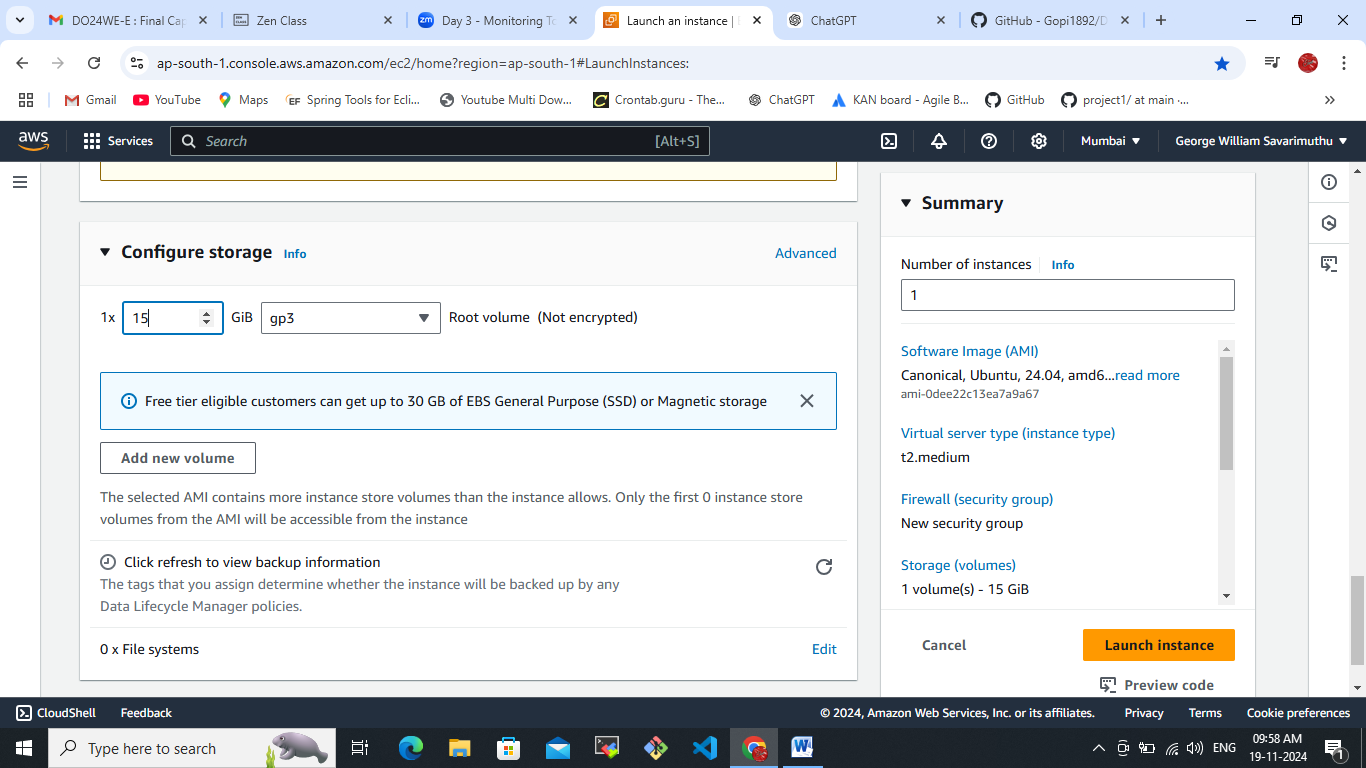
Create a instance with Ununtu image with t3 medium

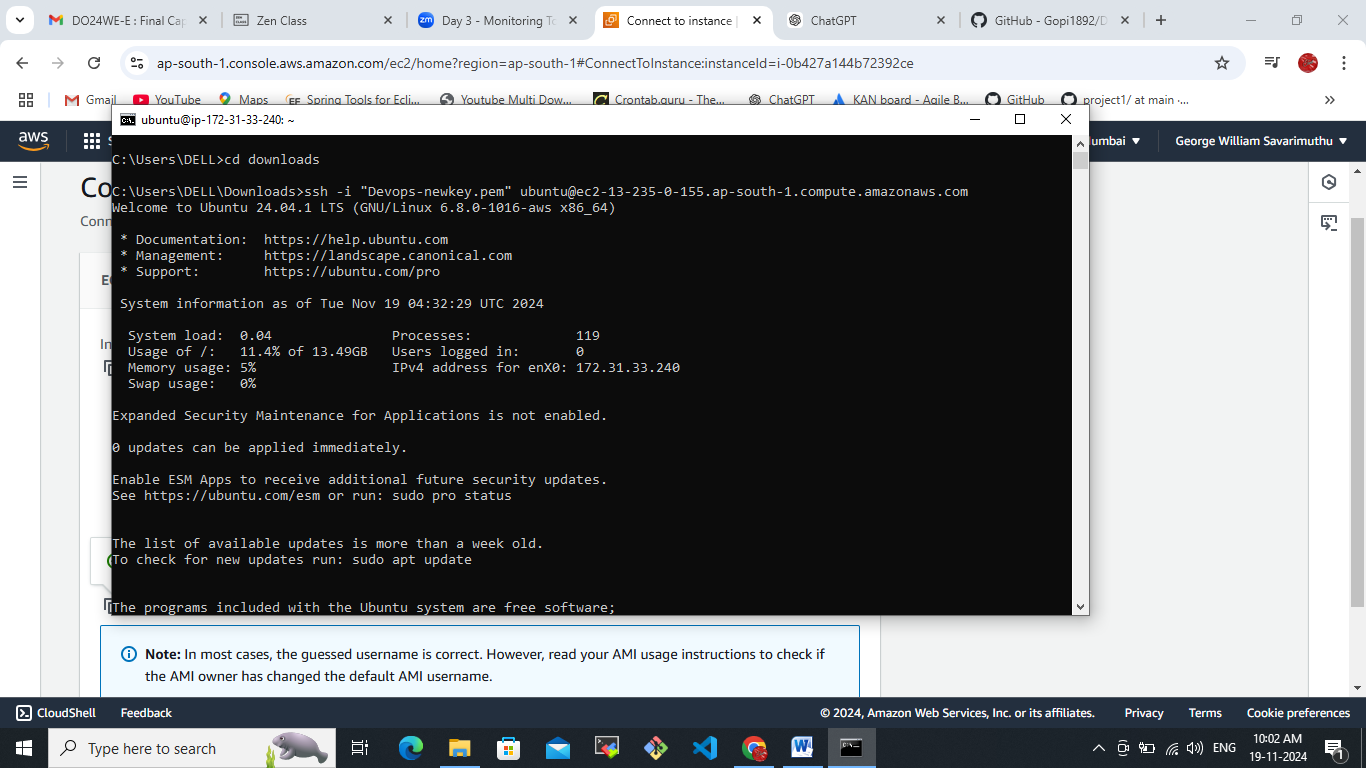


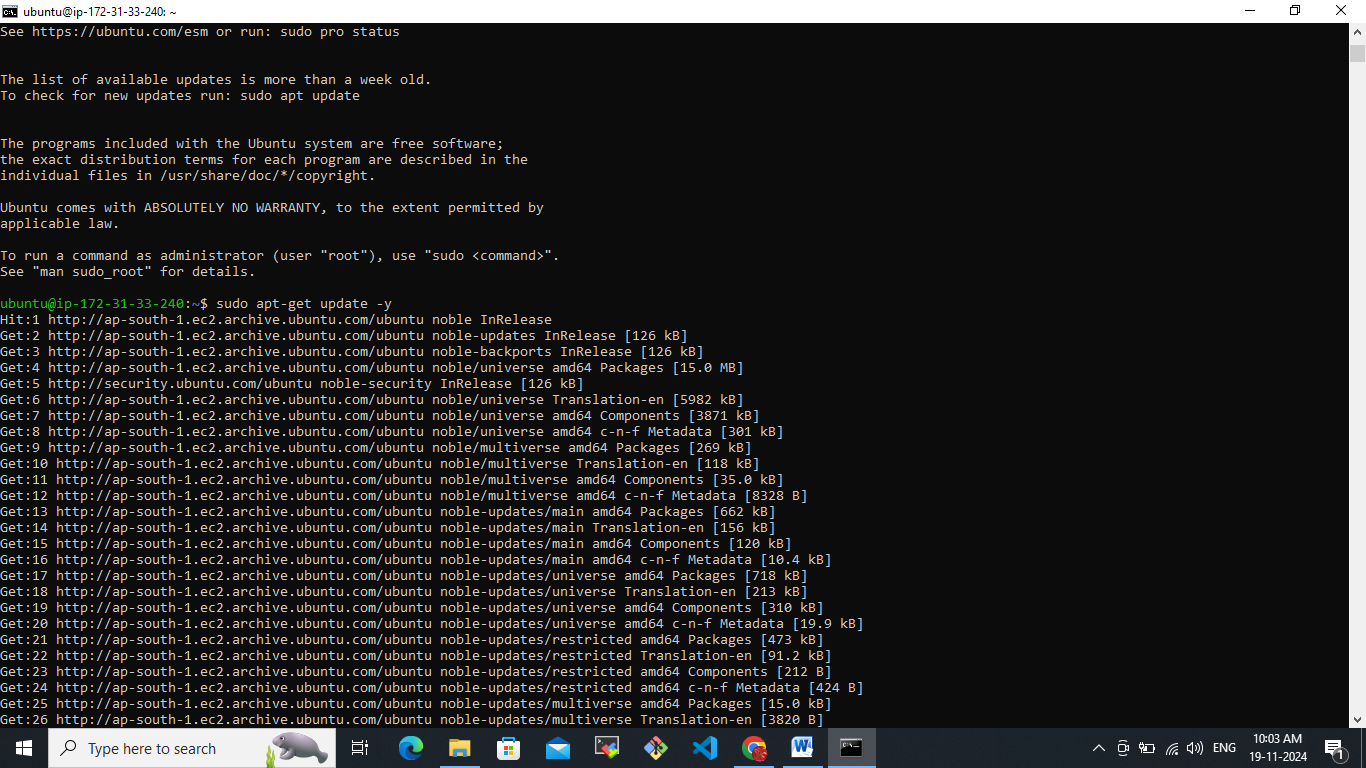


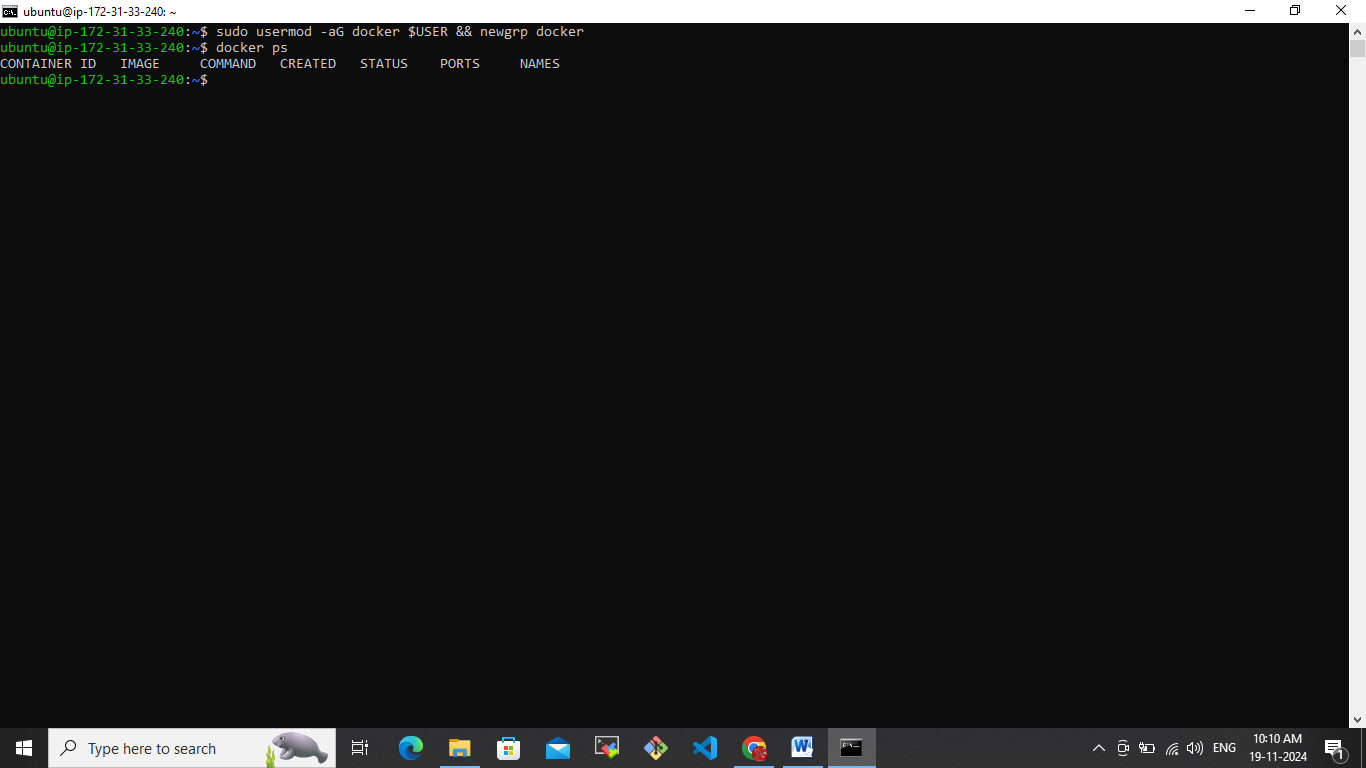


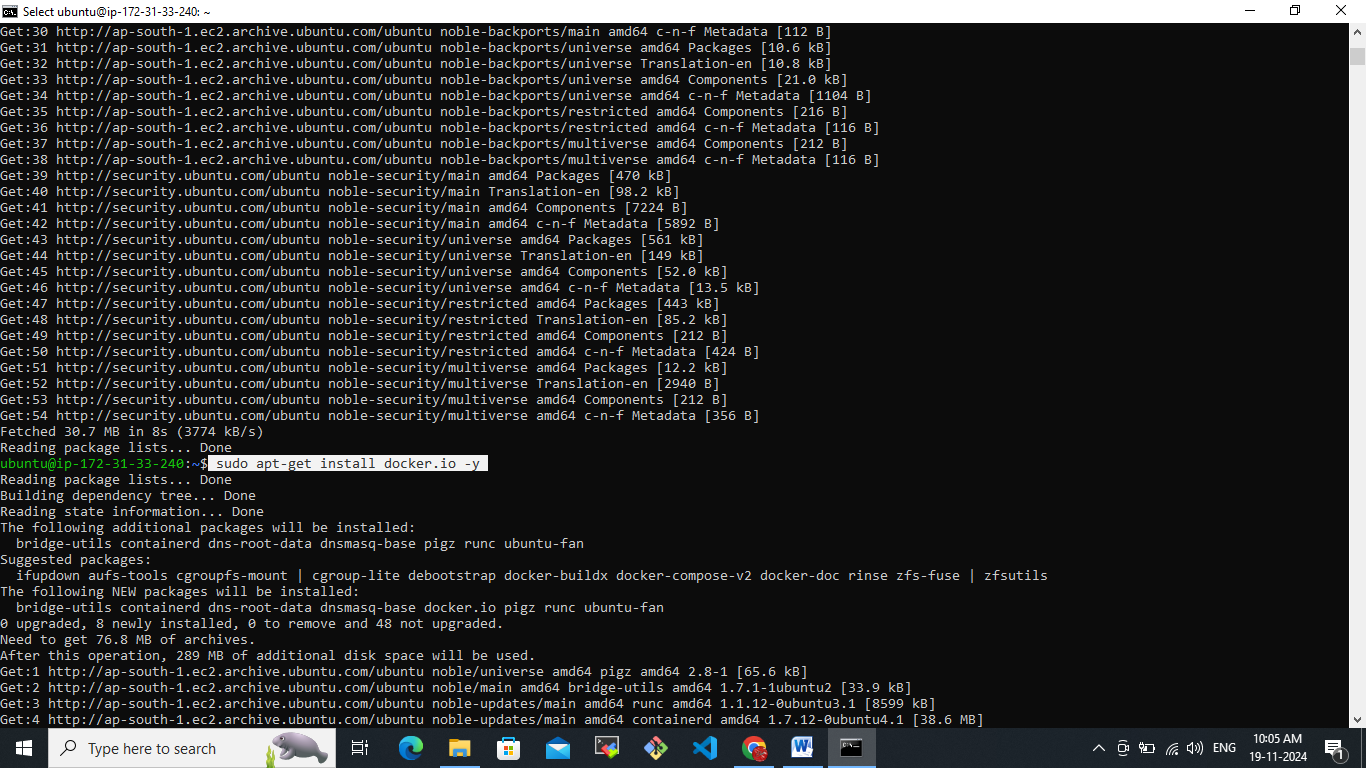


Instance created run cmd with administrator

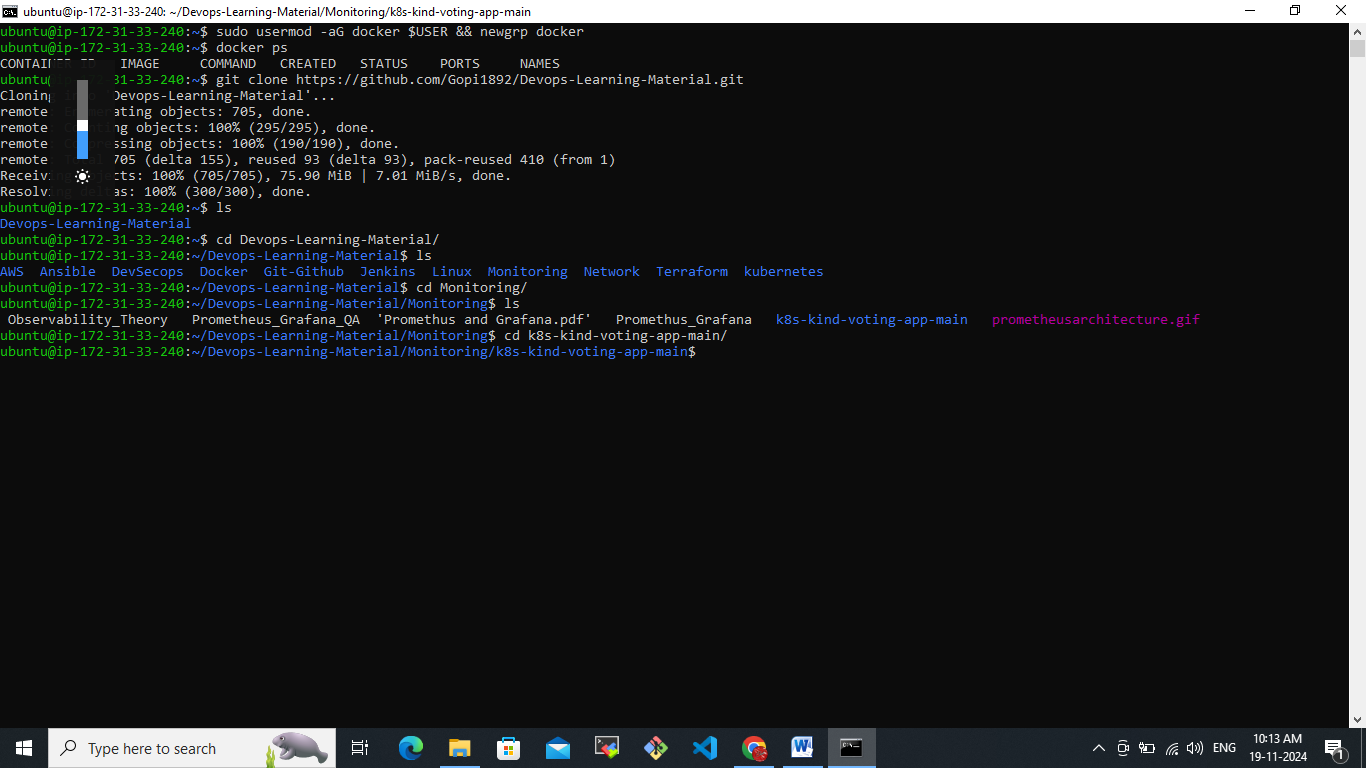
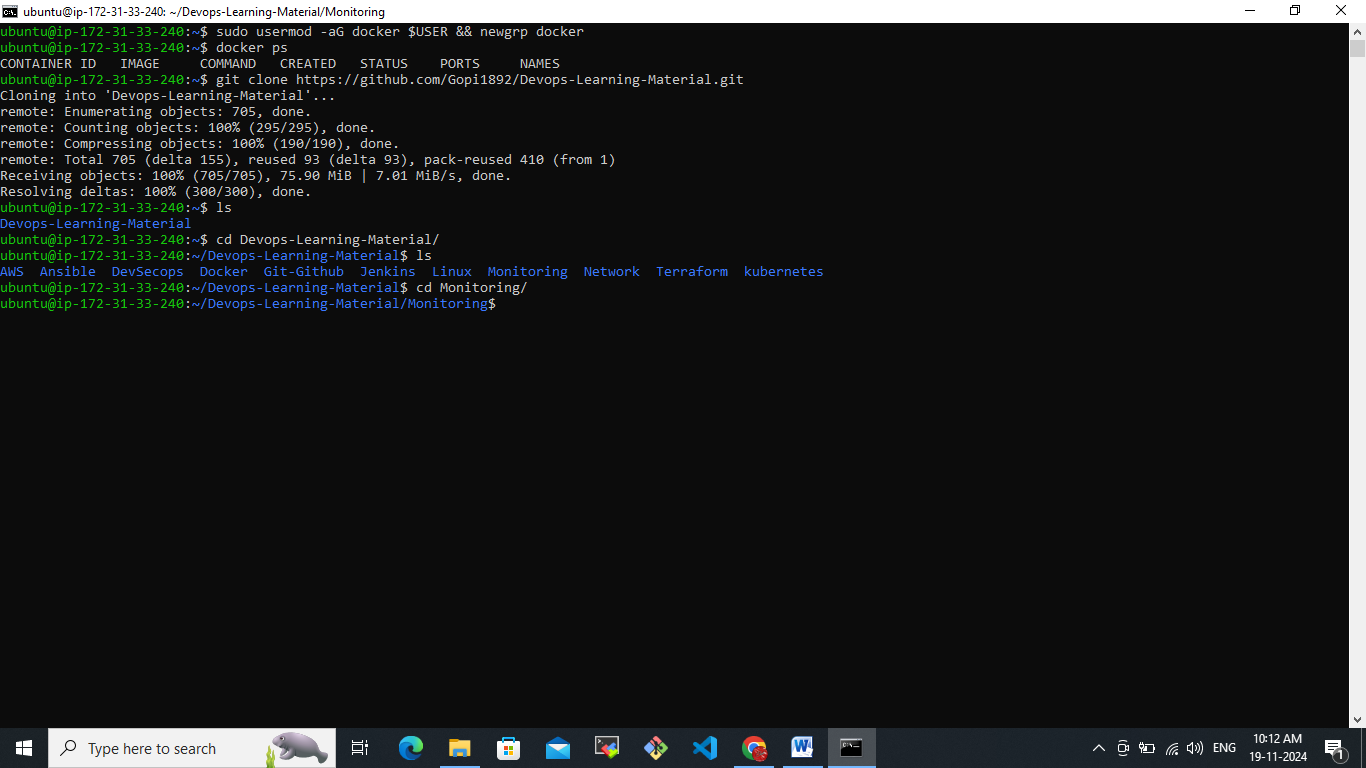


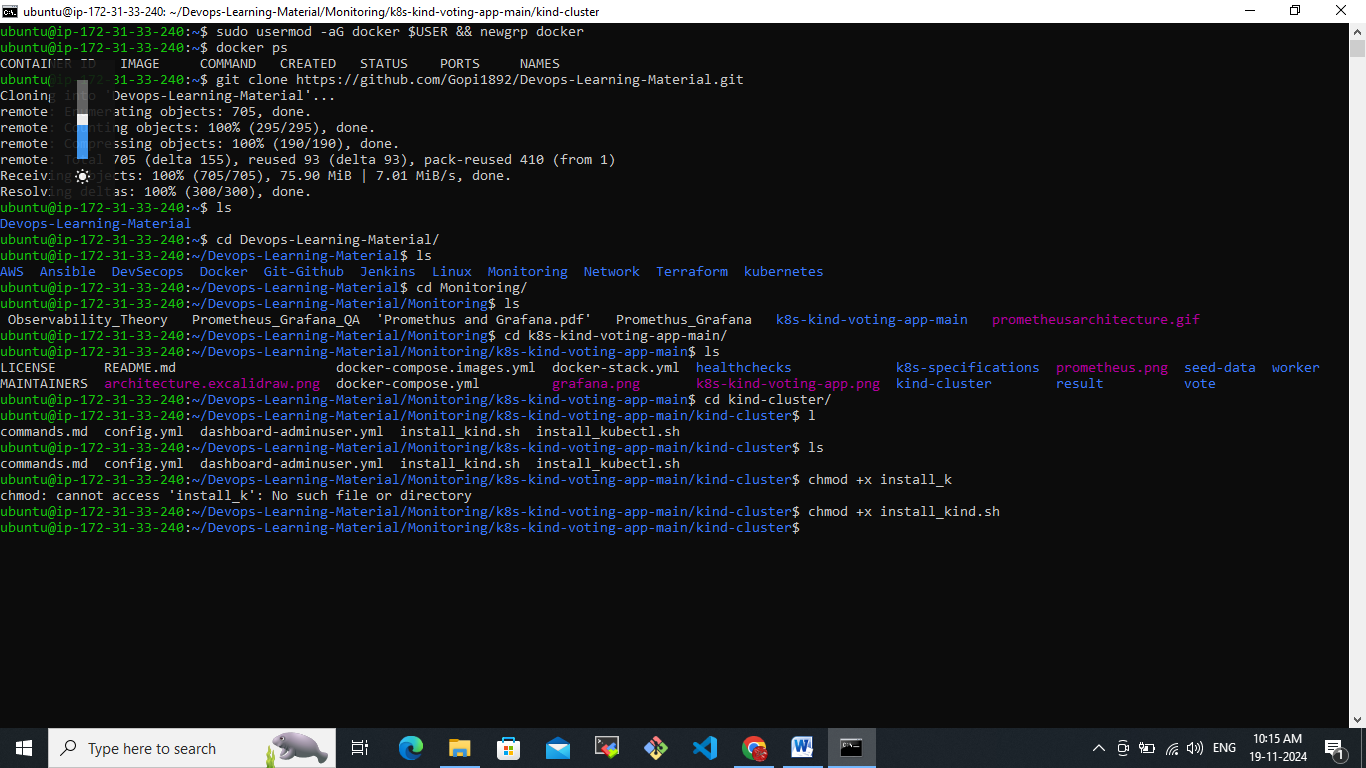


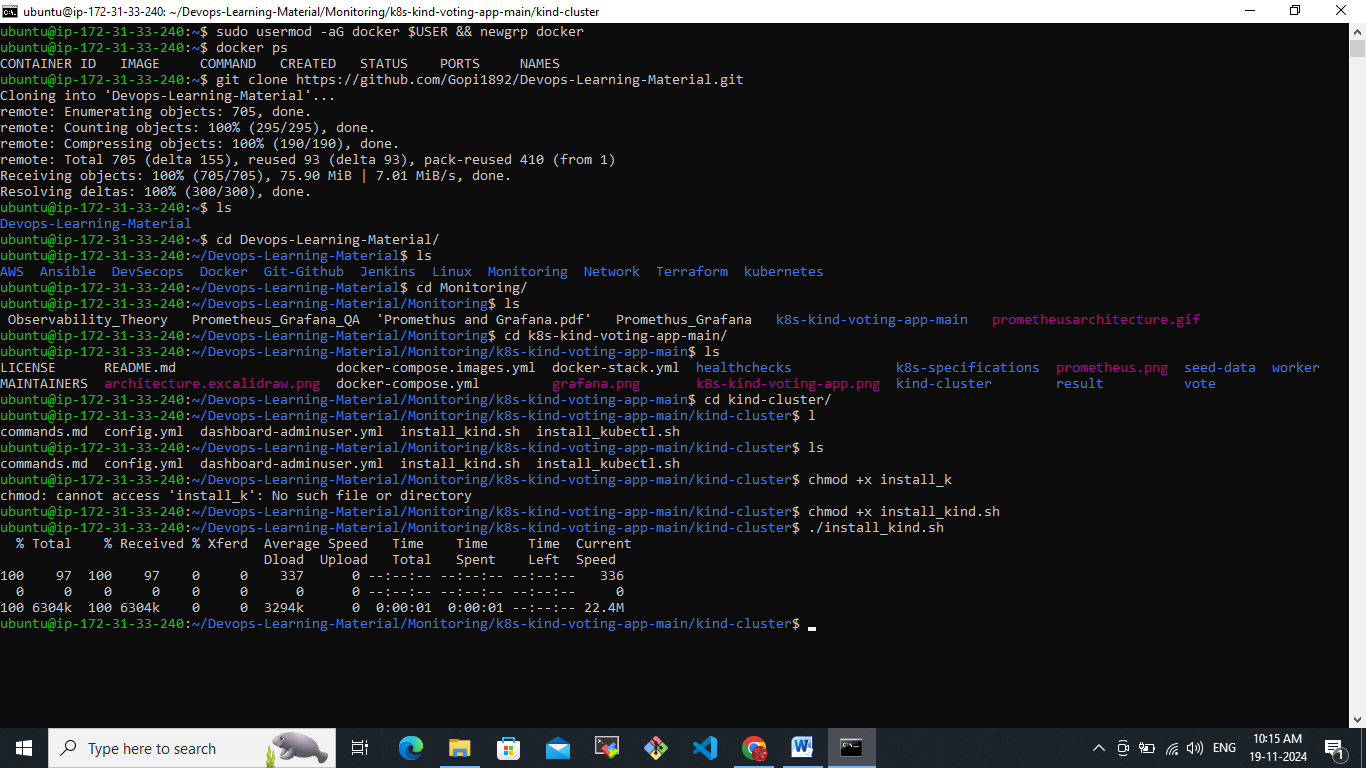


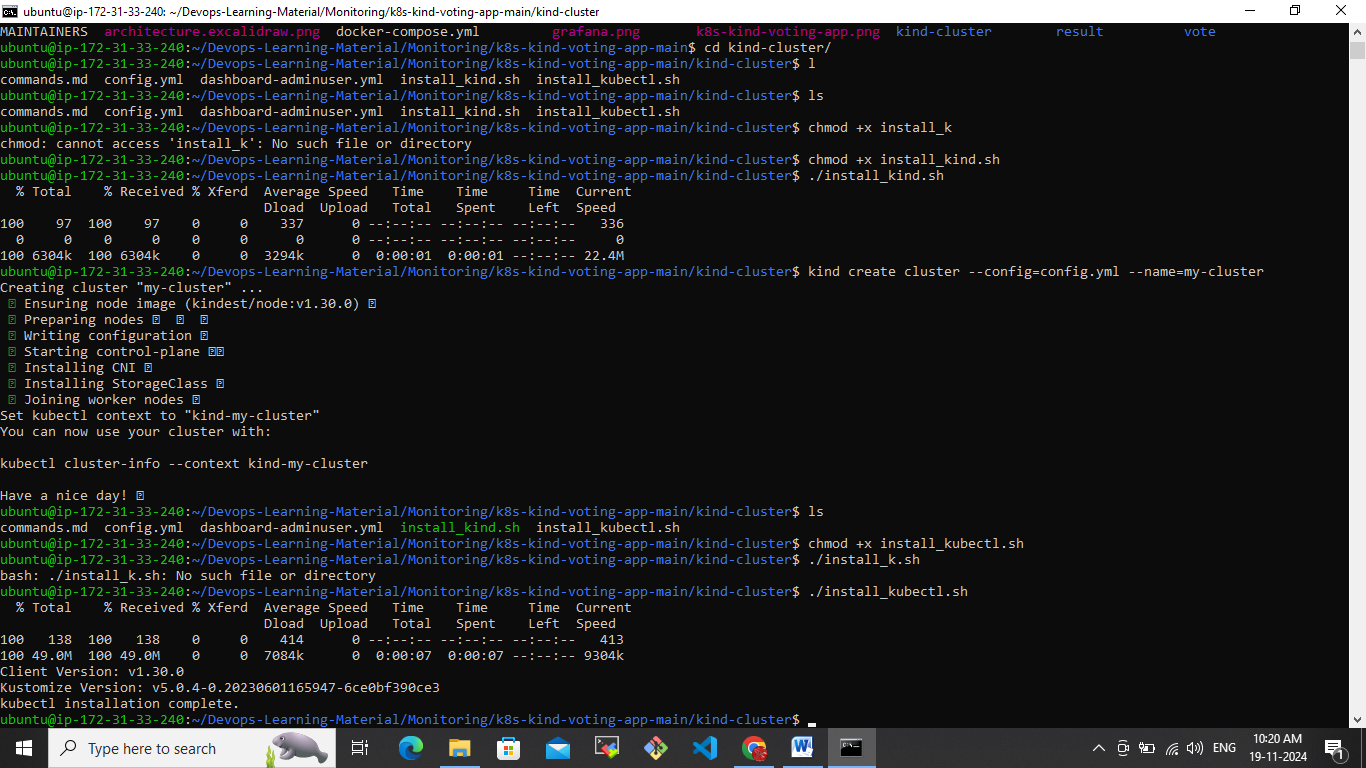
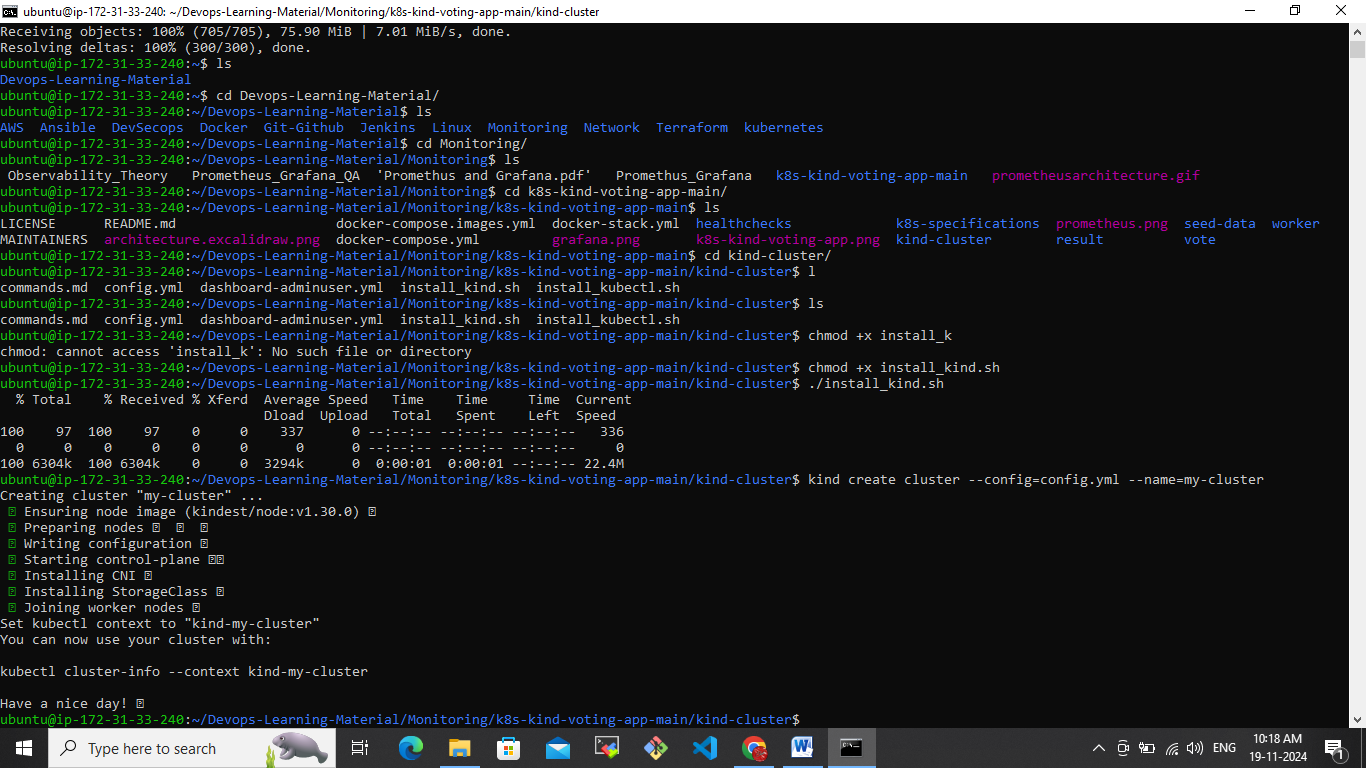


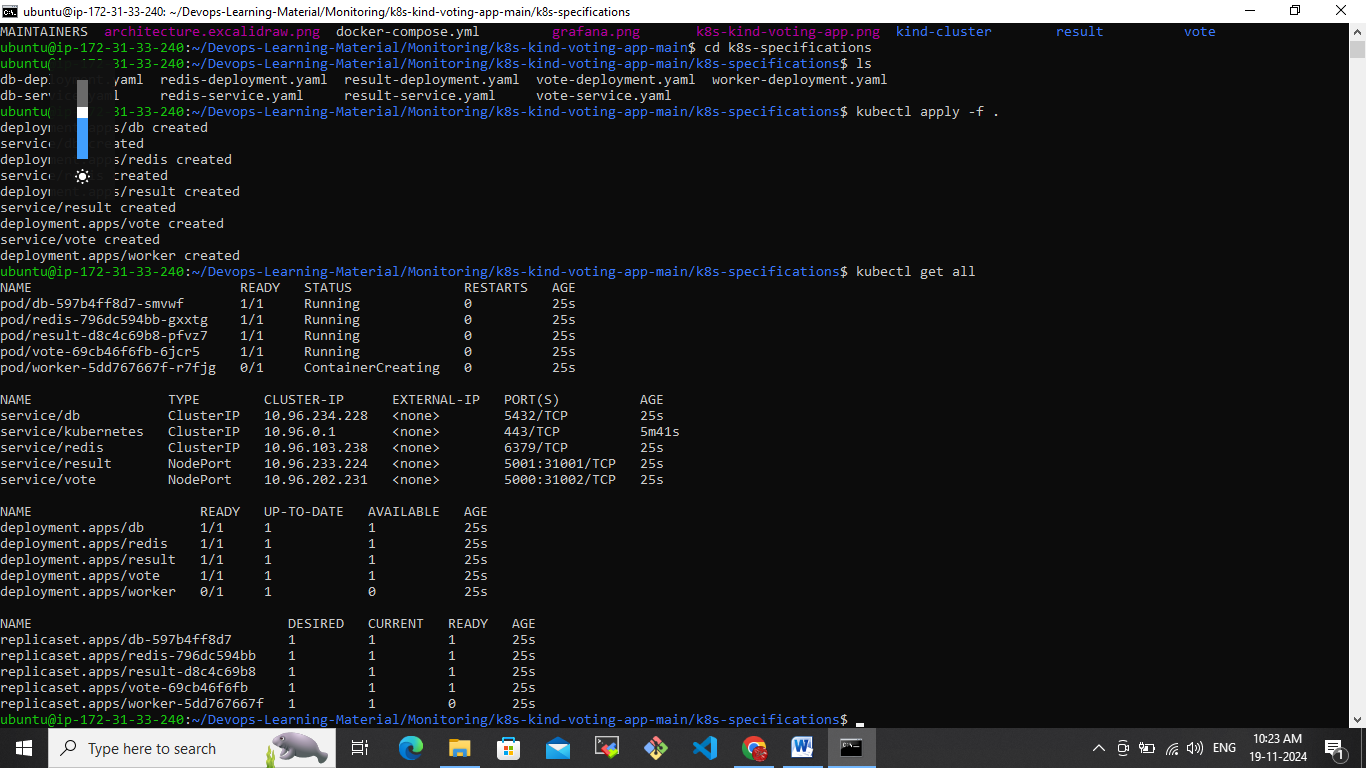
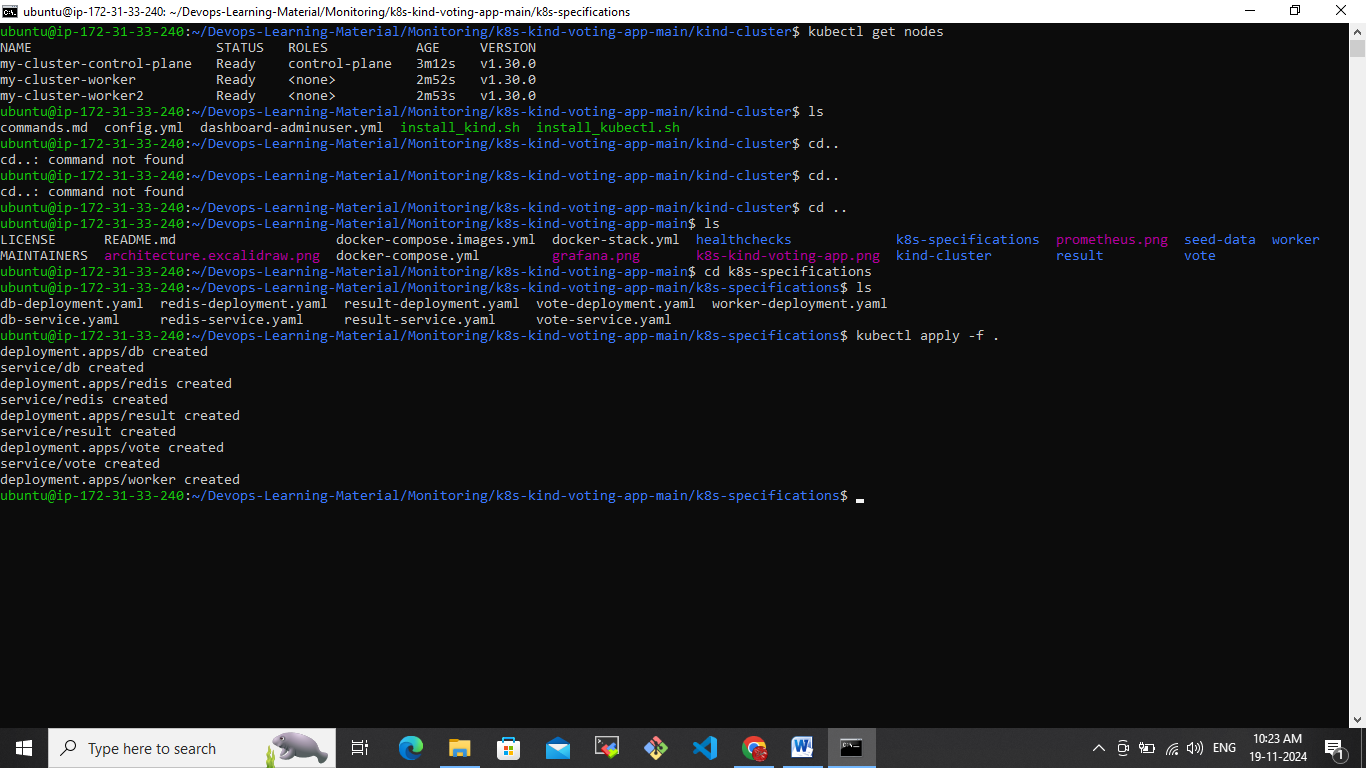
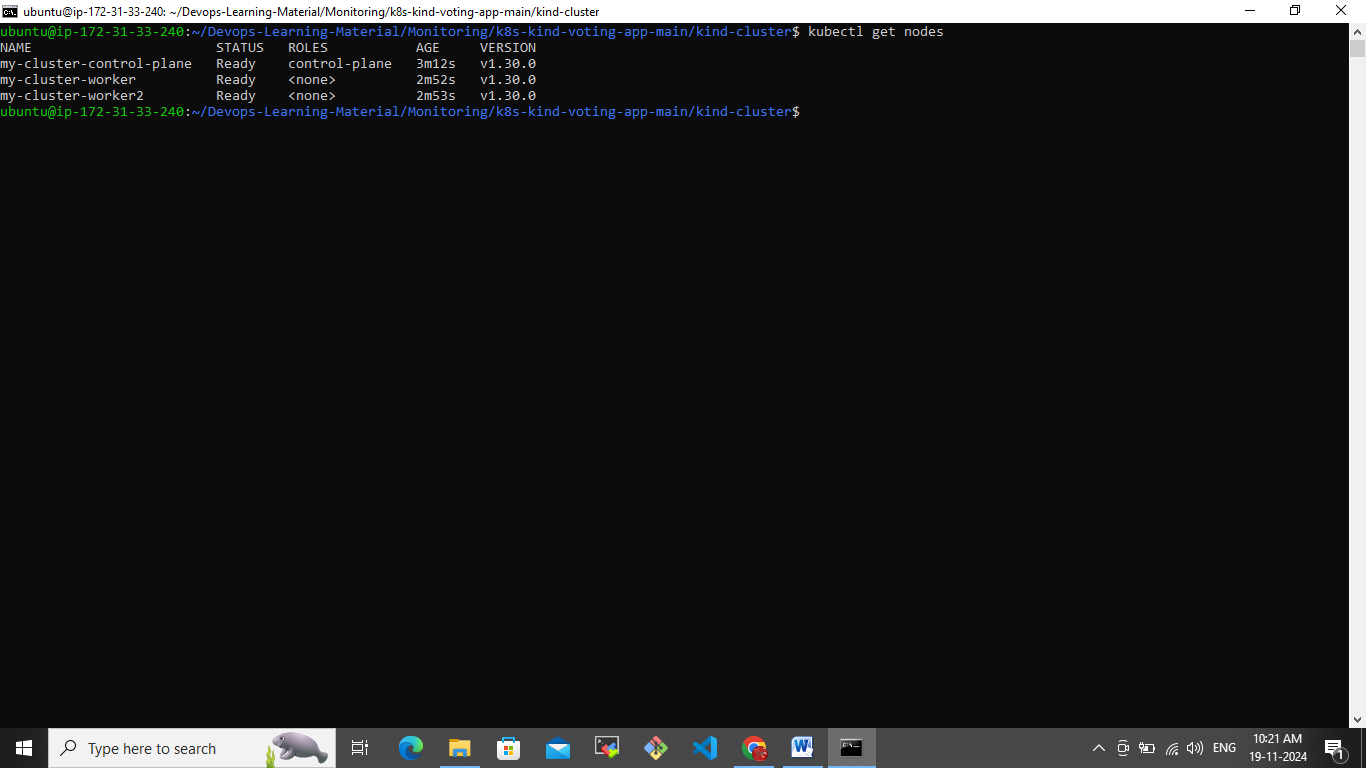
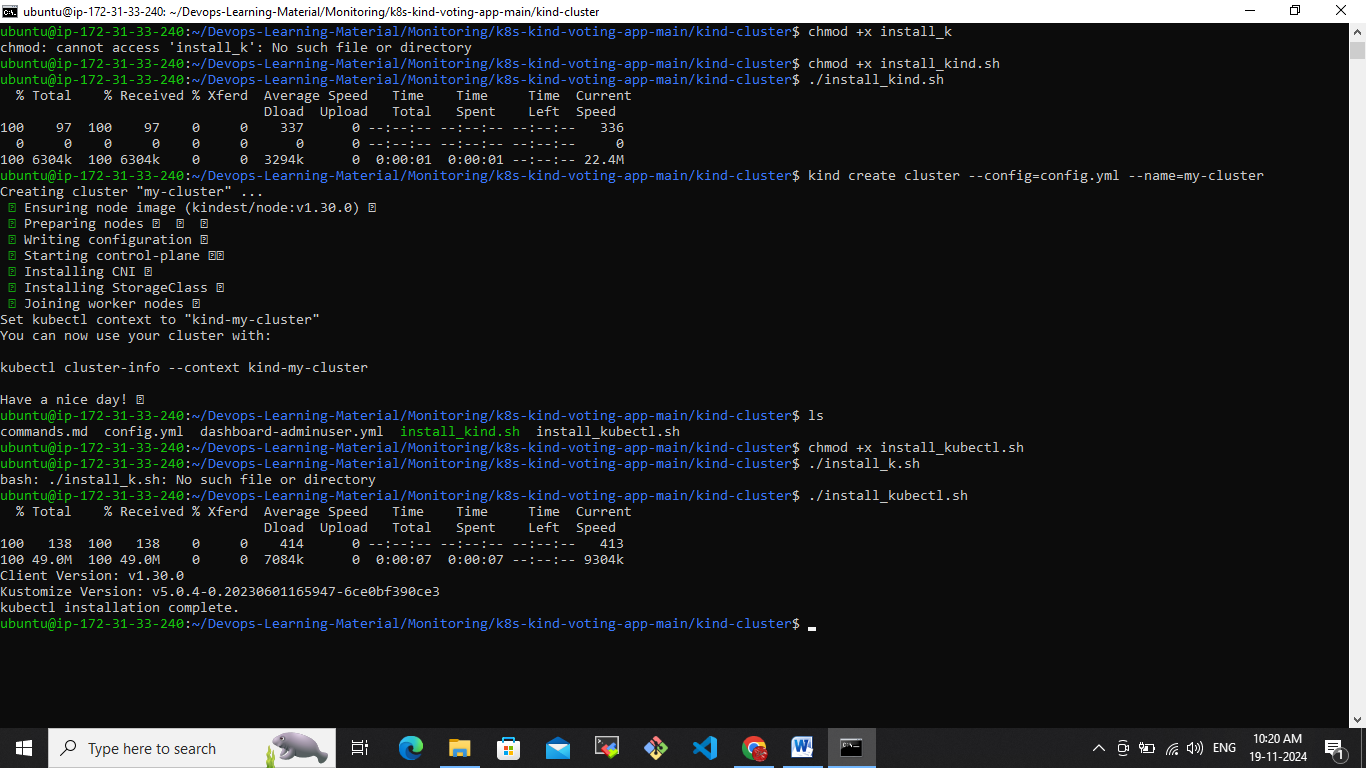
Git clone



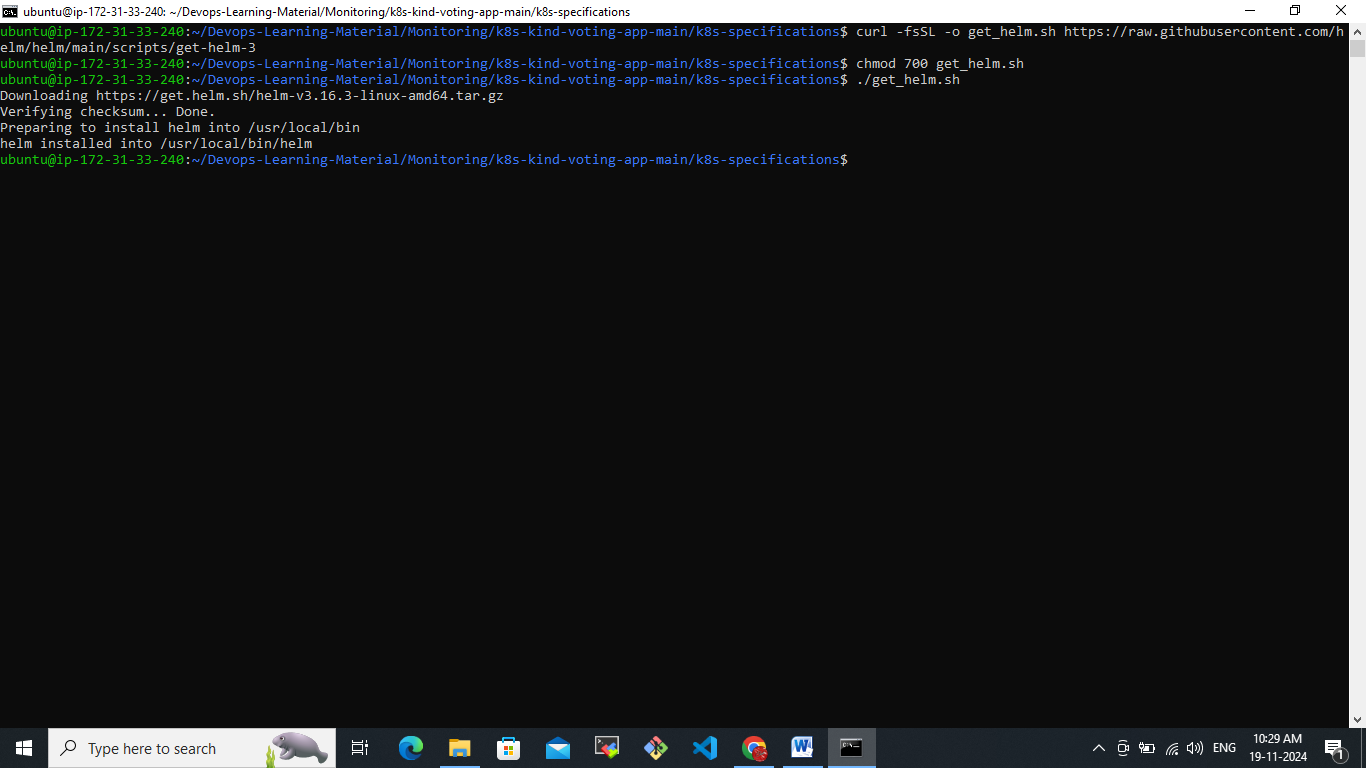


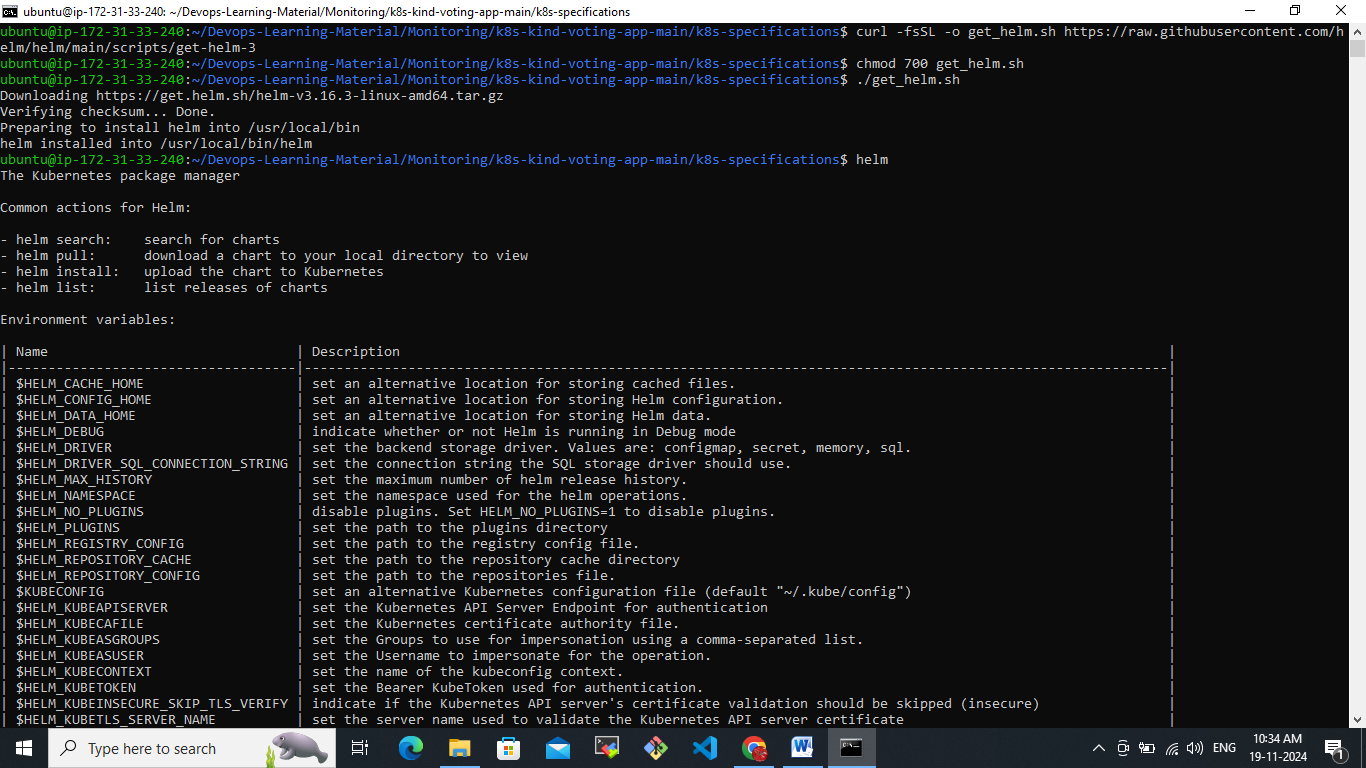




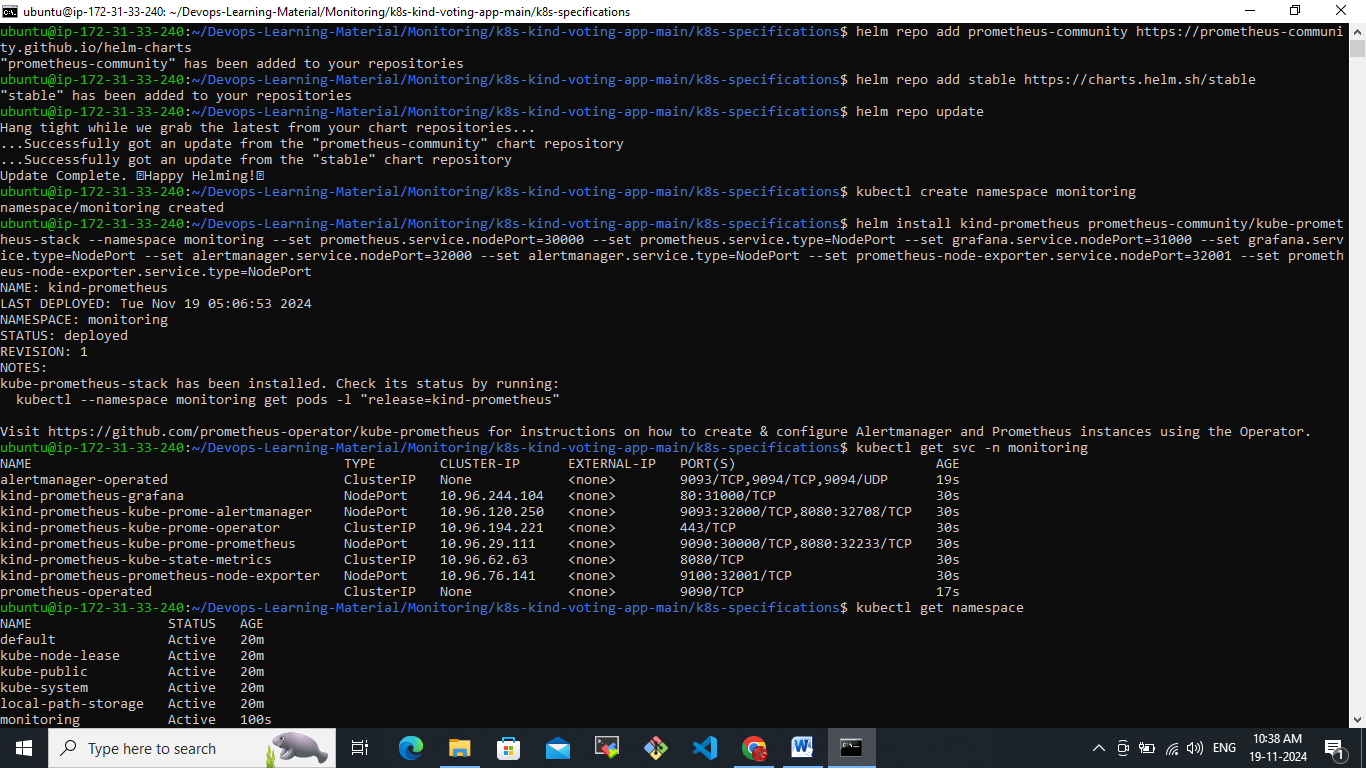


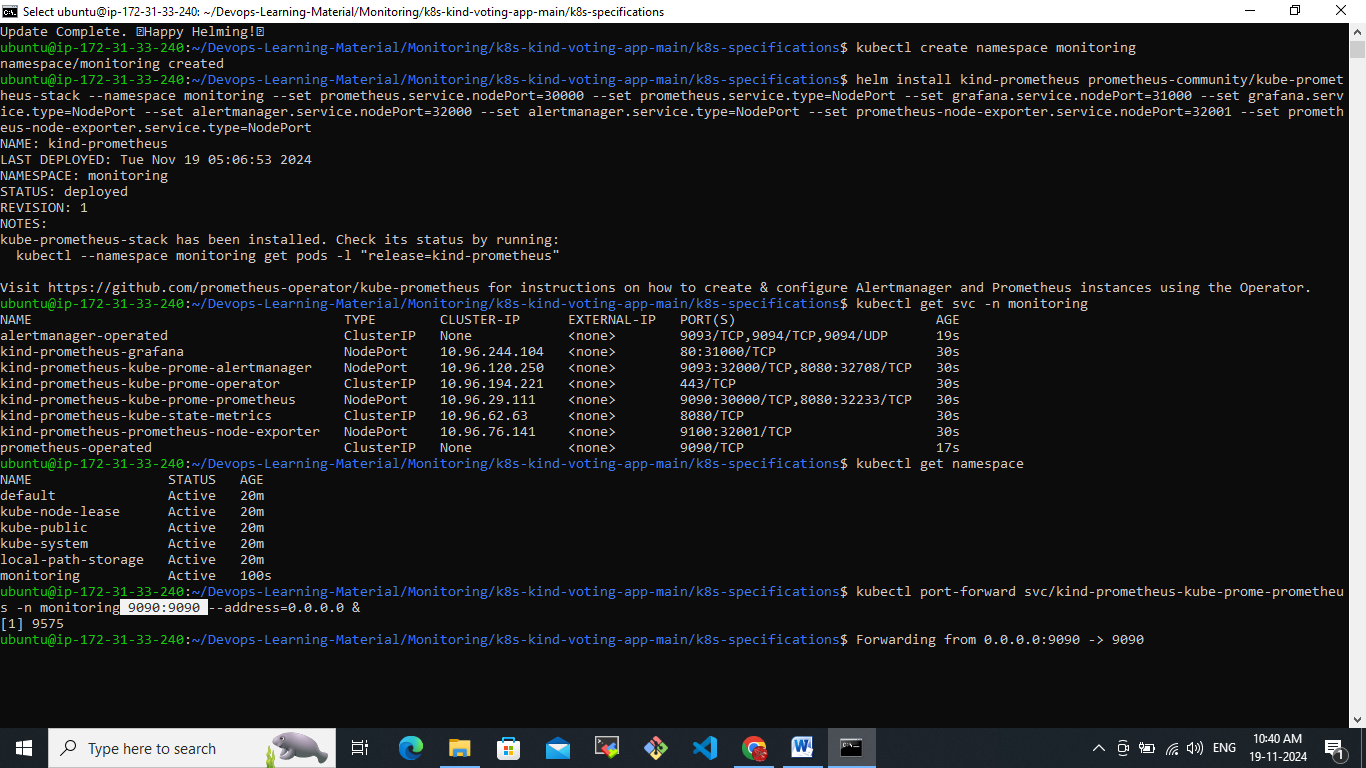
Let’s go ahead and install Helm



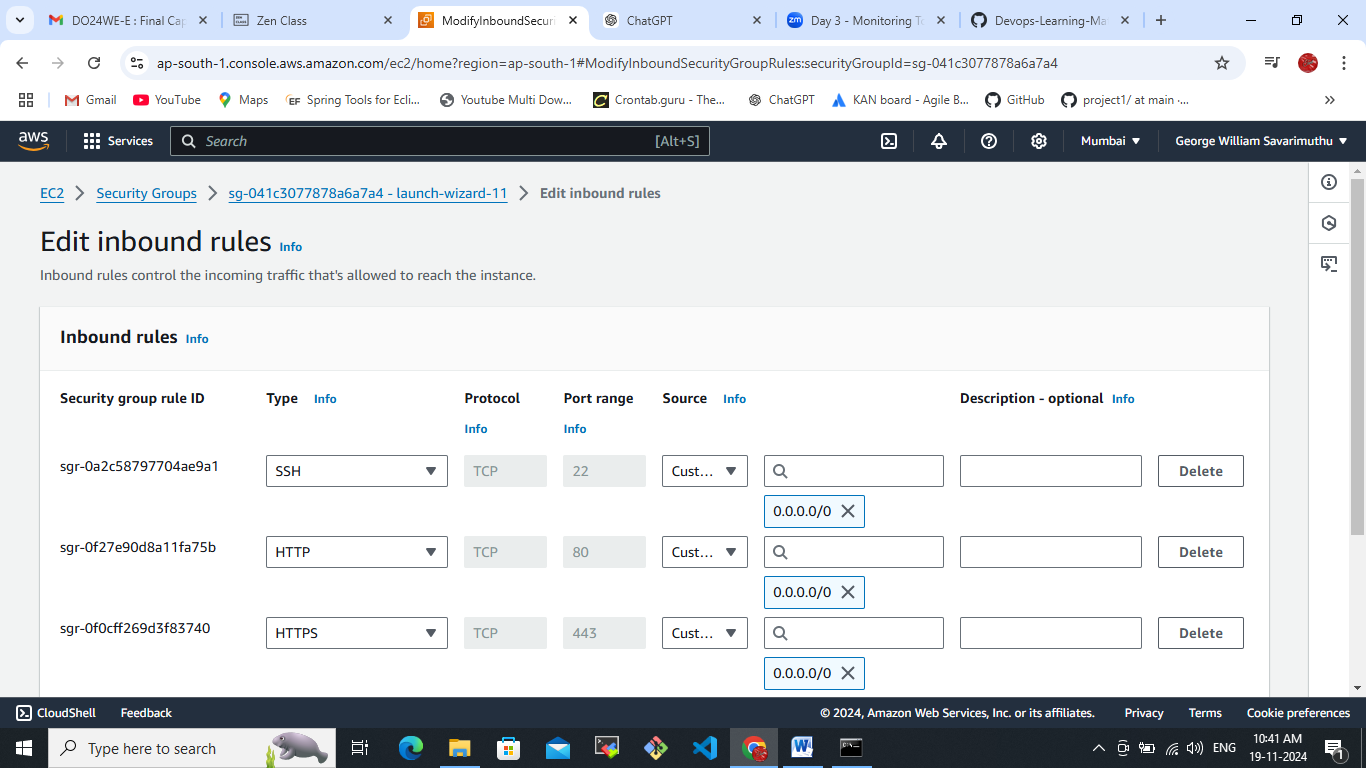


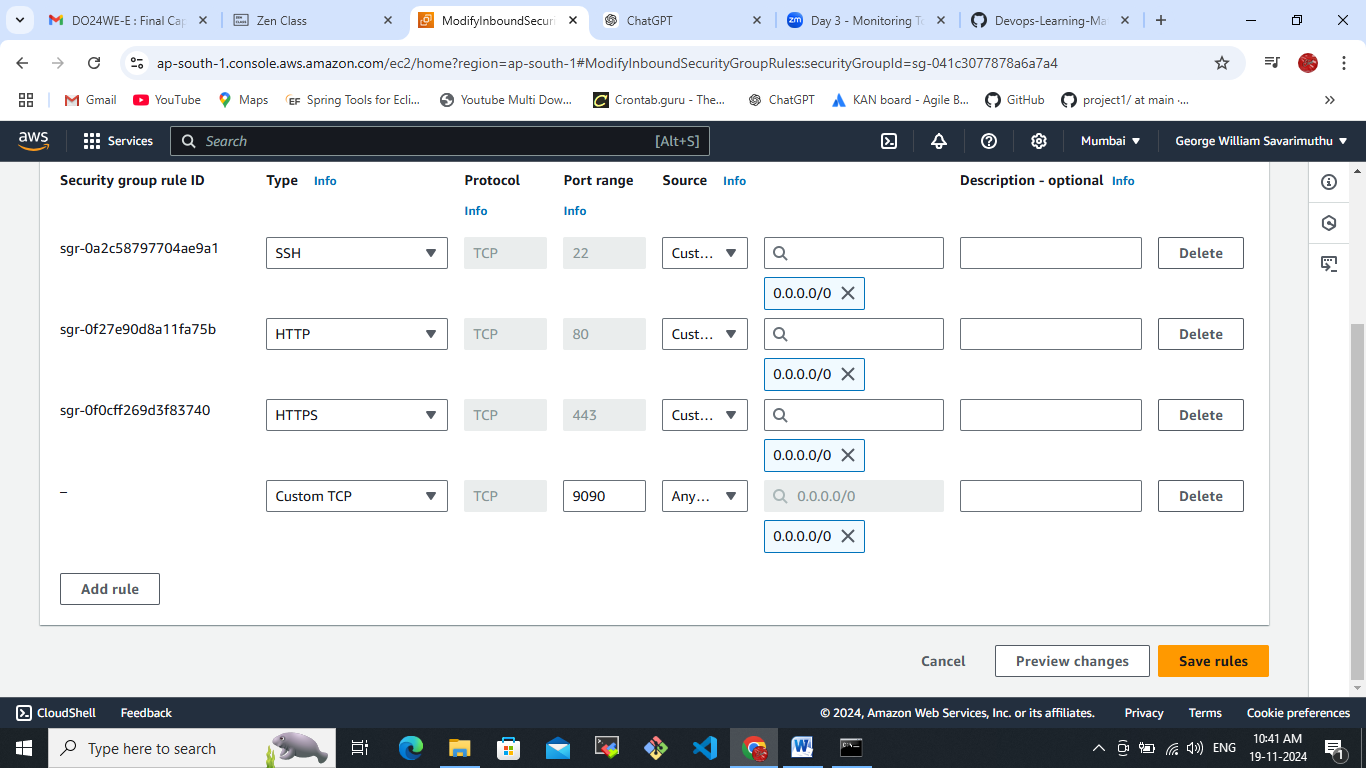
Now let’s install Prometheus stack



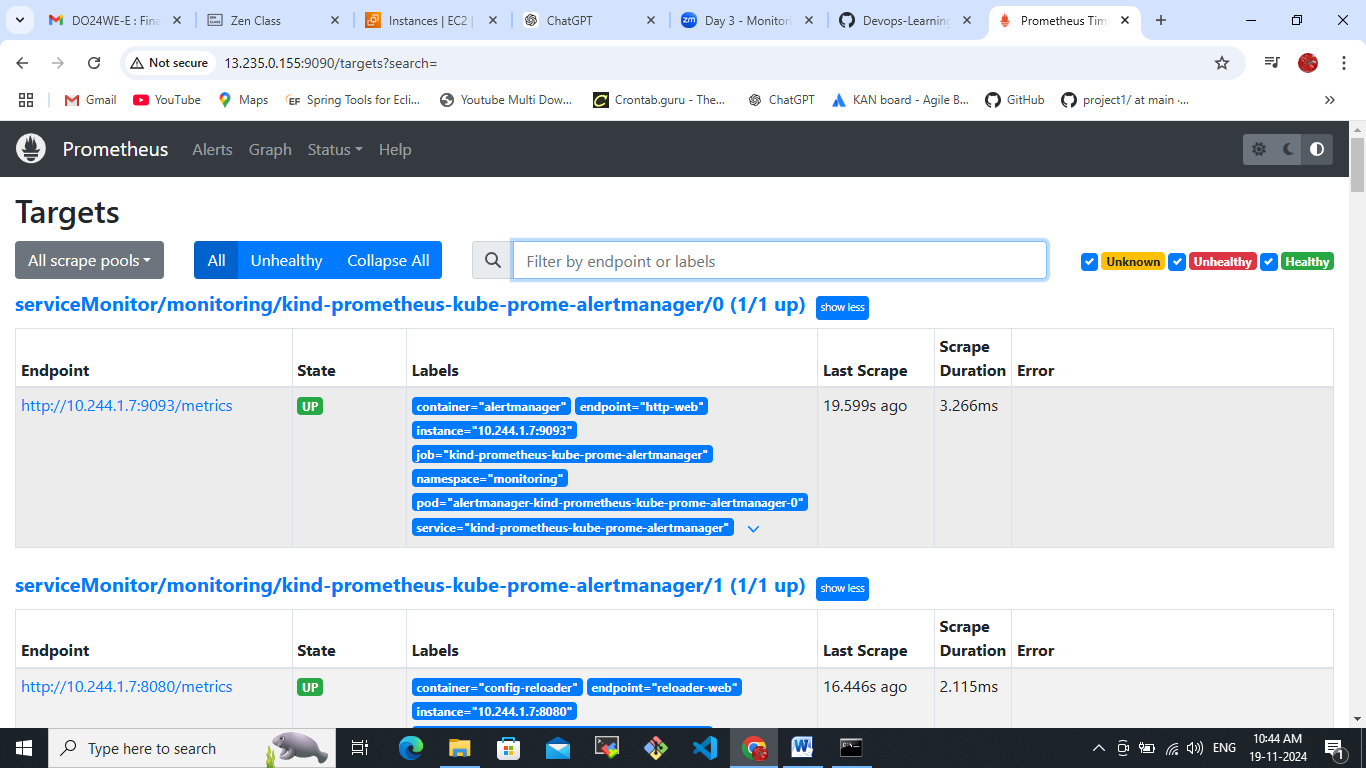


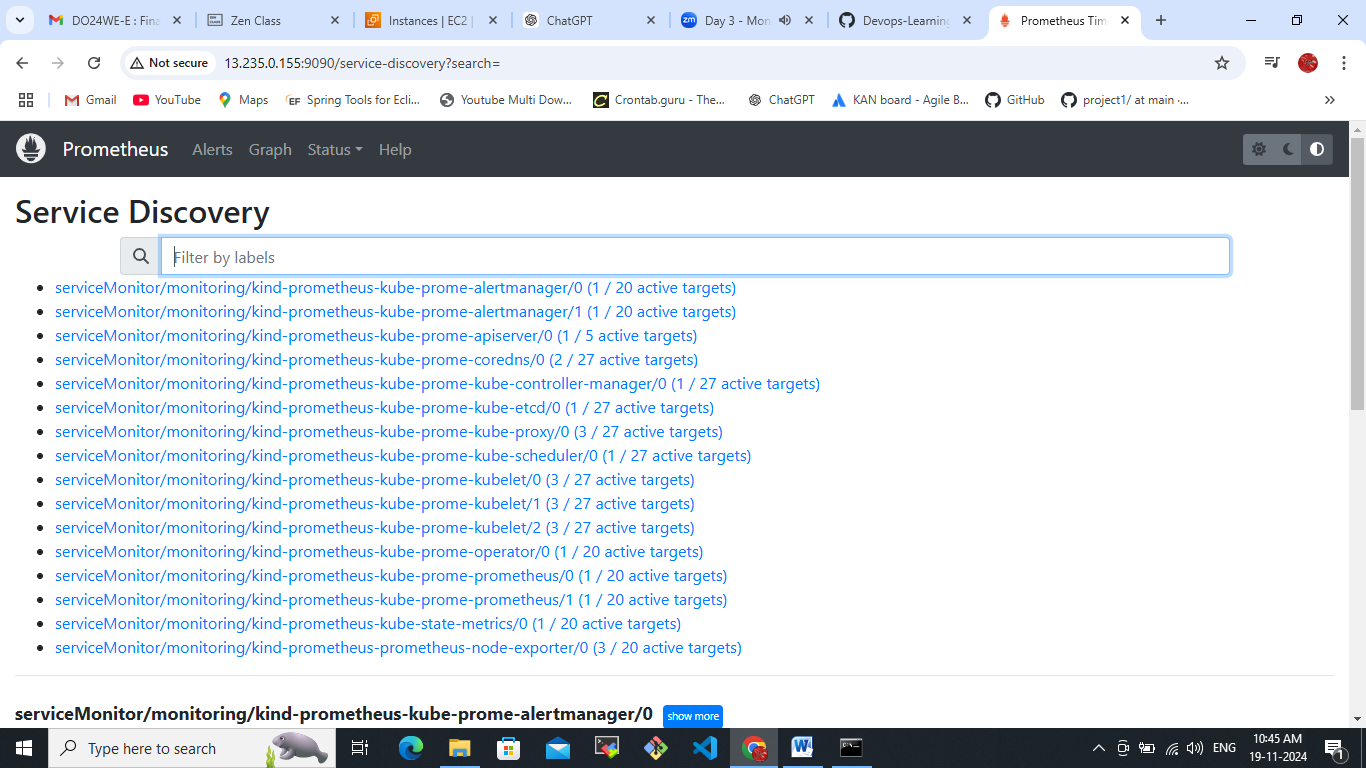
Now lets go ahed the security groups in aws whether the ports add already added or not in inbound section



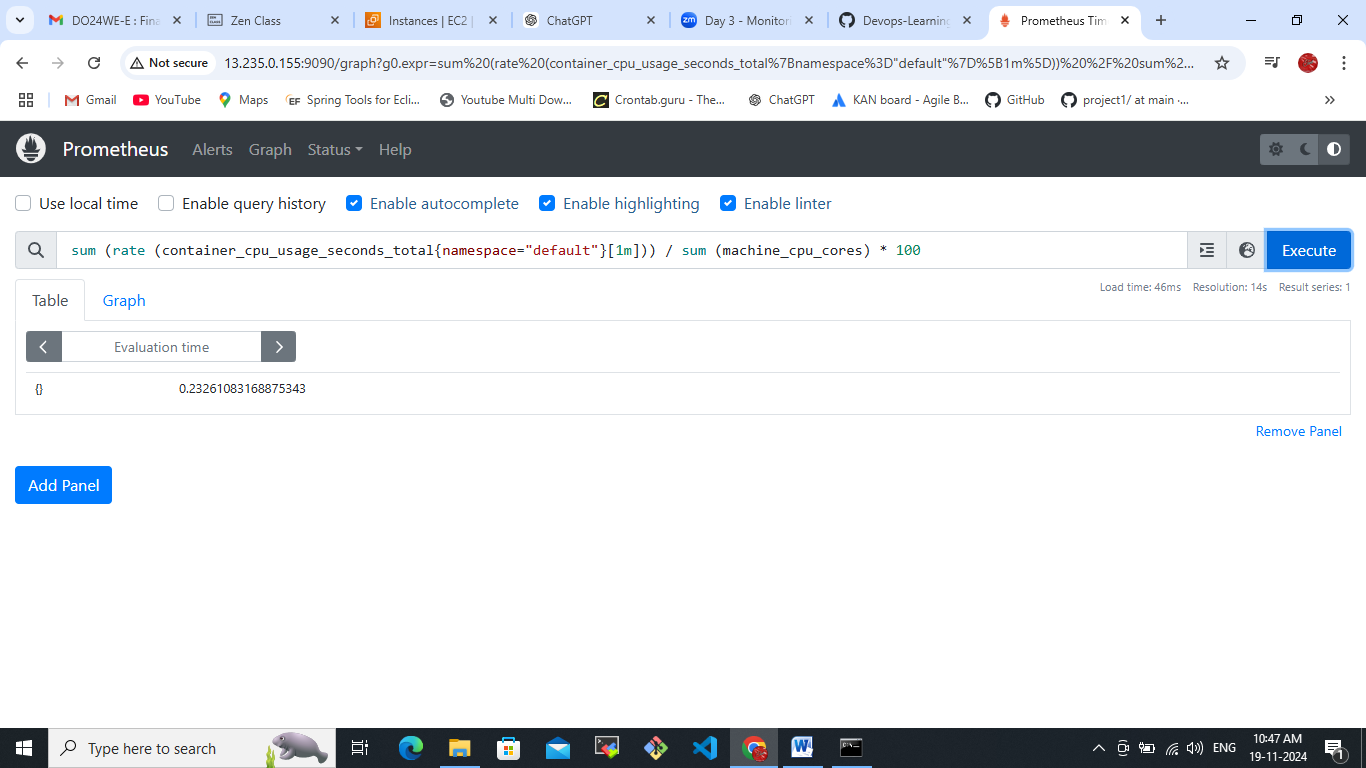


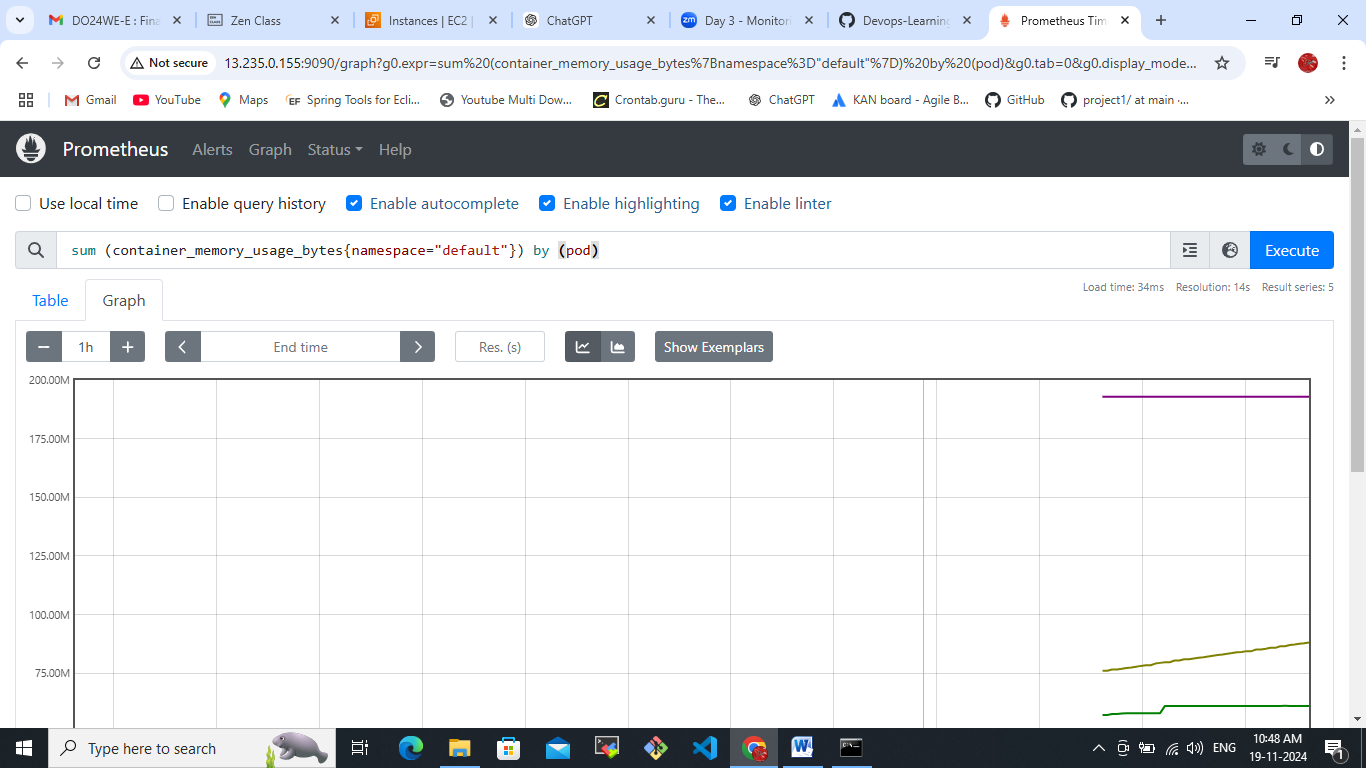
Done let try to access by using public ip

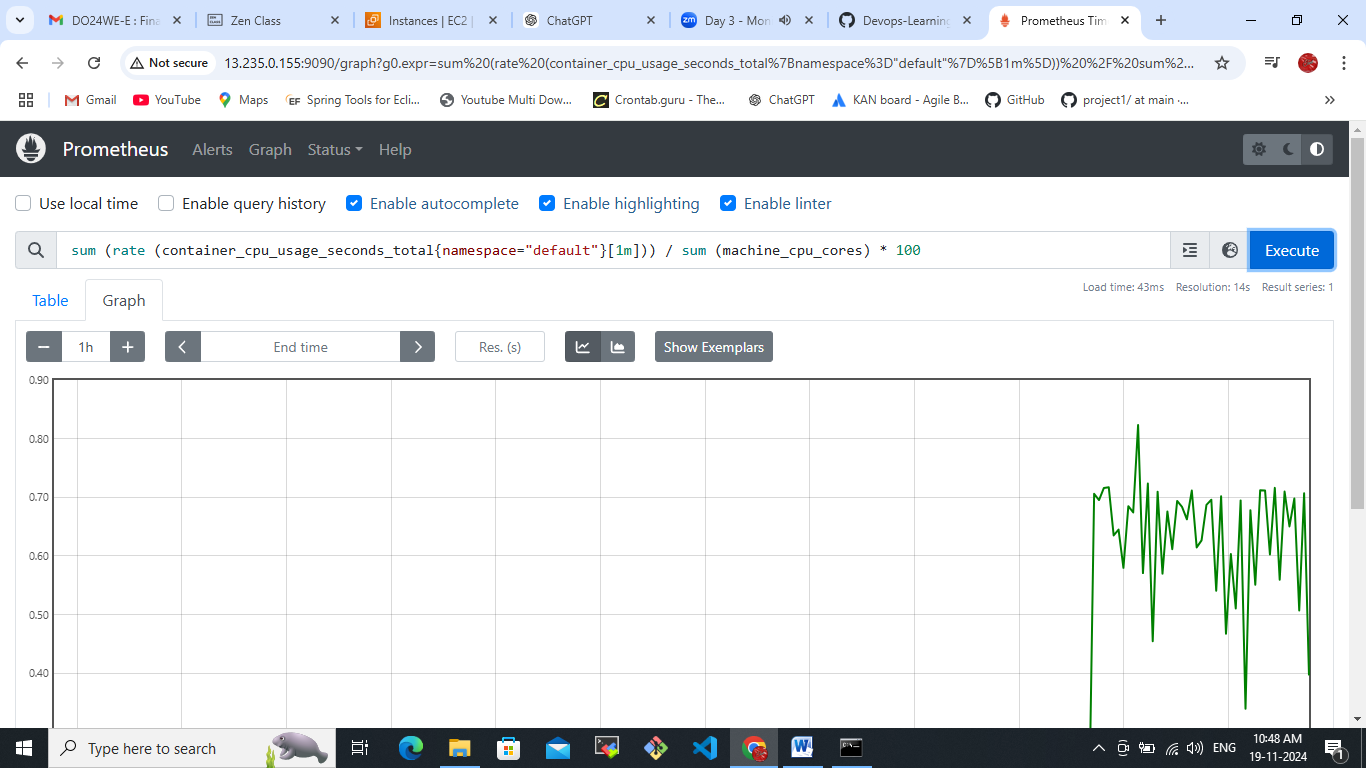
<http://amazon> public ip:9090  
http://13.235.0.155:9090/  
  


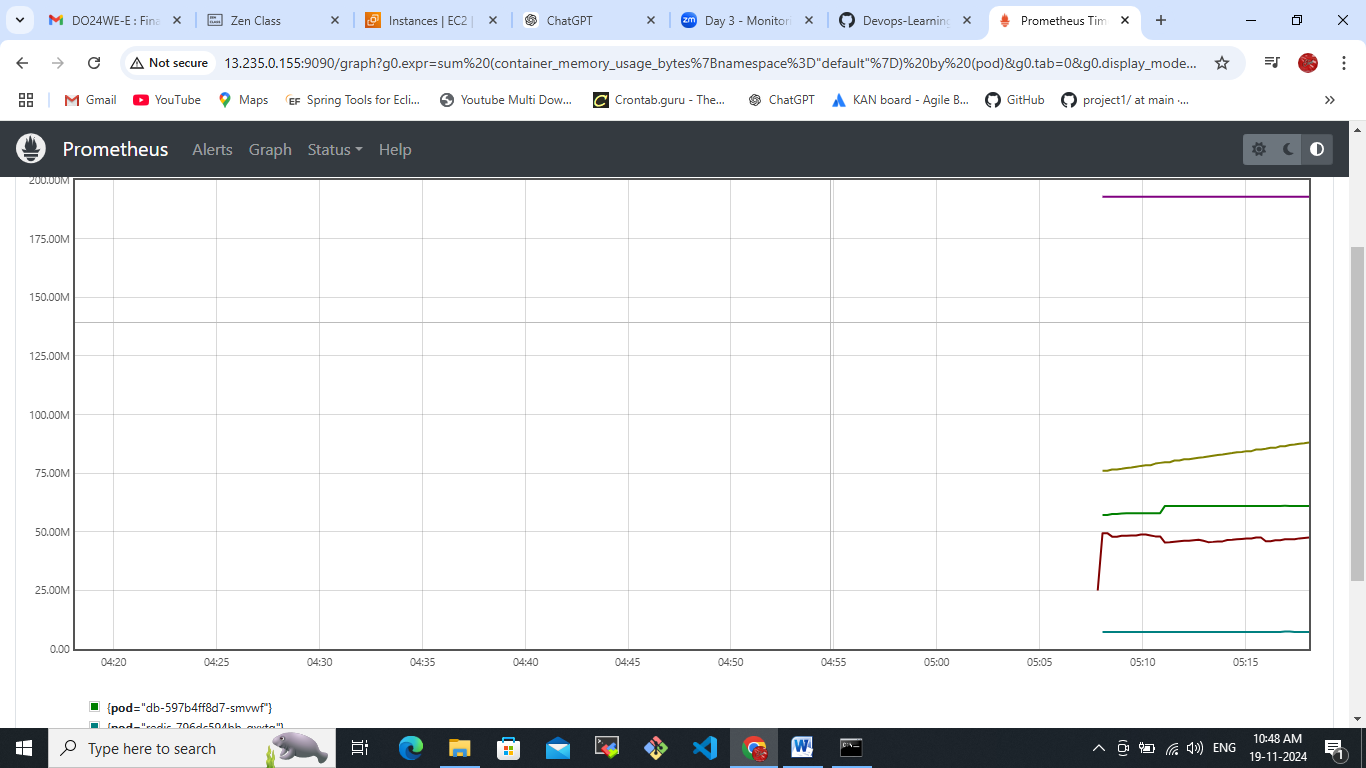


To find the CPU cores



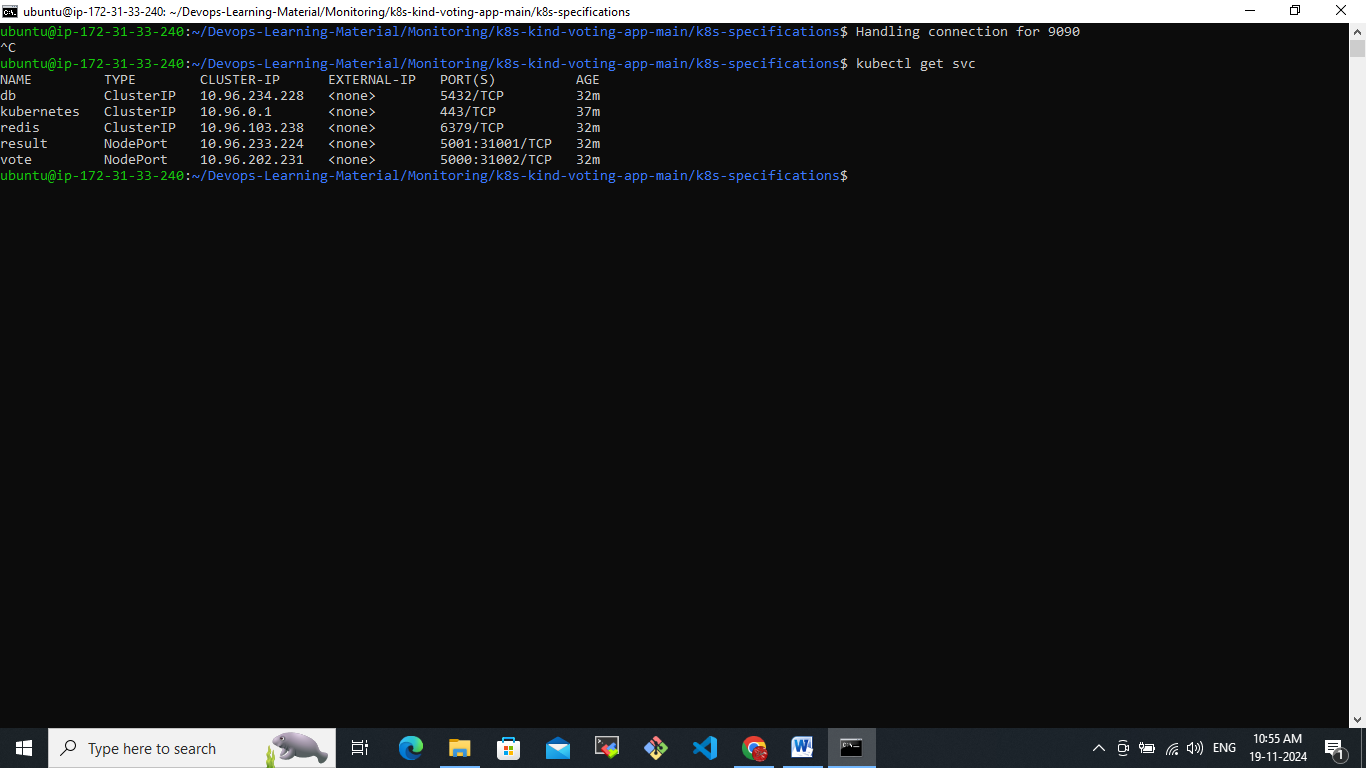




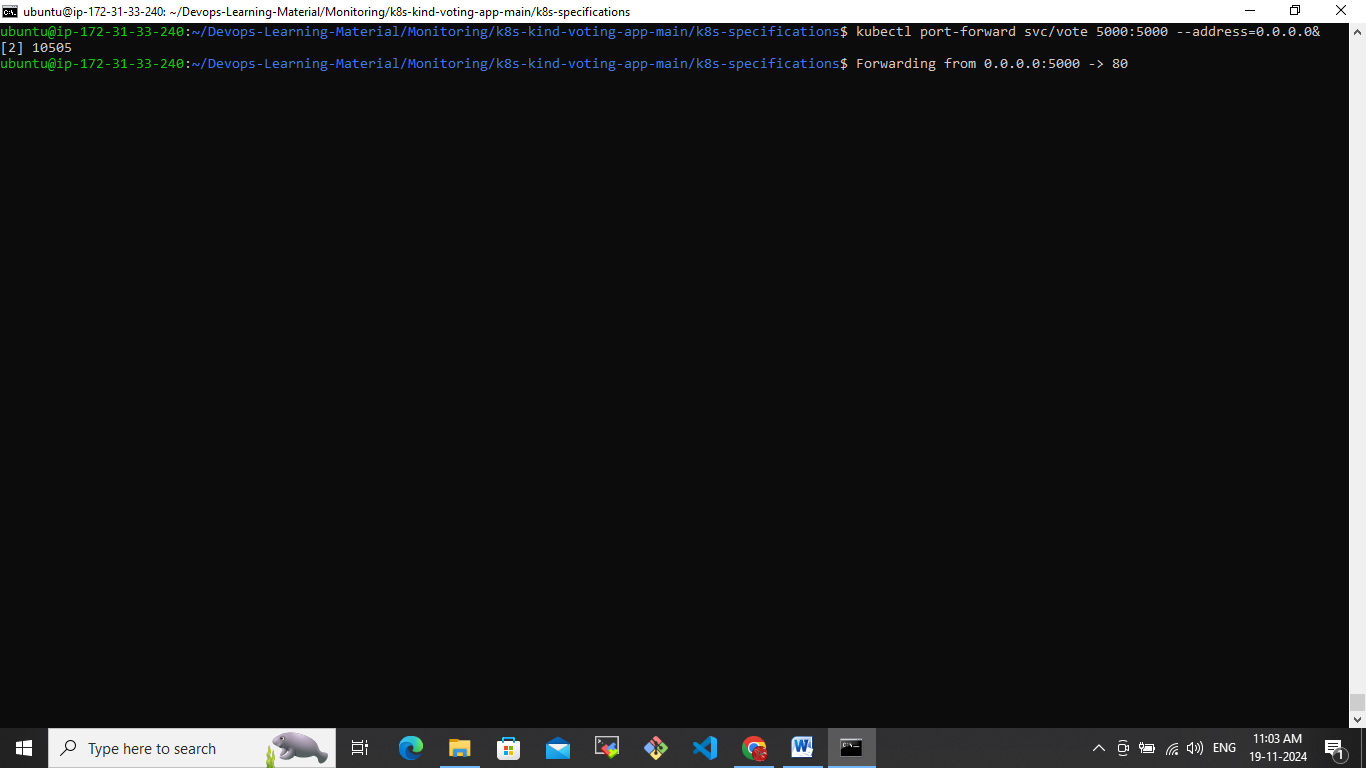


To expose our application

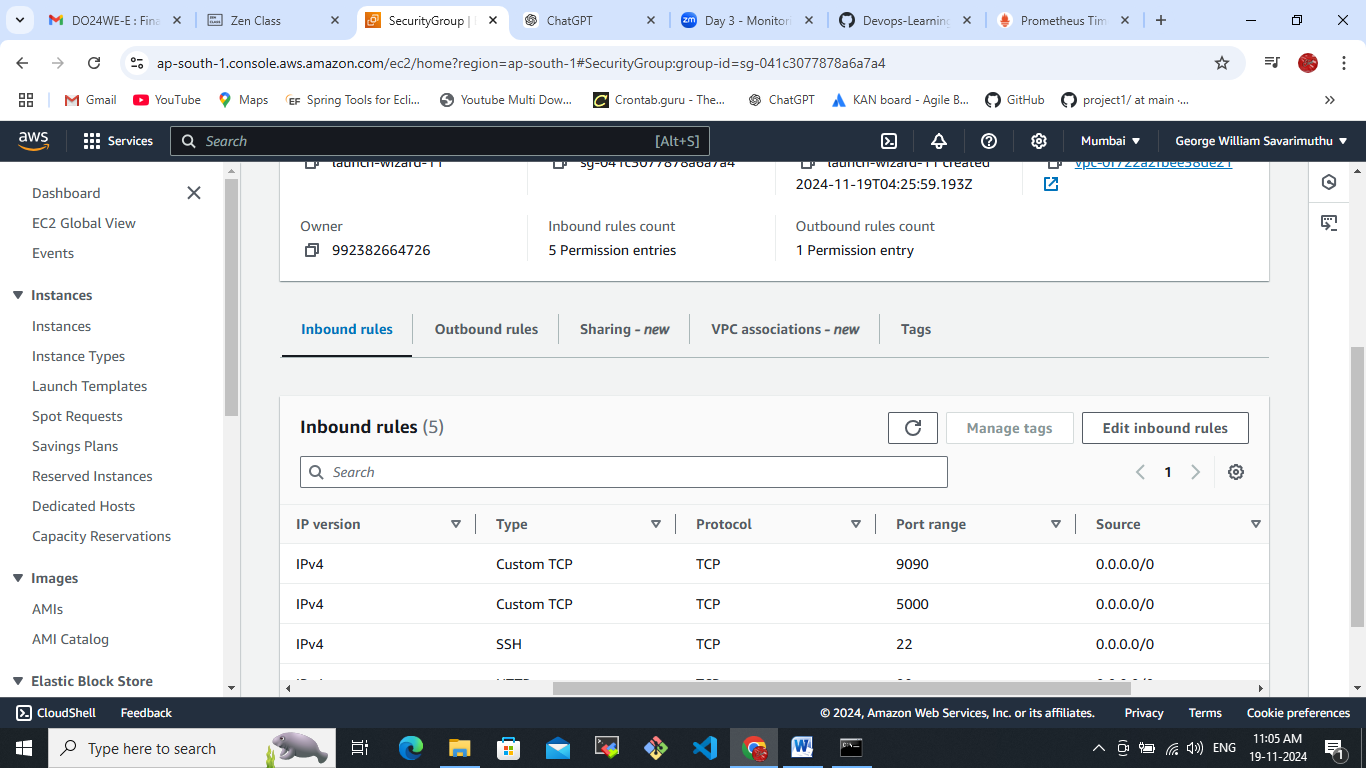
Kubectl get svc

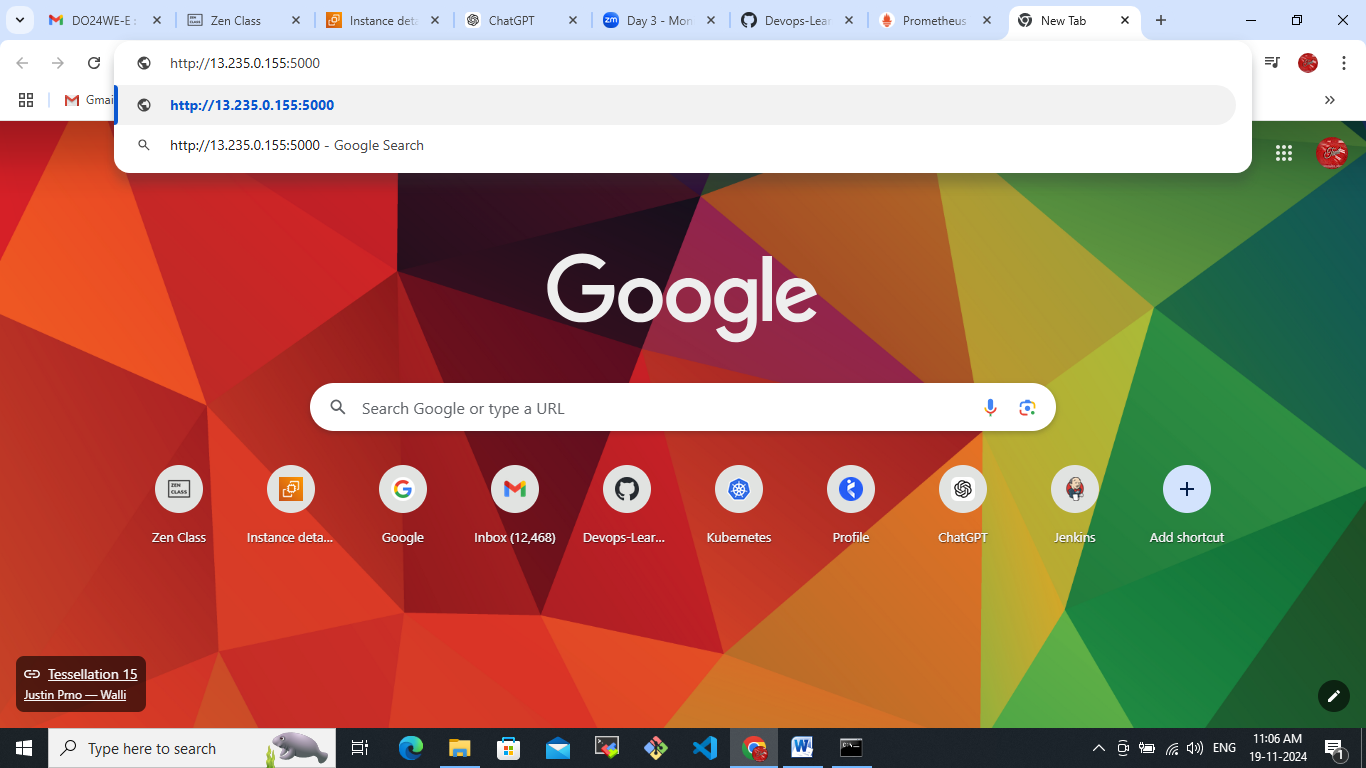


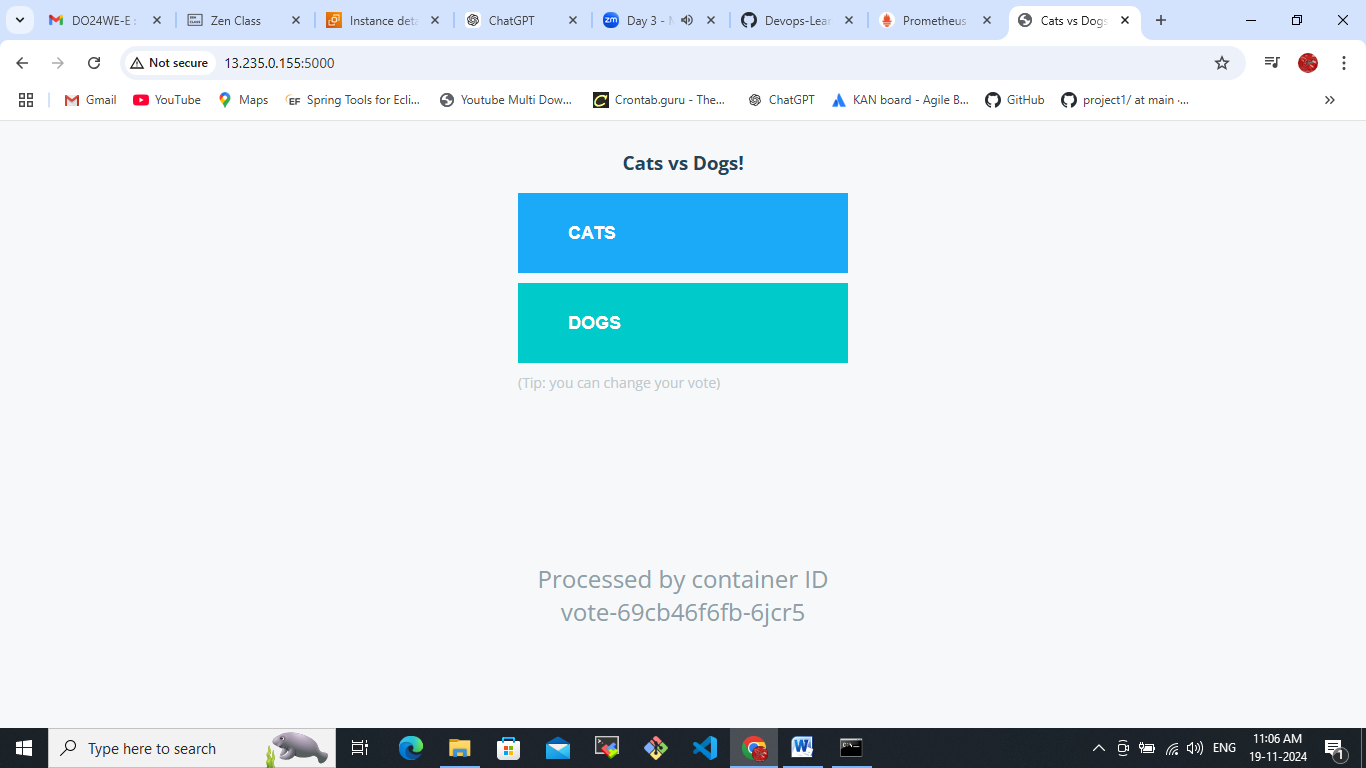
Kubectl port-forward svc/vote 5000:5000 --address=0.0.0.0&



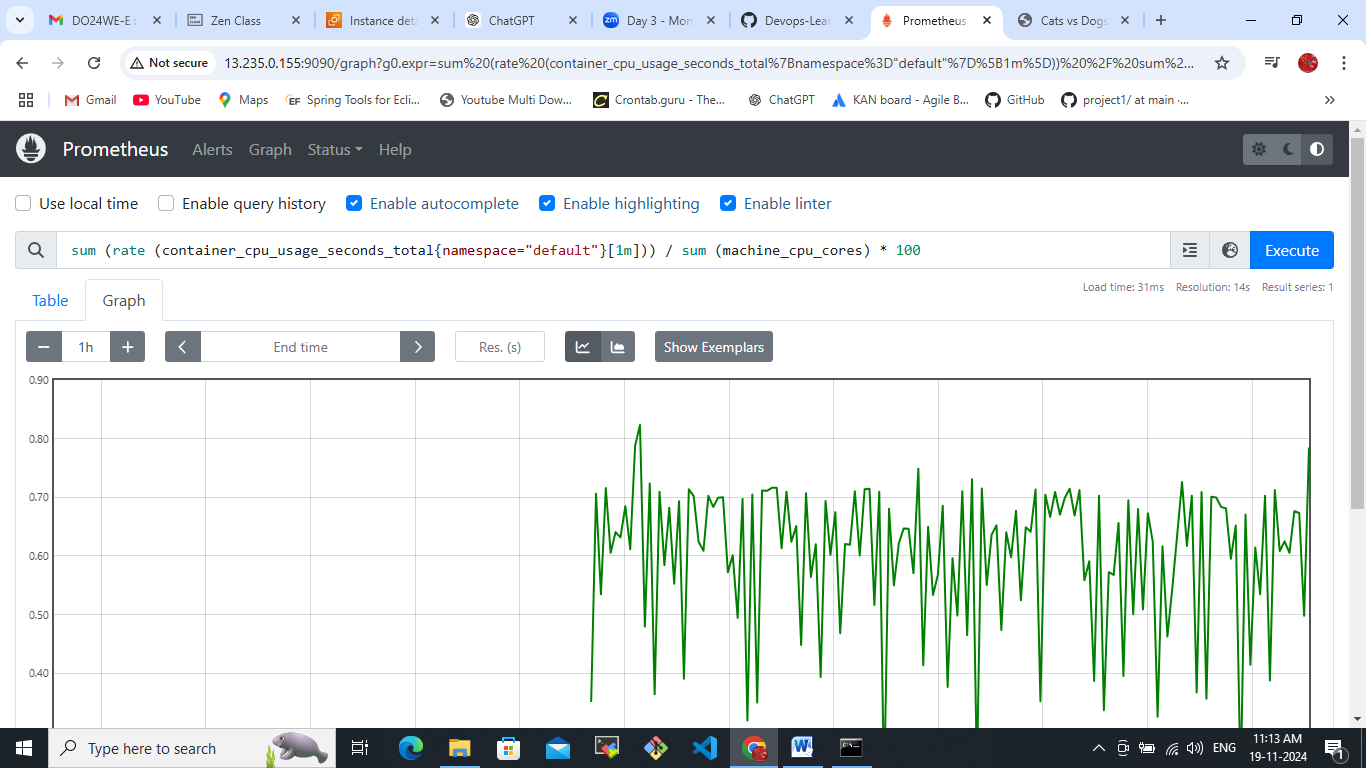
Now let’s add 5000 port in our inbound security groups





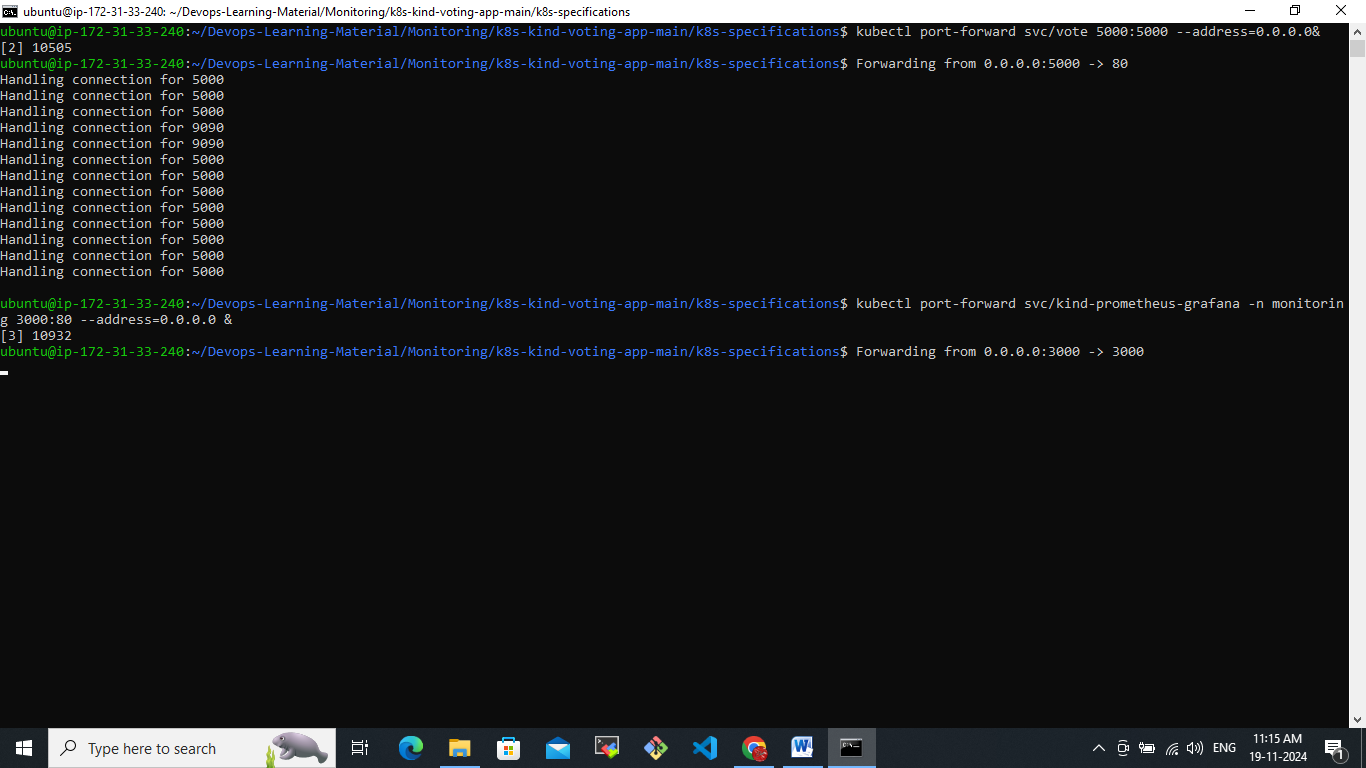


After running this application by just clicking cats or dogs application consumed the servers CPU now let s check same query with execute could see the more spike see since application used the servers utility



Now go to expose the grafana

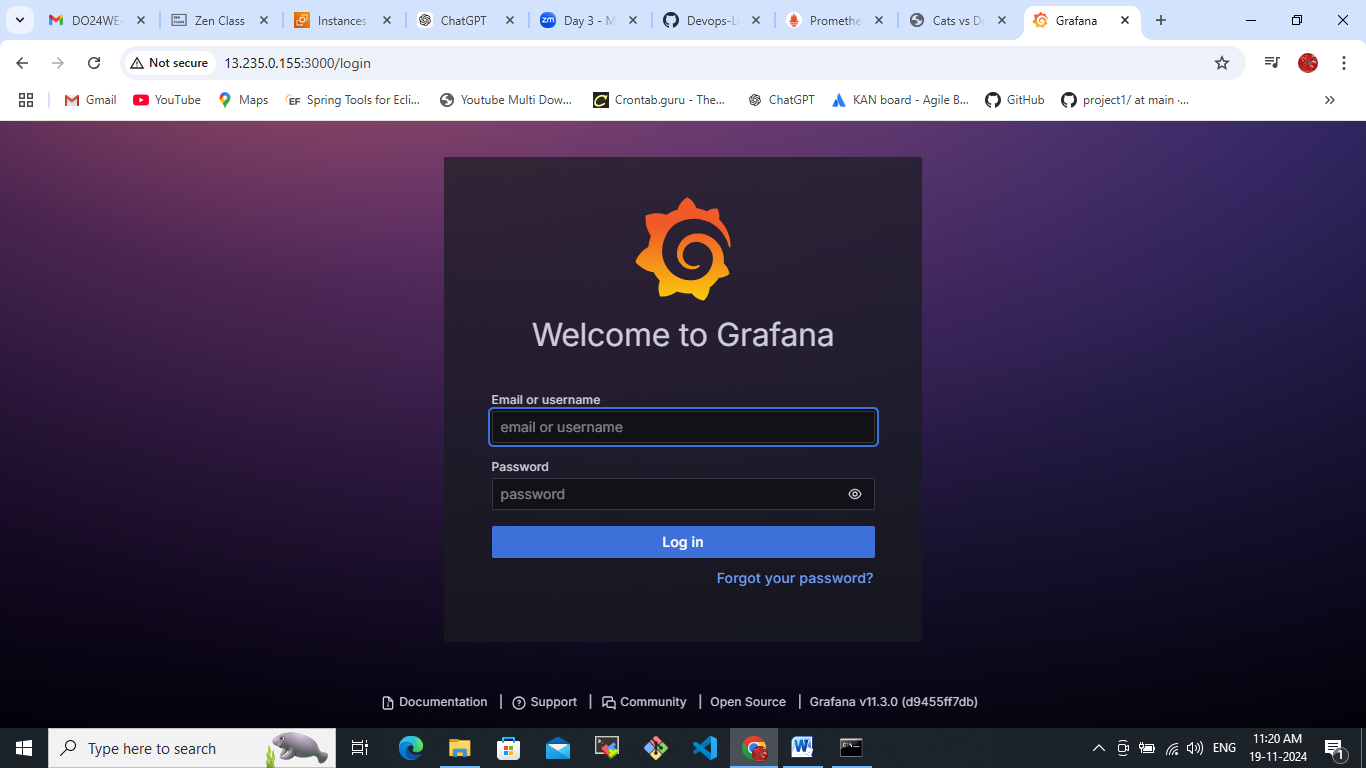
kubectl port-forward svc/kind-prometheus-grafana -n monitoring 3000:80 --address=0.0.0.0 &

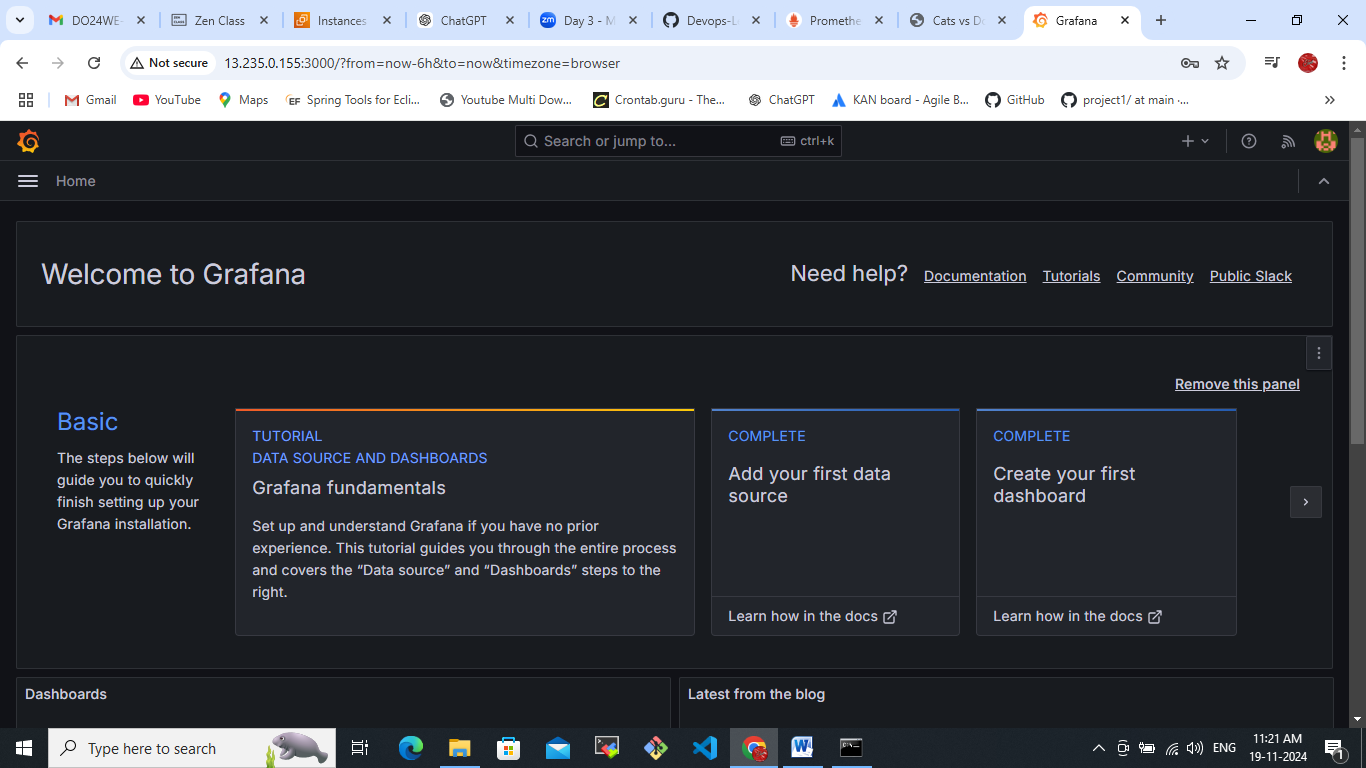


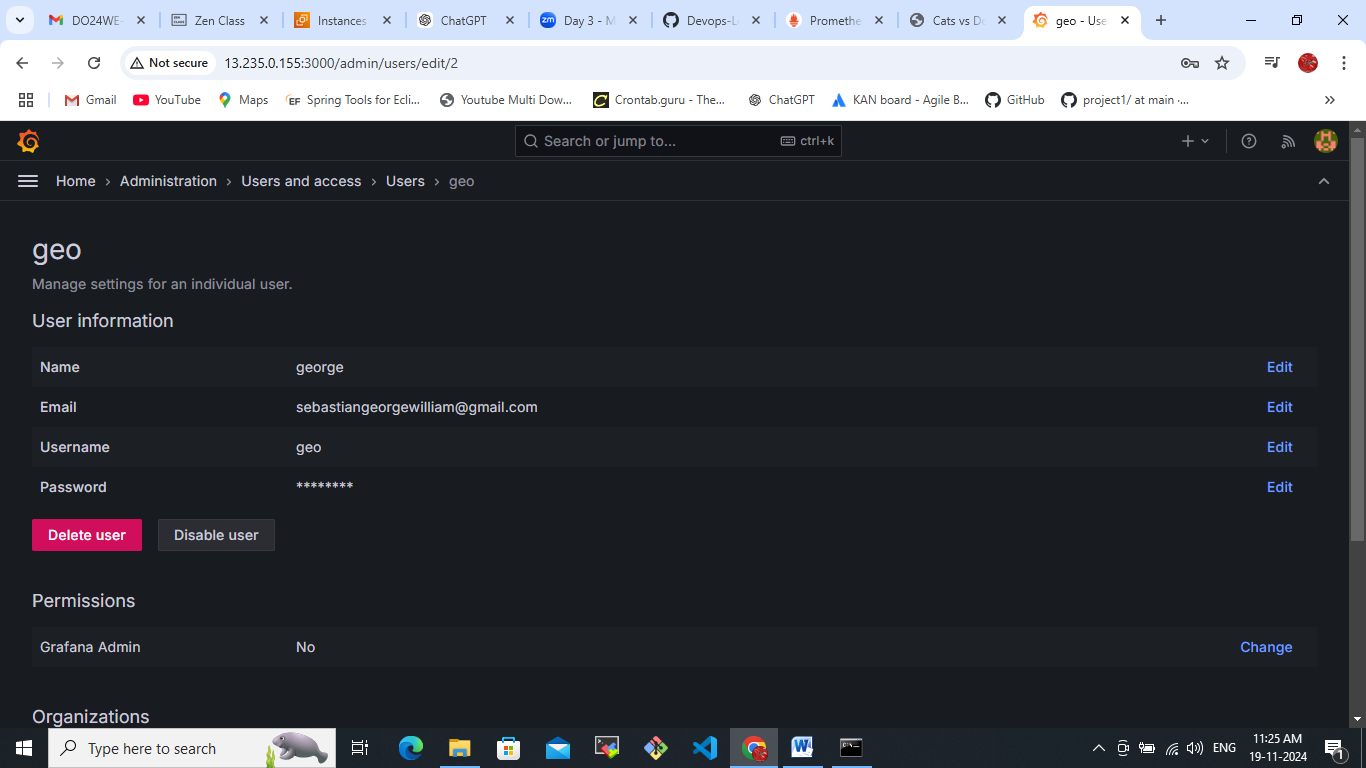
Now go ahead let’s see grafana dashboard by using port3000:80

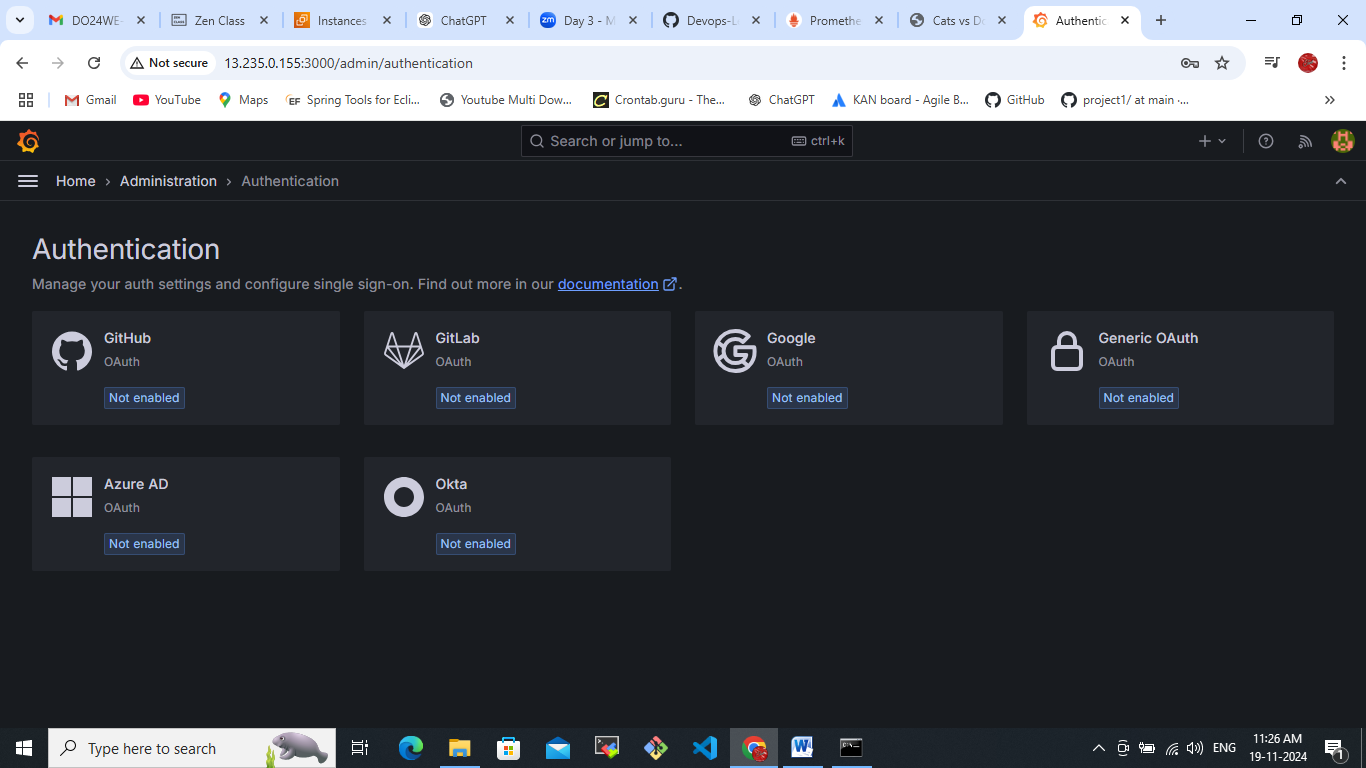
User name; admin

Password: prom-operator









We can enable to it for the authentication

Now add premotheus as new data source

