**Part1**

About proudest Achievement

**Part2**

High available and load-balanced cloud environment Kubernetes cluster using Kops

One of the first and biggest differences between EKS and kops is in how control and access are handled. With EKS, things such as managing the master node and configuring the cloud environment are handled by Amazon, leaving you with little to no control over them.

Kops, on the other hand, lets you configure your cloud environments the way you like them, including configuring them to meet specific needs. In the long run, the level of control provided by kops will enable better control over the efficiency of the cloud environment.

step:1

setting up the kops for creating three node cluster

For that creating and configuring IAM user with AdministratorAccess, creating the access key and the secret access key.

s3 bucket is created to store cluster state

With the kops create cluster command i created cluster

with --yes, kops print the list of the whole actions is going to do in my AWS account

step:2

Using infrastructure as a code tool called Jenkins to implement Continuous Integration and Continuous Deployment

creating the environment with docker registry and defining a variable docker image.

---stage1

pulling source code from my GitHub repository

---stage2

Building docker image

---stage3

Pushing image to Docker hub

stage4

Deploying mynginx.yaml to Kubernetes by adding cd plugin in Jenkins

**part3**

This part focuses on monitoring logs. It scans the log file and returns a report with errors in sessions along with the last three messages before the error, assuming the session ends when an error occurs.