Created EKS cluster with t3.micro 5 nodes

```
tamil@Aravind:~/kubetask$ kubectl get nodes
NAME                                          STATUS   ROLES    AGE   VERSION
ip-192-168-4-156.ap-south-1.compute.internal    Ready    <none>   71s   v1.32.9-eks-ecaa3a6
ip-192-168-43-193.ap-south-1.compute.internal   Ready    <none>   69s   v1.32.9-eks-ecaa3a6
ip-192-168-43-228.ap-south-1.compute.internal   Ready    <none>   70s   v1.32.9-eks-ecaa3a6
ip-192-168-69-137.ap-south-1.compute.internal   Ready    <none>   71s   v1.32.9-eks-ecaa3a6
ip-192-168-90-41.ap-south-1.compute.internal    Ready    <none>   71s   v1.32.9-eks-ecaa3a6
tamil@Aravind:~/kubetask$
```

Created deployment with nginx image

```
tamil@Aravind:~/kubetask$ kubectl create deployment nginx-deployment --image=nginx
deployment.apps/nginx-deployment created
tamil@Aravind:~/kubetask$ kubectl get deployments
NAME               READY   UP-TO-DATE   AVAILABLE   AGE
nginx-deployment   1/1     1            1           24s
tamil@Aravind:~/kubetask$
```

```
tamil@Aravind:~/kubetask$ kubectl get pods
NAME                                READY   STATUS    RESTARTS   AGE
nginx-deployment-6cfb98644c-rbgrp   1/1     Running   0          78s
tamil@Aravind:~/kubetask$
```

Created Loadbalancer service and exposed outside network

```
tamil@Aravind:~/kubetask$ kubectl get svc nginx-service
NAME            TYPE           CLUSTER-IP       EXTERNAL-IP                                                                PORT(S)        AGE
nginx-service   LoadBalancer   10.100.128.235   a64da97e1ec30444485bc1d2fc04e44c-2077906025.ap-south-1.elb.amazonaws.com   80:32316/TCP   12s
tamil@Aravind:~/kubetask$
```

Accessing application from browser