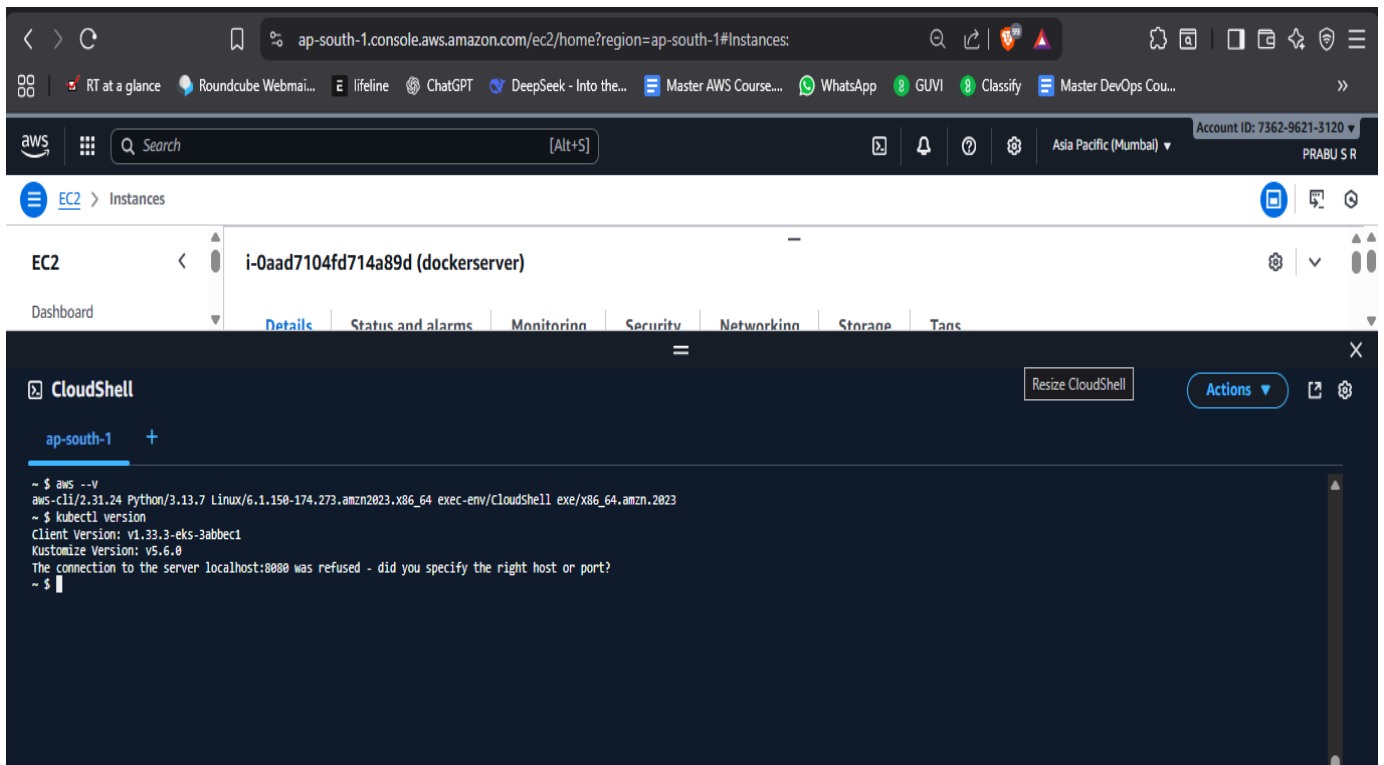


Kubernetes Task-2

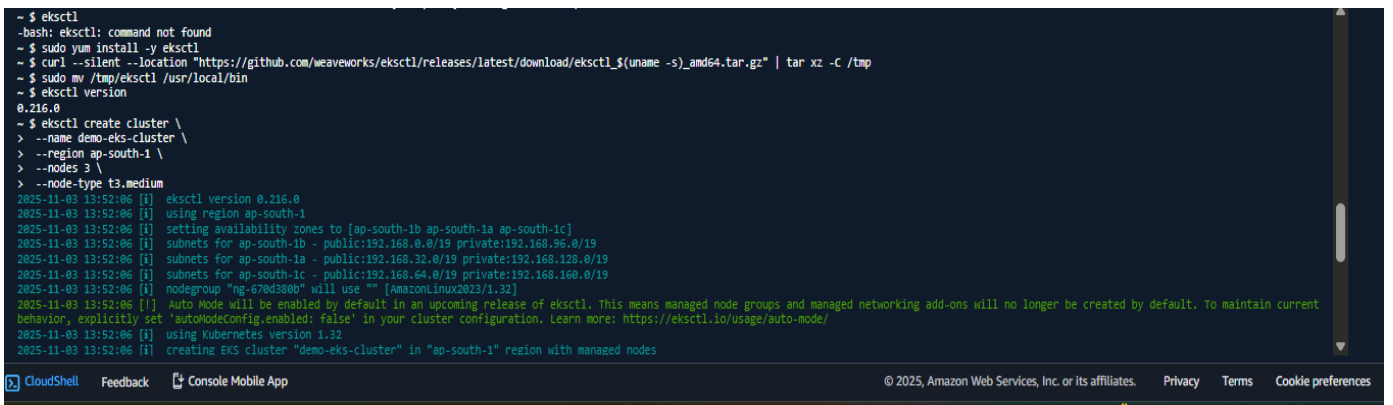
Task Description:

Create the K8s EKS, further you have to do the deployment of the Nginx application and access the application outside the cluster.

Kubernetes version check



eksctl version check and if not found, download manually and move to bin location and create cluster with 3 nodes, here t3.medium (free aws: t3.small only supported) i have given t3.small



CloudShell

ap-south-1



```
2025-11-03 13:52:06 [i] default addons coredns, metrics-server, vpc-cni, kube-proxy were not specified, will install them as EKS addons
2025-11-03 13:52:06 [i]
2 sequential tasks: { create cluster control plane "demo-eks-cluster",
  2 sequential sub-tasks: {
    2 sequential sub-tasks: {
      1 task: { create addons },
      wait for control plane to become ready,
    },
    create managed nodegroup "ng-670d380b",
  }
}
2025-11-03 13:52:06 [i] building cluster stack "eksctl-demo-eks-cluster-cluster"
2025-11-03 13:52:06 [i] deploying stack "eksctl-demo-eks-cluster-cluster"
2025-11-03 13:52:36 [i] waiting for CloudFormation stack "eksctl-demo-eks-cluster-cluster"
2025-11-03 13:53:06 [i] waiting for CloudFormation stack "eksctl-demo-eks-cluster-cluster"
2025-11-03 13:54:06 [i] waiting for CloudFormation stack "eksctl-demo-eks-cluster-cluster"
2025-11-03 13:55:06 [i] waiting for CloudFormation stack "eksctl-demo-eks-cluster-cluster"
2025-11-03 13:56:06 [i] waiting for CloudFormation stack "eksctl-demo-eks-cluster-cluster"
2025-11-03 13:57:07 [i] waiting for CloudFormation stack "eksctl-demo-eks-cluster-cluster"
2025-11-03 13:58:07 [i] waiting for CloudFormation stack "eksctl-demo-eks-cluster-cluster"
2025-11-03 13:59:07 [i] waiting for CloudFormation stack "eksctl-demo-eks-cluster-cluster"
2025-11-03 14:00:07 [i] waiting for CloudFormation stack "eksctl-demo-eks-cluster-cluster"
```

CloudShell

Actions



ap-south-1



```
2025-11-03 14:44:48 [i] no tasks
2025-11-03 14:44:48 [✓] all EKS cluster resources for "demo-eks-cluster" have been created
2025-11-03 14:44:49 [i] nodegroup "ng-bc9e0219" has 3 node(s)
2025-11-03 14:44:49 [i] node "ip-192-168-19-38.ap-south-1.compute.internal" is ready
2025-11-03 14:44:49 [i] node "ip-192-168-48-43.ap-south-1.compute.internal" is ready
2025-11-03 14:44:49 [i] node "ip-192-168-87-82.ap-south-1.compute.internal" is ready
2025-11-03 14:44:49 [i] waiting for at least 3 node(s) to become ready in "ng-bc9e0219"
2025-11-03 14:44:49 [i] nodegroup "ng-bc9e0219" has 3 node(s)
2025-11-03 14:44:49 [i] node "ip-192-168-19-38.ap-south-1.compute.internal" is ready
2025-11-03 14:44:49 [i] node "ip-192-168-48-43.ap-south-1.compute.internal" is ready
2025-11-03 14:44:49 [i] node "ip-192-168-87-82.ap-south-1.compute.internal" is ready
2025-11-03 14:44:49 [✓] created 1 managed nodegroup(s) in cluster "demo-eks-cluster"
2025-11-03 14:44:49 [i] creating addon: metrics-server
2025-11-03 14:44:49 [i] successfully created addon: metrics-server
2025-11-03 14:44:50 [i] kubectl command should work with "/home/cloudshell-user/.kube/config", try 'kubectl get nodes'
2025-11-03 14:44:50 [✓] EKS cluster "demo-eks-cluster" in "ap-south-1" region is ready
~ $
```

aws [Search] [Alt+S] Account ID: 7362-9621-3120 PRABU S R Asia Pacific (Mumbai)

EC2 > Instances

Instances (3) Info Last updated 9 minutes ago Connect Instance state Actions Launch instances

Find Instance by attribute or tag (case-sensitive) All states

<input type="checkbox"/>	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone
<input type="checkbox"/>	demo-eks-cluster-ng-bc9e0219-N...	i-0b9b9eac99a77923	Running	t3.small	Initializing	View alarms +	ap-south-1b
<input type="checkbox"/>	demo-eks-cluster-ng-bc9e0219-N...	i-0c1416699ac12212f	Running	t3.small	Initializing	View alarms +	ap-south-1a
<input type="checkbox"/>	demo-eks-cluster-ng-bc9e0219-N...	i-001dee657788f01a6	Running	t3.small	Initializing	View alarms +	ap-south-1c

i-0aad7104fd714a89d (dockerserver)

Details Status and alarms Monitoring Security Networking Storage Tags

▼ Instance summary Info

Instance ID i-0aad7104fd714a89d	Public IPv4 address -	Private IPv4 addresses -
IPv6 address	Instance state	Public DNS

create nginx-deployment.yaml file

aws [Search] [Alt+S] Account ID: 7362-9621-3120 PRABU S R Asia Pacific (Mumbai)

EC2 > Instances

CloudShell

ap-south-1 +

```
GNU nano 8.3 nginx-deployment.yaml
apiversion: apps/v1
kind: Deployment
metadata:
  name: nginx-deployment
spec:
  selector:
    matchLabels:
      app: nginx
  replicas: 2 # tells deployment to run 2 pods matching the template
  template:
    metadata:
      labels:
        app: nginx
    spec:
      containers:
        - name: nginx
          image: nginx:1.14.2
          ports:
            - containerPort: 80
```

CloudShell Feedback Console Mobile App © 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

a

```
2025-11-03 14:44:49 [i] Successfully created addon: metrics-server
2025-11-03 14:44:50 [i] kubectl command should work with "/home/cloudshell-user/.kube/config", try 'kubectl get nodes'
2025-11-03 14:44:50 [✓] EKS cluster "demo-eks-cluster" in "ap-south-1" region is ready
~ $ nano nginx-deployment.yaml
~ $ kubectl apply -f nginx-deployment.yaml
deployment.apps/nginx-deployment created
~ $
```

CloudShell Feedback Console Mobile App

```
deployment.apps/nginx-deployment created
~ $ kubectl get pods
NAME                                READY   STATUS    RESTARTS   AGE
nginx-deployment-647677fc66-bn9gv   1/1     Running   0           20s
nginx-deployment-647677fc66-wpxbg   1/1     Running   0           20s
~ $
```

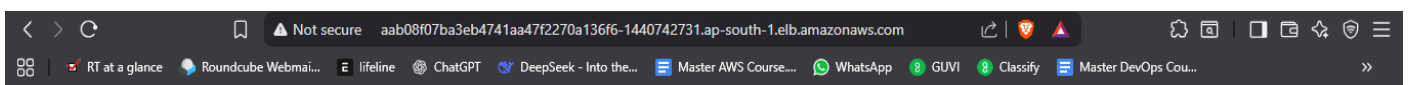
create nginx-service.yaml

```
~ $ kubectl apply -f nginx-service.yaml
error: error parsing nginx-service.yaml: error converting YAML to JSON: yaml: line 12: found a tab character that violates indentation
~ $ nano nginx-service.yaml
~ $ kubectl apply -f nginx-service.yaml
service/nginx-service created
~ $
```

run the service url(external ip)

```
~ $ kubectl get svc nginx-service --watch
NAME      TYPE        CLUSTER-IP      EXTERNAL-IP      PORT(S)      AGE
nginx-service  LoadBalancer  10.100.199.108   aab08f07ba3eb4741aa47f2270a136f6-1440742731.ap-south-1.elb.amazonaws.com  80:30911/TCP  58s
```

paste in browser and see nginx webpage



common cmds to check logs and datas inside clusters

```
^C~ $ kubectl get pods -o wide
NAME                                READY   STATUS    RESTARTS   AGE   IP              NODE                                NOMINATED NODE   READINESS GATES
nginx-deployment-647677fc66-bn9gv   1/1     Running   0           6m21s  192.168.74.59   ip-192-168-87-82.ap-south-1.compute.internal  <none>            <none>
nginx-deployment-647677fc66-wpxbg   1/1     Running   0           6m21s  192.168.20.211  ip-192-168-19-38.ap-south-1.compute.internal  <none>            <none>
~ $
```

```
~ $ kubectl get all
NAME                                READY   STATUS    RESTARTS   AGE
pod/nginx-deployment-647677fc66-bn9gv  1/1     Running   0           6m53s
pod/nginx-deployment-647677fc66-wpxbg   1/1     Running   0           6m53s

NAME                                TYPE        CLUSTER-IP      EXTERNAL-IP      PORT(S)      AGE
service/kubernetes                  ClusterIP    10.100.0.1       <none>            443/TCP      27m
service/nginx-service               LoadBalancer 10.100.199.108   aab08f07ba3eb4741aa47f2270a136f6-1440742731.ap-south-1.elb.amazonaws.com  80:30911/TCP  3m14s

NAME                                READY   UP-TO-DATE   AVAILABLE   AGE
deployment.apps/nginx-deployment    2/2     2             2           6m53s

NAME                                DESIRED   CURRENT   READY   AGE
replicaset.apps/nginx-deployment-647677fc66  2         2         2       6m53s
~ $
```

```

~ $ kubectl describe svc nginx-service
Name:          nginx-service
Namespace:     default
Labels:        <none>
Annotations:   <none>
Selector:      app=nginx
Type:          LoadBalancer
IP Family Policy: SingleStack
IP Families:   IPv4
IP:            10.100.199.108
IPs:           10.100.199.108
LoadBalancer Ingress: aab08f07ba3eb4741aa47f2270a136f6-1440742731.ap-south-1.elb.amazonaws.com
Port:          <unset> 80/TCP
TargetPort:    80/TCP
NodePort:      <unset> 30911/TCP
Endpoints:     192.168.20.211:80,192.168.74.59:80
Session Affinity: None
External Traffic Policy: Cluster
Internal Traffic Policy: Cluster
Events:
  Type     Reason              Age   From                      Message
  ----     -
  Normal   EnsuringLoadBalancer 4m2s  service-controller        Ensuring load balancer
  Normal   EnsuredLoadBalancer 3m59s service-controller        Ensured load balancer
~ $

```

resource cleanup process

CloudShell

ap-south-1



```

~ $ eksctl delete cluster --name demo-eks-cluster --region ap-south-1
2025-11-03 15:07:01 [i] deleting EKS cluster "demo-eks-cluster"
2025-11-03 15:07:01 [i] will drain 0 unmanaged nodegroup(s) in cluster "demo-eks-cluster"
2025-11-03 15:07:01 [i] starting parallel draining, max in-flight of 1
2025-11-03 15:07:01 [i] deleted 0 Fargate profile(s)
2025-11-03 15:07:01 [✓] kubeconfig has been updated
2025-11-03 15:07:01 [i] cleaning up AWS load balancers created by Kubernetes objects of Kind Service or Ingress

```