## **Phase-5 Practice Project: Assisted Practice -**

4. Deploy your Application:

wget https://amazon-eks.s3-us-west-2.amazonaws.com/1.10.3/2018-07-26/bin/linux/amd64/kubectl chmod +x

kubectl

./kubectl

mkdir bin cp ./kubectl \$HOME/bin/kubectl && export PATH=\$HOME/bin:\$PATH

kubectl version

kubectl version --short --client

```
root@ip-172-31-17-73:~# mkdir bin
root@ip-172-31-17-73:~# cp ./kubectl $HOME/bin/kubectl && export PATH=$HOME/bin:$PATH
root@ip-172-31-17-73:~# kubectl version
Client Version: version.Info{Major:"1", Minor:"10", GitVersion:"v1.10.3", GitCommit:"2bbaC-26T20:40:11Z", GoVersion:"go1.9.3", Compiler:"gc", Platform:"linux/amd64"}
```

wget https://amazon-eks.s3-us-west-2.amazonaws.com/1.10.3/2018-07-26/bin/linux/amd64/aws-iam-authenticator chmod +x ./aws-

iam-authenticator

cp ./aws-iam-authenticator \$HOME/bin/aws-iam-authenticator && export PATH=\$HOME/bin:\$PATH

aws-iam-authenticator help

curl --silent --location

# "https://github.com/weaveworks/eksctl/releases/download/latest\_release/ eksctl\_\$(uname -s)\_amd64.tar.gz" | tar xz -C /tmp mv /tmp/eksctl /usr/local/bin

### eksctl version

#### Access keys

Use access keys to make secure REST or HTTP Query protocol requests to AWS service APIs. For your protection, you should never share your secret keys with anyone. As a best practice, we recommend frequent key rotation. Learn more

#### Create access key

Acce	Access key ID	Created	Last used	Status	
	AKIAVORWYFFGC3WVPNWC	2019-07-24 08:28 UTC+0530	2019-07-26 13:51 UTC+0530 with sts in us-east-1	Active   Make inactive	×

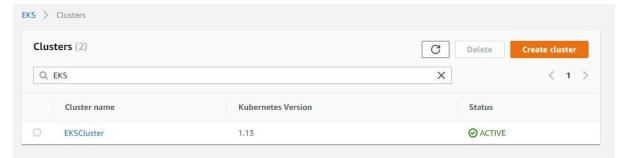
#### Create access key

Access key ID	Created 2019-07-24 08:28 UTC+0530	Last used 2019-07-26 13:51 UTC+0530 with sts in us-east-1	Status		
AKIAVORWYFFGC3WVPNWC			Active	Make inactive	×
AKIAVORWYFFGE3YTFZFZ	2019-07-28 07:49 UTC+0530	N/A	Active	Make inactive	×

```
root@ip-172-31-17-73:~# aws configure
AWS Access Key ID [None]: AKIAVORWYFFGE3YTFZFZ
AWS Secret Access Key [None]: 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
Default region name [None]: us-east-1
Default output format [None]: json
root@ip-172-31-17-73:~#
```

```
[â] all EKS cluster resource for "EKSCluster" had been created
[â] saved kubeconfig as "/root/.kube/config"
[â]
adding role "arn:aws:iam::130374862735:role/eksctl-EKSCluster-nodegroup-ng-c8-NodeInstanceRole-1FKZC9GNJUMU" to auth ConfigMap
[â]
nodegroup "ng-c8e07a6f" has 0 node(s)
[â]
1] waiting for at least 2 node(s) to become ready in "ng-c8e07a6f"
[â]
1] nodegroup "ng-c8e07a6f" has 2 node(s)
[â]
1] node "ip-192-168-28-149.us-west-2.compute.internal" is ready
[â]
1] node "ip-192-168-76-186.us-west-2.compute.internal" is ready
[â]
2]
2 kubectl command should work with "/root/.kube/config", try 'kubectl get nodes'
[â] EKS cluster "EKSCluster" in "us-west-2" region is ready
```

ot@ip-172-31-86-69:~# kubectl get node NAME STATUS AGE VERSION ip-192-168-28-149.us-west-2.compute.internal Ready <none> 5m v1.13.7-eks-c57ff8 ip-192-168-76-186.us-west-2.compute.internal v1.13.7-eks-c57ff8 Ready <none> 5m root@ip-172-31-86-69:~#



ot@ip-172-31-86-69:~‡ kubectl run kubernetes-bootcamp --image=docker.io/jocatalin/kubernetes-bootcamp:v1 --port=8080 deployment.apps "kubernetes-bootcamp" created root@ip-172-31-86-69:~# kubectl expose deployment/kubernetes-bootcamp --port=8080 --target-port=8080 --type=NodePort service "kubernetes-bootcamp" exposed root@ip-172-31-86-69:~# kubectl get pods READY STATUS RESTARTS NAME

kubernetes-bootcamp-6c5cfd894b-9jqzf 0/1 ContainerCreating
root@ip-172-31-86-69:~# kubectl get deployments

NAME

DESIRED CURRENT UP-TO-DATE AVAILABLE
kubernetes-bootcamp 1 1 1 1
root@ip-172-31-86-69:~# kubectl get pods ContainerCreating 63 NAME READY RESTARTS AGE kubernetes-bootcamp-6c5cfd894b-9jqzf 1/1 root@ip-172-31-86-69:~# kubectl get services Running 0 193 TYPE CLUSTER-IP ClusterIP 10.100.0.1 NAME EXTERNAL-IP PORT (S) kubernetes <none> 443/TCP 44m 1m kubernetes-bootcamp NodePort 10.100.33.238 <none> root@ip-172-31-86-69:~#