WEEK 7 ASSIGNMENT 7 - AWS EKS

Create Cluster using eksctl

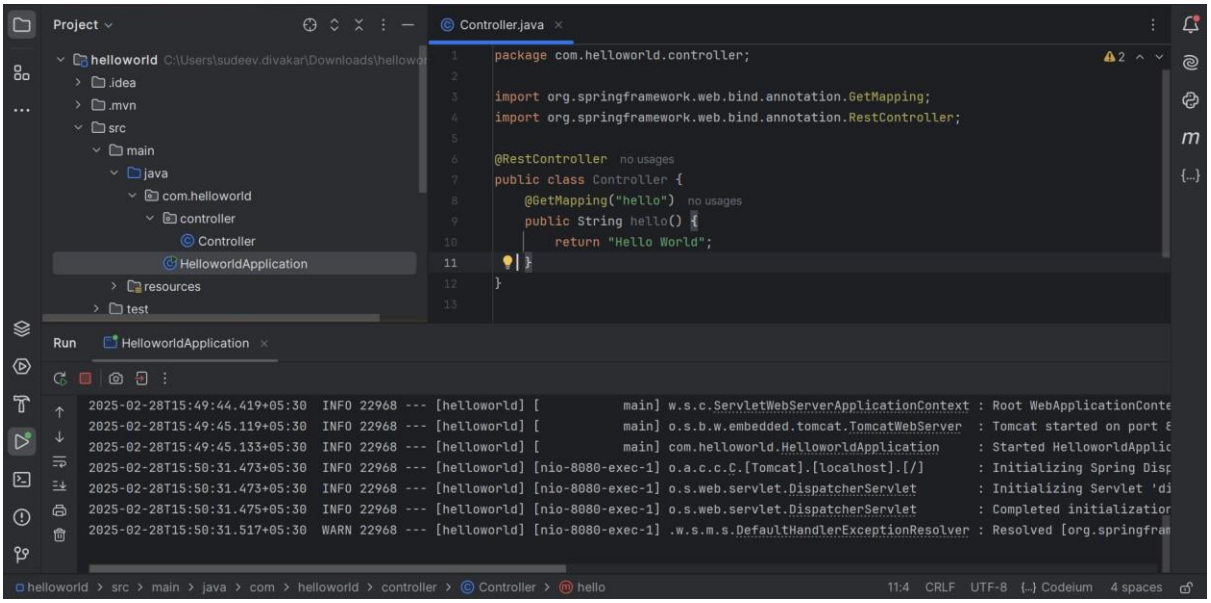
```
C:\Users\sudeev.divakar>eksctl create cluster --name sudeev-clus
ter --version 1.28 --nodes=1 --node-type=t2.small --region ap-so
uth-1
2025-02-28 15:27:43 [ℹ]  eksctl version 0.204.0
2025-02-28 15:27:43 [ℹ]  using region ap-south-1
2025-02-28 15:27:43 [ℹ]  skipping ap-south-1c from selection because it doesn't support the following instance type(s): t2.small
2025-02-28 15:27:43 [ℹ]  setting availability zones to [ap-south-1a ap-south-1b]
2025-02-28 15:27:43 [ℹ]  subnets for ap-south-1a - public:192.168.0.0/19 private:192.168.64.0/19
2025-02-28 15:27:43 [ℹ]  subnets for ap-south-1b - public:192.168.32.0/19 private:192.168.96.0/19
2025-02-28 15:27:43 [ℹ]  nodegroup "ng-883c7509" will use "" [AmazonLinux2/1.28]
2025-02-28 15:27:43 [ℹ]  using Kubernetes version 1.28
2025-02-28 15:27:43 [ℹ]  creating EKS cluster "sudeev-cluster" in "ap-south-1" region with managed nodes
2025-02-28 15:27:43 [ℹ]  will create 2 separate CloudFormation stacks for cluster itself and the initial managed nodegroup
2025-02-28 15:27:43 [ℹ]  if you encounter any issues, check CloudFormation console or try 'eksctl utils describe-stacks --region=ap-s
outh-1 --cluster=sudeev-cluster'
2025-02-28 15:27:43 [ℹ]  Kubernetes API endpoint access will use default of {publicAccess=true, privateAccess=false} for cluster "sud
eev-cluster" in "ap-south-1"
2025-02-28 15:27:43 [ℹ]  CloudWatch logging will not be enabled for cluster "sudeev-cluster" in "ap-south-1"
2025-02-28 15:27:43 [ℹ]  you can enable it with 'eksctl utils update-cluster-logging --enable-types={SPECIFY-YOUR-LOG-TYPES-HERE (e.g
. all)} --region=ap-south-1 --cluster=sudeev-cluster'
2025-02-28 15:27:43 [ℹ]  default addons metrics-server, vpc-cni, kube-proxy, coredns were not specified, will install them as EKS add
ons
2025-02-28 15:27:43 [ℹ]
2 sequential tasks: { create cluster control plane "sudeev-cluster",
    2 sequential sub-tasks: {
        2 sequential sub-tasks: {
            1 task: { create addons },
```

```
2025-02-28 15:36:48 [ℹ]  successfully created addon: kube-proxy
2025-02-28 15:36:49 [ℹ]  creating addon: coredns
2025-02-28 15:36:49 [ℹ]  successfully created addon: coredns
2025-02-28 15:38:50 [ℹ]  building managed nodegroup stack "eksctl-sudeev-cluster-nodegroup-ng-883c7509"
2025-02-28 15:38:50 [ℹ]  deploying stack "eksctl-sudeev-cluster-nodegroup-ng-883c7509"
2025-02-28 15:38:50 [ℹ]  waiting for CloudFormation stack "eksctl-sudeev-cluster-nodegroup-ng-883c7509"
2025-02-28 15:39:21 [ℹ]  waiting for CloudFormation stack "eksctl-sudeev-cluster-nodegroup-ng-883c7509"
2025-02-28 15:40:16 [ℹ]  waiting for CloudFormation stack "eksctl-sudeev-cluster-nodegroup-ng-883c7509"
2025-02-28 15:41:56 [ℹ]  waiting for CloudFormation stack "eksctl-sudeev-cluster-nodegroup-ng-883c7509"
2025-02-28 15:41:56 [ℹ]  waiting for the control plane to become ready
2025-02-28 15:41:58 [✔]  saved kubeconfig as "C:\\Users\\sudeev.divakar\\.kube\\config"
2025-02-28 15:41:58 [ℹ]  no tasks
2025-02-28 15:41:58 [✔]  all EKS cluster resources for "sudeev-cluster" have been created
2025-02-28 15:41:58 [ℹ]  nodegroup "ng-883c7509" has 1 node(s)
2025-02-28 15:41:58 [ℹ]  node "ip-192-168-29-93.ap-south-1.compute.internal" is ready
2025-02-28 15:41:58 [ℹ]  waiting for at least 1 node(s) to become ready in "ng-883c7509"
2025-02-28 15:41:58 [ℹ]  nodegroup "ng-883c7509" has 1 node(s)
2025-02-28 15:41:58 [ℹ]  node "ip-192-168-29-93.ap-south-1.compute.internal" is ready
2025-02-28 15:41:58 [✔]  created 1 managed nodegroup(s) in cluster "sudeev-cluster"
2025-02-28 15:42:02 [ℹ]  kubectl command should work with "C:\\Users\\sudeev.divakar\\.kube\\config", try 'kubectl get nodes'
2025-02-28 15:42:02 [✔]  EKS cluster "sudeev-cluster" in "ap-south-1" region is ready

C:\Users\sudeev.divakar>kubectl get nodes
NAME                                           STATUS   ROLES    AGE     VERSION
ip-192-168-29-93.ap-south-1.compute.internal   Ready    <none>   2m45s   v1.28.15-eks-aeac579

C:\Users\sudeev.divakar>
```

Simple Hello World Spring Boot Application

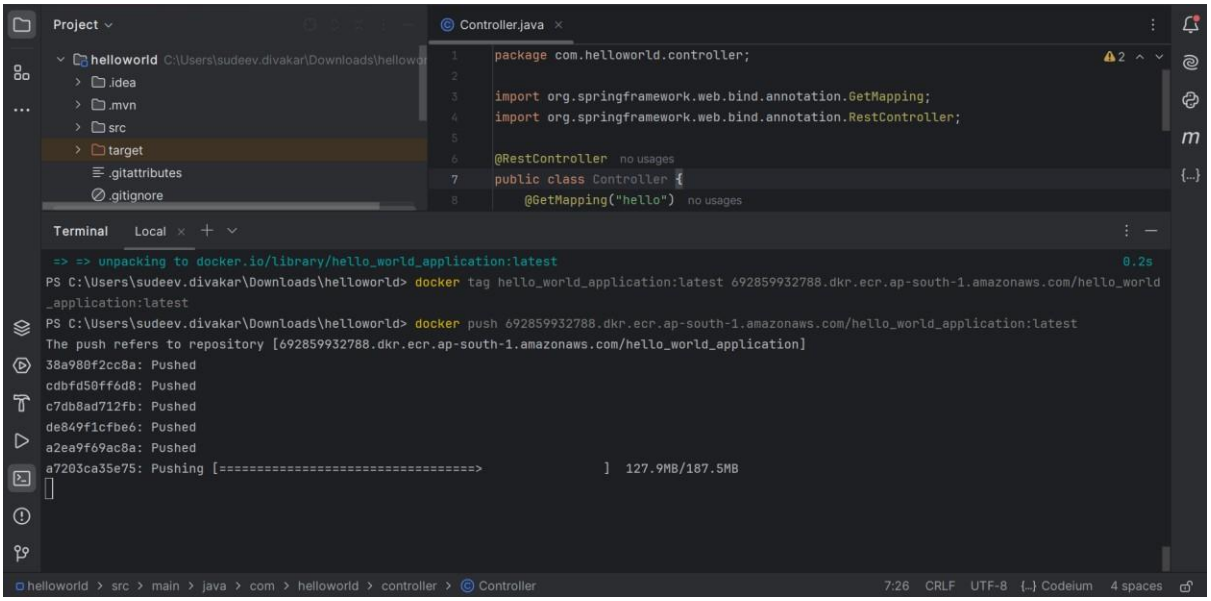


Create AWS ECR repository

```
C:\Users\sudeev.divakar>aws ecr create-repository --repository-name hello_world_application
-----------------------------------------------------------------------------------------------------
|                                        CreateRepository                                           |
+---------------------------------------------------------------------------------------------------+
||                                          repository                                             ||
|+--------------------+--------------------------------------------------------------------------+|
||  createdAt         |  2025-02-28T15:54:19.253000+05:30                                        ||
||  imageTagMutability|  MUTABLE                                                                 ||
||  registryId        |  692859932788                                                            ||
||  repositoryArn     |  arn:aws:ecr:ap-south-1:692859932788:repository/hello_world_application  ||
||  repositoryName    |  hello_world_application                                                 ||
||  repositoryUri     |  692859932788.dkr.ecr.ap-south-1.amazonaws.com/hello_world_application   ||
|+--------------------+--------------------------------------------------------------------------+|
||                                   encryptionConfiguration                                       ||
|+-----------------------------------------------------+-------------------------------------+|
||  encryptionType                                     |  AES256                             ||
|+-----------------------------------------------------+-------------------------------------+|
||                                imageScanningConfiguration                                      ||
|+-----------------------------------------------------+-------------------------------------+|
||  scanOnPush                                         |  False                              ||
|+-----------------------------------------------------+-------------------------------------+|


C:\Users\sudeev.divakar>aws ecr get-login-password --region ap-south-1 | docker login --username AWS --password-stdin 69
2859932788.dkr.ecr.ap-south-1.amazonaws.com
Login Succeeded
```

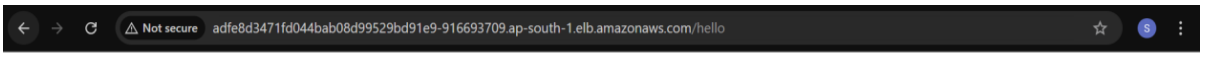
Dockerize application and push onto AWS ECR




```
PS C:\Users\sudeev.divakar\Downloads\helloworld> kubectl apply -f deployment.yaml
deployment.apps/hello-world-application created
PS C:\Users\sudeev.divakar\Downloads\helloworld> kubectl apply -f service.yaml
service/hello-world-service created
```

```
PS C:\Users\sudeev.divakar\Downloads\helloworld> kubectl get pods
NAME                                     READY   STATUS    RESTARTS        AGE
hello-world-application-67db74dddb-cpwvb   1/1     Running   0               14s
hello-world-application-67db74dddb-vflbb   1/1     Running   8 (6m8s ago)    17m
```

```
PS C:\Users\sudeev.divakar\Downloads\helloworld> kubectl get svc
NAME                TYPE           CLUSTER-IP      EXTERNAL-IP                                                              PORT(S)        AGE
hello-world-service   LoadBalancer   10.100.50.179   adfe8d3471fd044bab08d99529bd91e9-916693709.ap-south-1.elb.amazonaws.com   80:30099/TCP   18m
kubernetes            ClusterIP      10.100.0.1      <none>                                                                   443/TCP        65m
PS C:\Users\sudeev.divakar\Downloads\helloworld>
```


Not secure
adfe8d3471fd044bab08d99529bd91e9-916693709.ap-south-1.elb.amazonaws.com/hello

Hello World