Replace **<your-name>** with your **name** throughout the Lab.

1. SSH to the your **AWS WorkStation.**

|  |
| --- |
| $ sudo su  # vim java-app-<your-name>.yaml |

2. **Paste the below script in the java\_app.yaml**

Update the image: **lovescloud/java-app-merc:latest** and replace <yourname> with your namein the below script with your dockerhub image name that you uploaded to docker hub in docker lab Pushing images to docker Hub for java-app.

|  |
| --- |
| apiVersion: apps/v1  kind: Deployment  metadata:  name: java-app-<your-name>  spec:  selector:  matchLabels:  run: java-app-<your-name>  replicas: 2  template:  metadata:  labels:  run: java-app-<your-name>  spec:  containers:  - name: java-app-<your-name>  image: lovescloud/java-app-merc:latest  ports:  - name: port8080  containerPort: 8080  ---  apiVersion: v1  kind: Service  metadata:  name: java-app-<your-name>  labels:  run: java-app-<your-name>  spec:  type: NodePort  ports:  - name: port8080  port: 8080  protocol: TCP  selector:  run: java-app-<your-name> |

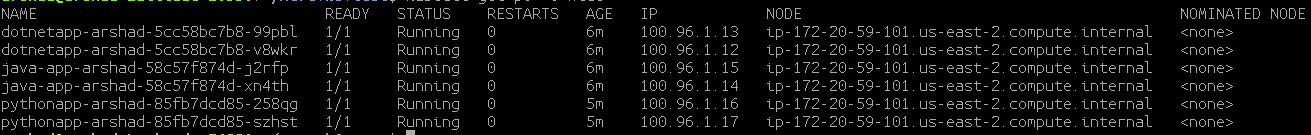
**Save and exit by pressing the ESC key and type wq to save and quit by pressing enter**

3. Run the below commands to deloy the JAVA application on your Kubernetes Cluster

|  |
| --- |
| # kubectl apply -f java-app-<your-name>.yaml |

4. Check the **NODE** where your app has been deployed.

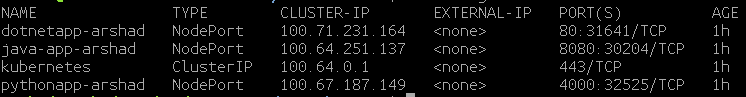
|  |
| --- |
| # kubectl get po -o wide |



In this example the app has been deployed to the NODE **ip-172-20-59-101.us-east-2.compute.internal**

5. Check the NODEPORT on which the application has been exposed

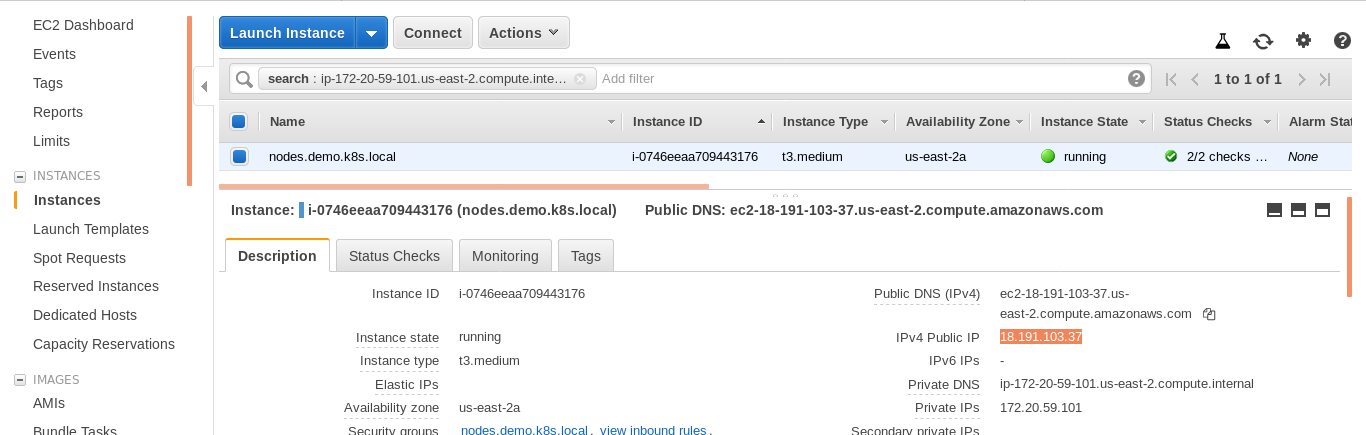
|  |
| --- |
| # kubectl get svc |

****

**Example**

**I**n this example the java application has been exposed on port **30204** as shown in the below.

6. Login to the **AWS** portal and check the public IP of the NODE (**ip-172-20-59-101.us-east-2.compute.internal**) to access the java application web page from the NODE Public IP address and Node Port on which it is exposed at.



**http://<NODE-PUBLIC-IP>:NODEPORT**

**Search for the NODE where your application has been deployed on the AWS EC2 Dashboard as shown below**

**Access the application from the public IP of the NODE and the NodePort as shown below**

Example.

[**http://18.191.103.37:30204/**](http://18.191.103.37:30204/)

