Assignment – 5

Platform Engineering

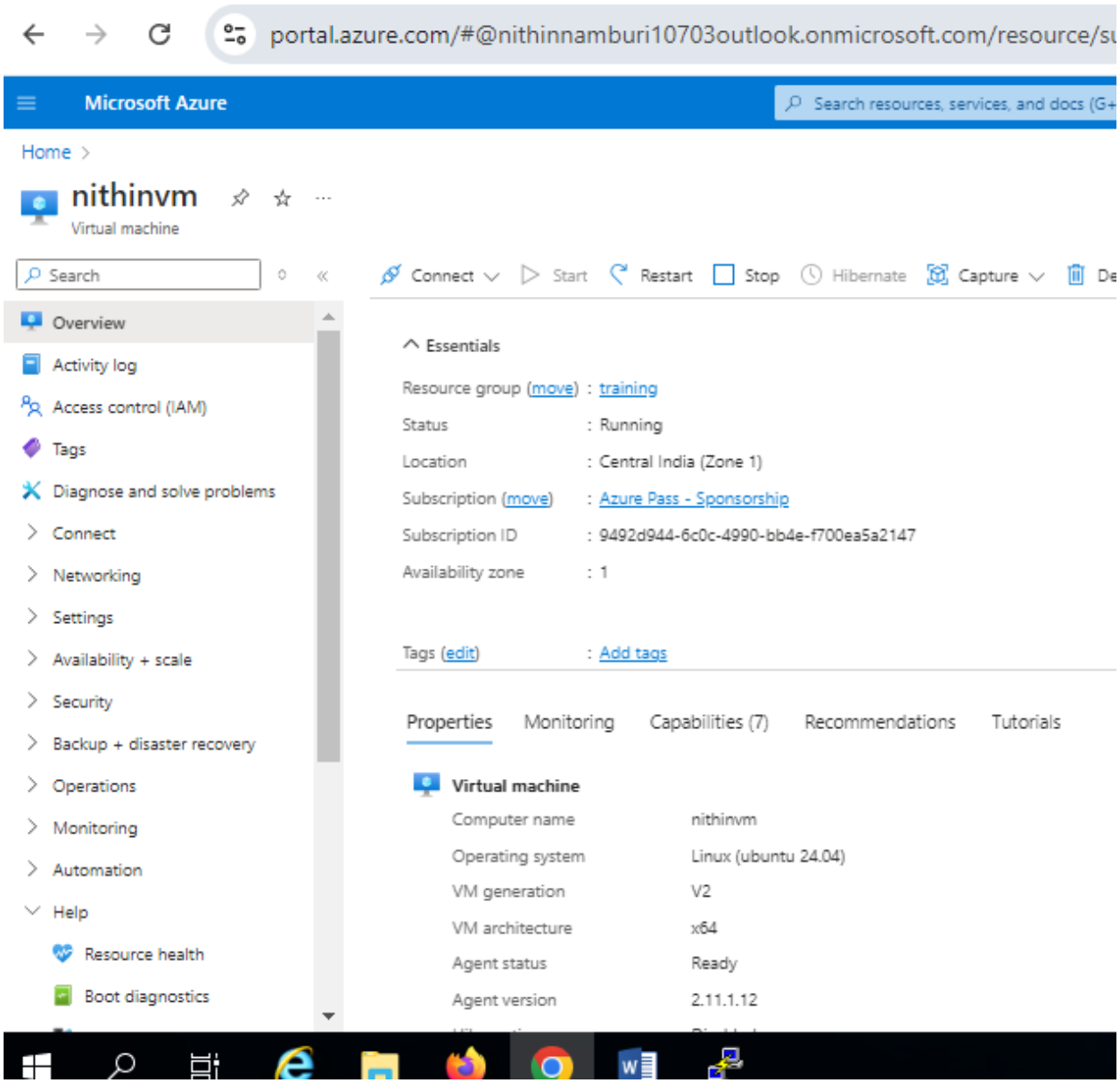
N.V.Nithin Kumar

1433832

1) Create a Docker for spring pet clinic.

Ans)

Step-1: Create a virtual Machine for Ubuntu.



Step-2: Connect to the putty and go to the root user.

- Clone Git Repository. https://github.com/nithinnamburi2003/spring-petclinic
- Goto the repository directory and then install docker.

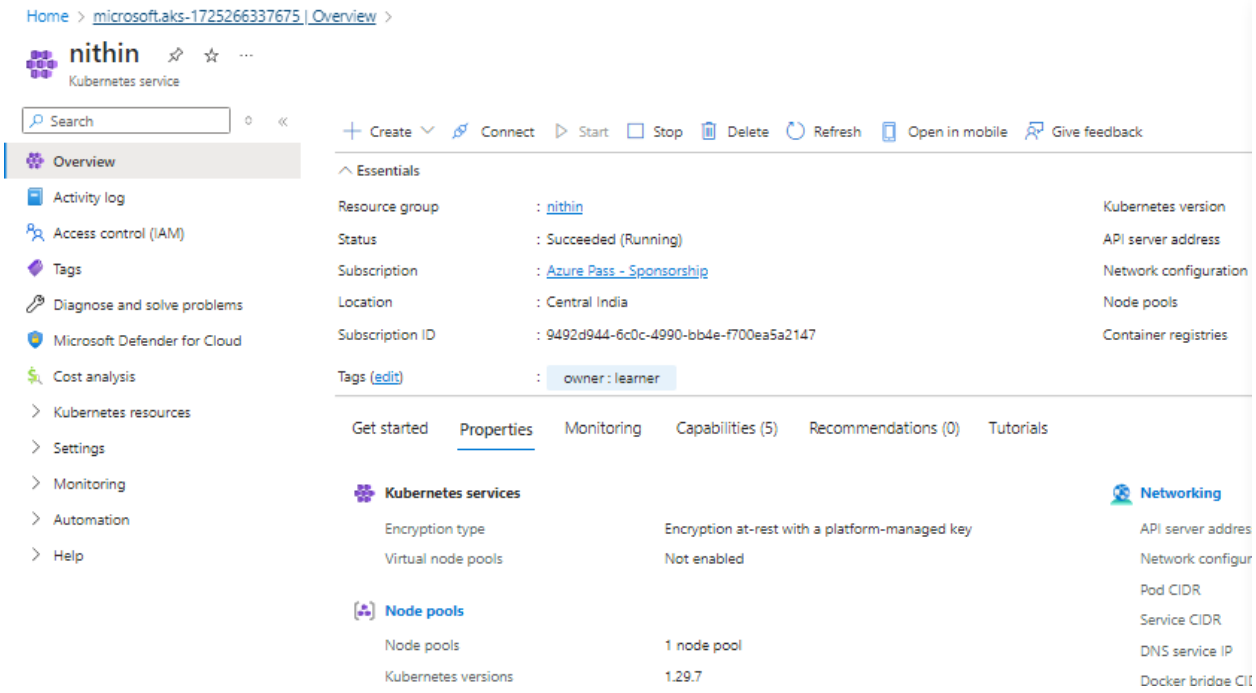
{

- sudo apt-get update
- sudo apt-get install ca-certificates curl
- sudo install -m 0755 -d /etc/apt/keyrings
- sudo curl -fsSL https://download.docker.com/linux/ubuntu/gpg -o /etc/apt/keyrings/docker.asc
- sudo chmod a+r /etc/apt/keyrings/docker.asc
-
- # Add the repository to Apt sources:
- echo \
- "deb [arch=$(dpkg --print-architecture) signed-by=/etc/apt/keyrings/docker.asc] https://download.docker.com/linux/ubuntu \
- $(. /etc/os-release && echo "$VERSION_CODENAME") stable" | \
- sudo tee /etc/apt/sources.list.d/docker.list > /dev/null
- sudo apt-get update}
- Install Java version 17 in the directory and make it ass default.
- Install maven
- Now create a dockerfile in the directory.
- {FROM openjdk:17-jdk-slim
- WORKDIR /app
- COPY target/spring-petclinic-*.jar app.jar
- EXPOSE 8080
- ENTRYPOINT ["java", "-jar", "app.jar"] }
- Build the dockerfile in the current directory.
- Now goto the chrome and search the VM IP with :8080
- The result will be as below

2) Host spring pet clinic in k8s.

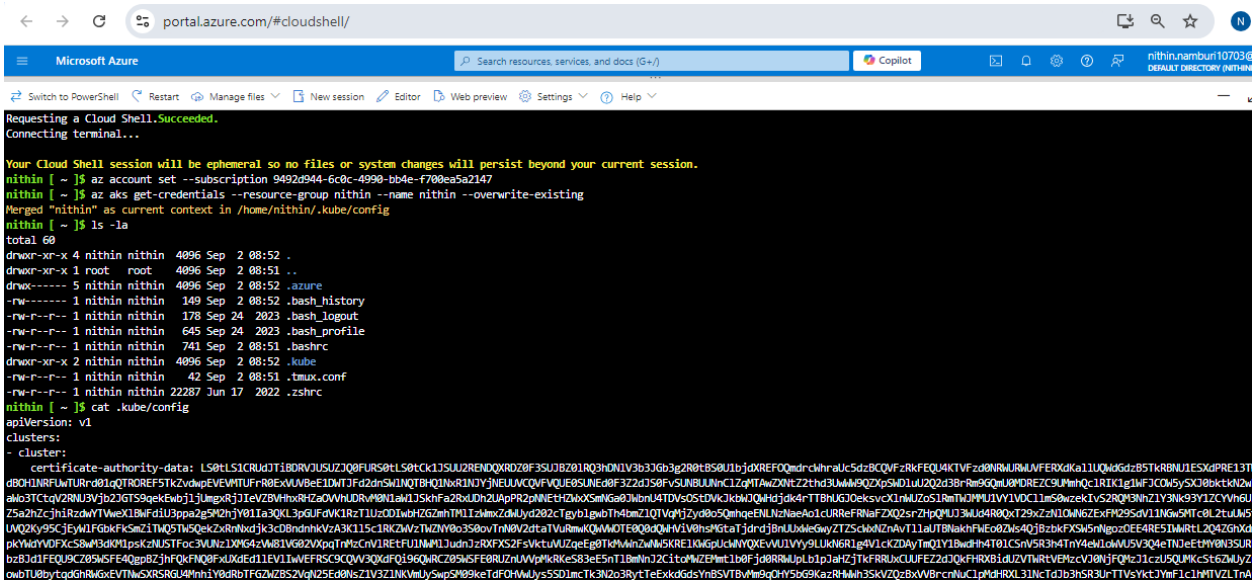
Ans)

Step-1: Create a kubernettes cluster.



Step – 2: Now click on the connect option and open azure CLI and run the commands.

- Copy the .kube file.



Step-3: Now connect to the Ubuntu putty and go to the root user home directory.

- Now run apt update.

```
login as: nubuntu
nubuntu@192.168.56.101's password:
Welcome to Ubuntu 22.04 LTS (GNU/Linux 6.8.0-40-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

239 updates can be applied immediately.
1 of these updates is a standard security update.
To see these additional updates run: apt list --upgradable

New release '24.04.1 LTS' available.
Run 'do-release-upgrade' to upgrade to it.

Last login: Mon Sep  2 13:51:48 2024 from 192.168.56.1
nubuntu@nubuntu-VirtualBox:~$ sudo su
[sudo] password for nubuntu:
root@nubuntu-VirtualBox:/home/nubuntu# cd
root@nubuntu-VirtualBox:~# apt update
Hit:1 http://security.ubuntu.com/ubuntu jammy-security InRelease
Hit:2 http://in.archive.ubuntu.com/ubuntu jammy InRelease
Hit:3 http://in.archive.ubuntu.com/ubuntu jammy-updates InRelease
Hit:4 http://in.archive.ubuntu.com/ubuntu jammy-backports InRelease
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
235 packages can be upgraded. Run 'apt list --upgradable' to see them.
root@nubuntu-VirtualBox:~# curl -LO https://dl.k8s.io/release/v1.29.7/bin/linux/
amd64/kubectl
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100   138  100   138    0     0    190      0 --:--:-- --:--:-- --:--:--   191
100 47.8M  100 47.8M    0     0  11.9M      0  0:00:03  0:00:03 --:--:-- 14.7M
root@nubuntu-VirtualBox:~# ls
```

Step – 4: Now install the Kubernettes in the binary file format and give the write permission and move it to the usr/bin.

```
root@nubuntu-VirtualBox:~# chmod +x kubectl
root@nubuntu-VirtualBox:~# mv kubectl /usr/bin
root@nubuntu-VirtualBox:~# kubectl
kubectl controls the Kubernetes cluster manager.

 Find more information at: https://kubernetes.io/docs/reference/kubectl/

Basic Commands (Beginner):
  create          Create a resource from a file or from stdin
  expose          Take a replication controller, service, deployment or pod and
expose it as a new Kubernetes service
  run             Run a particular image on the cluster
  set             Set specific features on objects

Basic Commands (Intermediate):
  explain         Get documentation for a resource
  get             Display one or many resources
  edit            Edit a resource on the server
  delete          Delete resources by file names, stdin, resources and names, or
by resources and label selector
```

Step-5 : Now create a deployment and service yaml files and apply them and run command kubectl get all to get the External IP addresses.

```
root@nubuntu-VirtualBox:~# vi .kube/config
root@nubuntu-VirtualBox:~# vi .kube/config
root@nubuntu-VirtualBox:~# kubectl get nodes
error: error loading config file "/root/.kube/config": no kind "Config" is regis
tered for version "v" in scheme "pkg/runtime/scheme.go:100"
root@nubuntu-VirtualBox:~# vi .kube/config
root@nubuntu-VirtualBox:~# kubectl get nodes
NAME                                STATUS   ROLES    AGE   VERSION
aks-agentpool-35370287-vmss000000   Ready    <none>   32m   v1.29.7
aks-agentpool-35370287-vmss000001   Ready    <none>   32m   v1.29.7
root@nubuntu-VirtualBox:~# vi deployment.yaml
root@nubuntu-VirtualBox:~# kubectl apply -f deployment.yaml
deployment.apps/spring-petclinic created
root@nubuntu-VirtualBox:~# vi service.yaml
root@nubuntu-VirtualBox:~# kubectl apply -f service.yaml
service/spring-petclinic created
root@nubuntu-VirtualBox:~# kubectl get all
NAME                                 READY   STATUS    RESTARTS   AGE
pod/spring-petclinic-699b946665-48qv2   1/1     Running   0          75s
pod/spring-petclinic-699b946665-gsg77   0/1     Pending   0          75s
pod/spring-petclinic-699b946665-k5g2j   1/1     Running   0          75s

NAME                       TYPE           CLUSTER-IP    EXTERNAL-IP     PORT(S)        AGE
service/kubernetes         ClusterIP      10.0.0.1      <none>          443/TCP        41m
service/spring-petclinic   LoadBalancer   10.0.48.179   4.224.110.220   80:30222/TCP   32s

NAME                               READY   UP-TO-DATE   AVAILABLE   AGE
deployment.apps/spring-petclinic   2/3     3            2           76s

NAME                                          DESIRED   CURRENT   READY   AGE
replicaset.apps/spring-petclinic-699b946665   3         3         2       76s
```

Deployment.Yaml:

```yaml
apiVersion: apps/v1
kind: Deployment
metadata:
 name: spring-petclinic
 labels:
   app: spring-petclinic
spec:
 replicas: 3  # Number of pods
 selector:
   matchLabels:
     app: spring-petclinic
 template:
   metadata:
     labels:
       app: spring-petclinic
   spec:
     containers:
     - name: spring-petclinic
       image: till0061/spring-petclinic:latest
       ports:
       - containerPort: 8080
       resources:
         requests:
           memory: "512Mi"
           cpu: "500m"
         limits:
           memory: "1Gi"
           cpu: "1"
~
~
~
~
~
```

Service.yaml:

root@nubuntu-VirtualBox: ~

```yaml
apiVersion: v1
kind: Service
metadata:
 name: spring-petclinic
spec:
 type: LoadBalancer
 ports:
 - port: 80
   targetPort: 8080
 selector:
   app: spring-petclinic
~
~
~
~
~
~
```

Step-6:Now copy the external IP and paste it in the browser.