DEVOPS

ASSIGNMENT 4

STEPS:

Step 1: Start Minikube

minikube start --driver=docker --force

Step 2: Create a Deployment

kubectl create deployment webapp --image=nginx --port=80

Step 3: Expose the Deployment as a Service

kubectl expose deployment webapp --type=NodePort --port=80 --target-port=80

Step 4: Verify the Running Pods

kubectl get pod

Step 5: Verify the Service

kubectl get svc

Step 6: Open the Service in a Web Browser

minikube service webapp

Step 7: Test the Service Using curl

curl http://192.168.49.2:31432

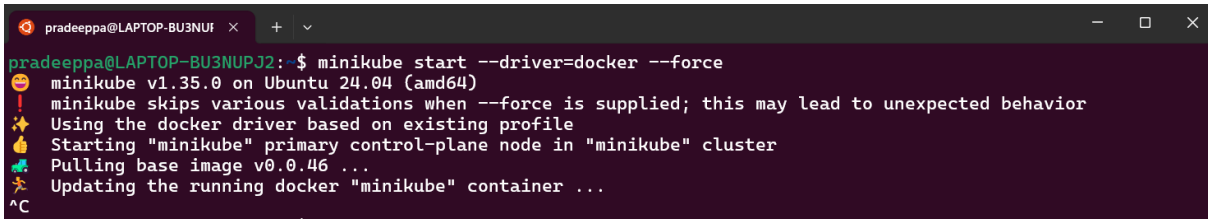
Step 8: Continuously Monitor the Pods

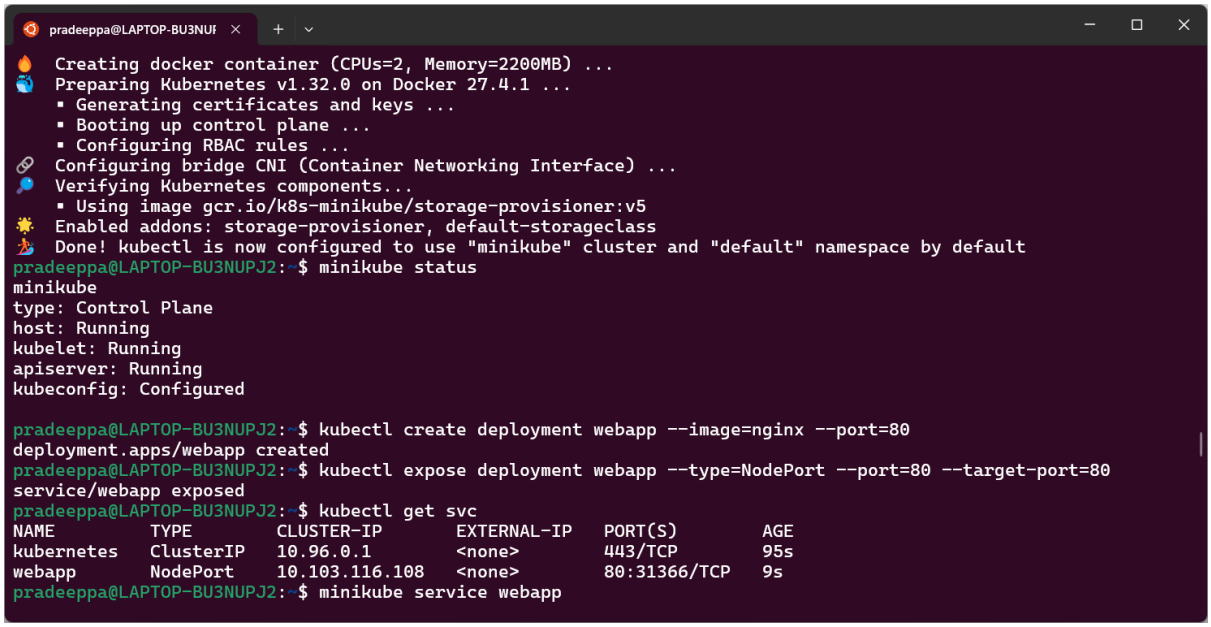
watch kubectl get pod

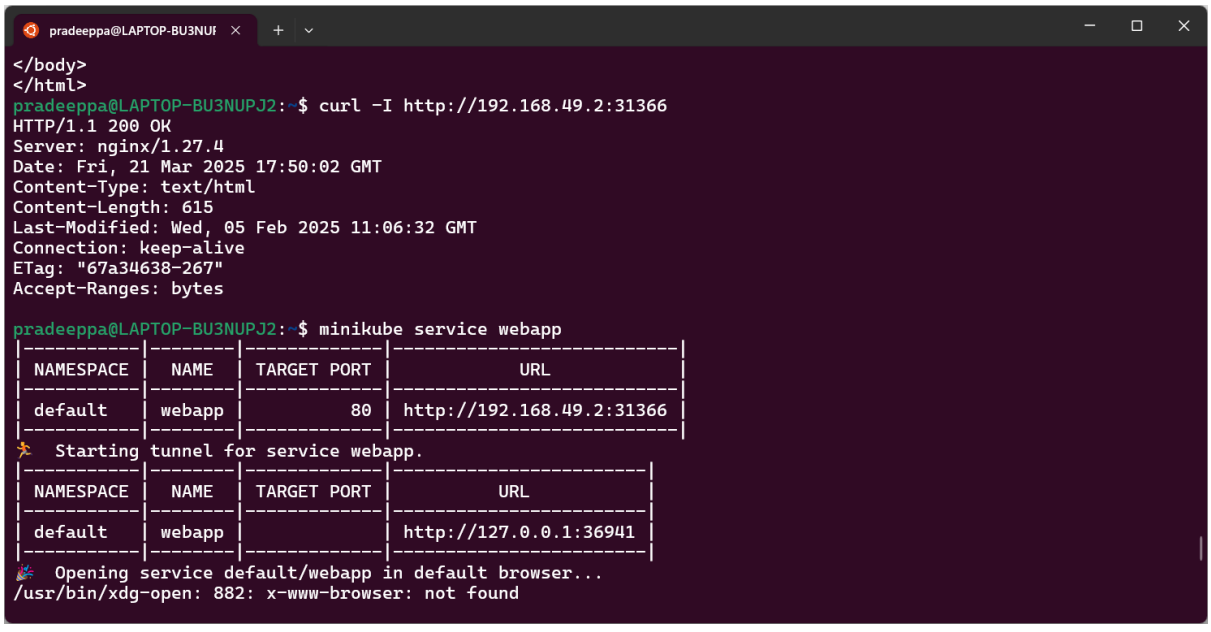
Step 9: Continuously Monitor Pod Logs

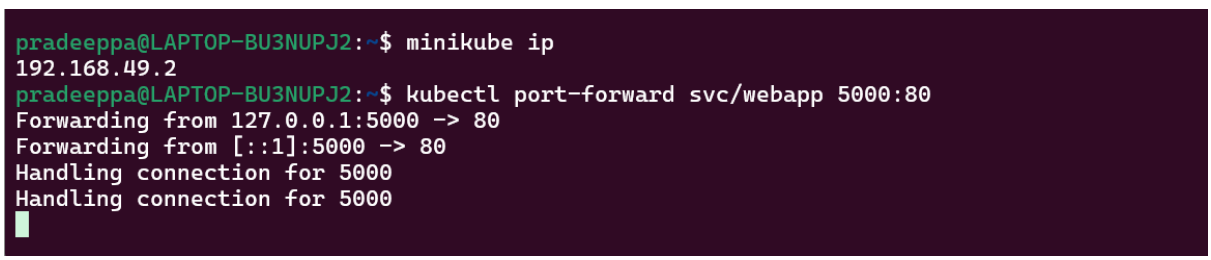
watch kubectl logs webapp-869b646d9f-b4hgr

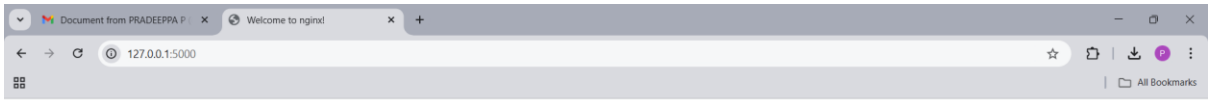
OUTPUT:


pradeeppa@LAPTOP-BU3NUI
pradeeppa@LAPTOP-BU3NUPJ2:~$ minikube start --driver=docker --force
minikube v1.35.0 on Ubuntu 24.04 (amd64)
minikube skips various validations when --force is supplied; this may lead to unexpected behavior
Using the docker driver based on existing profile
Starting "minikube" primary control-plane node in "minikube" cluster
Pulling base image v0.0.46 ...
Updating the running docker "minikube" container ...
^C


pradeeppa@LAPTOP-BU3NUI
Creating docker container (CPUs=2, Memory=2200MB) ...
Preparing Kubernetes v1.32.0 on Docker 27.4.1 ...
▪ Generating certificates and keys ...
▪ Booting up control plane ...
▪ Configuring RBAC rules ...
Configuring bridge CNI (Container Networking Interface) ...
Verifying Kubernetes components...
▪ Using image gcr.io/k8s-minikube/storage-provisioner:v5
Enabled addons: storage-provisioner, default-storageclass
Done! kubectl is now configured to use "minikube" cluster and "default" namespace by default
pradeeppa@LAPTOP-BU3NUPJ2:~$ minikube status
minikube
type: Control Plane
host: Running
kubelet: Running
apiserver: Running
kubeconfig: Configured
pradeeppa@LAPTOP-BU3NUPJ2:~$ kubectl create deployment webapp --image=nginx --port=80
deployment.apps/webapp created
pradeeppa@LAPTOP-BU3NUPJ2:~$ kubectl expose deployment webapp --type=NodePort --port=80 --target-port=80
service/webapp exposed
pradeeppa@LAPTOP-BU3NUPJ2:~$ kubectl get svc
NAME TYPE CLUSTER-IP EXTERNAL-IP PORT(S) AGE
kubernetes ClusterIP 10.96.0.1 <none> 443/TCP 95s
webapp NodePort 10.103.116.108 <none> 80:31366/TCP 9s
pradeeppa@LAPTOP-BU3NUPJ2:~$ minikube service webapp


pradeeppa@LAPTOP-BU3NUI
</body>
</html>
pradeeppa@LAPTOP-BU3NUPJ2:~$ curl -I http://192.168.49.2:31366
HTTP/1.1 200 OK
Server: nginx/1.27.4
Date: Fri, 21 Mar 2025 17:50:02 GMT
Content-Type: text/html
Content-Length: 615
Last-Modified: Wed, 05 Feb 2025 11:06:32 GMT
Connection: keep-alive
ETag: "67a34638-267"
Accept-Ranges: bytes
pradeeppa@LAPTOP-BU3NUPJ2:~$ minikube service webapp
| NAMESPACE | NAME | TARGET PORT | URL |
| default | webapp | 80 | http://192.168.49.2:31366 |
Starting tunnel for service webapp.
| NAMESPACE | NAME | TARGET PORT | URL |
| default | webapp | | http://127.0.0.1:36941 |
Opening service default/webapp in default browser...
/usr/bin/xdg-open: 882: x-www-browser: not found


pradeeppa@LAPTOP-BU3NUPJ2:~$ minikube ip
192.168.49.2
pradeeppa@LAPTOP-BU3NUPJ2:~$ kubectl port-forward svc/webapp 5000:80
Forwarding from 127.0.0.1:5000 -> 80
Forwarding from [::1]:5000 -> 80
Handling connection for 5000
Handling connection for 5000



Welcome to nginx!

If you see this page, the nginx web server is successfully installed and working. Further configuration is required.

For online documentation and support please refer to nginx.org.
Commercial support is available at nginx.com.

Thank you for using nginx.