

## DAY 3 -Task

### MINIKUBE INSTALLATION

```
deepika@DEEPI:/mnt/c/WINDOWS/system32
deepika@DEEPI:/mnt/c/WINDOWS/system32$ docker --version
Docker version 28.0.1, build 068a01e
deepika@DEEPI:/mnt/c/WINDOWS/system32$ sudo apt update && sudo apt upgrade -y
[sudo] password for deepika:
Ign:1 https://pkg.jenkins.io/debian-stable binary/ InRelease
Hit:2 https://download.docker.com/linux/ubuntu noble InRelease
Hit:3 https://pkg.jenkins.io/debian-stable binary/ Release
Hit:5 http://archive.ubuntu.com/ubuntu noble InRelease
Get:6 http://security.ubuntu.com/ubuntu noble-security InRelease [126 kB]
Get:7 http://archive.ubuntu.com/ubuntu noble-updates InRelease [126 kB]
Get:8 http://archive.ubuntu.com/ubuntu noble-backports InRelease [126 kB]
Get:9 http://security.ubuntu.com/ubuntu noble-security/main amd64 Components [8964 B]
Get:10 http://archive.ubuntu.com/ubuntu noble-updates/main amd64 Packages [921 kB]
Get:11 http://security.ubuntu.com/ubuntu noble-security/universe amd64 Components [52.0 kB]
Get:12 http://security.ubuntu.com/ubuntu noble-security/restricted amd64 Components [212 B]
Get:13 http://security.ubuntu.com/ubuntu noble-security/multiverse amd64 Components [208 B]
Get:14 http://archive.ubuntu.com/ubuntu noble-updates/main amd64 Components [151 kB]
Get:15 http://archive.ubuntu.com/ubuntu noble-updates/universe amd64 Packages [1040 kB]
Get:16 http://archive.ubuntu.com/ubuntu noble-updates/universe amd64 Components [364 kB]
Get:17 http://archive.ubuntu.com/ubuntu noble-updates/restricted amd64 Components [212 B]
Get:18 http://archive.ubuntu.com/ubuntu noble-updates/multiverse amd64 Components [940 B]
Get:19 http://archive.ubuntu.com/ubuntu noble-backports/main amd64 Components [208 B]
Get:20 http://archive.ubuntu.com/ubuntu noble-backports/universe amd64 Components [20.0 kB]
Get:21 http://archive.ubuntu.com/ubuntu noble-backports/restricted amd64 Components [212 B]
Get:22 http://archive.ubuntu.com/ubuntu noble-backports/multiverse amd64 Components [212 B]
Fetched 2938 kB in 4s (722 kB/s)
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
11 packages can be upgraded. Run 'apt list --upgradable' to see them.
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Calculating upgrade... Done
The following upgrades have been deferred due to phasing:
  libnss-systemd libpam-systemd libsystemd-shared libsystemd0 libudev1 systemd systemd-dev systemd-resolved systemd-sysv systemd-timesyncd udev
0 upgraded, 0 newly installed, 0 to remove and 11 not upgraded.
deepika@DEEPI:/mnt/c/WINDOWS/system32$ sudo apt install -y curl wget apt-transport-https
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
curl is already the newest version (8.5.0-2ubuntu10.6).
wget is already the newest version (1.21.4-1ubuntu4.1).
wget set to manually installed.
```

```
deepika@DEEPI:/mnt/c/WINDOWS/system32
deployment.apps/nginx-login unchanged
service/nginx-login-service unchanged
deepika@DEEPI:/mnt/c/WINDOWS/system32$ kubectl get pods
NAME                READY   STATUS    RESTARTS   AGE
nginx-login-b6fd8c4f-c9ntm  0/1     ContainerCreating   0           8m27s
deepika@DEEPI:/mnt/c/WINDOWS/system32$ kubectl describe pod nginx-login-b6fd8c4f-c9ntm
Name:               nginx-login-b6fd8c4f-c9ntm
Namespace:          default
Priority:             0
Service Account:     default
Node:               minikube/192.168.49.2
Start Time:          Thu, 20 Mar 2025 04:32:07 +0000
Labels:              app=nginx-login
                    pod-template-hash=b6fd8c4f
Annotations:         <none>
Status:              Pending
IP:                 <none>
IPs:                 <none>
Controlled By:       ReplicaSet/nginx-login-b6fd8c4f
Containers:
  nginx-login:
    Container ID:
    Image:         nginx:latest
    Image ID:
    Port:          80/TCP
    Host Port:     0/TCP
    State:         Waiting
      Reason:      ContainerCreating
    Ready:         False
    Restart Count: 0
    Environment:   <none>
    Mounts:
      /var/run/secrets/kubernetes.io/serviceaccount from kube-api-access-z9lpt (ro)
Conditions:
  Type              Status
  PodReadyToStartContainers  False
  Initialized        True
```

## MINIKUBE VERSION AND STATUS

```
deepika@DEEPI:/mnt/c/WINDOWS/system32$ sudo apt install wsl
deepika@DEEPI:/mnt/c/WINDOWS/system32$ jenkins --version
2.492.2
deepika@DEEPI:/mnt/c/WINDOWS/system32$ java -version
openjdk version "21.0.6" 2025-01-21
OpenJDK Runtime Environment (build 21.0.6+7-Ubuntu-124.04.1)
OpenJDK 64-Bit Server VM (build 21.0.6+7-Ubuntu-124.04.1, mixed mode, sharing)
deepika@DEEPI:/mnt/c/WINDOWS/system32$ minikube version
minikube version: v1.35.0
commit: dd5d320e41b5451cdf3c01891bc4e13d189586ed-dirty
deepika@DEEPI:/mnt/c/WINDOWS/system32$ minikube status
minikube: command not found
deepika@DEEPI:/mnt/c/WINDOWS/system32$ minikube status
minikube
type: Control Plane
host: Stopped
kubelet: Stopped
apiserver: Stopped
kubeconfig: Stopped
deepika@DEEPI:/mnt/c/WINDOWS/system32$ minikube start
👉 minikube v1.35.0 on Ubuntu 24.04 (amd64)
🌟 Using the docker driver based on existing profile
👉 Starting "minikube" primary control-plane node in "minikube" cluster
👉 Pulling base image v0.0.46 ...
👉 Restarting existing docker container for "minikube" ...
👉 Preparing Kubernetes v1.32.0 on Docker 27.4.1 ...
🔎 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
deepika@DEEPI:/mnt/c/WINDOWS/system32$ minikube status
minikube
type: Control Plane
host: Running
kubelet: Running
apiserver: Running
kubeconfig: Configured
deepika@DEEPI:/mnt/c/WINDOWS/system32$ Sudo nano nginx-deployment.yml

deepika@DEEPI:/mnt/c/WINDOWS/system32$ kubectl version
Client Version: v1.32.3
Kustomize Version: v5.5.0
Server Version: v1.32.0
deepika@DEEPI:/mnt/c/WINDOWS/system32$
```

## OUTPUT

```
deepika@DEEPI:/mnt/c/WINDOWS/system32
nginx-login-b6fd8c4f-c4qf4 1/1 Running 0 26s
deepika@DEEPI:/mnt/c/WINDOWS/system32$ minikube service nginx-login-service --url
http://127.0.0.1:32807
! Because you are using a Docker driver on linux, the terminal needs to be open to run it.
^Cdeepika@DEEPI:/mnt/c/WINDOWS/system32$ minikube ip
192.168.49.2
deepika@DEEPI:/mnt/c/WINDOWS/system32$ curl http

^C
deepika@DEEPI:/mnt/c/WINDOWS/system32$ curl http://192.168.49.2:30008
<!DOCTYPE html>
<html>
<head>
<title>Welcome to nginx!</title>
<style>
html { color-scheme: light dark; }
body { width: 35em; margin: 0 auto;
font-family: Tahoma, Verdana, Arial, sans-serif; }
</style>
</head>
<body>
<h1>Welcome to nginx!</h1>
<p>If you see this page, the nginx web server is successfully installed and
working. Further configuration is required.</p>

<p>For online documentation and support please refer to
<a href="http://nginx.org/">nginx.org</a>.<br>
Commercial support is available at
<a href="http://nginx.com/">nginx.com</a>.</p>

<p><em>Thank you for using nginx.</em></p>
</body>
</html>
deepika@DEEPI:/mnt/c/WINDOWS/system32$ kubectl get svc
NAME         TYPE        CLUSTER-IP   EXTERNAL-IP   PORT(S)          AGE
kubernetes   ClusterIP   10.96.0.1    <none>        443/TCP          22h
nginx-login-service NodePort    10.99.244.192 <none>        80:30008/TCP    24m
deepika@DEEPI:/mnt/c/WINDOWS/system32$
```

