

THIVISHA S
22CSR225

TASK 3

```
varsh_12@VARSH:~$ code --install-extension <extension-name>
```

```
-bash: syntax error near unexpected token 'newline'
```

```
varsh_12@VARSH:~$ wsl --install -d ubuntu
```

```
Command 'wsl' not found, but can be installed with:
```

```
sudo apt install wsl
```

```
varsh_12@VARSH:~$ docker images
```

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
test1	latest	53664373f5d4	23 minutes ago	195MB
varsh1204/devops	latest	53664373f5d4	23 minutes ago	195MB

```
varsh_12@VARSH:~$ docker run -itd -p 70:80 test1
```

```
51394a8cbe9ebf596b5f471098474a79781603b9ced7cfb1e329029f3b3c5912
```

```
varsh_12@VARSH:~$ sudo apt-get install docker-buildx-plugin
```

```
[sudo] password for varsh_12:
```

```
Reading package lists... Done
```

```
Building dependency tree... Done
```

```
Reading state information... Done
```

```
The following additional packages will be installed:
```

```
docker-ce-cli docker-compose-plugin
```

```
The following packages will be upgraded:
```

```
docker-buildx-plugin docker-ce-cli docker-compose-plugin
```

```
3 upgraded, 0 newly installed, 0 to remove and 28 not upgraded.
```

```
Need to get 65.2 MB of archives.
```

```
After this operation, 853 kB of additional disk space will be used.
```

```
Do you want to continue? [Y/n] y
```

```
Get:1 https://download.docker.com/linux/ubuntu noble/stable amd64 docker-buildx-plugin amd64 0.22.0-1~ubuntu.24.04~noble [35.3 MB]
```

```
Get:2 https://download.docker.com/linux/ubuntu noble/stable amd64 docker-ce-cli amd64 5:28.0.2-1~ubuntu.24.04~noble [15.8 MB]
```

```
Get:3 https://download.docker.com/linux/ubuntu noble/stable amd64 docker-compose-plugin amd64 2.34.0-1~ubuntu.24.04~noble [14.1 MB]
```

```
Fetched 65.2 MB in 13s (5124 kB/s)
```

```
(Reading database ... 42381 files and directories currently installed.)
```

```
Preparing to unpack .../docker-buildx-plugin_0.22.0-1~ubuntu.24.04~noble_amd64.deb ...
```

```
Unpacking docker-buildx-plugin (0.22.0-1~ubuntu.24.04~noble) over (0.21.1-1~ubuntu.24.04~noble) ...
```

```
Preparing to unpack .../docker-ce-cli_5%3a28.0.2-1~ubuntu.24.04~noble_amd64.deb ...
```

```
Unpacking docker-ce-cli (5:28.0.2-1~ubuntu.24.04~noble) over (5:28.0.1-1~ubuntu.24.04~noble) ...
```

```
Preparing to unpack .../docker-compose-plugin_2.34.0-1~ubuntu.24.04~noble_amd64.deb ...
```

```
Unpacking docker-compose-plugin (2.34.0-1~ubuntu.24.04~noble) over (2.33.1-1~ubuntu.24.04~noble) ...
```

```
Setting up docker-buildx-plugin (0.22.0-1~ubuntu.24.04~noble) ...
```

```
Setting up docker-compose-plugin (2.34.0-1~ubuntu.24.04~noble) ...
```

```
Setting up docker-ce-cli (5:28.0.2-1~ubuntu.24.04~noble) ...
```

```
Processing triggers for man-db (2.12.0-4build2) ...
```

```
varsh_12@VARSH:~$
```

```
varsh_12@VARSH: ~  
sudo install minikube-linux-amd64 /usr/local/bin/minikube  
% Total      % Received % Xferd  Average Speed   Time    Time     Time  Current  
           % Dload  % Upload   Total     Spent    Left     Speed  
100 119M 100 119M    0     0 4698k      0  0:00:26  0:00:26 --:--:-- 4830k  
varsh_12@VARSH:~$ minikube version  
minikube version: v1.35.0  
commit: dd5d320e41b5451cdf3c01891bc4e13d189586ed-dirty  
varsh_12@VARSH:~$ minikube start --driver=docker  
🐳 minikube v1.35.0 on Ubuntu 24.04 (amd64)  
💡 Using the docker driver based on user configuration  
  
❌ Requested memory allocation (1874MB) is less than the recommended minimum 1900MB. Deployments may fail.  
  
🔥 The requested memory allocation of 1874MiB does not leave room for system overhead (total system memory: 1874MiB). You may face stability issues.  
💡 Suggestion: Start minikube with less memory allocated: 'minikube start --memory=1874mb'  
  
🚀 Using Docker driver with root privileges  
❗ For an improved experience it's recommended to use Docker Engine instead of Docker Desktop.  
Docker Engine installation instructions: https://docs.docker.com/engine/install/#server  
👍 Starting "minikube" primary control-plane node in "minikube" cluster  
🐳 Pulling base image v0.0.46 ...  
📦 Downloading Kubernetes v1.32.0 preload ...  
> preloaded-images-k8s-v18-v1...: 333.57 MiB / 333.57 MiB 100.00% 3.50 Mi  
> gcr.io/k8s-minikube/kicbase...: 500.31 MiB / 500.31 MiB 100.00% 2.81 Mi  
🔥 Creating docker container (CPUs=2, Memory=1874MB) ...  
🐳 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  
varsh_12@VARSH:~$ minikube status  
minikube  
type: Control Plane  
host: Running  
kubelet: Running  
apiserver: Running  
kubeconfig: Configured  
  
varsh_12@VARSH:~$ |
```

```
varsh_12@VARSH: ~  
varsh_12@VARSH:~$ minikube version  
minikube version: v1.35.0  
commit: dd5d320e41b5451cdf3c01891bc4e13d189586ed-dirty  
varsh_12@VARSH:~$ minikube start  
🐳 minikube v1.35.0 on Ubuntu 24.04 (amd64)  
🌟 Using the docker driver based on existing profile  
  
🚫 Exiting due to PROVIDER_DOCKER_NOT_RUNNING: deadline exceeded running "docker version --format <no value>--<no value>:<no value>": signal: killed  
💡 Suggestion: Restart the Docker service  
📄 Documentation: https://minikube.sigs.k8s.io/docs/drivers/docker/  
  
varsh_12@VARSH:~$ kubectl get nodes  
NAME      STATUS    ROLES    AGE     VERSION  
minikube   Ready     control-plane  158m    v1.32.0  
varsh_12@VARSH:~$ kubectl create deployment r1 --image=varsh1204/varsh1204/devops --port=80  
deployment.apps/r1 created  
varsh_12@VARSH:~$ |
```

💡 Restarting the docker service may improve performance.

NAMESPACE	NAME	TARGET PORT	URL
default	r3	80	http://192.168.49.2:30987

✗ Exiting due to SVC_UNREACHABLE: service not available: no running pod for service r3 found

🐱 If the above advice does not help, please let us know:
👉 <https://github.com/kubernetes/minikube/issues/new/choose>

Please run `minikube logs --file=logs.txt` and attach logs.txt to the GitHub issue.
Please also attach the following file to the GitHub issue:
- /tmp/minikube_service_1f1376dc6c5b6e954e9491fb61dab5e54fc00912_0.log

varsh_12@VARSH:~\$ kubectl get nodes

NAME	STATUS	ROLES	AGE	VERSION
minikube	Ready	control-plane	172m	v1.32.0

varsh_12@VARSH:~\$ kubectl get svc

NAME	TYPE	CLUSTER-IP	EXTERNAL-IP	PORT(S)	AGE
kubernetes	ClusterIP	10.96.0.1	<none>	443/TCP	172m
r1	NodePort	10.109.150.162	<none>	80:30649/TCP	9m20s
r3	NodePort	10.103.187.39	<none>	80:30987/TCP	71s

varsh_12@VARSH:~\$ minikube service r3

NAMESPACE	NAME	TARGET PORT	URL
default	r3	80	http://192.168.49.2:30987









🏃 Starting tunnel for service r3.

NAMESPACE	NAME	TARGET PORT	URL
default	r3		http://127.0.0.1:44433

🌐 Opening service default/r3 in default browser...

👉 <http://127.0.0.1:44433>

❗ Because you are using a Docker driver on linux, the terminal needs to be open to run it.

<p>OnePlus 9 5G</p>  <p>5.4 inch display 399</p> <p>Add to Cart</p>	<p>Iphone 13 mini</p>  <p>5.4 inch display 399</p> <p>Add to Cart</p>	<p>Samsung s21 ultra</p>  <p>5.4 inch display 399</p> <p>Add to Cart</p>	<p>xiomi mi 11</p>  <p>5.4 inch display 399</p> <p>Add to Cart</p>
<p>OnePlus 9 5G</p>  <p>5.4 inch display 399</p> <p>Add to Cart</p>	<p>Iphone 13 mini</p>  <p>5.4 inch display 399</p> <p>Add to Cart</p>	<p>Samsung s21 ultra</p>  <p>5.4 inch display 399</p> <p>Add to Cart</p>	<p>xiomi mi 11</p>  <p>5.4 inch display 399</p> <p>Add to Cart</p>