

TASK 3-Minikube Deployment Task

Step 1: Install Kubectl

Since Kubectl is not found, install it with the following command:

sudo snap install kubectl --classic

Alternatively, you can download it using curl:

```
sowbaraniga_k@DESKTOP-73QEITE:~$ curl -LO https://github.com/kubernetes/minikube/releases/latest/download/minikube-linux-amd64
sudo install minikube-linux-amd64 /usr/local/bin/minikube && rm minikube-linux-amd64
```

% Total	% Received	% Xferd	Average Speed	Time	Time Current	Left	Speed
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
100	119M	100	119M	0	0	4185K	0:00:29 0:00:29 4206K

Step 2: Start Minikube

Start the Minikube cluster using the following command:

minikube start

```
sowbaraniga_k@DESKTOP-73QEITE:~$ minikube start
minikube v1.35.0 on Ubuntu 24.04 (amd64)
* Automatically selected the docker driver
* 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 v0.46...
Downloading Kubernetes v1.32.0 preload ...
> gcr.io/k8s-minikube/kicbase...: 0 B [ ] 7% ? p/ > gcr.io/k8s-minikube/kicbase...: 0 B [ ] 7% ? p/ > gcr.io/k8s-minikube/kicbase...: 0 B [ ] 7% ? p/
: 0 B [ ] 7% ? p/ > preloaded-images-k8s-v18-v1...: 0 B / 333.57 MiB [ ] 0.00% ? p/ > gcr.io/k8s-minikube/kicbase...: 0 B [ ] 7% ? p/ > preloaded-images-k8s-v18-v1...: 0 B [ ] 7% ? p/
? p/ > preloaded-images-k8s-v18-v1...: 76.76 KiB / 333.57 MiB [ ] 0.02% ? p/ > gcr.io/k8s-minikube/kicbase...: 0 B [ ] 7% ? p/ > preloaded-images-k8s-v18-v1...: 110.76 KiB / 333.57 MiB 0.03
1... 94.76 KiB / 333.57 MiB [ ] 0.03% ? p/ > gcr.io/k8s-minikube/kicbase...: 1.61 KiB / 500.31 MiB [ ] 0.00% ? p/ > preloaded-images-k8s-v18-v1...: 206.76 KiB / 333.57 MiB 0.06% 184.76 > gcr.io/k8s-minikube/kicbase...: 1.61 KiB / 500.31 MiB 0.00% 2.68 KiB > preloaded-images-k8s-v18-v1...: 350.76 KiB / 333.57 MiB 0.10% 184.76 > gcr.io/k8s-minikube/kicbase...: 1.61 KiB / 500.31 MiB 0.00% 2.68 KiB > preloaded-images-k8s-v18-v1...: 446.76 KiB / 333.57 MiB 0.13% 208.96 > gcr.io/k8s-minikube/kicbase...: 1.61 KiB / 500.31 MiB 0.00% 2.51 KiB > preloaded-images-k8s-v18-v1...: 606.76 KiB / 333.57 MiB 0.18% 208.96 > gcr.io/k8s-minikube/kicbase...: 17.76 KiB / 500.31 MiB 0.00% 2.51 KiB > preloaded-images-k8s-v18-v1...: 766.76 KiB / 333.57 MiB 0.22% 208.96 > gcr.io/k8s-minikube/kicbase...: 81.75 KiB / 500.31 MiB 0.02% 2.51 KiB > preloaded-images-k8s-v18-v1...: 830.76 KiB / 333.57 MiB 0.24% 236.79 > gcr.io/k8s-minikube/kicbase...: 97.75 KiB / 500.31 MiB 0.02% 12.70 K > preloaded-images-k8s-v18-v1...: 926.76 KiB / 333.57 MiB 0.27% 236.79 > gcr.io/k8s-minikube/kicbase...: 129.75 KiB / 500.31 MiB 0.03% 12.70 > preloaded-images-k8s-v18-v1...: 1.01 MiB / 333.57 MiB 0.30% 236.79 K > gcr.io/k8s-minikube/kicbase...: 193.75 KiB / 500.31 MiB 0.04% 12.70 > preloaded-images-k8s-v18-v1...: 1.14 MiB / 333.57 MiB 0.34% 257.66 K > gcr.io/k8s-minikube/kicbase...: 241.72 KiB / 500.31 MiB 0.05% 27.33 > preloaded-images-k8s-v18-v1...: 1.22 MiB / 333.57 MiB 0.36% 257.66 K > gcr.io/k8s-minikube/kicbase...: 289.72 KiB / 500.31 MiB 0.06% 27.33 > preloaded-images-k8s-v18-v1...: 1.34 MiB / 333.57 MiB 0.40% 257.66 K > gcr.io/k8s-minikube/kicbase...: 337.72 KiB / 500.31 MiB 0.07% 27.33 > preloaded-images-k8s-v18-v1...: 1.41 MiB / 333.57 MiB 0.42% 270.25 K > gcr.io/k8s-minikube/kicbase...: 385.72 KiB / 500.31 MiB 0.08% 41.05 > preloaded-images-k8s-v18-v1...: 1.53 MiB / 333.57 MiB 0.46% 270.25 K > gcr.io/k8s-minikube/kicbase...: 401.69 KiB / 500.31 MiB 0.08% 41.05 > preloaded-images-k8s-v18-v1...: 1.66 MiB / 333.57 MiB 0.50% 270.25 K > gcr.io/k8s-minikube/kicbase...: 465.70 KiB / 500.31 MiB 0.09% 41.05 > preloaded-images-k8s-v18-v1...: 1.75 MiB / 333.57 MiB 0.52% 290.65 K > gcr.io/k8s-minikube/kicbase...: 513.70 KiB / 500.31 MiB 0.10% 52.18 > preloaded-images-k8s-v18-v1...: 1.70 MiB / 333.57 MiB 0.53% 290.65 K > gcr.io/k8s-minikube/kicbase...: 545.70 KiB / 500.31 MiB 0.11% 52.18 > preloaded-images-k8s-v18-v1...: 1.91 MiB / 333.57 MiB 0.57% 290.65 K > gcr.io/k8s-minikube/kicbase...: 609.70 KiB / 500.31 MiB 0.12% 52.18 > preloaded-images-k8s-v18-v1...: 2.03 MiB / 333.57 MiB 0.61% 302.91 K > gcr.io/k8s-minikube/kicbase...: 657.70 KiB / 500.31 MiB 0.13% 64.30 > preloaded-images-k8s-v18-v1...: 2.09 MiB / 333.57 MiB 0.63% 302.91 K > gcr.io/k8s-minikube/kicbase...: 705.07 KiB / 500.31 MiB 0.14% 64.30 > preloaded-images-k8s-v18-v1...: 2.19 MiB / 333.57 MiB 0.66% 302.91 K > gcr.io/k8s-minikube/kicbase...: 753.67 KiB / 500.31 MiB 0.15% 64.30 > preloaded-images-k8s-v18-v1...: 2.31 MiB / 333.57 MiB 0.69% 314.33 K > gcr.io/k8s-minikube/kicbase...: 817.67 KiB / 500.31 MiB 0.16% 77.35 > preloaded-images-k8s-v18-v1...: 2.44 MiB / 333.57 MiB 0.73% 314.33 K > gcr.io/k8s-minikube/kicbase...
```

Step 3: Verify Minikube Installation

```
sowbaraniga_k@DESKTOP-73QEITE:~$ kubectl get nodes
```

NAME	STATUS	ROLES	AGE	VERSION
minikube	Ready	control-plane	19s	v1.32.0

Step 4: Create a Deployment

Create a deployment named `r2` with the image `sowbaranigak/dev`:

kubectl create deployment r2 --image=sowbaranigak/dev --port=80

```
sowbaraniga_k@DESKTOP-73QEITE:~$ kubectl create deployment r2 --image=sowbaranigak/dev --port=80
deployment.apps/r2 created
```

Step 5: Expose the Deployment

Expose the deployment as a NodePort service:

kubectl expose deployment r2 --port=80 --type=NodePort

```
sowbaraniga_k@DESKTOP-73QEITE:~$ kubectl expose deployment r2 --port=80 --type=NodePort
service/r2 exposed
```

Step 6: Verify the Pod

Check the running pods:

kubectrl get pods

```
sowbaraniga_k@DESKTOP-73QEITE:~$ kubectrl get pods
NAME                                READY   STATUS    RESTARTS   AGE
r1-77c5b5bbd7-w5rct                0/1     ImagePullBackOff    0          25m
r2-867d7797f8-9v7s2                1/1     Running             0          5m4s
```

Step 7: Access the Service

Expose the service using Minikube and get the URL:

minikube service r2

```
sowbaraniga_k@DESKTOP-73QEITE:~$ minikube service r2
```

NAMESPACE	NAME	TARGET PORT	URL
default	r2	80	http://192.168.49.2:32207

🌟 Starting tunnel for service r2.

NAMESPACE	NAME	TARGET PORT	URL
default	r2		http://127.0.0.1:44351

🚀 Opening service default/r2 in default browser...
👉 http://127.0.0.1:44351
! Because you are using a Docker driver on linux, the terminal needs to be open to run it.

Step 8: Output in the Web Browser

