

# Setup minikube at your local and explore creating namespaces (Go through official documentation)

## STEP 1 - minikube status

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

PS P:\Aws Devops\Sample codes> minikube status
minikube
type: Control Plane
host: Stopped
kublet: Stopped
apiserver: Stopped
kubeconfig: Stopped

PS P:\Aws Devops\Sample codes> minikube start
🐳 minikube v1.34.0 on Microsoft Windows 11 Home Single Language 10.0.22631.4391 Build 22631.4391
🔗 Using the docker driver based on existing profile
👉 Starting "minikube" primary control-plane node in "minikube" cluster
📥 Pulling base image v0.0.45 ...
🔄 Restarting existing docker container for "minikube" ...
❗ Failing to connect to https://registry.k8s.io/ from inside the minikube container
💡 To pull new external images, you may need to configure a proxy: https://minikube.sigs.k8s.io/docs/reference/networking/proxy/
🔄 Preparing Kubernetes v1.31.0 on Docker 27.2.0 ...
🔍 Verifying Kubernetes components...
  ▪ Using image docker.io/kubernetes/dashboard:v2.7.0
  ▪ Using image gcr.io/k8s-minikube/storage-provisioner:v5
  ▪ Using image docker.io/kubernetes/metrics-scraper:v1.0.8
💡 Some dashboard features require the metrics-server addon. To enable all features please run:
```

## STEP 2 - minikube start

```
minikube addons enable metrics-server

🌟 Enabled addons: storage-provisioner, default-storageclass, dashboard
👉 Done! kubectl is now configured to use "minikube" cluster and "default" namespace by default
PS P:\Aws Devops\Sample codes> minikube status
minikube
type: Control Plane
host: Running
kublet: Running
apiserver: Running
kubeconfig: Configured

PS P:\Aws Devops\Sample codes> 
```

### STEP 3 - kubectl create namespace guvi

### STEP 4 - kubectl get namespace

```
PS P:\Aws Devops\Sample codes> kubectl create namespace guvi
namespace/guvi created
PS P:\Aws Devops\Sample codes> kubectl get namespace
NAME                STATUS    AGE
default             Active    2d22h
guvi                 Active    13s
kube-node-lease     Active    2d22h
kube-public         Active    2d22h
kube-system         Active    2d22h
kubernetes-dashboard Active    2d21h
```

### STEP 5 - kubectl create deployments team --image=nginx

### --namespace=guvi

### STEP 6 - kubectl get deployments

### STEP 7 - kubectl get pods

```
PS P:\Aws Devops\Sample codes> kubectl create deployment team --image=nginx --namespace=guvi
deployment.apps/team created
PS P:\Aws Devops\Sample codes> kubectl get deployments
NAME    READY   UP-TO-DATE   AVAILABLE   AGE
team    1/1     1            1           59s
PS P:\Aws Devops\Sample codes> kubectl get pods
NAME                                READY   STATUS    RESTARTS   AGE
nginx                                1/1     Running   0           3m39s
team-5987748b5b-n8jlf              1/1     Running   0           2m24s
```

### STEP 8 - minikube dashboard

#### Metadata

Name	Namespace	Created	Age	UID
team-5987748b5b-n8jlf	guvi	Nov 10, 2024	10 minutes ago	61b30050-5146-4cfa-8c58-ea4cae0c75f8

#### Labels

app: team   pod-template-hash: 5987748b5b