

# Lab Exercise 7- Install Minikube on Linux (Ubuntu /MacOS/Windows)

Name – Ayush Bhardwaj  
500124917

B2 DevOps

---

## Install Minikube on Linux (Ubuntu)

```
sudo apt install -y docker.io # Ubuntu
```

```
sudo systemctl start docker  
sudo systemctl enable docker  
sudo usermod -aG docker $USER  
newgrp docker
```

---

### Step 2: Install kubectl

```
curl -LO https://storage.googleapis.com/kubernetes-release/release/\$\(curl -s https://storage.googleapis.com/kubernetes-release/release/stable.txt\)/bin/linux/amd64/kubectl  
chmod +x kubectl  
sudo mv kubectl /usr/local/bin/
```

Verify:

```
kubectl version --client
```

---

### Step 3: Install Minikube

```
curl -LO https://storage.googleapis.com/minikube/releases/latest/minikube-linux-amd64  
chmod +x minikube-linux-amd64  
sudo mv minikube-linux-amd64 /usr/local/bin/minikube
```

Verify:

```
minikube version
```

---

### Step 4: Start Minikube

```
minikube start --driver=docker
```

Check status:

```
minikube status
```

---

## Install Minikube on Windows

### Prerequisites

- Docker Desktop installed

```

PS C:\Users\ASUS> docker --version
Docker version 29.2.0, build 0b9d198
PS C:\Users\ASUS> docker run hello-world

Hello from Docker!
This message shows that your installation appears to be working correctly.

To generate this message, Docker took the following steps:
1. The Docker client contacted the Docker daemon.
2. The Docker daemon pulled the "hello-world" image from the Docker Hub.
   (amd64)
3. The Docker daemon created a new container from that image which runs the
   executable that produces the output you are currently reading.
4. The Docker daemon streamed that output to the Docker client, which sent
   it to your terminal.

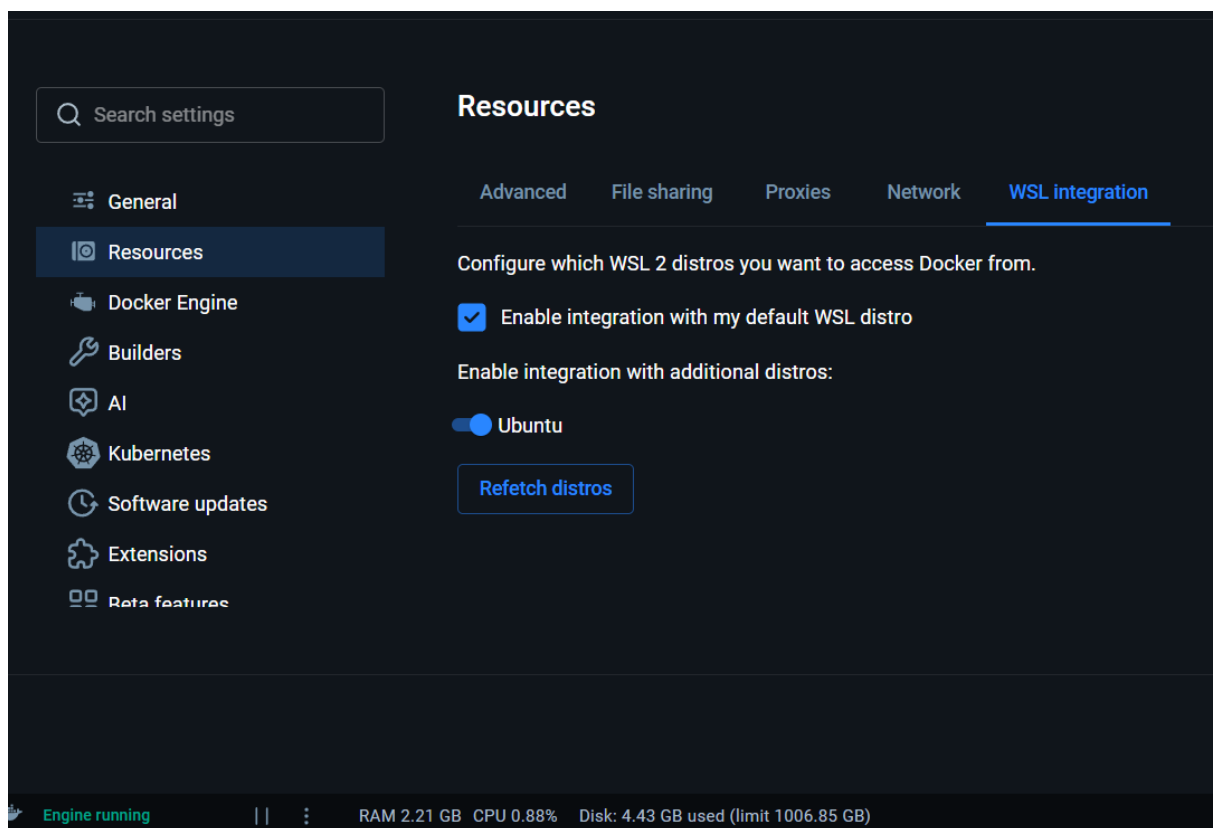
To try something more ambitious, you can run an Ubuntu container with:
$ docker run -it ubuntu bash

Share images, automate workflows, and more with a free Docker ID:
https://hub.docker.com/

For more examples and ideas, visit:
https://docs.docker.com/get-started/

```

- Enable WSL2



## Install Minikube

```
choco install minikube -y
```

```
Administrator: Windows PowerShell
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\Users\ASUS> choco install minikube -y
Chocolatey v2.6.0
Installing the following packages:
minikube
By installing, you accept licenses for the packages.
Downloading package from source 'https://community.chocolatey.org/api/v2/'
Progress: Downloading kubernetes-cli 1.35.1... 100%

kubernetes-cli v1.35.1 [Approved]
kubernetes-cli package files install completed. Performing other installation steps.
Extracting 64-bit C:\ProgramData\chocolatey\lib\kubernetes-cli\tools\kubernetes-client-windows-amd64.tar.gz to C:\ProgramData\chocolatey\lib\kubernetes-cli\tools...
C:\ProgramData\chocolatey\lib\kubernetes-cli\tools
Extracting 64-bit C:\ProgramData\chocolatey\lib\kubernetes-cli\tools\kubernetes-client-windows-amd64.tar to C:\ProgramData\chocolatey\lib\kubernetes-cli\tools...
C:\ProgramData\chocolatey\lib\kubernetes-cli\tools
ShimGen has successfully created a shim for kubectl-convert.exe
ShimGen has successfully created a shim for kubectl.exe
The install of kubernetes-cli was successful.
Deployed to 'C:\ProgramData\chocolatey\lib\kubernetes-cli\tools'
Downloading package from source 'https://community.chocolatey.org/api/v2/'
Progress: Downloading Minikube 1.38.1... 100%

Minikube v1.38.1 [Approved]
Minikube package files install completed. Performing other installation steps.
ShimGen has successfully created a shim for minikube.exe
The install of Minikube was successful.
Deployed to 'C:\ProgramData\chocolatey\lib\Minikube'

Chocolatey installed 2/2 packages.
See the log for details (C:\ProgramData\chocolatey\logs\chocolatey.log).
PS C:\Users\ASUS> |
```

OR download exe:

<https://github.com/kubernetes/minikube/releases/latest>

Start:

minikube start

```
PS C:\Users\ASUS> minikube start
🐳 minikube v1.38.1 on Microsoft Windows 11 Home Single Language 25H2
🌟 Automatically selected the docker driver. Other choices: hyperv, ssh
❗ Starting v1.39.0, minikube will default to "containerd" container runtime. See #21973 for more info.
🔧 Using Docker Desktop driver with root privileges
👍 Starting "minikube" primary control-plane node in "minikube" cluster
📦 Pulling base image v0.0.50 ...
📦 Downloading Kubernetes v1.35.1 preload ...
> preloaded-images-k8s-v18-v1...: 30.78 KiB / 272.45 MiB [>] 0.01% ? p/
> preloaded-images-k8s-v18-v1...: 46.78 KiB / 272.45 MiB [>] 0.02% ? p/
> gcr.io/k8s-minikube/kicbase...: 0 B [-----] ?% ? p/
> preloaded-images-k8s-v18-v1...: 78.77 KiB / 272.45 MiB [>] 0.03% ? p/
> gcr.io/k8s-minikube/kicbase...: 0 B [-----] ?% ? p/
> preloaded-images-k8s-v18-v1...: 126.77 KiB / 272.45 MiB 0.05% 160.08
> gcr.io/k8s-minikube/kicbase...: 0 B [-----] ?% ? p/
> preloaded-images-k8s-v18-v1...: 206.77 KiB / 272.45 MiB 0.07% 160.08
> gcr.io/k8s-minikube/kicbase...: 0 B [-----] ?% ? p/
> preloaded-images-k8s-v18-v1...: 334.77 KiB / 272.45 MiB 0.12% 160.08
> gcr.io/k8s-minikube/kicbase...: 1.60 KiB / 519.58 MiB [>_] 0.00% ? p/
> preloaded-images-k8s-v18-v1...: 558.77 KiB / 272.45 MiB 0.20% 196.16
> gcr.io/k8s-minikube/kicbase...: 1.60 KiB / 519.58 MiB [>_] 0.00% ? p/
> preloaded-images-k8s-v18-v1...: 974.77 KiB / 272.45 MiB 0.35% 196.16
> gcr.io/k8s-minikube/kicbase...: 1.60 KiB / 519.58 MiB 0.00% 2.67 KiB
> preloaded-images-k8s-v18-v1...: 1.70 MiB / 272.45 MiB 0.62% 196.16 K
> gcr.io/k8s-minikube/kicbase...: 1.60 KiB / 519.58 MiB 0.00% 2.67 KiB
> preloaded-images-k8s-v18-v1...: 3.06 MiB / 272.45 MiB 1.12% 460.85 K
> gcr.io/k8s-minikube/kicbase...: 97.75 KiB / 519.58 MiB 0.02% 2.67 Ki
> preloaded-images-k8s-v18-v1...: 5.47 MiB / 272.45 MiB 2.01% 460.85 K
> gcr.io/k8s-minikube/kicbase...: 465.72 KiB / 519.58 MiB 0.09% 52.41
> preloaded-images-k8s-v18-v1...: 8.82 MiB / 272.45 MiB 3.24% 460.85 K
> gcr.io/k8s-minikube/kicbase...: 1.55 MiB / 519.58 MiB 0.30% 52.41 Ki
> preloaded-images-k8s-v18-v1...: 11.87 MiB / 272.45 MiB 4.36% 1.37 Mi
> gcr.io/k8s-minikube/kicbase...: 3.21 MiB / 519.58 MiB 0.62% 52.41 Ki
> preloaded-images-k8s-v18-v1...: 14.59 MiB / 272.45 MiB 5.36% 1.37 Mi
> gcr.io/k8s-minikube/kicbase...: 5.56 MiB / 519.58 MiB 1.07% 611.50 K
> preloaded-images-k8s-v18-v1...: 16.51 MiB / 272.45 MiB 6.06% 1.37 Mi
> gcr.io/k8s-minikube/kicbase...: 7.17 MiB / 519.58 MiB 1.38% 611.50 K
> preloaded-images-k8s-v18-v1...: 17.67 MiB / 272.45 MiB 6.48% 1.90 Mi
> gcr.io/k8s-minikube/kicbase...: 13.06 MiB / 519.58 MiB 2.51% 611.50
> preloaded-images-k8s-v18-v1...: 20.01 MiB / 272.45 MiB 7.35% 1.90 Mi
> gcr.io/k8s-minikube/kicbase...: 16.67 MiB / 519.58 MiB 3.21% 1.75 Mi
> preloaded-images-k8s-v18-v1...: 21.39 MiB / 272.45 MiB 7.85% 1.90 Mi
> gcr.io/k8s-minikube/kicbase...: 20.14 MiB / 519.58 MiB 3.88% 1.75 Mi
> preloaded-images-k8s-v18-v1...: 23.64 MiB / 272.45 MiB 8.68% 2.42 Mi
> gcr.io/k8s-minikube/kicbase...: 23.63 MiB / 519.58 MiB 4.55% 1.75 Mi
```

```
🔥 Creating docker container (CPUs=2, Memory=3900MB) ...
🐳 Preparing Kubernetes v1.35.1 on Docker 29.2.1 ...
🔗 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
```

# Install Minikube on macOS

```
brew install minikube
```

```
brew install kubectl
```

Start:

```
minikube start
```

---

## Verify Kubernetes Cluster

```
kubectl get nodes
```

Expected output:

```
NAME      STATUS  ROLES    AGE  VERSION
minikube  Ready   control-plane  xx   v1.xx
```

```
PS C:\Users\ASUS> kubectl get nodes
NAME      STATUS  ROLES    AGE  VERSION
minikube  Ready   control-plane  27s   v1.35.1
PS C:\Users\ASUS>
```

---

## Useful Minikube Commands (Lab Ready)

| Command            | Purpose        |
|--------------------|----------------|
| minikube dashboard | Open K8s UI    |
| minikube stop      | Stop cluster   |
| minikube delete    | Delete cluster |
| minikube ssh       | Access node    |

| Command             | Purpose       |
|---------------------|---------------|
| kubectl get pods -A | View all pods |