

Docker Containers and Kubernetes Fundamentals

Docker / Docker Hub Account

You'll need a free Docker account. This account will also give you access to Docker Hub.

<https://hub.docker.com/>

Laptop – Hardware

A laptop supporting virtualization.

You'll need Administrator rights on the machine.

Note that corporate locked down machines may not grant you enough privileges to run some tools used in this course.

Windows Requirements

A laptop supporting virtualization, with Windows 10 or 11 and the latest updates.

OPTION 1 - Windows Subsystem for Linux 2

The recommended way of running Docker Desktop on a Windows machine by running it using WSL 2.

The requirements are:

- Windows 10 Version 2004 or higher, with Build 19041 or higher
or
- Windows 11
- Virtualization enabled

Refer to the documentation for the complete step-by-step installation instructions:

<https://docs.microsoft.com/en-us/windows/wsl/install>

It's important to limit the max memory and CPU that WSL can use by creating a config file.

Create a file named **.wslconfig** (that's dot wslconfig, no extension) in your **C:\users\[YourUser]** folder

Paste this suggested configuration:

```
[wsl2]
memory=2GB
processors=2
```

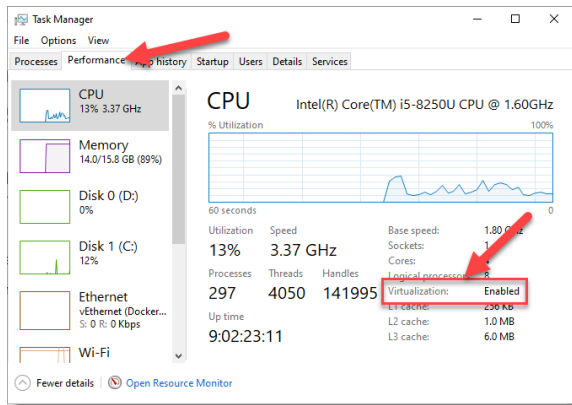
Save the file and restart WSL by typing:

```
wsl --shutdown
wsl
```

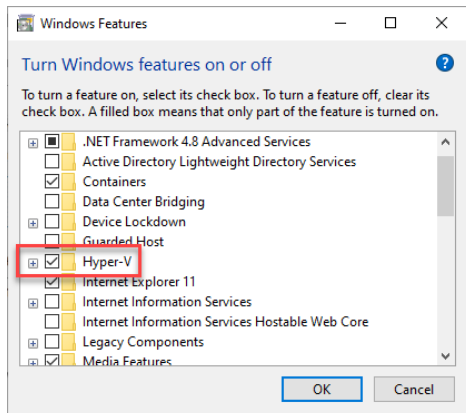
OPTION 2 – Hyper-V

Docker Desktop will also run by using an Hyper-V virtual machine on laptops/PC that don't have WSL 2 installed.

You can see if virtualization is enabled in Task manager.



To enable it, press the Windows key and type "**turn windows features on or off**" and click on it to open it. Check **Hyper-V** and reboot.



Chocolatey

Chocolatey is a package manager for Windows. Installation instructions are found here:

<https://chocolatey.org/install>

Note that corporate locked down machines may not grant you enough privileges to install Chocolatey.

Make sure to run a Command or a Powershell Prompt as an admin when installing packages.

Already using Chocolatey? Make sure to update your already installed packages.

```
choco upgrade all -y
```



macOS 10.15 and higher. Using the latest version is recommended.

Mac with Intel chip or Apple Silicon can run Docker Desktop.

Refer to the Docker Desktop system requirements page: <https://docs.docker.com/docker-for-mac/install/>

Note that other tools used in this course may not yet work on Apple Silicon.

Brew

Brew is a package manager for macOS. Installation instructions are found here:

<https://brew.sh/>



Ubuntu Desktop 18.04.x or later.

Snap

Snap is a package manager for Ubuntu and it should already be installed. To validate open a terminal, type `snap` to validate. If not, install it using:

```
sudo apt update
```

```
sudo apt install snapd
```

Docker



On Windows and macOS, install Docker Desktop.

<https://www.docker.com/products/docker-desktop>

Docker Compose will be installed by this installation.



On Ubuntu, you can install the Docker engine or Docker Desktop

Docker Engine

Refer to the documentation for the latest installation instructions:

<https://docs.docker.com/engine/install/ubuntu/>

Next install Docker Compose:

```
sudo apt install docker-compose
```

Docker Desktop

<https://docs.docker.com/desktop/install/linux-install/>



Install Visual Studio Code on your machine.

<https://code.visualstudio.com/>

After installing Code, you need to install the Docker extension.

Launch Code and install the Docker extension

1. Click on the Extensions tabs
2. Search for Docker
3. Select the **Docker** extension
4. Install it

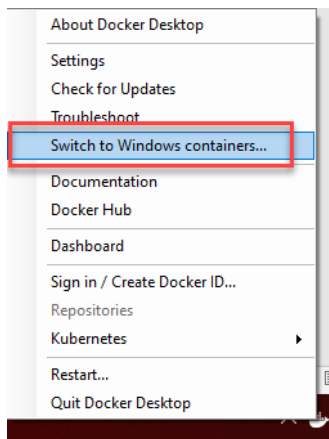


Kubernetes – Docker Desktop

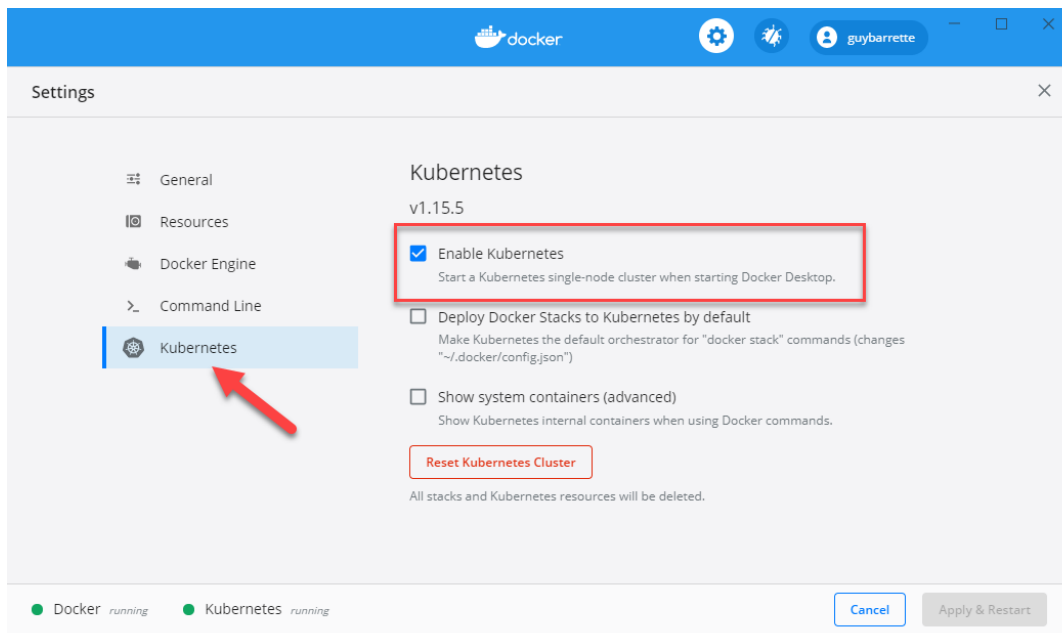
Open Docker Desktop and enable Kubernetes. This will download and install additional files and containers in your Docker Desktop installation.

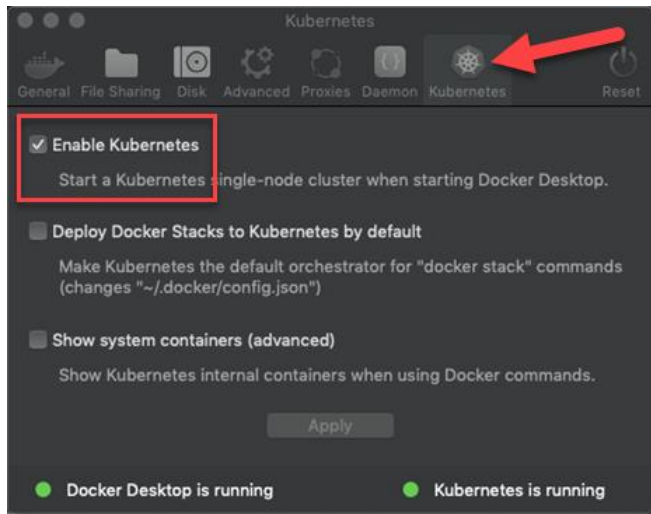
Windows

On Windows, make sure that Docker is using Linux containers. To validate, right-click on the Docker icon in the system tray and if you see **Switch to Windows containers...**, you're OK 😊



Open Docker Desktop and enable Kubernetes.







Kubernetes – Docker Engine

If you installed Docker Engine, you have multiple choices: Minikube, MicroK8s and Kind amongst many other. Minikube requires virtualization while MicroK8s and Kind do not. MicroK8s is super easy to install but the choice is yours!

MicroK8s

<https://ubuntu.com/kubernetes/install>

```
sudo snap install microk8s --classic
sudo snap alias microk8s.kubectl kubectl
```

Minikube (requires virtualization)

<https://minikube.sigs.k8s.io/docs/start/>

Install Kubernetes with Minikube on VirtualBox using these commands:

```
sudo apt-get update
sudo apt-get install -y apt-transport-https
sudo apt-get install -y virtualbox virtualbox-ext-pack
curl -Lo minikube https://storage.googleapis.com/minikube/releases/v0.28.2/minikube-linux-amd64
chmod +x minikube && sudo mv minikube /usr/local/bin/
minikube start
```

Kind

<https://kind.sigs.k8s.io/docs/user/quick-start/>

```
curl -Lo ./kind https://kind.sigs.k8s.io/dl/v0.11.1/kind-linux-amd64
chmod +x ./kind

# move to a folder in your path
mv ./kind /some-dir-in-your-PATH/kind

# create a cluster with the default name "kind"
kind create cluster
```

Kubernetes CLI and other tools

Install the following tools:



The Kubernetes CLI should be installed by Docker Desktop. To confirm, open a terminal and type **kubectl**. If not installed, you can install it using chocolatey.

```
choco install kubernetes-cli
```

Also install these tools:

```
choco install k9s
```

```
choco install lens
```



The Kubernetes CLI should be installed by Docker Desktop. To confirm, open a terminal and type **kubectl**. If not installed, you can install it using Brew.

```
brew install kubectl
```

Also install these tools:

```
brew install --cask lens
```

```
brew install derailed/k9s/k9s
```



```
snap install kubectl --classic
```

Install K9s using the instructions found here:

<https://github.com/derailed/k9s>

Windows – Optional tools

Optionally, you can install Windows versions of these Unix tools. They will prove very useful.

Before installing these, open a terminal and type the commands (curl, grep and sed) to see if they are already installed or not.

```
choco install curl
```

```
choco install grep
```

```
choco install sed
```










Windows Terminal

It is highly recommended that you use the Windows Terminal. It's much better than the old Command Prompt and it's available for free from the Windows Store. On Windows 11, Terminal is installed by default.

<https://docs.microsoft.com/en-us/windows/terminal/get-started>

Checklist

Use this checklist to validate your installation.

		✓
	I have a Docker account and I can log to https://hub.docker.com	
	On Windows, I have installed Chocolatey and if I type choco at the command prompt/terminal I get some information and no errors	
	On macOS, I have installed Brew and if I type brew at the command prompt/terminal I get some information and no errors	
	I have installed Visual Studio Code and can launch it	
	I have installed Docker Desktop and if I type docker at the command prompt/terminal I get some information and no errors	
	I have installed the Docker Engine instead of Docker Desktop and if I type docker at the command prompt/terminal I get some information and no errors	
	On Windows or macOS, I have enabled Kubernetes in Docker Desktop	
	I have enabled Kubernetes in Docker Desktop	
	I have kubectl and if I type kubectl at the command prompt/terminal I get some information and no errors	