Kubernetes Workshop
Software installation

If you face any issues during the installation, please contact me directly at victor.tan.33@gmail.com

Please ensure that you complete the installation **PRIOR** to the workshop, as the installation process may take over an hour to complete: this will depend on your machine specifications and broadband connection speed.

1 Install a PDF Reader

This is for viewing the lab manuals which are in PDF format. You can also view PDF files in a browser, but this can crash abruptly and disrupt your lab session. A dedicated PDF reader is more stable.

Two good options are:

https://www.adobe.com/acrobat/pdf-reader.html

https://www.foxit.com/pdf-reader/

2 Install a full-featured text editor

You will need a good text editor to open text files.

For Windows, a good option is:
Notepad++
https://notepad-plus-plus.org/downloads/v8.1.1/

For Linux/MacOS:
There should already be many preinstalled editors such as GEdit, Nano, Vim, Emacs, etc. You can also use VS Code, Atom, Brackets or any of the other popular IDEs out there.

3 Install Visual Studio Code

We need an IDE to work with the source code in the sample projects.

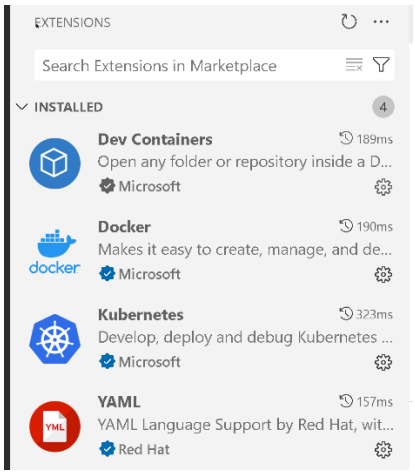
A popular open-source IDE is Visual Studio (VS) Code
https://code.visualstudio.com/

VS Code also has some useful extensions that facilitate working with Docker and Kubernetes.

https://code.visualstudio.com/docs/editor/extension-marketplace

You can install the following extensions for this purpose

- Docker
- Kubernetes
- Dev Containers
- YAML



If you are using your own IDE and are comfortable with that, feel free to stick with it.
All the VS code extension functionality can also be achieved from standard CLI commands.

4 Install PowerShell 7

This is only necessary if you are working on a Window system. For Linux, you can use the standard Bash shell terminal.
NOTE: Powershell 7 is a **different** application from the standard Windows PowerShell that comes preinstalled on all Windows distribution.

The MSI installation executable can be downloaded from:
https://learn.microsoft.com/en-us/powershell/scripting/install/installing-powershell-on-windows?view=powershell-7.4

The installation process is outlined below:

https://www.youtube.com/watch?v=3E-iBL4Nyt0

5 Install a REST API client (Postman)

We need a REST API client to send HTTP POST and GET requests to the sample apps that will be running in containers and pods.

There are many REST API client options available for Windows and Linux.
A popular option is Postman:

https://www.postman.com/downloads/

You can register to use this application for free by signing up with an account (or using your existing personal GMail account).

6 Install Docker engine

You should have at least 8GB on your machine for Docker to run smoothly.

There are two possible options for installing a Docker engine:

- Docker Desktop (most widely used package, but specific licensing requirements apply)
- Rancher Desktop (most popular open source package)

Regardless of which option you are using, you will need WSL2 (Windows Subsystem for Linux) if you are installing on a Windows system, which is a requirement for the Docker engine.

You can check whether WSL is installed on your Windows system with:
https://pureinfotech.com/check-wsl-version-windows-10/

If not installed, try to install the latest version (WSL2):
https://www.omgubuntu.co.uk/how-to-install-wsl2-on-windows-10
https://gcore.com/learning/how-to-install-wsl-2-on-windows/

You may also need to enable BIOS-level hardware virtualization support (if it is not already enabled) in order to run a Docker engine using WSL2. If you have virtualization errors reported while installing

Docker Desktop or Rancher Desktop, then you will probably need to reboot your PC and access the BIOS menu to turn on virtualization. Some sample tutorials on doing this:

https://www.youtube.com/watch?v=ZDeje9wgDp4
https://www.youtube.com/watch?v=MOuTxfzCvMY

6.1 Rancher Desktop

https://rancherdesktop.io/

A video demonstrating installation (if you need it) is available at:
https://www.youtube.com/watch?v=MwwF-dWtEAk

6.2 Docker Desktop

IMPORTANT: Please read Docker Subscription Policy if you are working in a large company / organization prior to installation **to avoid any legal issues.**
https://www.docker.com/pricing/

Installing on Windows:
https://docs.docker.com/docker-for-windows/install/

Installing on Linux:
https://docs.docker.com/desktop/install/linux-install/

Some Youtube videos that demonstrate the installation process step by step
https://www.youtube.com/watch?v=ZyBBv1JmnWQ
https://www.youtube.com/watch?v=WDEdRmTCSs8

WSL2 backend is the more recent and efficient way for installing and running Docker on Windows. However, if for some reason, you can't install WSL2, you can still use the Hyper-V backend. However, this can cause problems if you are also using virtualization platform software like Virtual Box or VMWare Workstation on Windows. If you have installed Docker using Hyper-V instead of WSL, you will need to disable Hyper-V first to run VMWare Workstation, and then reenable it when you want to run Docker:

https://kb.vmware.com/s/article/2146361
https://docs.microsoft.com/en-us/troubleshoot/windows-client/application-management/virtualization-apps-not-work-with-hyper-v

7 Install Kubernetes

If you installed **Rancher Desktop**, this **is already a complete Kubernetes installation, so no additional installation is necessary.**

If you installed Docker Desktop, the recommended option for installing Kubernetes is Minikube.

Make sure that Docker Desktop is now running and active first before attempting a Minikube installation.
https://minikube.sigs.k8s.io/docs/start

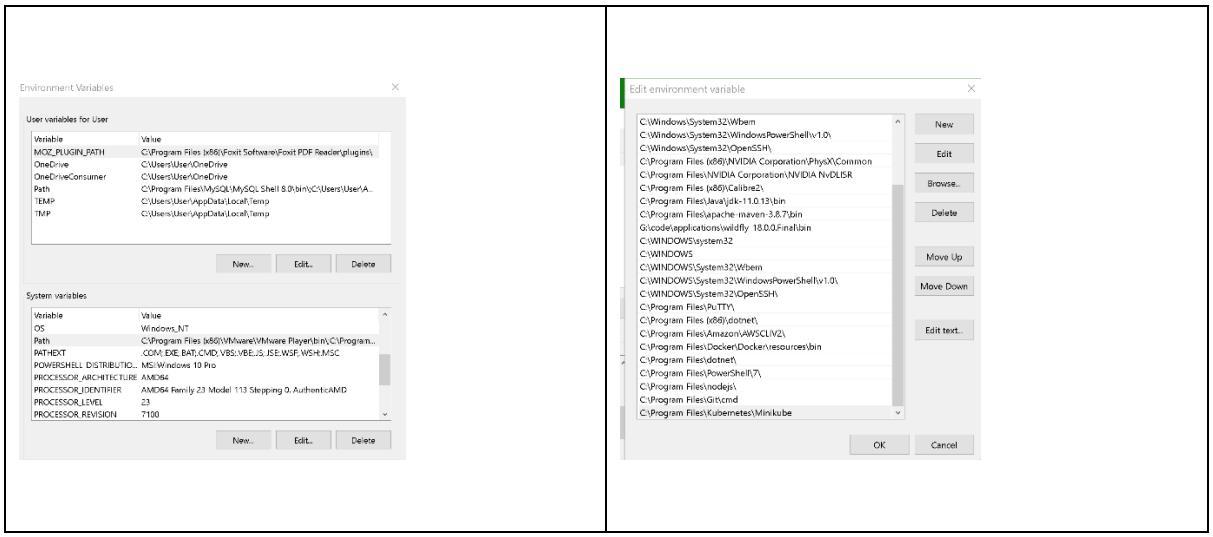
There are 2 main ways to install:

Approach 1: Select and use `.exe download` installer type (this is the default chosen and is also the easiest).
https://storage.googleapis.com/minikube/releases/latest/minikube-installer.exe

Approach 2: Copy and paste the provided PowerShell commands into a PowerShell terminal
This approach is demonstrated here
https://www.youtube.com/watch?v=xNefZ51jHKg

The easiest is Approach 1: download and run the installer executable. When installation is complete, the `minikube.exe` binary executable will be by default located in this directory.
`C:\Program Files\Kubernetes\Minikube`
This directory should have been automatically added to your system PATH environment variable as part of the installation process. You can verify by checking the PATH environment variable and placing this directory value in there if it is not already automatically set.

https://www.wikihow.com/Change-the-PATH-Environment-Variable-on-Windows



Open a Powershell 7 terminal and type:

`minikube`

You should see some console output regarding the commands available

```
PS G:\> minikube
minikube provisions and manages local Kubernetes clusters optimized for development workflows.

Basic Commands:
  start            Starts a local Kubernetes cluster
  status           Gets the status of a local Kubernetes cluster
  stop             Stops a running local Kubernetes cluster
  delete           Deletes a local Kubernetes cluster
  dashboard        Access the Kubernetes dashboard running within the minikube cluster
  pause            pause Kubernetes
  unpause          unpause Kubernetes
```

Next type:

```
minikube status
```

To verify that that the Minikube single node cluster is in fact up and running.

If the above installation approach is not working, follow the previous Youtube video link demonstrating installation process by copying and pasting the Powershell commands from the installation page.

8 Sign up for a DockerHub account

If you don't already have one, sign up for a free DockerHub account at:

https://hub.docker.com/

Instructions are outlined here
https://docs.docker.com/docker-id/

9 Preparations

9.1 Broadband connection bandwidth

A significant portion of lab activity involves downloading images from the DockerHub public registry. Some of these images may be up to 20MB or more in size. If your broadband connection has low bandwidth, these downloads may take excessively long to complete and delay your progress in the lab. If possible, try to ensure a connection speed of at least 5Mb/s during the Docker lab sessions.

9.2 Basic Linux command line knowledge

You will need some knowledge of basic Linux shell commands to work inside Docker Linux containers. If you have no experience at the command line in Linux, you can optionally prepare beforehand as we will not have time to do a beginner level Linux tutorial in the course.

Basic Linux tutorials:
https://ubuntu.com/tutorials/command-line-for-beginners
https://ryanstutorials.net/linuxtutorial/
http://www.ee.surrey.ac.uk/Teaching/Unix/