

Installing Docker on Linux

Installation of Docker Desktop on Native Hardware

Native Hardware means a physical laptop or desktop computer. If you are using WSL on Windows, then you should be installing Docker Desktop for Windows NOT Linux. If you are installing within a VM like VirtualBox or Parallels, or, a cloud server such as AWS, then, you must use the Installation on Cloud Servers or inside Virtual Machines instructions instead. Docker Desktop does not work with nested virtualization.

Currently Docker Desktop currently only works with the **Ubuntu**, **Debian**, or **Fedora** distributions. If you are using any other distribution, you will need to follow the "Installation on Cloud Servers or inside Virtual Machines" instructions.

1. Create Dockerhub account

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

2. Install Docker Desktop for Linux

Simply follow the generic installation instructions for your particular distribution:

<https://docs.docker.com/desktop/install/linux-install/#generic-installation-steps>

3. Login to the Dockerhub

In order to push and pull images, you will need to log in to the Dockerhub. In your terminal, run `docker login` and then enter your Dockerhub account username and password.

4. Test Docker installation

After completing the installation steps, test out Docker by running `docker run hello-world`. This should download and run the test container printing "hello world" to your console.

Installation on Cloud Servers or inside Virtual Machines

The steps listed below are for Ubuntu Desktop LTS. You can find the full official docs and steps for other Linux distributions here:

Ubuntu: <https://docs.docker.com/install/linux/docker-ce/ubuntu/>

CentOS: <https://docs.docker.com/install/linux/docker-ce/centos/>

(Note: Students have encountered issues when using CentOS or RHEL as the host related to Docker container communication. You may need to research some workaround for the errors you run into or search the QA for already posted solutions)

Debian: <https://docs.docker.com/install/linux/docker-ce/debian/>

1. **Create Dockerhub account**

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

2. **Install Docker**

The Docker docs suggest setting up a Docker repository to install and update from.

This is where you should

start: <https://docs.docker.com/engine/install/ubuntu/#install-using-the-repository>

3. **Login to the Dockerhub**

In order to push and pull images, you will need to log in to the Dockerhub. In your terminal, run `docker login` and then enter your Dockerhub account username and password.

4. **Test Docker installation**

After completing the installation steps, test out Docker by running `sudo docker run hello-world`. This should download and run the test container printing "hello world" to your console.

5. **Testing Docker Compose**

Important! The version of Docker Compose that is now installed with Docker does not include a symlink to the `docker-compose` (with a hyphen) command. This only exists in Docker Desktop. So, all commands should be run without a hyphen.

After completing, test your installation by running:

`docker compose -v`

This should print the version and build numbers to your console.

6. **Run without Sudo**

Follow these instructions to run Docker commands without sudo:

<https://docs.docker.com/install/linux/linux-postinstall/#manage-docker-as-a-non-root-user>

7. **Start on Boot**

Follow these instructions so that Docker and its services start automatically on boot:

<https://docs.docker.com/install/linux/linux-postinstall/#configure-docker-to-start-on-boot>

You may need to restart your system before starting the class material.