

Installing Docker on Ubuntu

Warning: this tutorial requires you to restart your computer.

For other distributions of Linux (for example Debian, Fedora, etc) see the link <https://docs.docker.com/desktop/setup/install/linux/>.

1. Install Docker from the Command Line

Unlike Windows or Mac, the best option for installing Docker on Ubuntu is through the command line. You can see below the commands you should copy and paste in the **terminal** in order to install the tool.

- a. Ensure that old versions of docker are removed to start fresh with this installation.

```
sudo apt remove docker docker-engine docker.io containerd  
run
```

→ When using **sudo** the terminal will ask you for the password you use to access your user to allow changes to be made to your computer.

- b. Install required packages

In this step we will update all system packages, and then install the Docker-related packages.

```
sudo apt update  
sudo apt install apt-transport-https ca-certificates curl  
software-properties-common
```

After the second command starts, it will ask you for **permission** to install the packages. You will need to type 'Yes' into the terminal to start the installation process.

You will be asked to select your region by entering the corresponding number from the list.

You will also need to then select your city (or the closest one available), and next your correspondent timezone.

c. Install Docker

In this step we will use the command line to install docker engine from the official repository. First, it adds a key to match the Docker version with your operating system (OS), and then it finally installs the software that includes all Docker tools. Don't worry about the details, just copy and paste the commands below.

```
# Add Docker's GPG key
```

```
curl -fsSL https://download.docker.com/linux/ubuntu/gpg |  
sudo gpg --dearmor -o  
/usr/share/keyrings/docker-archive-keyring.gpg
```

```
# Add Docker's official repository
```

```
echo "deb [arch=$(dpkg --print-architecture)  
signed-by=/usr/share/keyrings/docker-archive-keyring.gpg]  
https://download.docker.com/linux/ubuntu $(lsb_release -cs)  
stable" | sudo tee /etc/apt/sources.list.d/docker.list >  
/dev/null
```

```
# Install Docker Engine
```

```
sudo apt update  
sudo apt install docker-ce docker-ce-cli containerd.io
```

d. For Ubuntu 24.04, add your user to the docker group

To be able to use Docker with your user account, you need to add your user to the Docker group, by running the following command:

```
sudo usermod -aG docker $USER
```

- e. **Restart your computer** for the changes to take effect
-

2. Verify Docker Install

Open a **terminal** and run the code below (copy and paste it in the terminal, and press enter):

```
docker --version
```

```
docker run hello-world
```

The first command should output the docker version, for example:

Docker version 28.1.1

The second command should **download a test image** and **run a simple container**, verifying everything is working. Below is an example of the expected output.

```
Unable to find image 'hello-world:latest' locally
latest: Pulling from library/hello-world
e6590344b1a5: Pull complete
Digest: sha256:c41088499908a59aae84b0a49c70e86f4731e588a737f1637e73c8c09d995654
Status: Downloaded newer image for hello-world:latest

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/
```



3. Use Docker

You can now **pull the tabulizer image** we will use for the workshop (paste the command below in the terminal):

```
docker pull vpnagraj/tabulizer:latest
```