

Getting Started with Docker

Introduction

Prerequisites:

- General exposure to command line
- Basic knowledge of networking
- Basic Linux Shell/Bash scripting (Good to have, not necessary)

What is Docker?

Docker carves up a running Linux system into small independent self-sufficient executable packages of software called containers. Containers have their own programs, dependencies and everything that is required to run the particular software. This helps separate applications from the infrastructure, thereby enabling you to deliver software quickly.

These containers are portable between different systems, and Docker helps you create, run, and move your containers to and from other systems. Docker is NOT the same as virtual machines - there is a single operating system that is running.

Defining a container: A container is a self contained sealed unit of software. It has everything in it that is needed to run that code i.e. the operating system, all of the code, the configs, and it contains all the processes within that container. Additionally, it has all of the networking to allow these containers to talk to the other containers they're supposed to be able to talk to, and no other containers. It has all the dependencies that your system needs, bundled up in that container. And it even includes just enough of the operating system to run your code.

Installation:

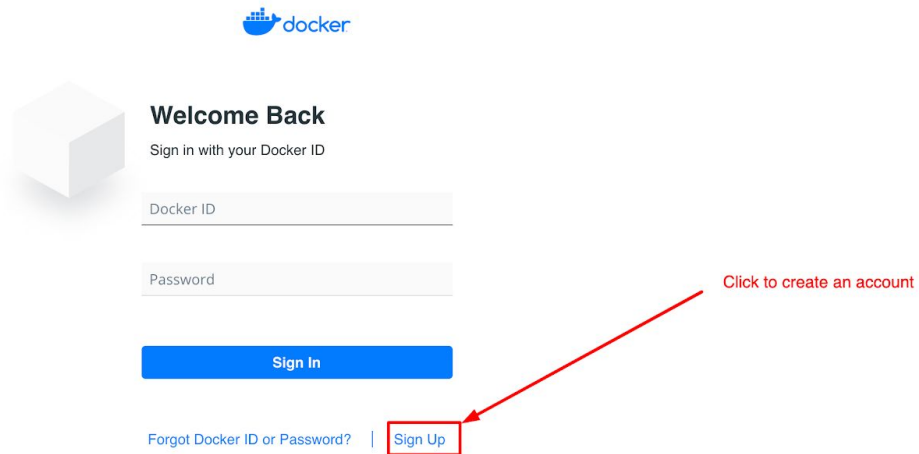
Note: If you have the old version of boot2docker installed - Please uninstall it, and follow the steps below to install the latest version of Docker via Docker Desktop.

Link to official documentation: <https://docs.docker.com/get-docker/>

Mac OS

URL: <https://docs.docker.com/docker-for-mac/install/>

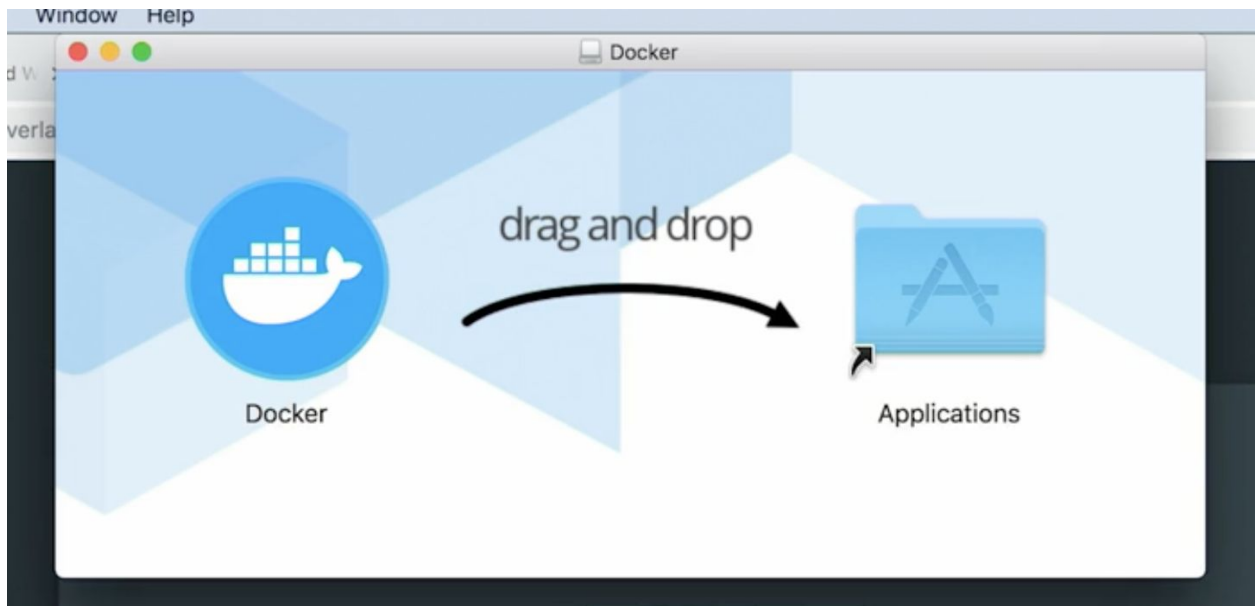
1. Create/Sign up for a docker account if you do not already have one - <http://hub.docker.com/sso/start>



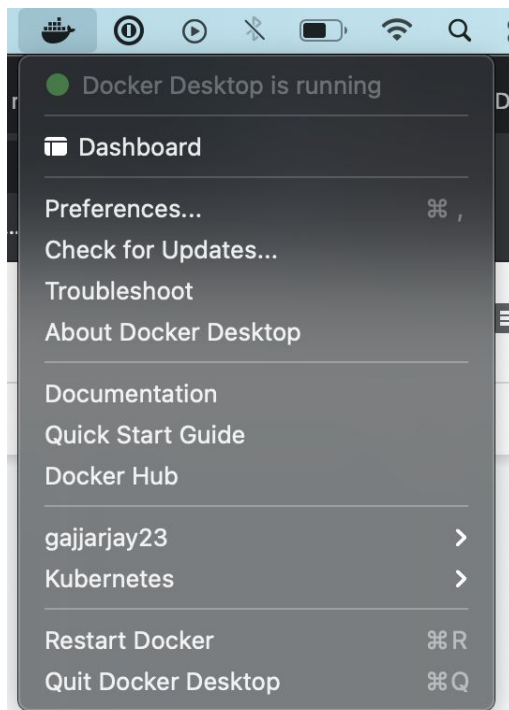
2. Go to Google, and type in Docker Desktop, click on the first link - which will take you to the official docker desktop page - <https://www.docker.com/products/docker-desktop> (this link may change over the years)



3. Choose Download for Mac - and this should start a download for a disk image. Once downloaded - click on it, and this should start the installation process. Drag and drop the icon into the applications folder to complete the installation.

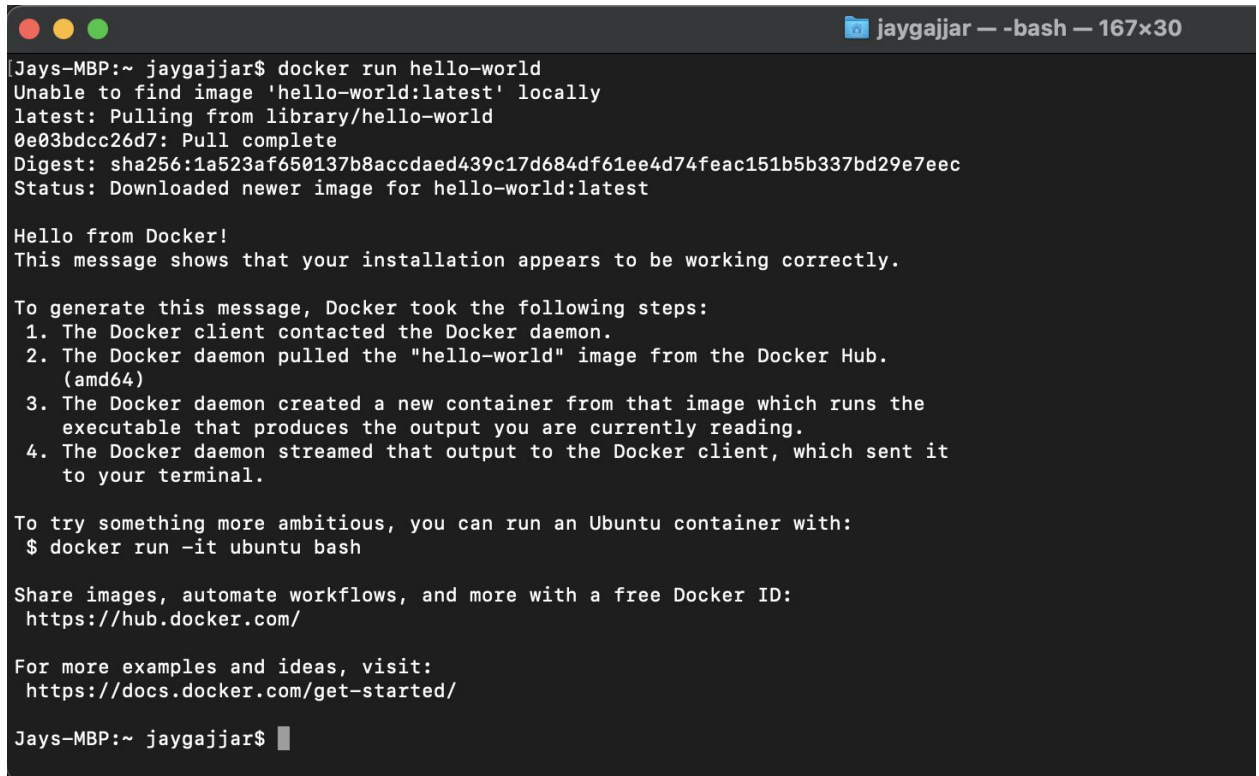


4. You can now go ahead, and start docker from the the finder.
5. Docker Desktop should now ask you to login - you can go ahead and login with the credentials that you used to create your account in the first step.



6. You can click on the whale at the top of your screen to get quick links and information about your docker. You can use the "Restart Docker" option to refresh your docker settings, just in case something does not work as usual, or as it should. In extreme cases - you can choose to quit Docker Desktop, and then start it back up.

7. To test the running of docker - Go ahead to the terminal - and execute the command:
docker run hello-world.



```
Jays-MBP:~ jaygajjar$ docker run hello-world
Unable to find image 'hello-world:latest' locally
latest: Pulling from library/hello-world
0e03bdcc26d7: Pull complete
Digest: sha256:1a523af650137b8accdaed439c17d684df61ee4d74feac151b5b337bd29e7eec
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/

Jays-MBP:~ jaygajjar$
```

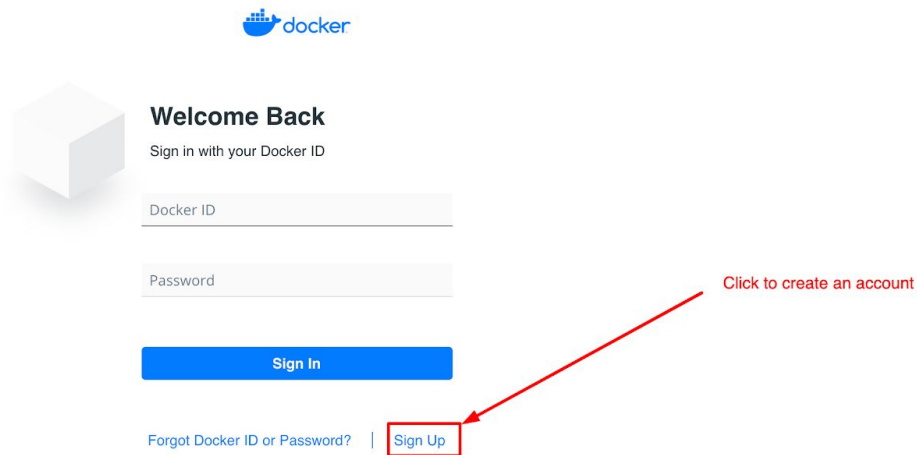
8. You should get an output as shown above. If the hello-world image does not exist locally, docker will pull it from the web, and then execute it.

In case of any errors while running the command (for example: cannot connect, or socket not available etc.) - you can just click on the whale icon, and double check that Docker is running, and restart it as mentioned in step 6.

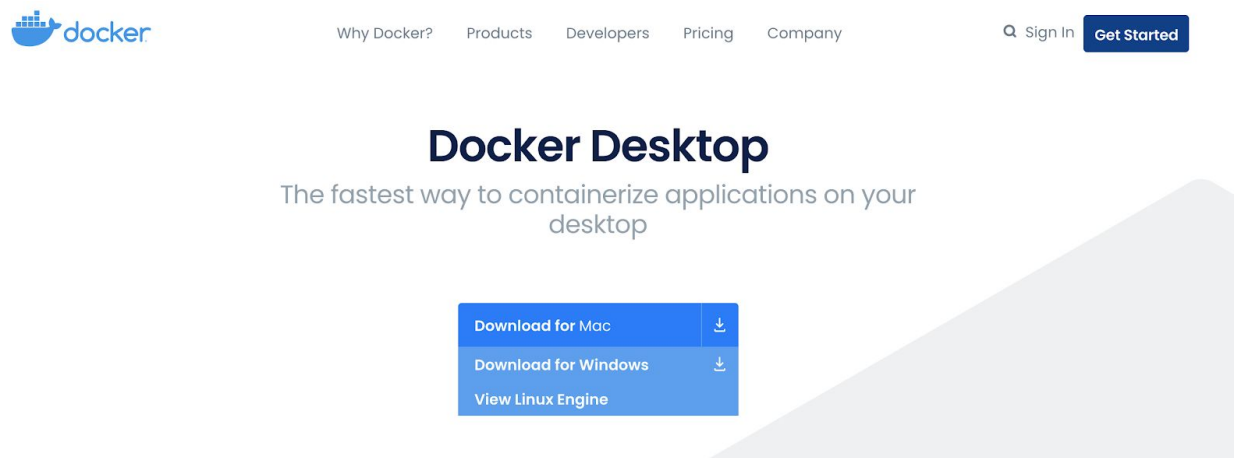
Windows

URL: <https://docs.docker.com/docker-for-windows/install/>

1. Create/Sign up for a docker account if you do not already have one - <http://hub.docker.com/sso/start>

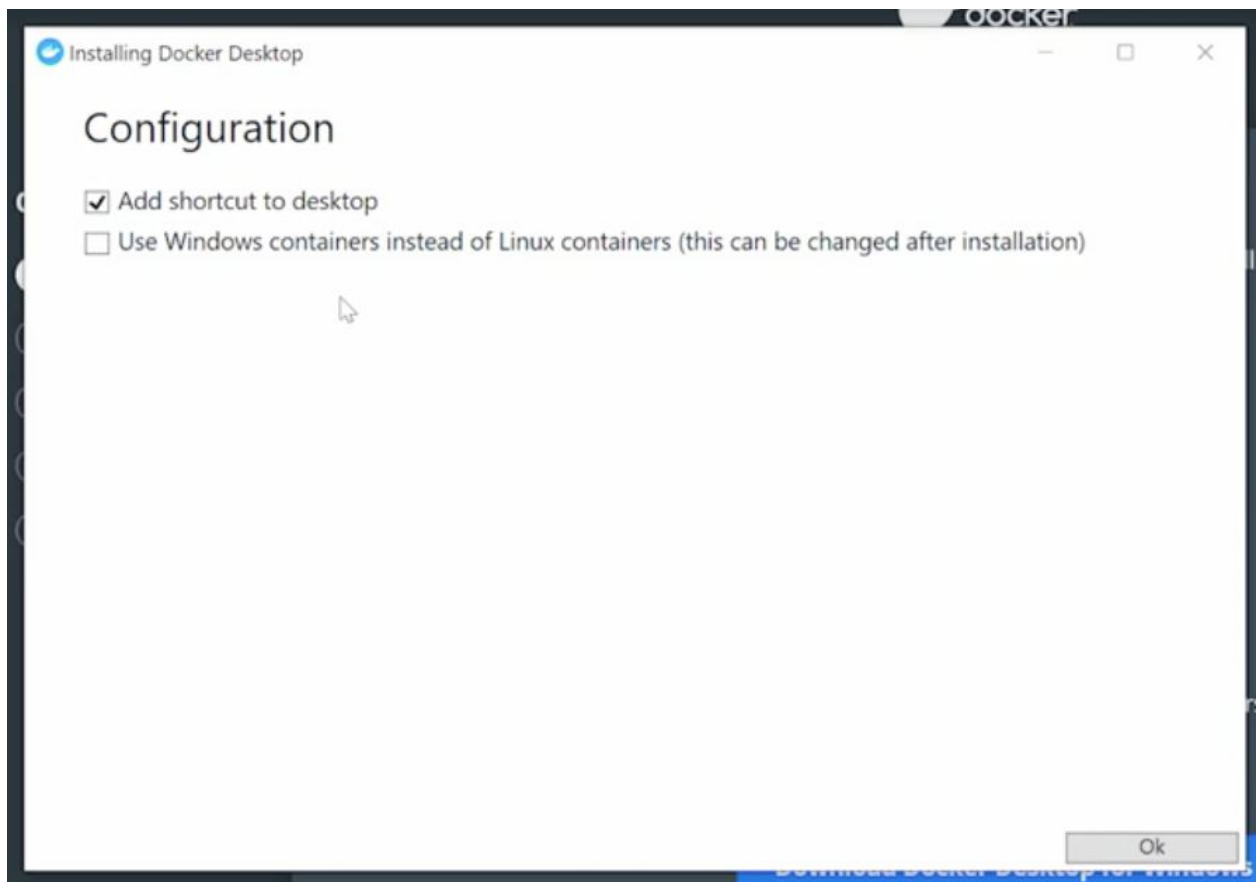


2. Go to Google, and type in Docker Desktop, click on the first link - which will take you to the official docker desktop page - <https://www.docker.com/products/docker-desktop> (this link may change over the years)



3. Select Download for Windows - and start the installation, once the download is complete. Once it starts installing, there will be a popup - "Use Windows containers "instead of Linux containers"? - Ensure that this option is **unchecked**. One of the key points of

Docker is that it allows you to be cross-platform.



4. On Windows, Docker needs to activate Hyper-V, Windows' virtualization platform, so that it can create Linux containers on a Windows environment. This will probably require you to reboot your computer.
5. To test if Docker is installed correctly - run the command - *docker run hello-world*.

```
C:\Users\Arthur Ulfeldt>docker run hello-world
Unable to find image 'hello-world:latest' locally
latest: Pulling from library/hello-world
1b930d010525: Pull complete
Digest: sha256:b8ba256769a0ac28dd126d584e0a2011cd2877f3f76e093a7ae560f2a5301c00
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/
```

In case of any errors while running the command (for example: cannot connect, or socket not available etc.) - you can just click on the whale icon, and double check that Docker is running, and restart it. Restarting it should fix most of the issues - and if it does not, then quit it, and then open docker again.

Docker for Linux (Ubuntu)

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

1. Installation of Docker is just like installation of any other program on Linux. You will need to install Docker CE.
2. If you have any older versions of docker, you can remove them with the help of the following command: `sudo apt-get remove docker docker-engine docker.io containerd runc`.
3. Ensure your system has its dependencies up to date with - `sudo apt-get update`
4. Next, we install the docker dependencies that need to be in place:
`sudo apt-get install \
 apt-transport-https \
 ca-certificates \
 curl \
 gnupg-agent \
 software-properties-common`
5. The next command makes sure that the system trusts the software provided by docker:
`curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo apt-key add -`
 - a. You can verify that you have the key with the fingerprint 9DC8 5822 9FC7 DD38 854A E2D8 8D81 803C 0EBF CD88, by searching for the last 8 characters of the fingerprint - by using the following command: `sudo apt-key fingerprint 0EBFCD88`
6. Next, go ahead and run the command below. What this does is - call the `apt-add-repository`, which adds an official source of software, saying we're going to add a source of Debian packages for the, in this case, amd64 architecture, which is one of the most commonly used architectures. The `lsb_release` command inside a subshell, is just a little bit of shell magic to automatically detect which version of Ubuntu your system is running and fetch the appropriate packages.
`sudo add-apt-repository \
 "deb [arch=amd64] https://download.docker.com/linux/ubuntu \
 $(lsb_release -cs) \
 stable"`
7. You can now go ahead and refresh the dependencies using `sudo apt-get update` and then install the latest version of Docker CE using `sudo apt-get install docker-ce docker-ce-cli containerd.io`.
8. Another step - that might be useful is to add your user to the list of users, who are allowed to use the docker command in the shell without using sudo. For this, you can create a docker group and add your user/list of users to it.

9. To create the group, run the command - `sudo groupadd docker`. Add your user to this group by using `sudo usermod -aG docker $USER`. `$USER` will take consider the currently logged in user.
10. Simply log out as the current user, and log back in for the changes to take effect.
11. Run the command `docker run hello-world` to test the setup!

```
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/engine/userguide/
```


References:

- [1] Docker Documentation - <https://docs.docker.com/>
- [2] Docker MacOS Installation - <https://docs.docker.com/docker-for-mac/install/>
- [3] Docker Windows Installation - <https://docs.docker.com/docker-for-windows/install/>
- [4] Docker Linux Installation - <https://docs.docker.com/engine/install/linux-postinstall/>