

Install Docker Desktop on macOS without Admin

2023-11-16

Contents

1 Summary	1
2 Procedure	1
2.1 Docker web site	1
2.2 macOS	2
2.3 Docker Desktop installation	2
2.4 Terminal	6

1 Summary

Instructions to install Docker on macOS for which user does not have Admin privileges. These instructions are meant for iMacs in Biochem 201 but should work elsewhere.

2 Procedure

1. download Docker Desktop
2. create local **Applications** directory
3. launch and uncheck options requiring admin password

Optional: register with Docker

2.1 Docker web site

1. Go to docker.com and click “Get Started” button at the top right.
2. Download Docker for Mac. Select appropriate version: “Apple Chip” or “Intel Chip”



Figure 1: Click GetStarted and download Chip version.

3. Double-click on the downloaded file `Docker.dmg`. This will open a window suggesting to copy the Docker icon onto the **Applications** folder. However, this action would require Admin privileges.

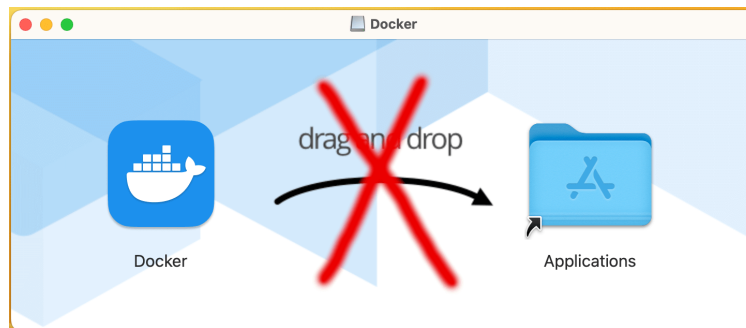


Figure 2: Do not move there unless you have Admin privileges.

2.2 macOS

The easiest way to install applications is to create a new folder within your username area, which can also be called **Applications**.

One way is to right-click and request a new directory.

4. Copy the Docker icon into the Applications folder just created.

2.3 Docker Desktop installation

5. Double click on the Docker icon
6. If asked "... Are you sure you want to open it?" click on "Open" button

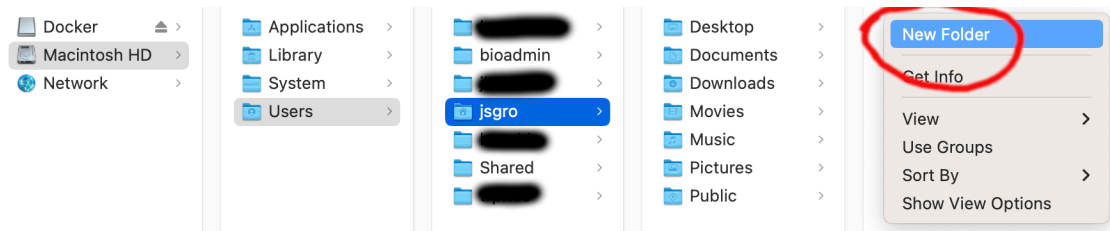


Figure 3: Create an Applications directory in your Users home.

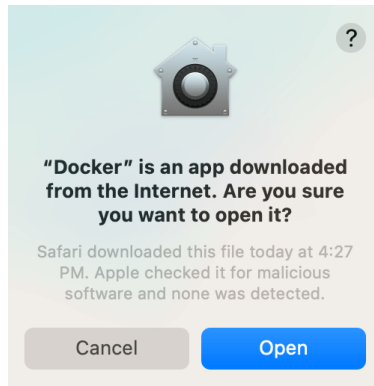


Figure 4: Click Open button to confirm.

7. Select non-Admin settings from the presented window.

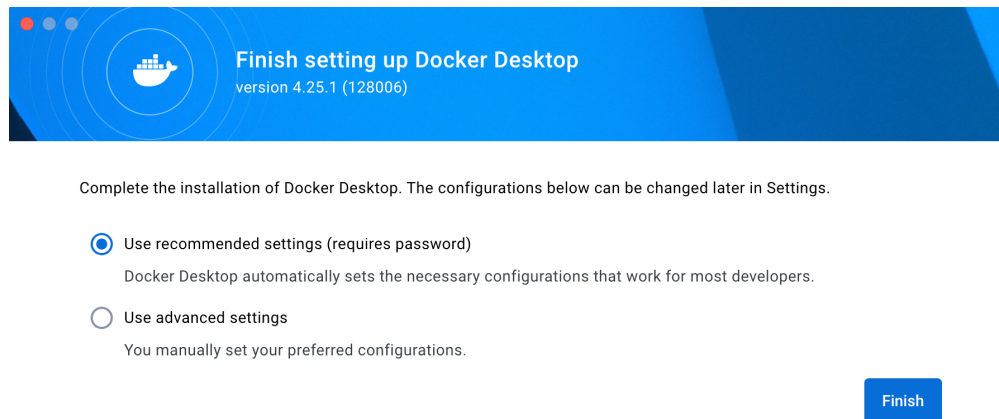
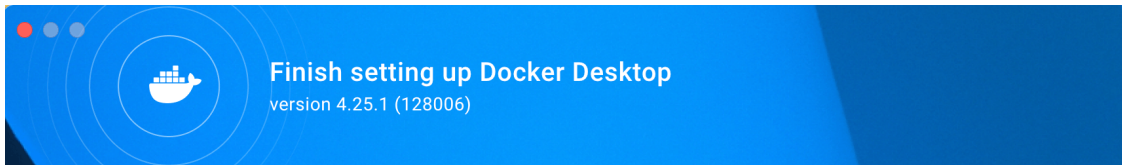


Figure 5: Select Use Advanced Settings.

8. In the next option offered unselect System and choose User button.

9. Uncheck the next 2 options that require password.



Complete the installation of Docker Desktop. The configurations below can be changed later in Settings.

- Use recommended settings (requires password)
Docker Desktop automatically sets the necessary configurations that work for most developers.
- Use advanced settings
You manually set your preferred configurations.

Choose how to configure the installation of Docker's CLI tools:

- System (requires password)
Docker CLI tools are installed under `/usr/local/bin`.
- User
Docker CLI tools are installed under `$HOME/.docker/bin`. Note: You need to manually add `$HOME/.docker/bin` to your PATH. [Learn more](#)
- Allow the default Docker socket to be used (requires password)
Creates `/var/run/docker.sock` which some third party clients may use to communicate with Docker Desktop.
- Allow privileged port mapping (requires password)
Starts the privileged helper process which binds privileged ports that are between 1 and 1024. [Learn more](#)
- Automatically check configuration
Regularly checks your configuration to ensure no unexpected changes have been made by another application.

Finish

Figure 6: Select User and uncheck options requiring password. Click Finish.

- on the next panel, choose “Sign-in” if you created a Docker user previously, or select “Continue without signing in”.



Figure 7: Log-in or continue without signing in.

- Fill-in or Skip the short survey.
- Docker is now ready to use. Check that the engine is running at the bottom left of the main window.

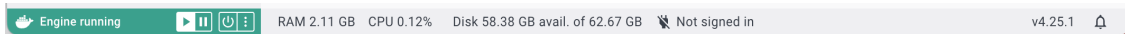


Figure 8: Check that engine is running.

A top-menu icon is available to access Docker Desktop, offering detailed menus.

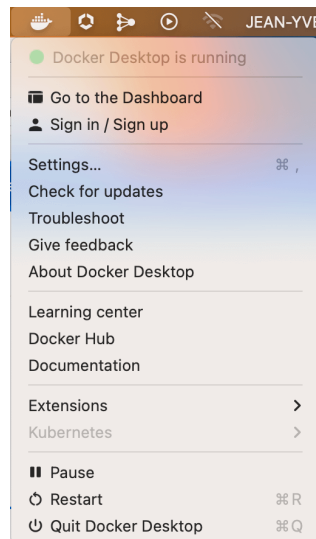


Figure 9: Top menu icon.

2.4 Terminal

It is important to follow the instructions given above within the **User** settings to provide the necessary information to macOS since the Users **Applications** folder we created is not in a standard location.

The instructions were: You need to manually add `$HOME/.docker/bin` to your `PATH`.

Unfortunately, the “Learn more” link does not contain the clear instructions needed, which are detailed below.

To make this a permanent change, the instruction has to be added to a hidden file specific to the shell used. On newer macOS, the default shell is `zsh` but some users keep the older `bash` option. Select below the command matching your shell.

Start by opening **Terminal** and inquire which shell it is using:

```
echo $SHELL
```

The answer may be either `/bin/bash` or `/bin/zsh`

Add the command to the “resource” file with the command below matching the shell in use:

```
# for bash  
echo "export PATH=$HOME/.docker/bin:$PATH" >> ~/.bashrc  
  
# for zsh  
echo "export PATH=$HOME/.docker/bin:$PATH" >> ~/.zshrc
```

This means that this line will be activated the next time a Terminal is open.

Docker is now ready to be used with the usual commands.