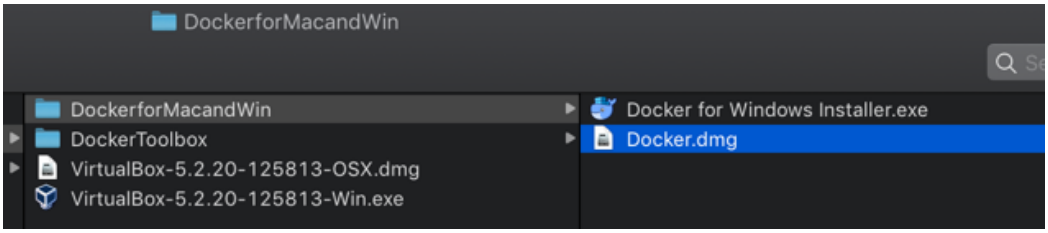
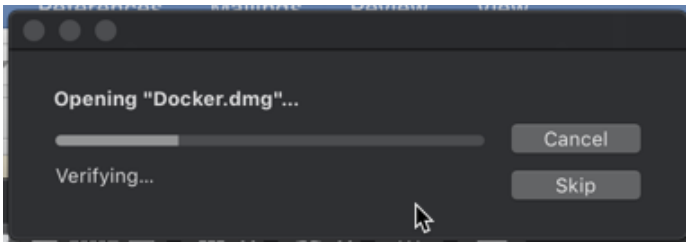
Manual

Install Docker for Mac OSX

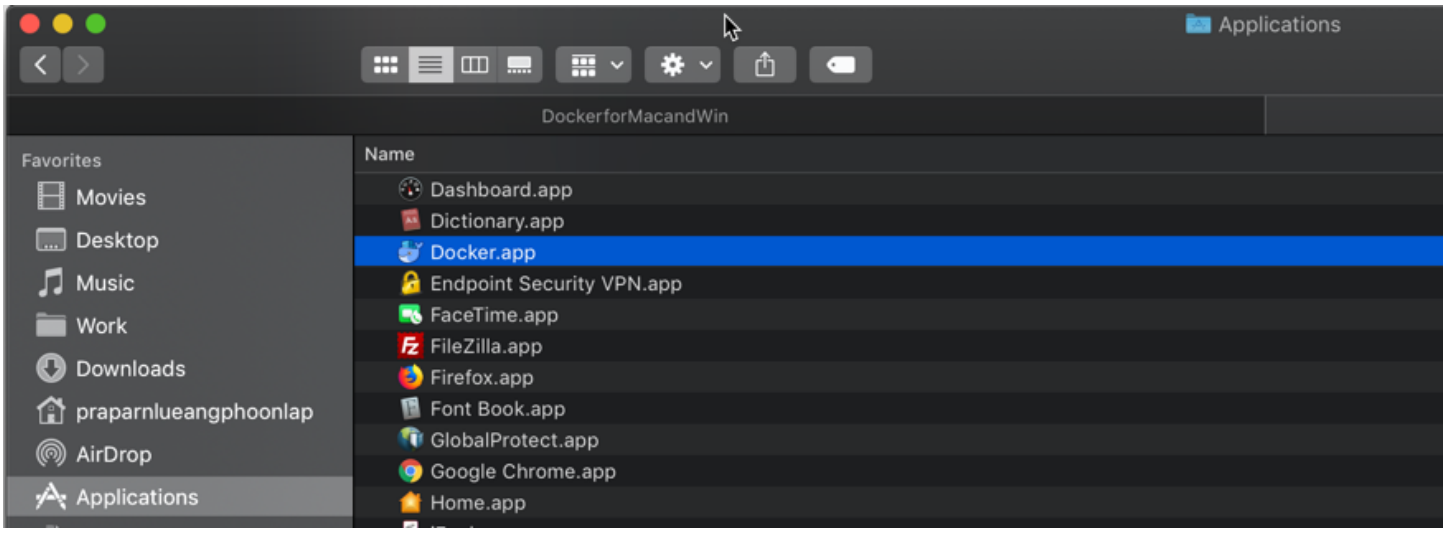
Install Docker for Mac OS

1. Double click "Docker.dmg" from path "software" for start installation

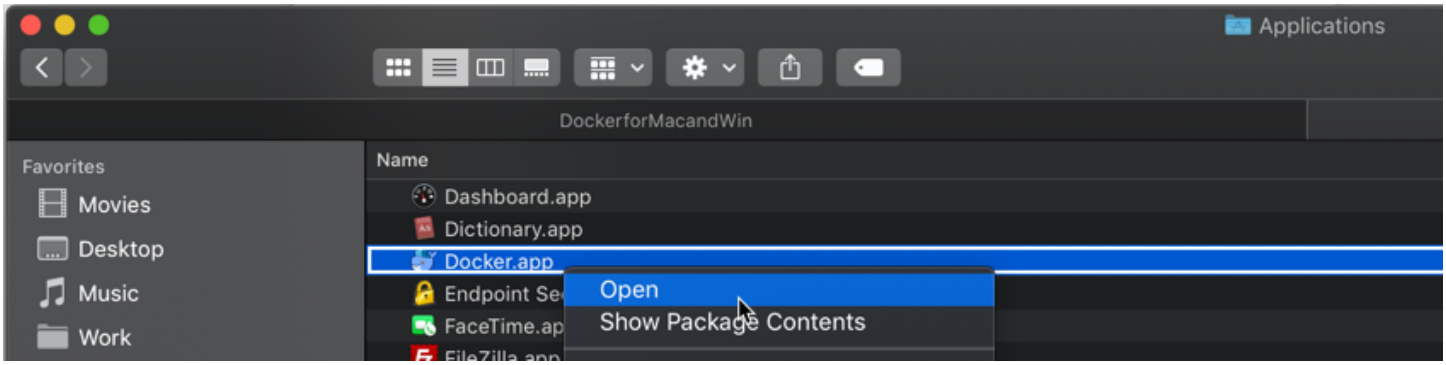


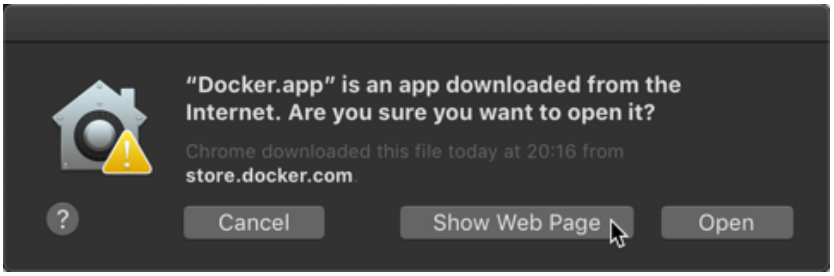


2. Drag docker.app to Applications

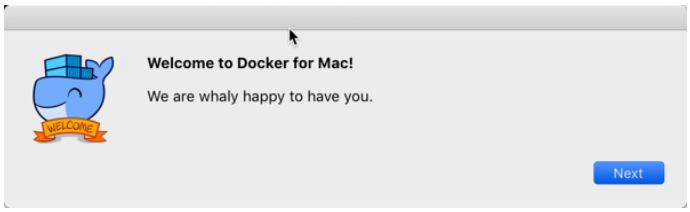


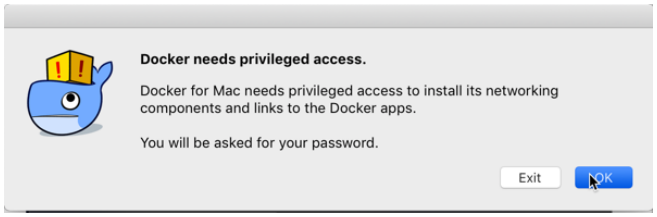
3. Start docker by click "Docker" and "Open" or from sportlight search "docker" and Click "Open"

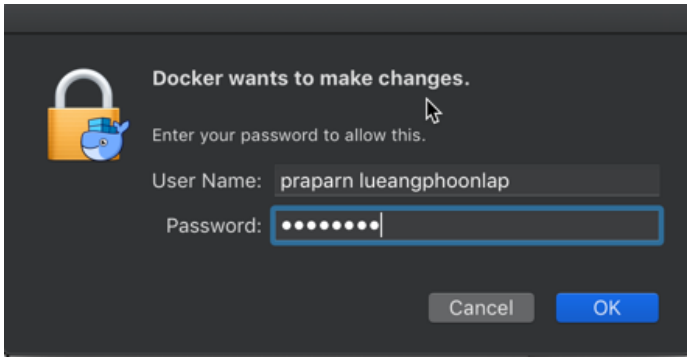




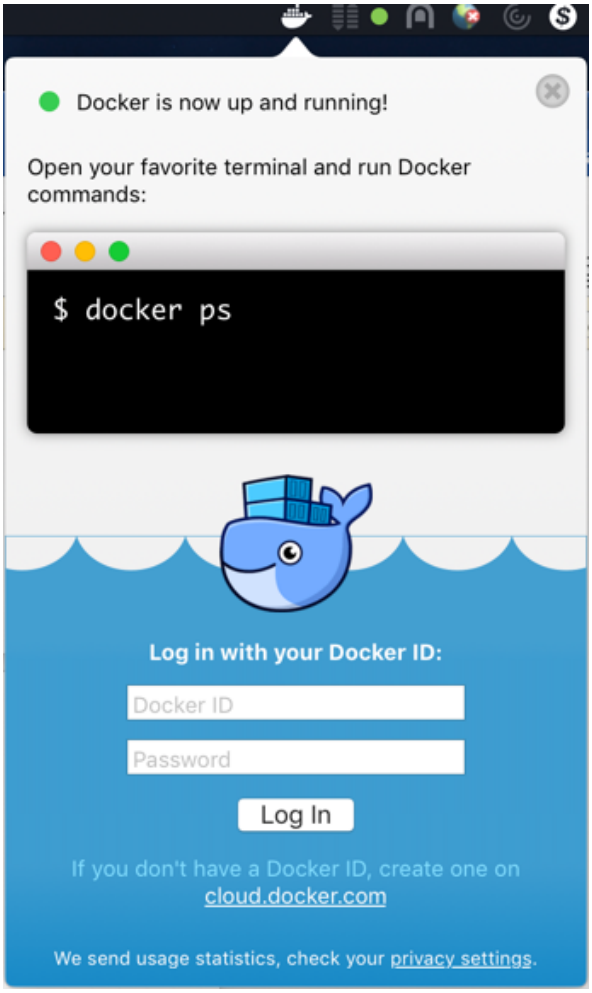
4. Following screen for setup docker for mac

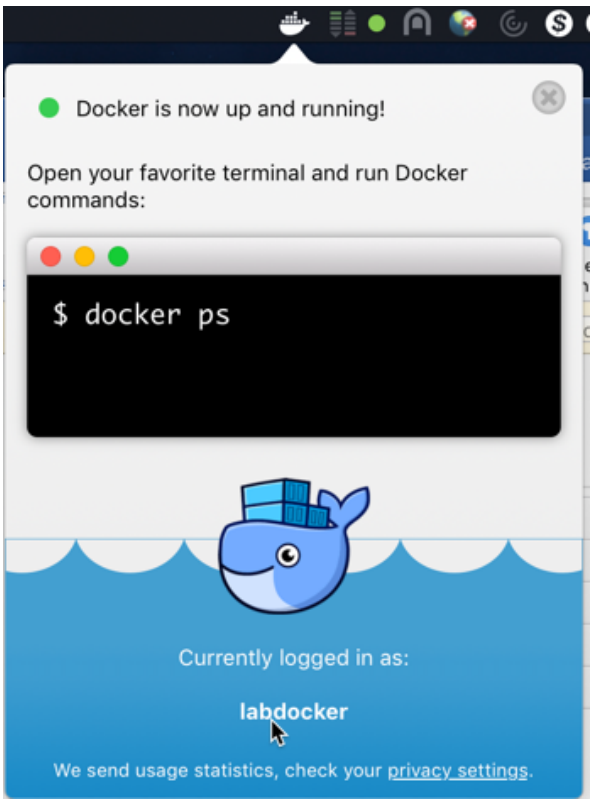




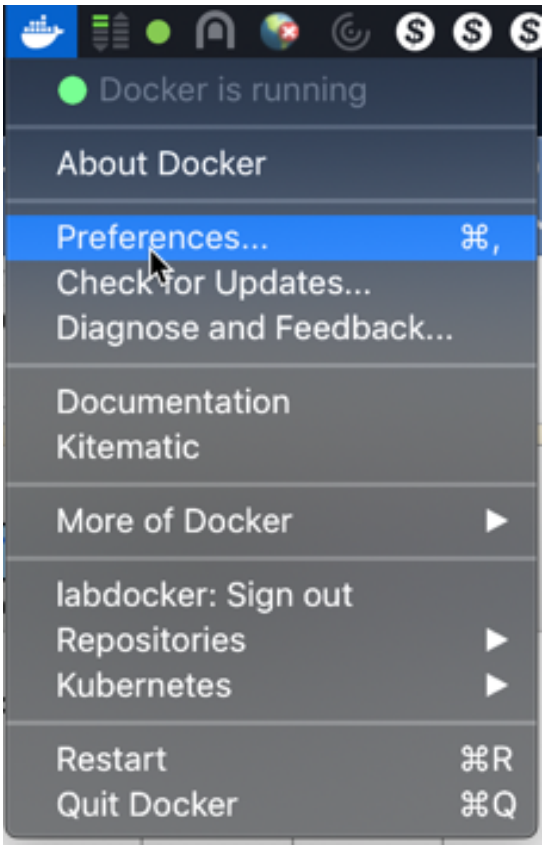


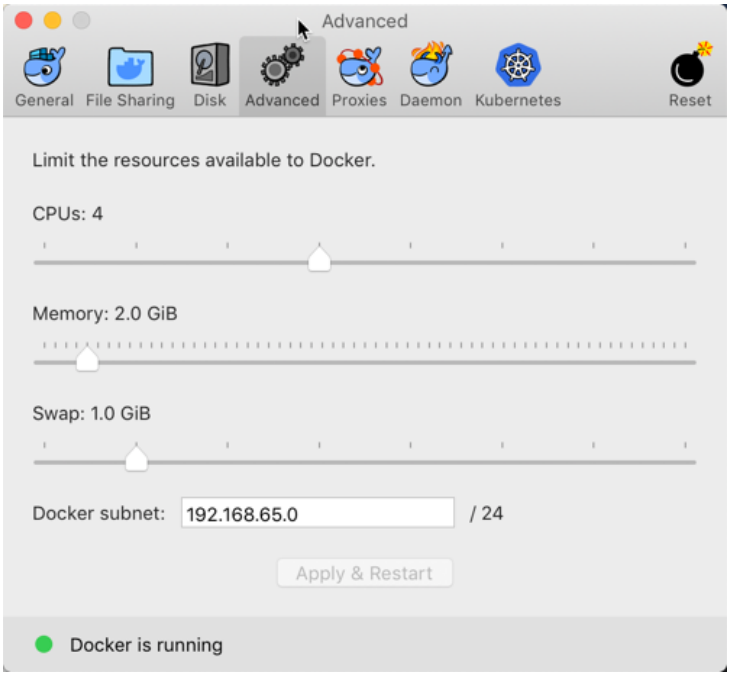
5. Docker will show starting method and prompt for login. If you already have user on "hub.docker.com" you can sign-in here



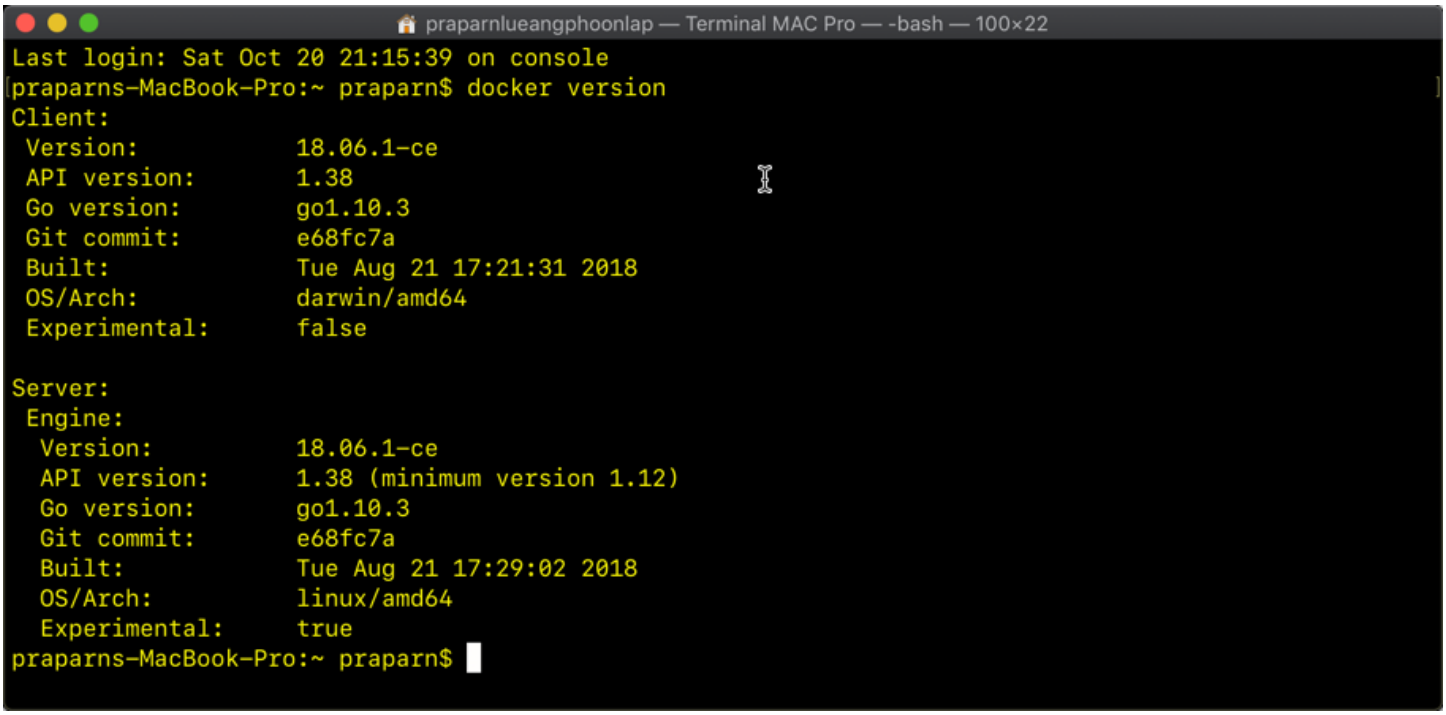


6. Right click on Docker's icon select "Preference" ➔ "Advance" and make sure that the resource was allocated for CPUs: at least 3 and Memory 2.0 GiB (If not you can adjust and click "Apply & Restart"

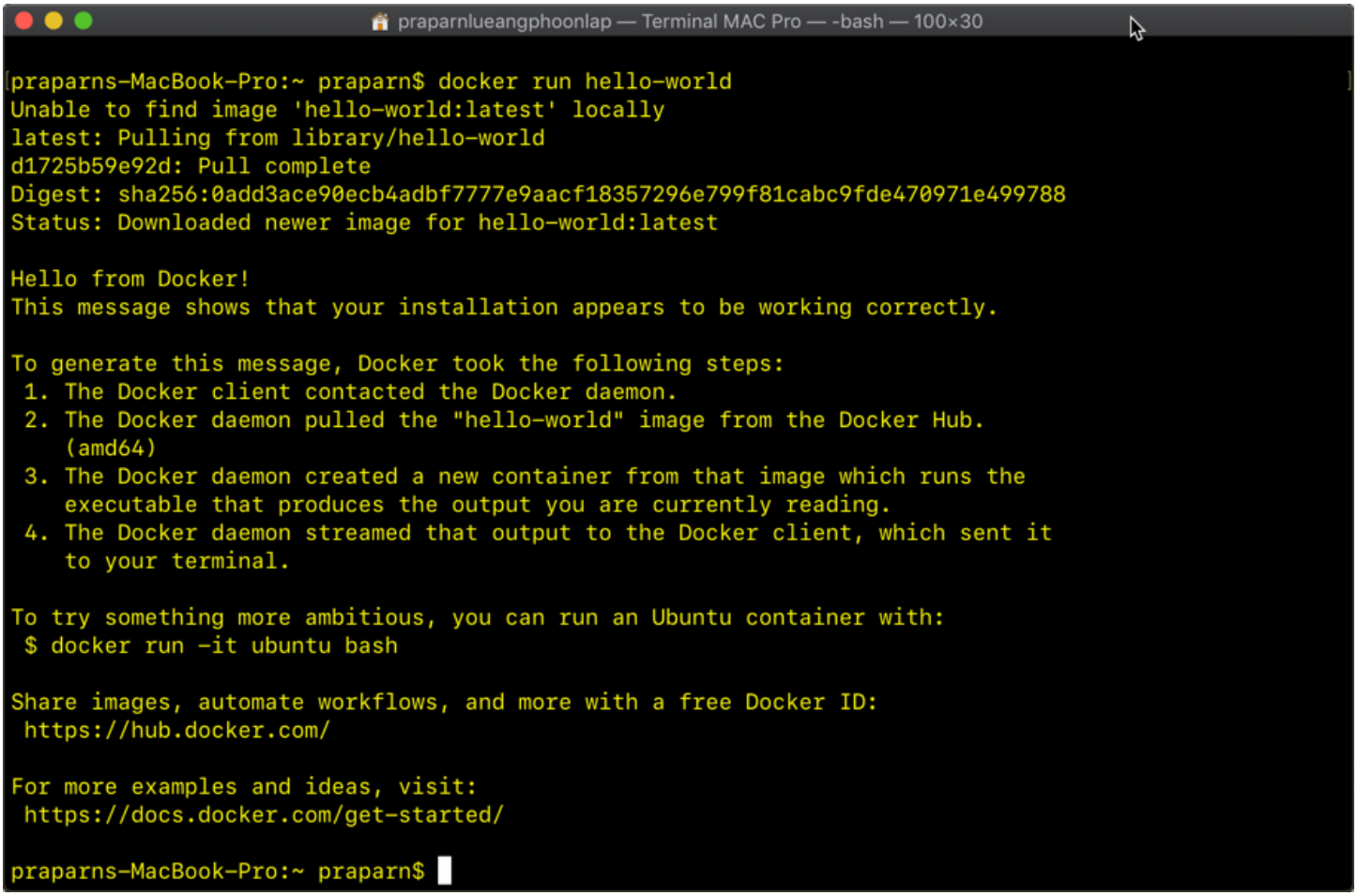




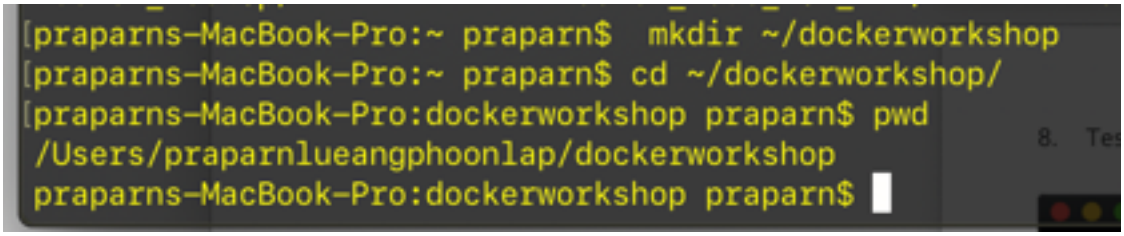
7. Check version of docker by command: docker version (On terminal)



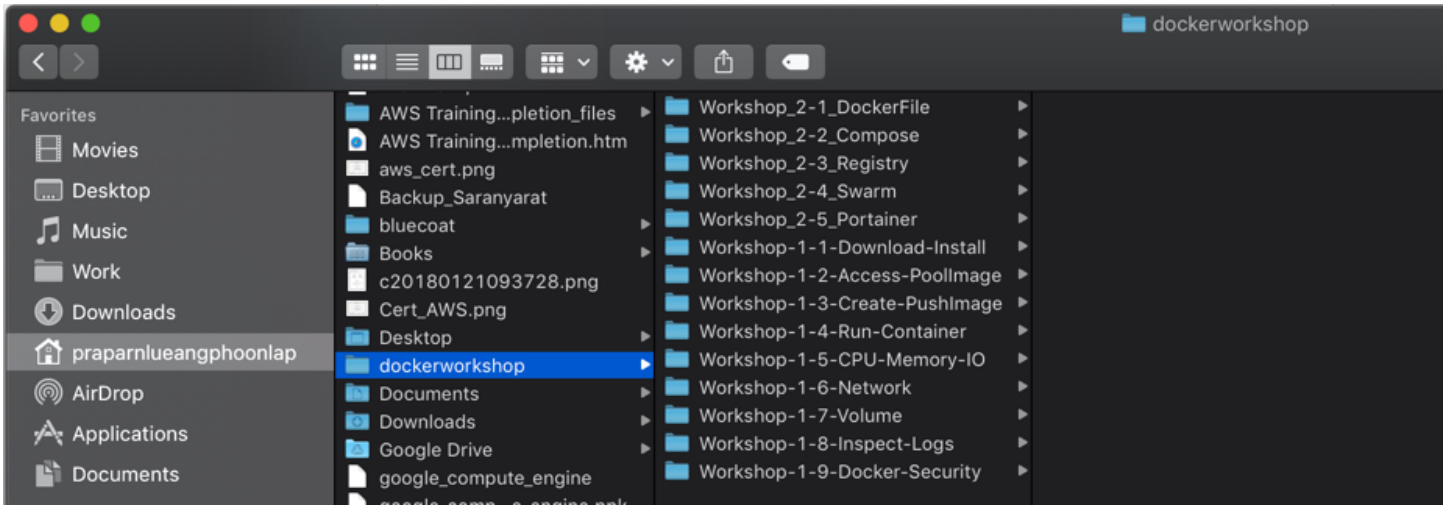
8. Test run first container by command: docker run hello-world on terminal



9. Make directory "dockerworkshop" on /Users/<your username> by command: mkdir ~/dockerworkshop

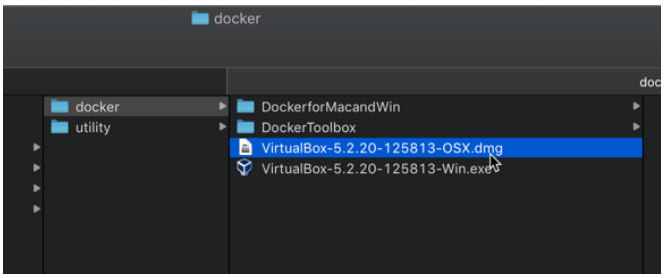


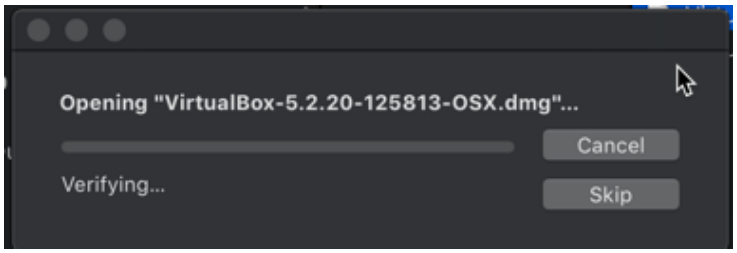
10. Copy all workshop folder to this folder



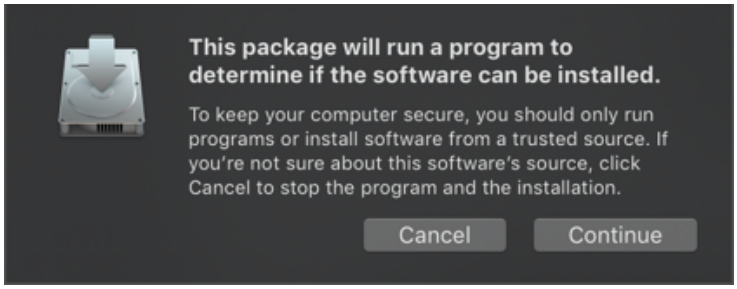
Install Oracle VirtualBox (Optional)

1. Double click "DockerToolbox.pkg

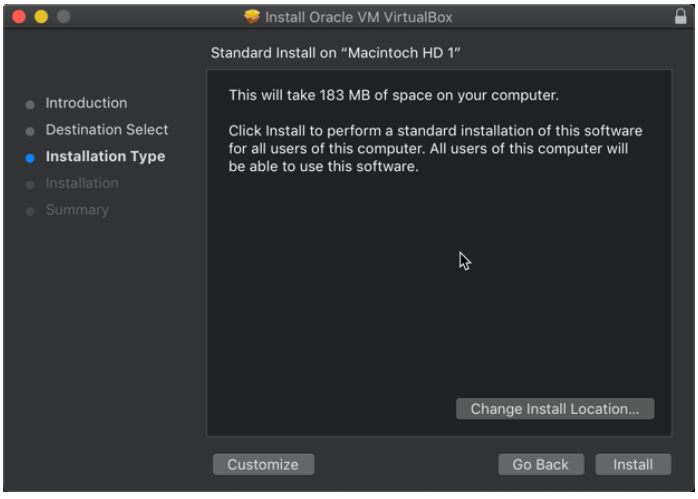


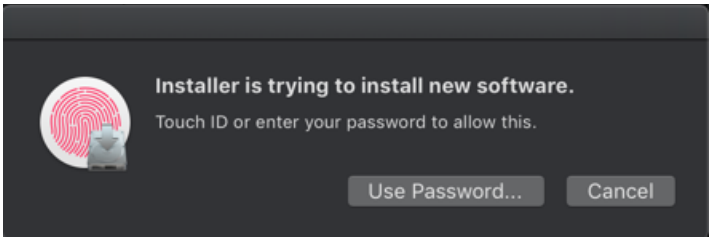


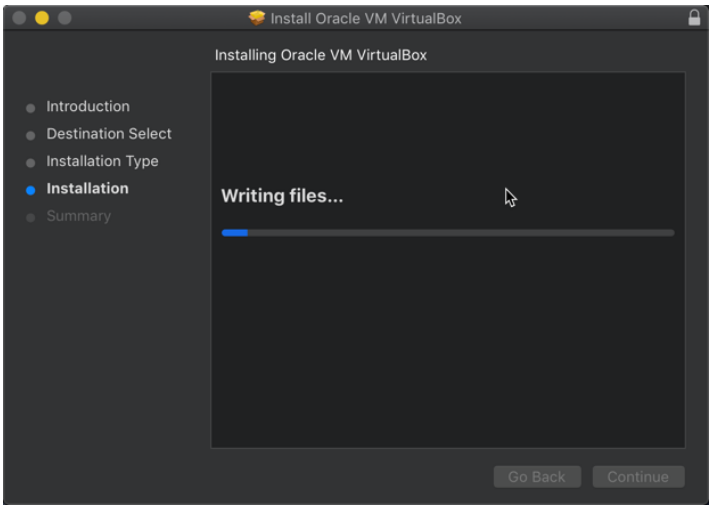
2. Double click icon "No.1" for Setup and Click "Continue"

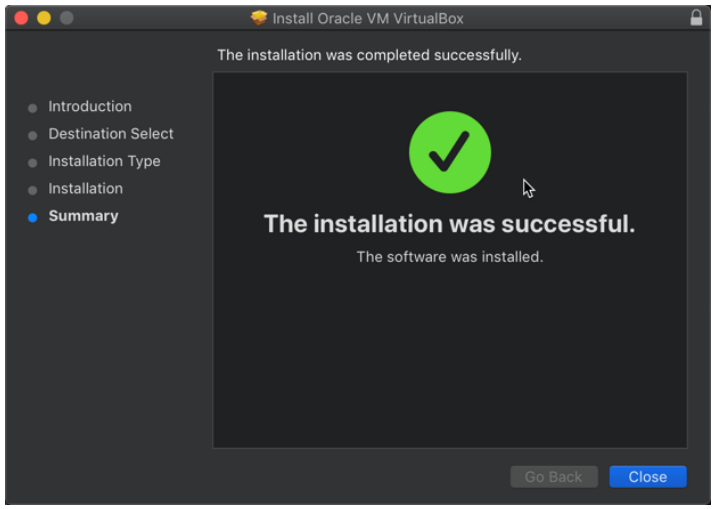


3. Following screen for setup



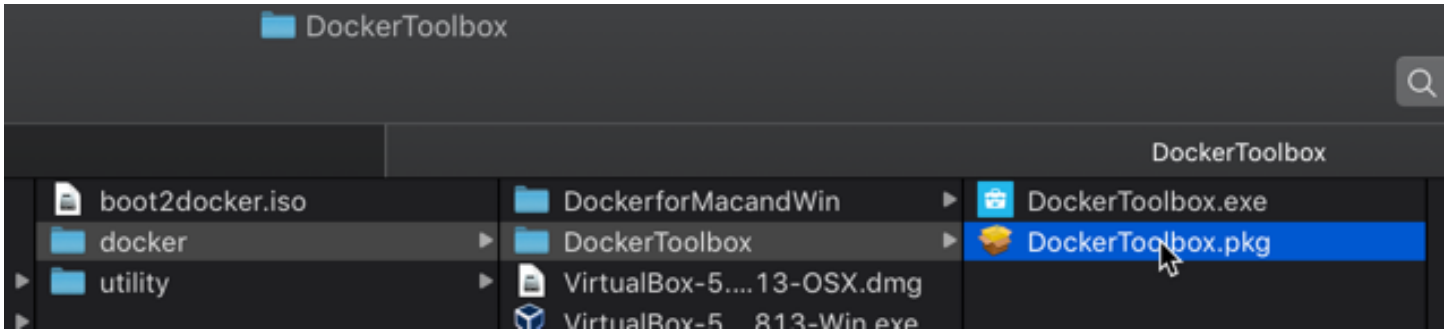

Installer is trying to install new software.
Touch ID or enter your password to allow this.
Use Password...
Cancel


Install Oracle VM VirtualBox
Installing Oracle VM VirtualBox
Introduction
Destination Select
Installation Type
Installation
Summary
Writing files...
Go Back
Continue


Install Oracle VM VirtualBox
The installation was completed successfully.
Introduction
Destination Select
Installation Type
Installation
Summary
The installation was successful.
The software was installed.
Go Back
Close

Create docker-machine (Optional)

1. Install dockertoolbox as usual



2. Create docker-machine call "labdocker" by command: "docker-machine create --driver=virtualbox labdocker

```
praparns-MacBook-Pro:nodejs praparn$ docker-machine create --driver=virtualbox labdocker
Running pre-create checks...
(labdocker) No default Boot2Docker ISO found locally, downloading the latest release...
(labdocker) Latest release for github.com/boot2docker/boot2docker is v18.06.1-ce
(labdocker) Downloading /Users/praparnlueangphoonlap/.docker/machine/cache/boot2docker.iso from https://github.com/boot2
docker/boot2docker/releases/download/v18.06.1-ce/boot2docker.iso...
(labdocker) 0%....10%....20%....30%....40%....50%....60%....70%....80%....90%....100%
Creating machine...
(labdocker) Copying /Users/praparnlueangphoonlap/.docker/machine/cache/boot2docker.iso to /Users/praparnlueangphoonlap/.
docker/machine/machines/labdocker/boot2docker.iso...
(labdocker) Creating VirtualBox VM...
(labdocker) Creating SSH key...
(labdocker) Starting the VM...
(labdocker) Check network to re-create if needed...
(labdocker) Waiting for an IP...
Waiting for machine to be running, this may take a few minutes...
Detecting operating system of created instance...
Waiting for SSH to be available...
Detecting the provisioner...
Provisioning with boot2docker...
Copying certs to the local machine directory...
Copying certs to the remote machine...
Setting Docker configuration on the remote daemon...
Checking connection to Docker...
Docker is up and running!
To see how to connect your Docker Client to the Docker Engine running on this virtual machine, run: docker-machine env l
abdocker
praparns-MacBook-Pro:nodejs praparn$
```

3. Check status of docker-machine by command: docker-machine ls

```
praparns-MacBook-Pro:nodejs praparn$ docker-machine ls
NAME        ACTIVE   DRIVER       STATE     URL                         SWARM   DOCKER        ERRORS
labdocker   -        virtualbox   Running   tcp://192.168.99.100:2376           v18.06.1-ce
praparns-MacBook-Pro:nodejs praparn$
```

4. Access to docker-machine by command: docker-machine ssh labdocker and Test run docker image by command: docker run hello-world

```
praparns-MacBook-Pro:nodejs praparn$ docker-machine ssh labdocker
                        ##         .
                  ## ## ##        ==
               ## ## ## ## ##    ===
           /"""""""""""""""""\___/ ===
      ~~~ {~~ ~~~~ ~~~ ~~~~ ~~~ ~ /  ===- ~~~
           \______ o           __/
             \    \         __/
              \____\_______/
 _                 _   ____     _            _
| |__   ___   ___ | |_|___ \ __| | ___   ___| | _____ _ __
| '_ \ / _ \ / _ \| __| __) / _` |/ _ \ / __| |/ / _ \ '__|
| |_) | (_) | (_) | |_ / __/ (_| | (_) | (__|   <  __/ |
|_.__/ \___/ \___/ \__|_____\__,_|\___/ \___|_|\_\___|_|
Boot2Docker version 18.06.1-ce, build HEAD : c7e5c3e - Wed Aug 22 16:27:42 UTC 2018
Docker version 18.06.1-ce, build e68fc7a
docker@labdocker:~$ docker run hello-world
Unable to find image 'hello-world:latest' locally
latest: Pulling from library/hello-world
d1725b59e92d: Pull complete
Digest: sha256:0add3ace90ecb4adbf7777e9aacf18357296e799f81cabc9fde470971e499788
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/

docker@labdocker:~$
```