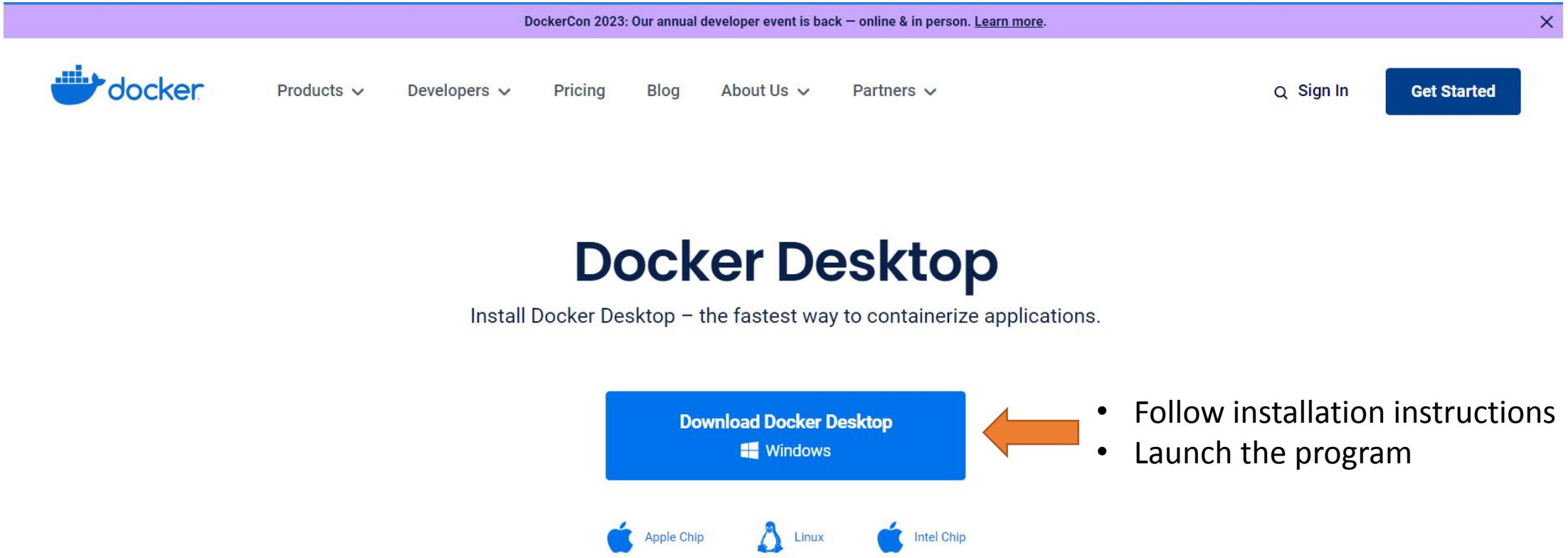
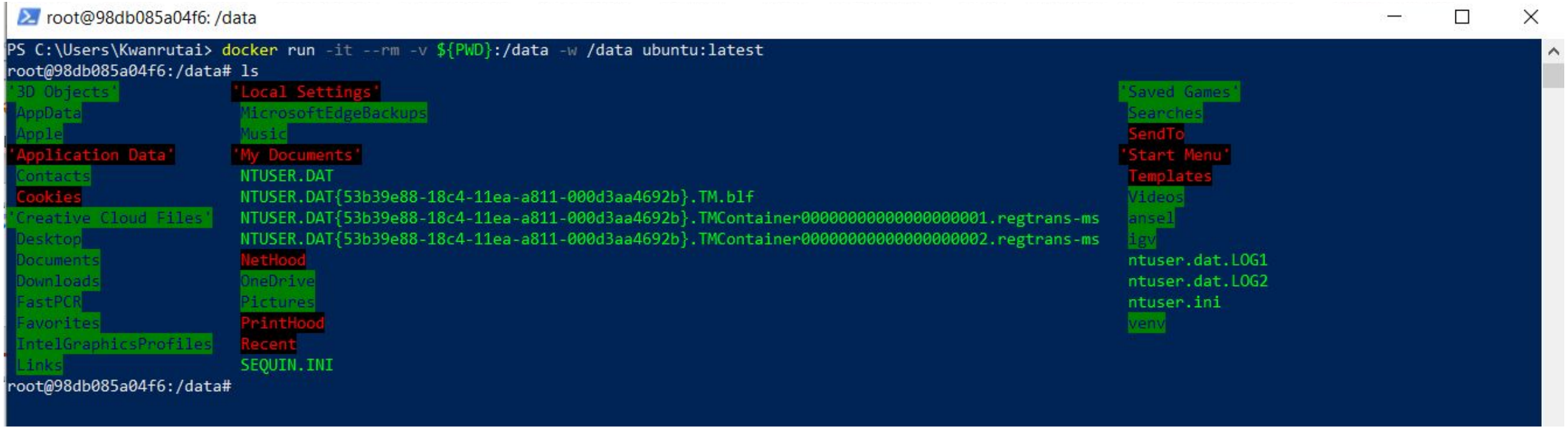
Run Docker on Windows



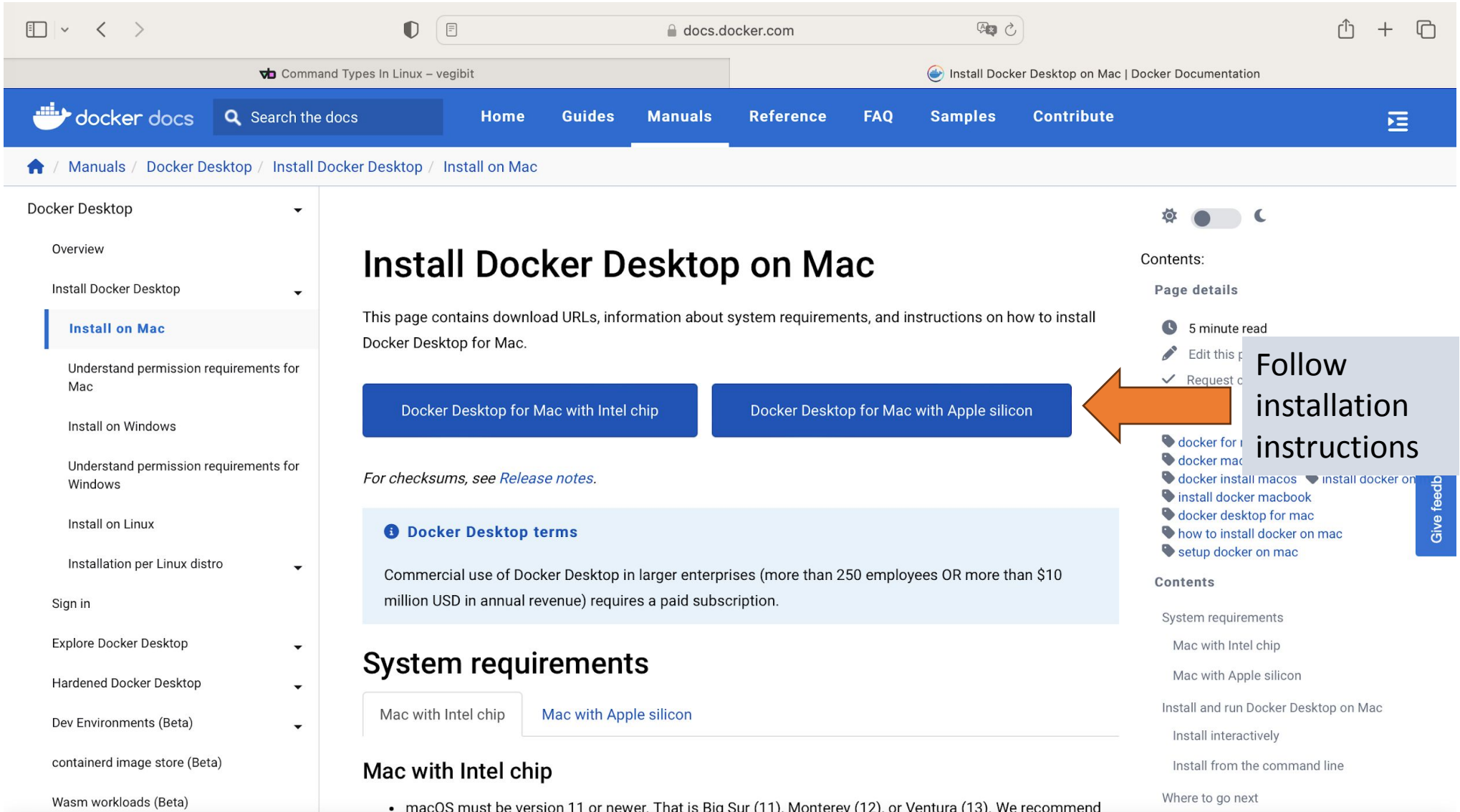
- Follow installation instructions
- Launch the program

Run Docker on Windows

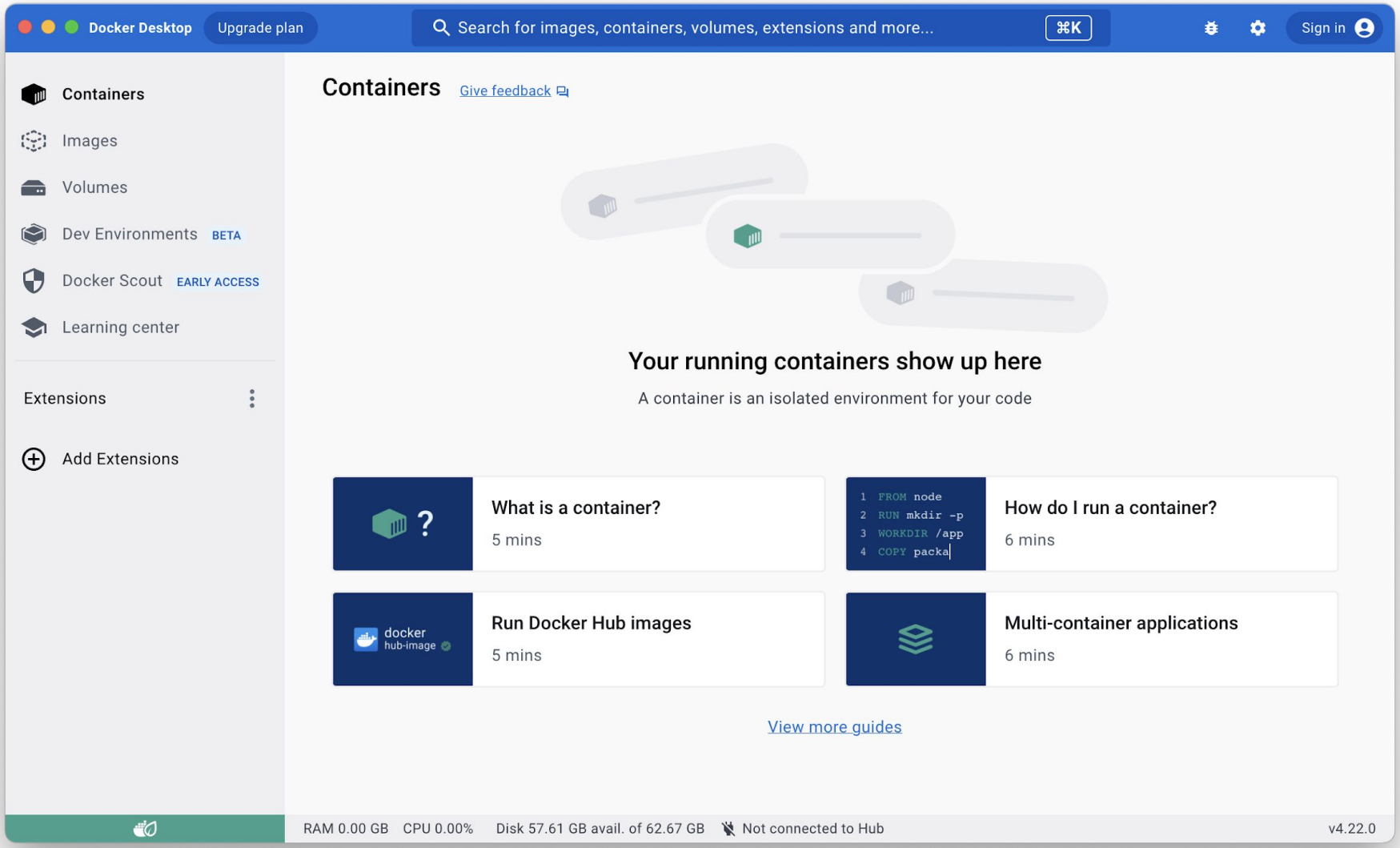
- Run in Windows PowerShell (Recommended)



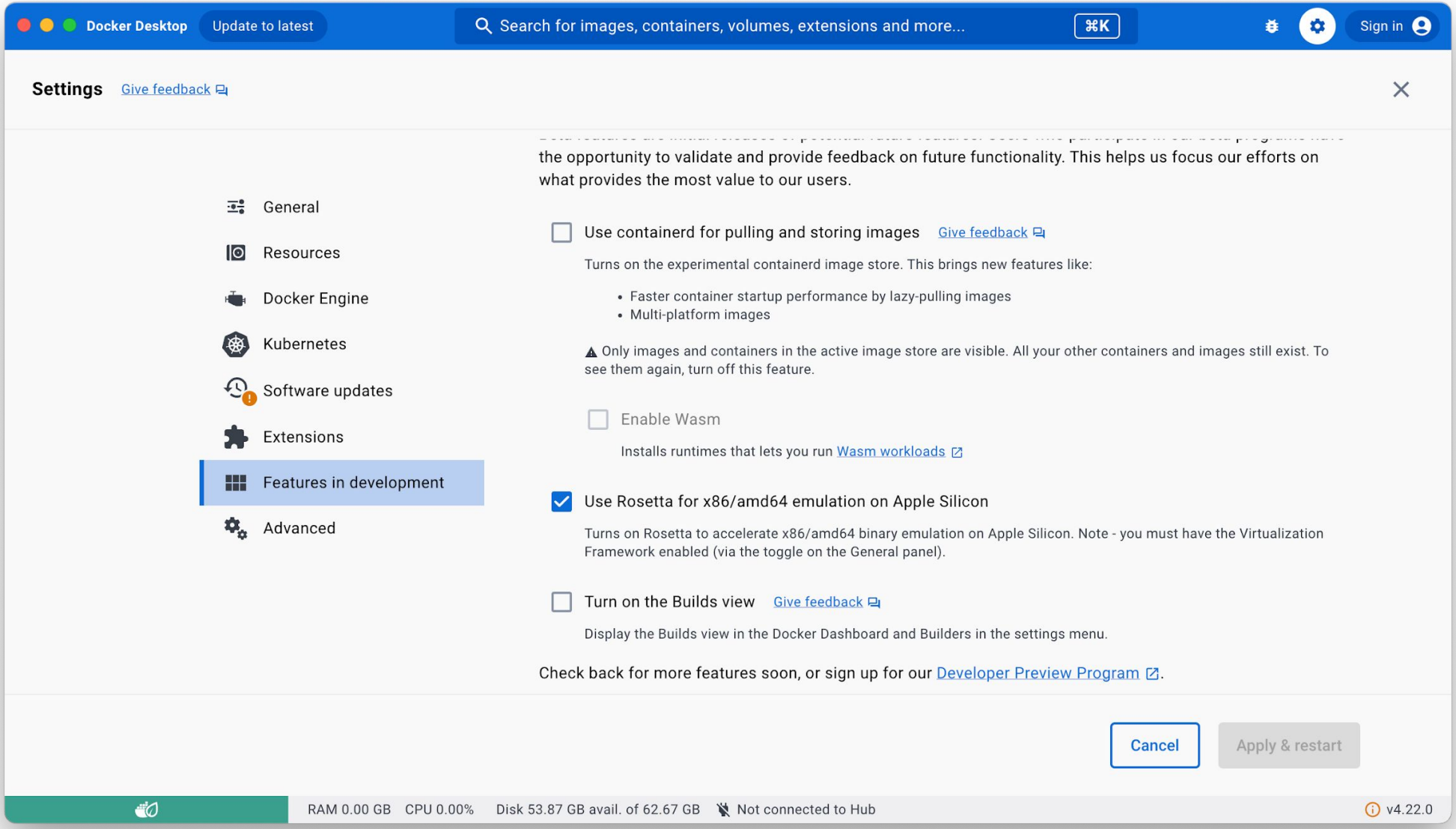
Install Docker Desktop for MacOS



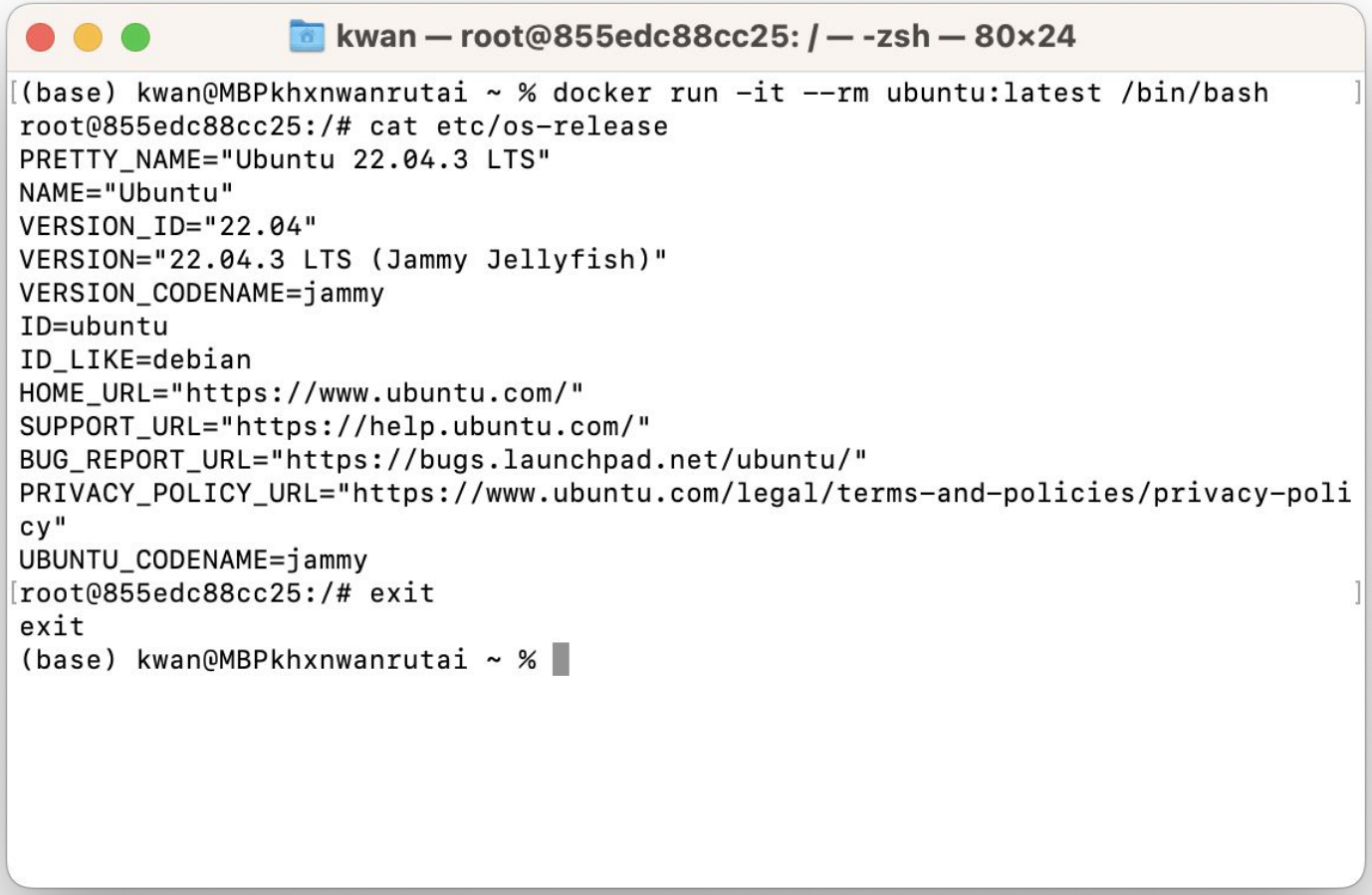
Launch Docker Desktop



For Mac with Apple Silicon



Run Docker in Terminal



Basic Docker commands

- `docker run -it --rm ubuntu:latest /bin/bash`
  - `docker run` → Create and run a new container from an image
  - `-it` → Interactive session
  - `--rm` → Automatically remove the container when it exits
  - `ubuntu:latest` → Image name and tag (i.e. version)
  - `/bin/bash` → Command to run within the container (/bin/bash = command prompt. Could be omitted.)

Basic Docker commands

- `docker run -it --rm -v /my/local/path:/data -w /data ubuntu:latest /bin/bash`
  - `-v /my/local/path:/data` → Mount /my/local/path to /data in the container
  - `-w` → Set /data of the container as the working directory

Basic Docker commands

- `docker ps` → Check all running instances
- `docker ps -a` → Check all running and exited instances
- `docker rm [CONTAINER ID]` → Remove an instance
- `docker images` → Check locally available images
- `docker rmi [IMAGE ID]` → Remove an image

Basic Docker commands

- `docker run -it -v /my/local/path:/data -w /data ubuntu:latest` → `exit` → Exited container (see instance ID with docker ps –a)
- `docker start -a -i [instance ID]` → Start and access an exited container (-a = attach to access the container; -i = interactive)

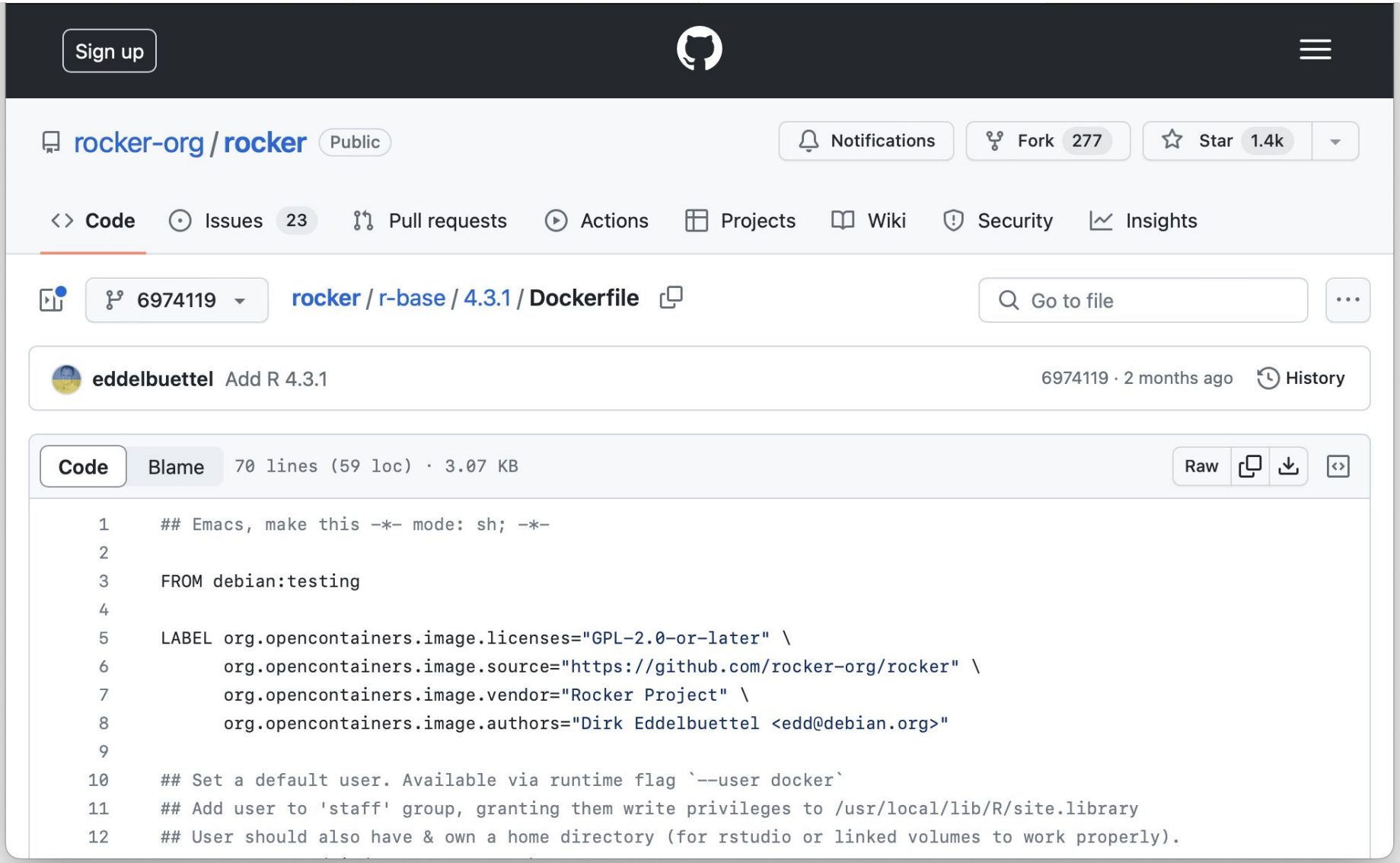
For MacOS with Apple Silicon (M1,M2,M3,...)

Images may not be available for ARM64 CPU.

Use '--platform linux/amd64' to run AMD64 images via an emulator (i.e., Rosetta)

- `docker run --rm -it --platform linux/amd64 ubuntu:latest`

Dockerfile


Sign up
rocker-org / rocker
Public
Notifications
Fork 277
Star 1.4k
Code
Issues 23
Pull requests
Actions
Projects
Wiki
Security
Insights
6974119
rocker / r-base / 4.3.1 / Dockerfile
Go to file
eddelbuettel Add R 4.3.1
6974119 · 2 months ago
History
Code
Blame
70 lines (59 loc) · 3.07 KB
Raw
Emacs, make this -*- mode: sh; -*-
FROM debian:testing
LABEL org.opencontainers.image.licenses="GPL-2.0-or-later" \
org.opencontainers.image.source="https://github.com/rocker-org/rocker" \
org.opencontainers.image.vendor="Rocker Project" \
org.opencontainers.image.authors="Dirk Eddelbuettel <edd@debian.org>"
Set a default user. Available via runtime flag `--user docker`
Add user to 'staff' group, granting them write privileges to /usr/local/lib/R/site.library
User should also have & own a home directory (for rstudio or linked volumes to work properly).

Simple docker file Test building

```
$ cd /your/path/
$ nano Dockerfile

# Simple Ubuntu container with htop and nano
installed
FROM ubuntu:latest
LABEL AUTHOR=dumrong.mai@mahidol.ac.th
RUN apt-get update && apt-get install -y
--no-install-recommends \
   nano \
   htop
```

Build image from a Dockerfile

- `cd /path/to/Dockerfile`
- `docker build -t image_name:tag .` (Do not forget '.')

For building AMD64 images on ARM64 CPU

- `docker build --platform linux/amd64 -t image_name:tag .` (Do not forget '.')