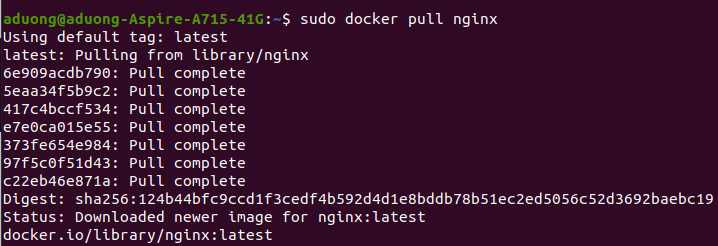
1 docker –version

kiểm tra phiên bản của docker đang được cài đặt trên máy

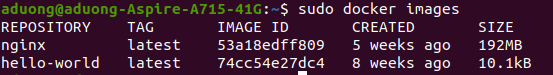
2 docker run hello-world

chạy hello-world có sẳn ở Docker

3 docker pull nginx

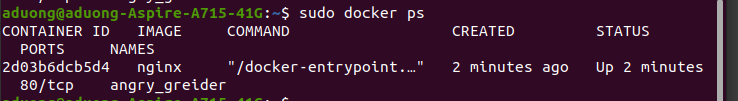
Pull về 1 image nginx từ DockerHub nginx

4 docker images

kiểm tra các image có trong máy local

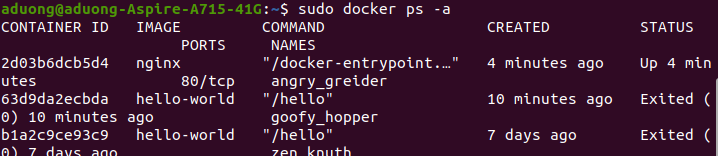
5 docker run -d nginx

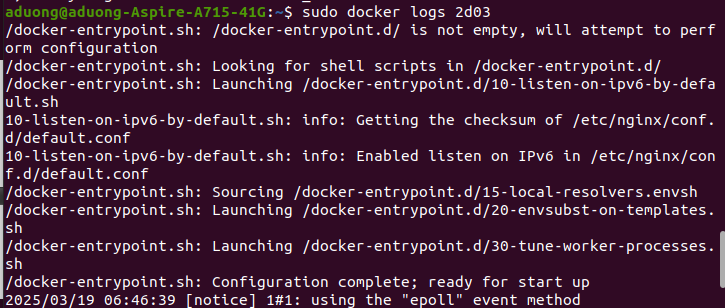
chạy image nginx dưới nền => chạy ngầm  
docker trả về containerID 2d03b6dcb5d4003fcf5c24f03a188f3449f5e0ad8a5c85dfb0c0f330a0083aea

6 docker ps

kiểm tra các container đang chạy = docker container ls

7 docker ps -a

lấy ra tất cả các container có trong máy local cả đang hoạt động hoặc không

8 docker logs <container\_id>  
xem logs của 1 container được chỉ định

9 docker exec -it <container\_id> /bin/sh

mở 1 shell để tương tác bên trong container được chỉ định

muốn out ra thì gõ exit

10 docker stop <container\_id>

stop container đang chạy

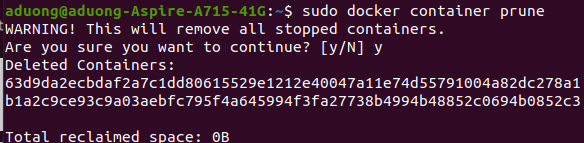
11 docker restart <container\_id>

restart lại container đang stop

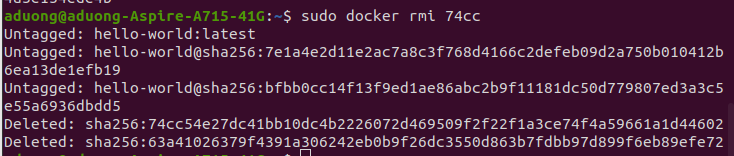
12 docker rm <container\_id>

xóa 1 container, lưu ý muốn xóa thì phải stop trước

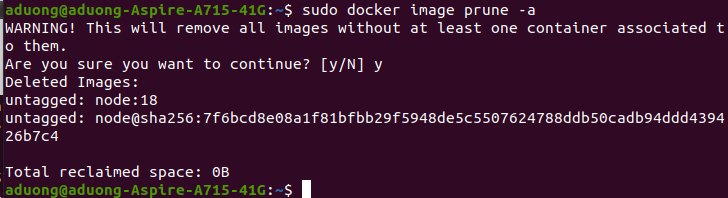
13 docker container prune

xóa toàn bộ container có trong hệ thống

14 docker rmi <image\_id>

xóa image theo

15 docker image prune -a

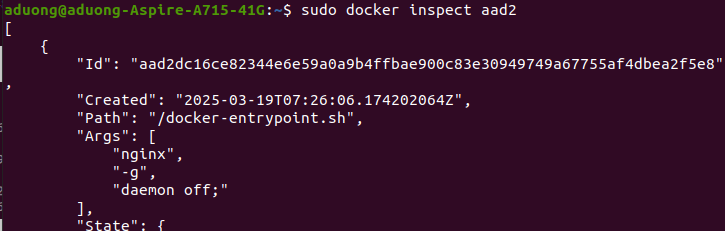
xóa tất cả image trong hệ thống

16 docker run -d -p 8080:80 nginx

chạy ngầm container nginx qua cổng 8080 port 80

có thể ở browser để kiểm tra http://localhost:8080/

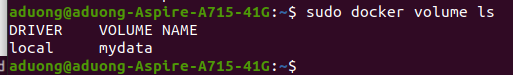
17 docker inspect <container\_id>

trả về các thông số của container dưới dạng JSON

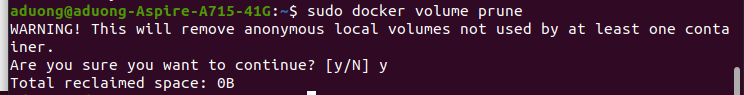
18 docker run -d -v mydata:/data nginx

chạy container nginx với 1 volumn chỉ định ở đây là mydata -> nếu chưa có mydata thì docker sẽ tự tạo mới

19 docker volume ls

trả về danh sách volume có trong hệ thống

20 docker volume prune

xóa toàn bộ các volome trong local

21 docker run -d --name my\_nginx nginx

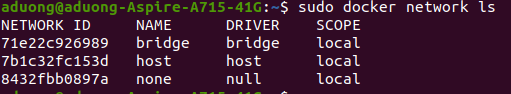
tạo và chạy một container nginx ở chế độ nền với một tên cụ thể

22 docker stats

ống

hiển thị thông tin về tài nguyên (CPU, RAM, Network, I/O) mà các container đang sử dụng theo thời gian thực

23 docker network ls



trả về danh sách các mang (network) của hệ thống Docker

24 docker network create my\_network

tạo 1 mạng mới trong hệ thống docker có tên my\_network

25 docker run -d --network my\_network --name my\_container nginx



chạy ngầm container nginx có trên my\_container gắn mạng my\_network vừa tạo cho container này

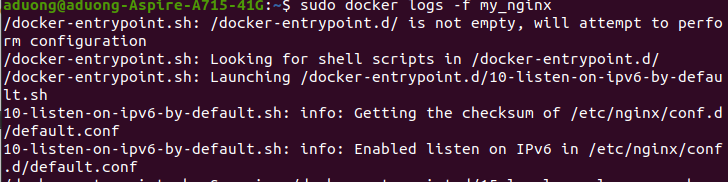
26 docker network connect my\_network my\_nginx

kết nối container my\_nginx vào mạng my\_network, cho phép nó giao tiếp với các container khác trong cùng network

27 docker run -d -e MY\_ENV=hello\_world nginx

chạy một container Nginx ở chế độ nền (-d) và thiết lập biến môi trường (-e MY\_ENV=hello\_world) bên trong container

28 docker logs -f my\_nginx

hiển thị log (nhật ký) của container my\_nginx theo thời gian thực.

29 FROM nginx

COPY index.html /usr/share/nginx/html/index.html

+ Dùng nginx làm base image.

+ Chép file index.html từ thư mục hiện tại vào thư mục chứa tài nguyên web của Nginx (/usr/share/nginx/html/index.html).

30 docker build -t my\_nginx\_image .

build image có tên my\_nginx\_ Dockerfile từ thư mục hiện tại

31 docker run -d -p 8080:80 my\_nginx\_image

chạy image my\_nginx\_image dưới nền qua cổng 8080 port 80