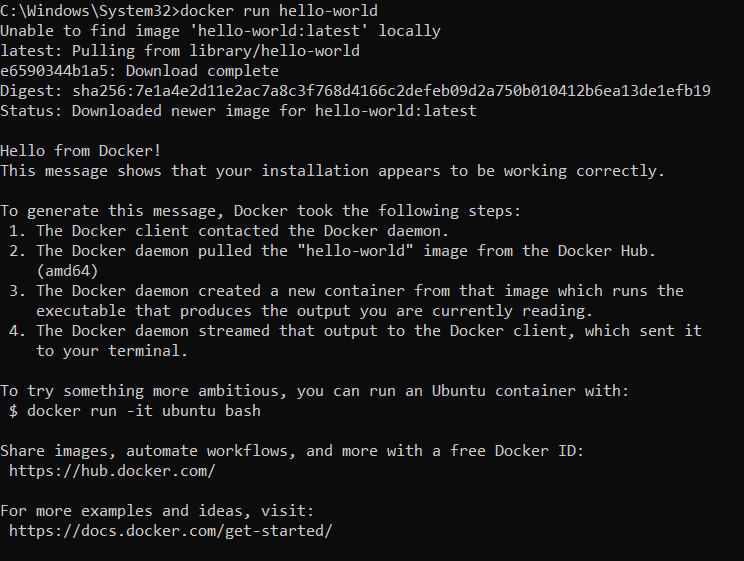
Phần 1: Các lệnh cơ bản thao tác với Docker

1 docker --version



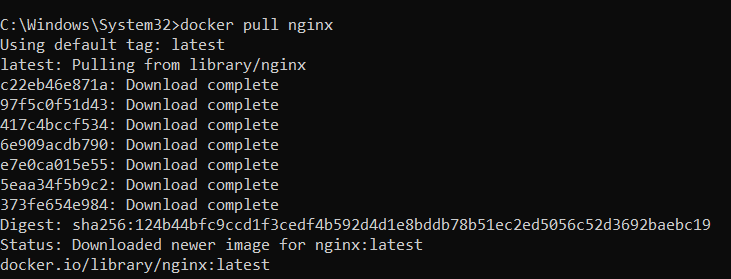
* Xem phiên bản hiện tại của docker

2 docker run hello-world



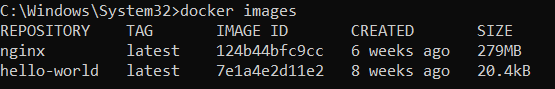
* Tải image **hello-world** về máy trong trường hợp máy chưa có

3 docker pull nginx



* Tải image **nginx** về máy nếu chưa có, nếu đã có thì sẽ kiểm tra phiên bản cập nhật

4 docker images



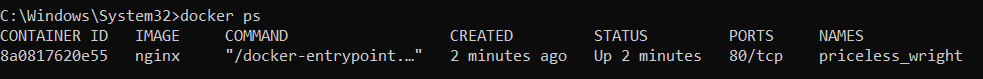
* Liệt kê các images đã được tải về máy

5 docker run -d nginx



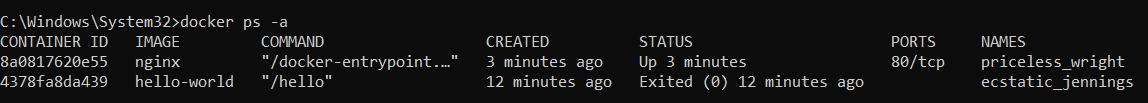
* Tạo một container từ image nginx và chạy trong chế độ nền và hiển thị containerID

6 docker ps



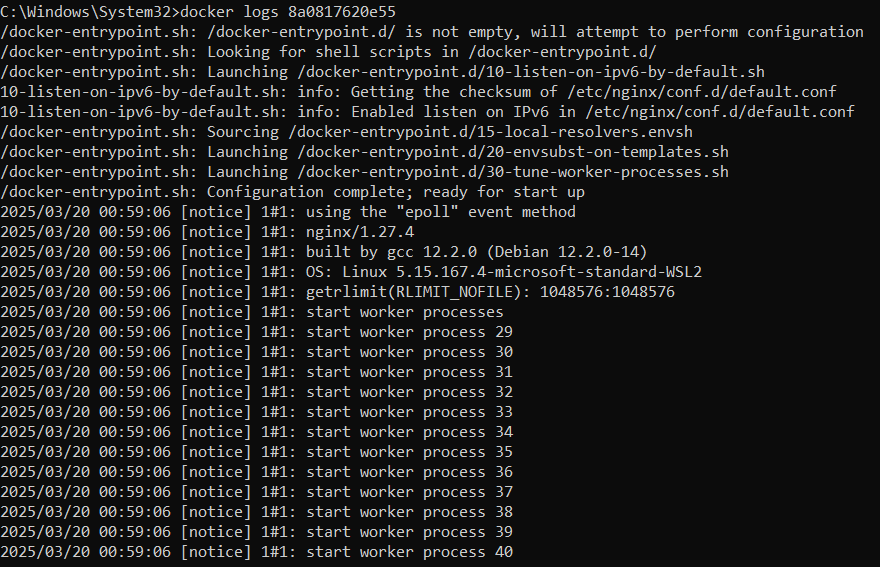
* Hiển thị danh sách các container đang chạy

7 docker ps -a



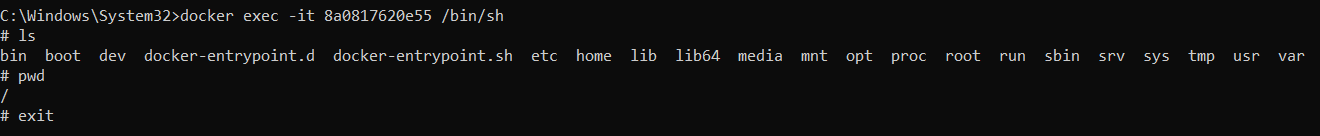
* Hiển thị danh sách tất cả các container, bao gồm cả container đang chạy(Up) và container đã dừng(Exited)

8 docker logs <container\_id>



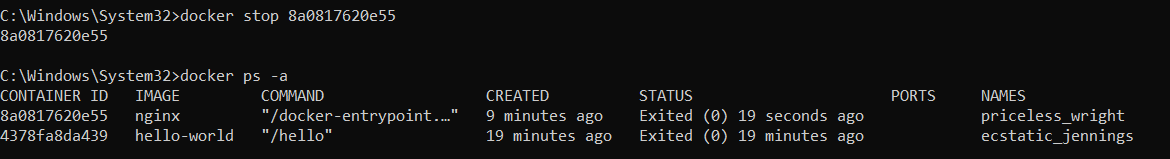
* Hiển thị nhật ký (log) của 1 container - 8a0817620e55 nginx

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



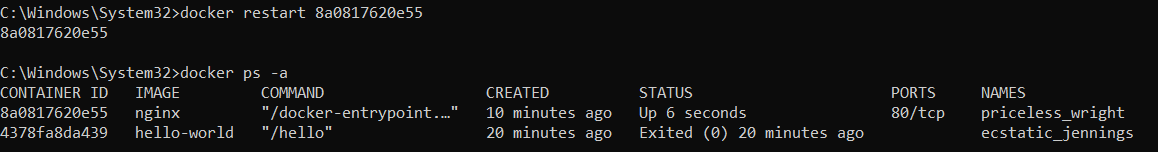
* Truy cập vào container và thực hiện các lệnh trong container

10 docker stop <container\_id>



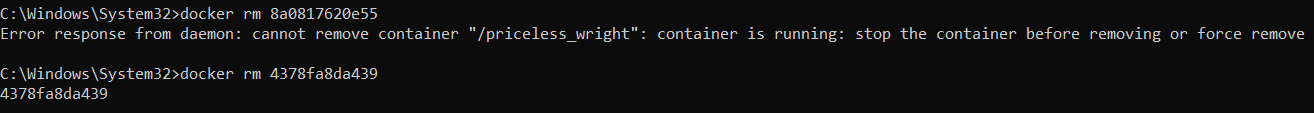
* Dừng một container đang chạy

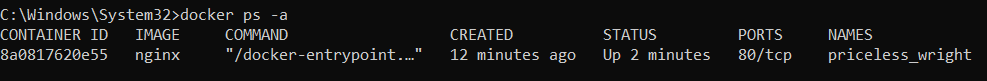
11 docker restart <container\_id>



* Khởi động lại container đã dừng

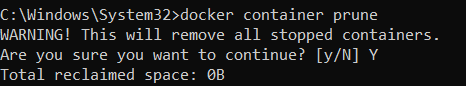
12 docker rm <container\_id>





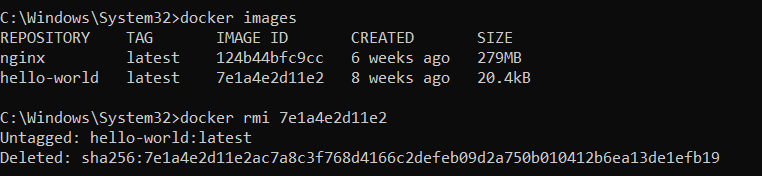
* Xóa 1 container đã dừng. Không thể xóa container khi đang chạy

13 docker container prune



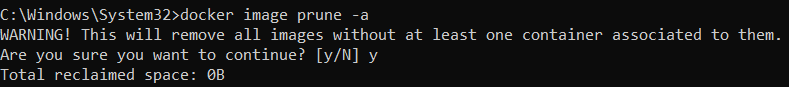
* Xóa hết tất cả container đã dừng

14 docker rmi <image\_id>



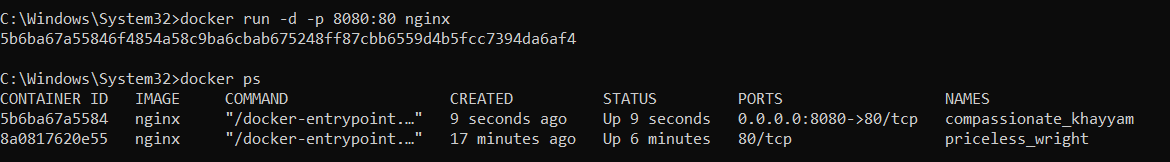
* Xóa 1 ảnh trong docker theo imageID

15 docker image prune -a

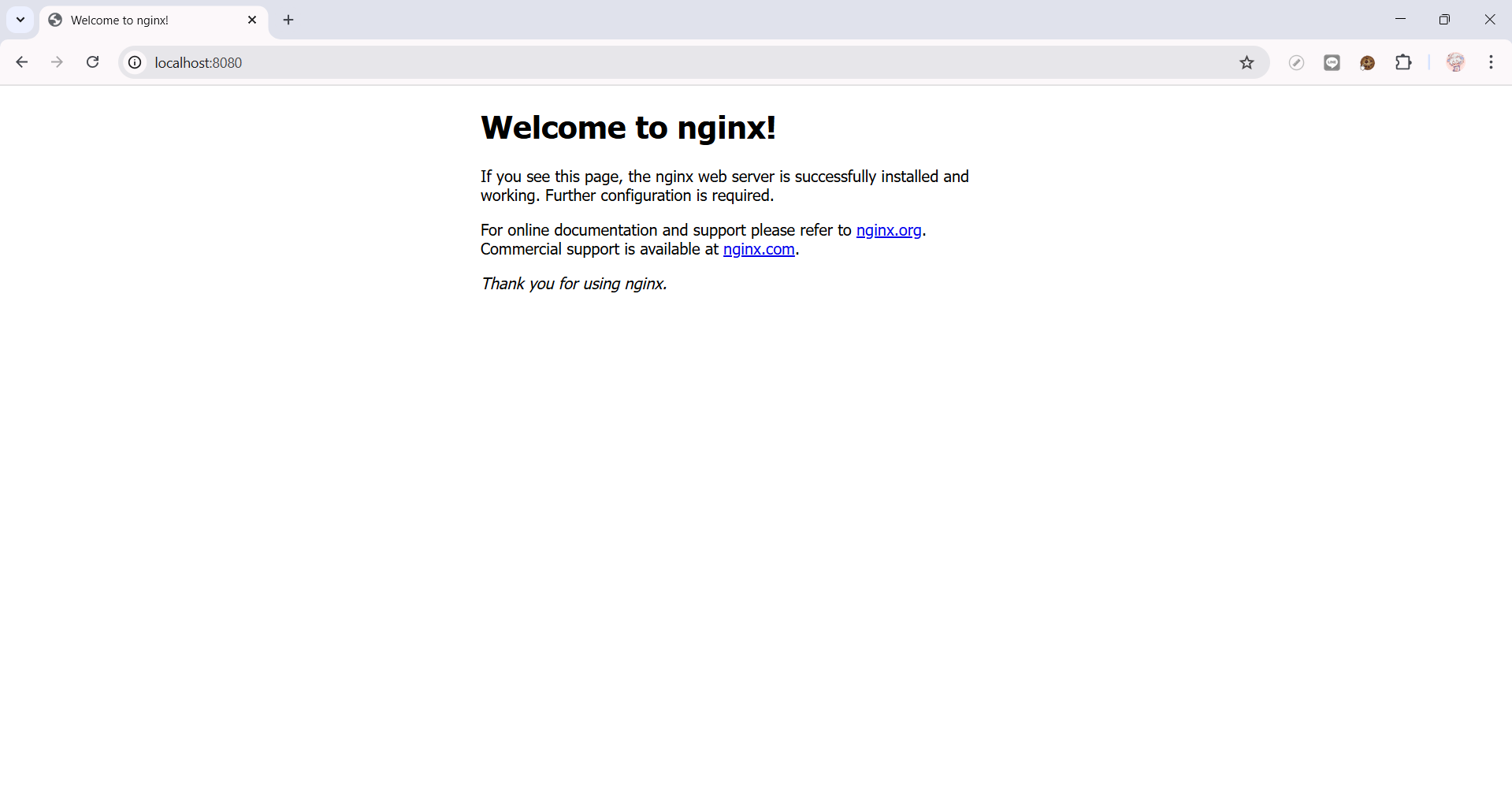


* Xóa những ảnh không sử dụng khỏi docker

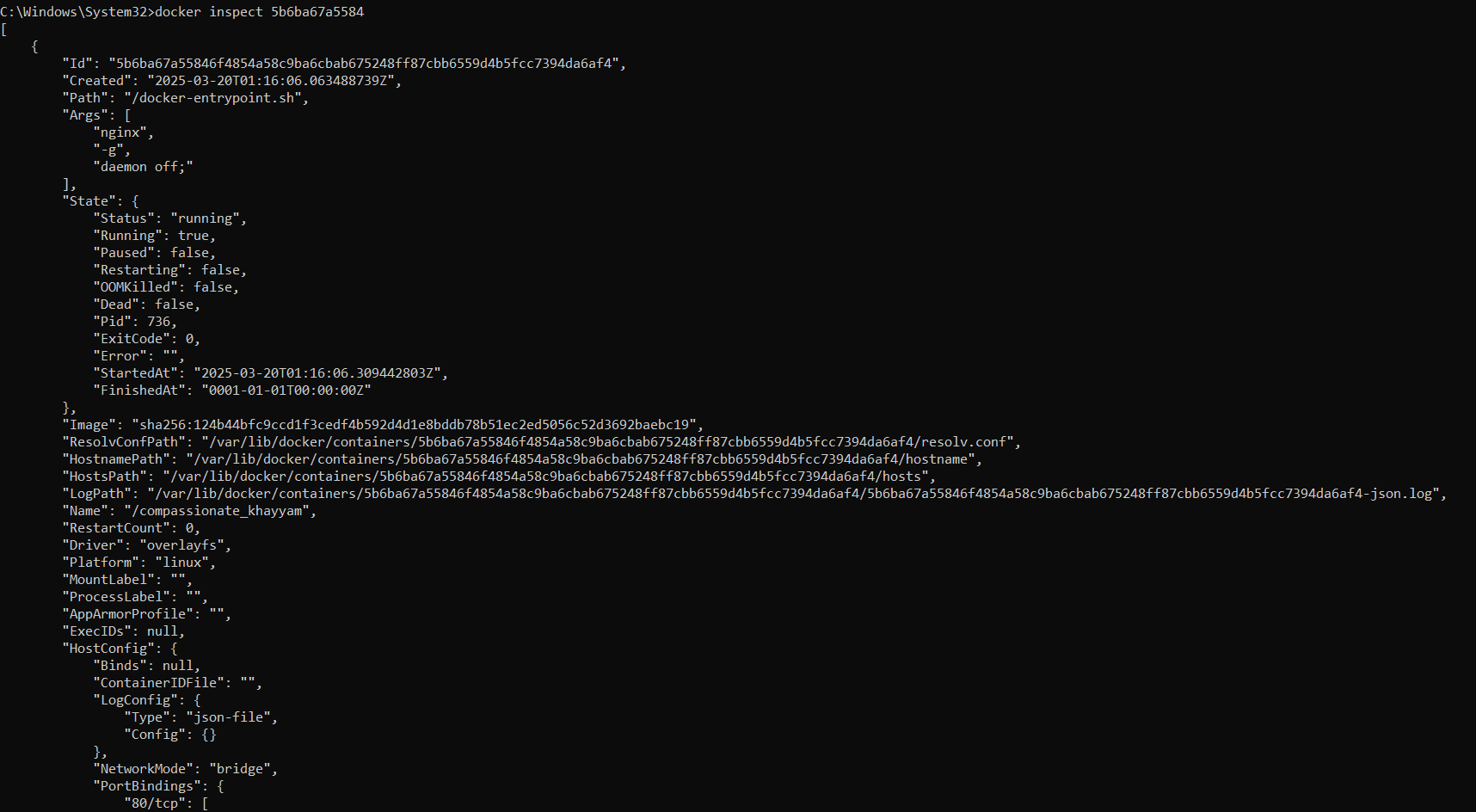
16 docker run -d -p 8080:80 nginx



* Chạy 1 container Nginx trong chế độ nền và ánh xạ vào cổng 8080 trên máy tính và cổng 80 của container

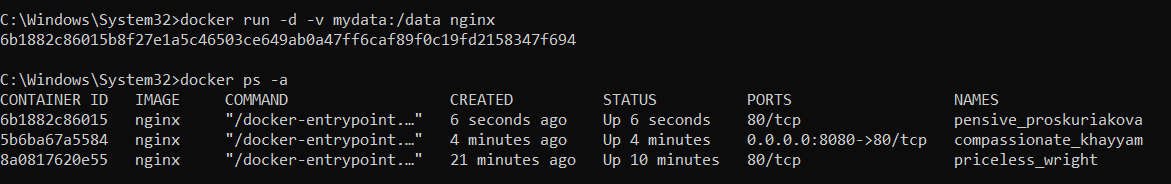


17 docker inspect <container\_id>



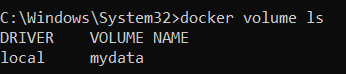
* Hiển thị thông tin của container dưới dạng JSON

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



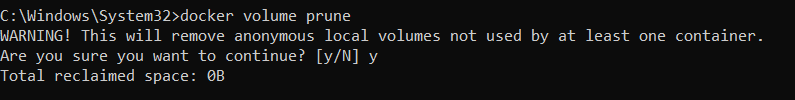
* Chạy một container Nginx trong chế độ nền và gắn (mount) volume có tên mydata vào thư mục /data trong container

19 docker volume ls



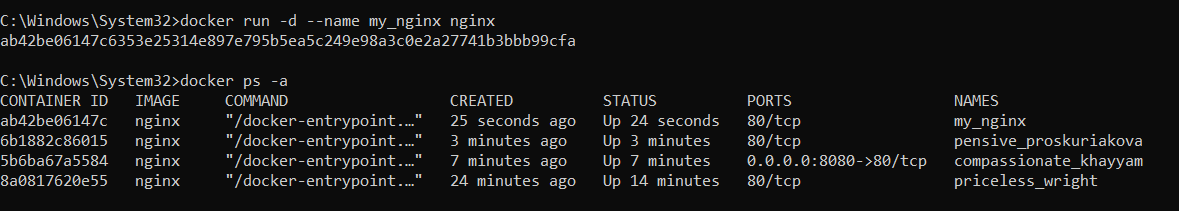
* Hiển thị danh sách volume

20 docker volume prune



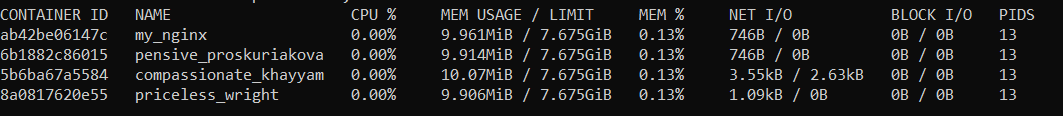
* Xóa các volume không sử dụng

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



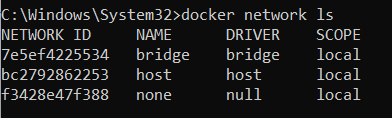
* Tạo và chạy 1 container đặt tên là my\_nginx

22 docker stats



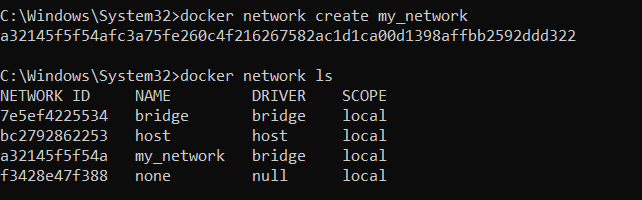
* Hiển thị thông tin hiệu suất và tài nguyên mà các container đang sử dụng

23 docker network ls



* Hiển thị danh sách các network

24 docker network create my\_network



* Tạo 1 network với tên my\_network

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



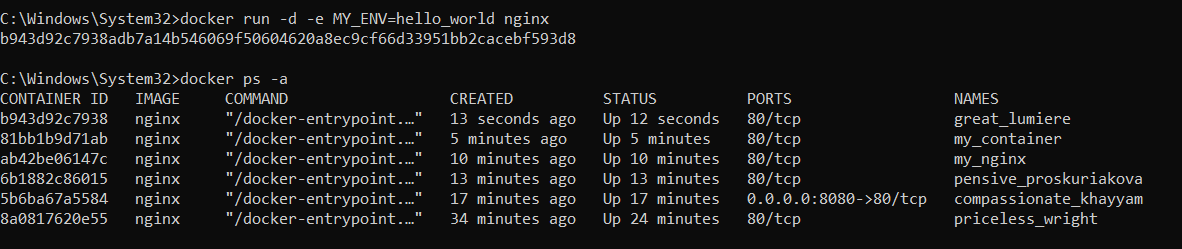
* Tạo và chạy một container Nginx được kết nối vào mạng my\_network

26 docker network connect my\_network my\_nginx



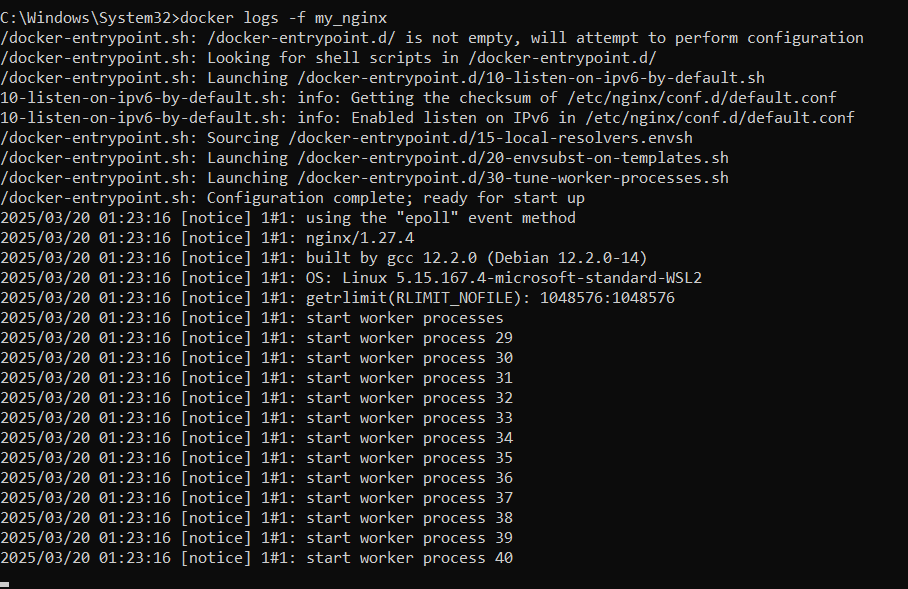
* Kết nối container my\_nginx vào mạng Docker có tên my\_network

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



* Khởi chạy một container Nginx trong chế độ nền và thiết lập biến môi trường MY\_ENV với giá trị "hello\_world"

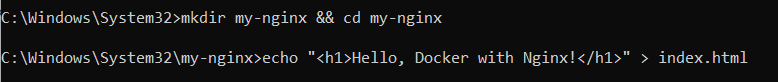
28 docker logs -f my\_nginx



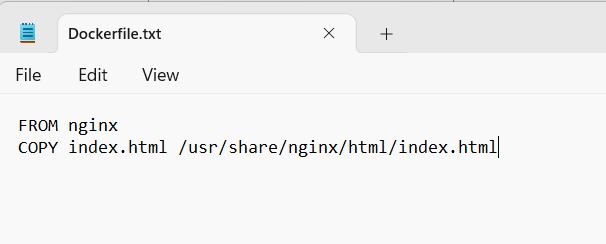
* Hiển thị nhật ký của container my\_nginx

29 FROM nginx

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

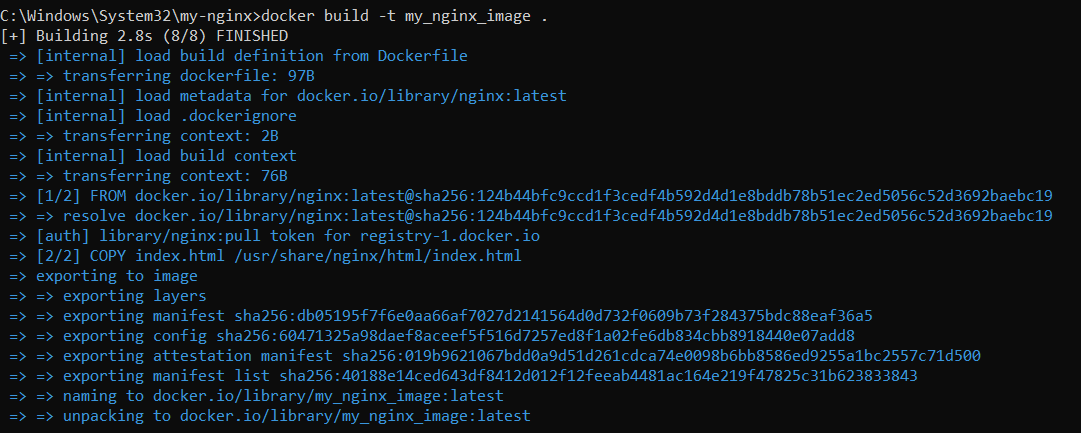






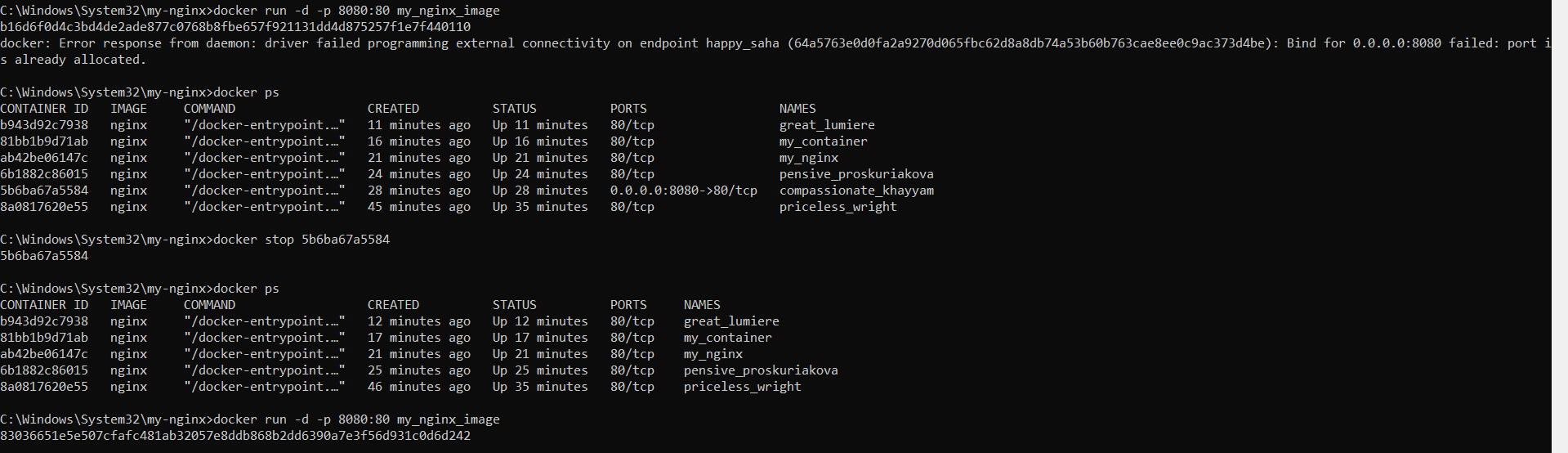
* Tạo các file index, dockerfile

30 docker build -t my\_nginx\_image .



* Tạo image **my\_nginx\_image** chứa file index.html

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



* Dừng 1 container đang dung cổng 8080
* Khởi chạy một container từ image my\_nginx\_image và ánh xạ trên cổng 8080 của máy host và 80 của container

