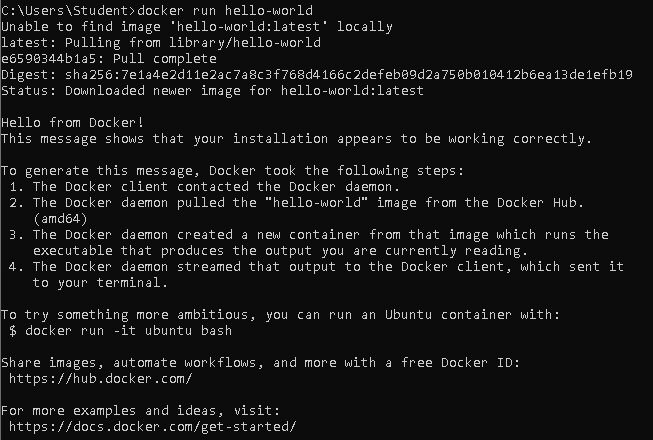
# 1. Kiểm tra phiên bản Docker

docker --version



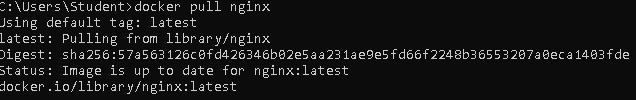
# 2. Chạy container kiểm tra

docker run hello-world



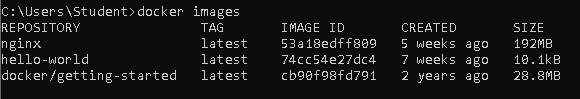
# 3. Tải image nginx từ Docker Hub

docker pull nginx



# 4. Liệt kê các image đã tải

docker images



# 5. Chạy container nginx ở chế độ nền

docker run -d nginx



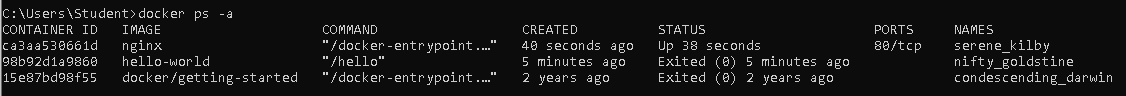
# 6. Xem danh sách container đang chạy

docker ps



# 7. Xem tất cả container (đã chạy hoặc dừng)

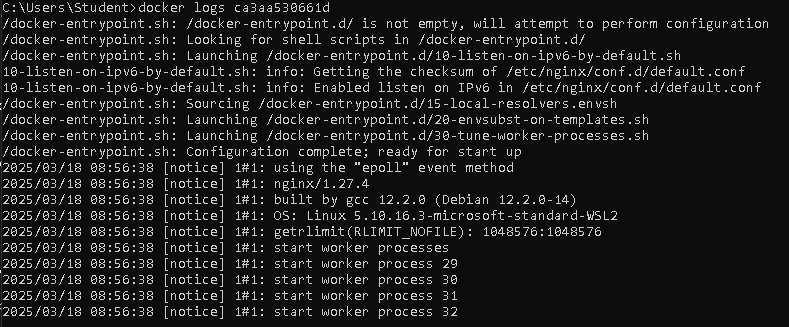
docker ps -a



# 8. Xem logs của container

docker logs <container\_id>

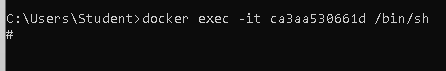
docker logs ca3aa530661d



# 9. Truy cập shell trong container

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

docker exec -it ca3aa530661d /bin/sh



# 10. Dừng container

docker stop <container\_id>

docker stop ca3aa530661d



# 11. Khởi động lại container

docker restart <container\_id>

docker restart ca3aa530661d



# 12. Xóa container

docker rm <container\_id>

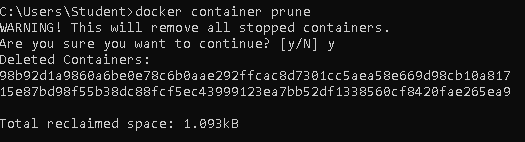
docker stop ca3aa530661d

docker rm ca3aa530661d



# 13. Xóa tất cả container đã dừng

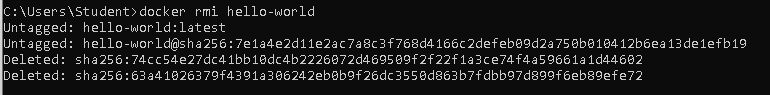
docker container prune



# 14. Xóa image theo ID

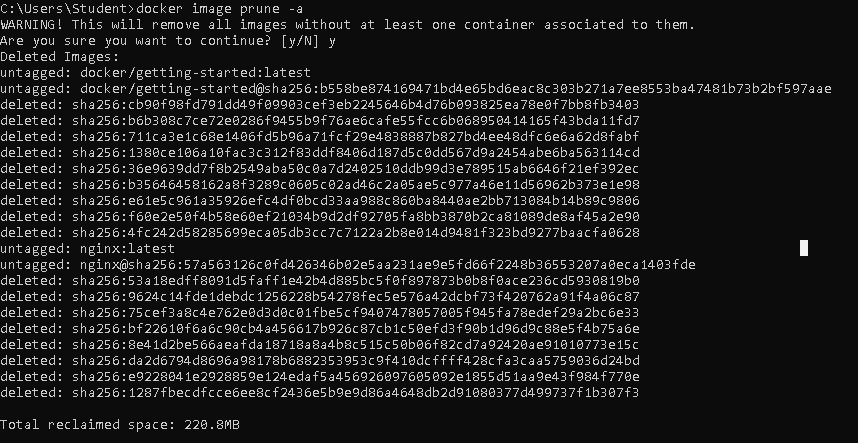
docker rmi <image\_id>

docker rmi hello-world



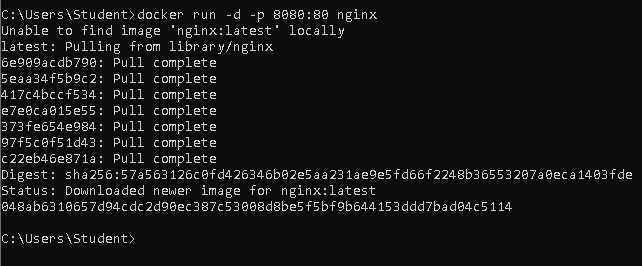
# 15. Xóa tất cả image không sử dụng

docker image prune -a



# 16. Chạy nginx và ánh xạ cổng 8080 -> 80

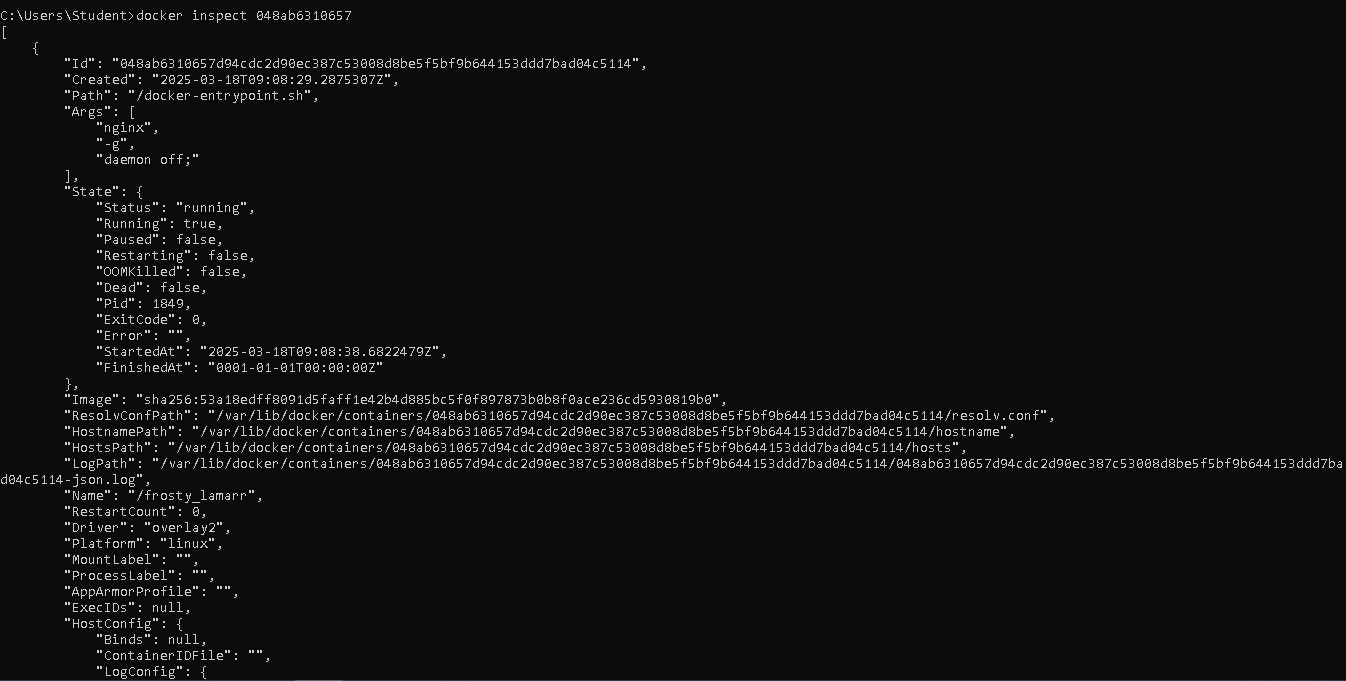
docker run -d -p 8080:80 nginx



# 17. Xem chi tiết container

docker inspect <container\_id>

docker inspect 048ab6310657



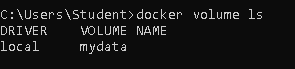
# 18. Chạy container với volume

docker run -d -v mydata:/data nginx



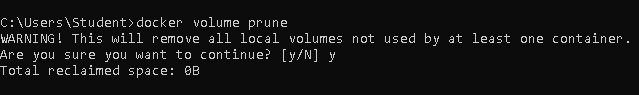
# 19. Liệt kê các volumes

docker volume ls



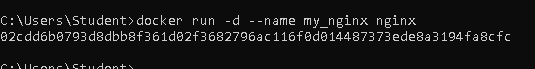
# 20. Xóa tất cả volumes không sử dụng

docker volume prune



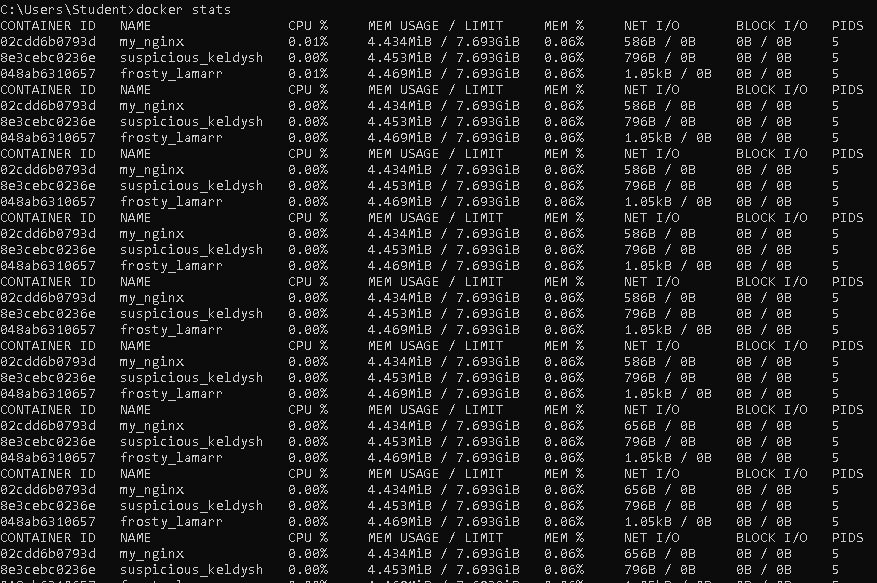
# 21. Chạy container và đặt tên

docker run -d --name my\_nginx nginx



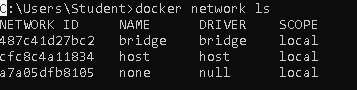
# 22. Theo dõi tài nguyên sử dụng

docker stats



# 23. Liệt kê các networks

docker network ls



# 24. Tạo network

docker network create my\_network



# 25. Chạy container trong network

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



# 26. Kết nối container vào network

docker network connect my\_network my\_nginx



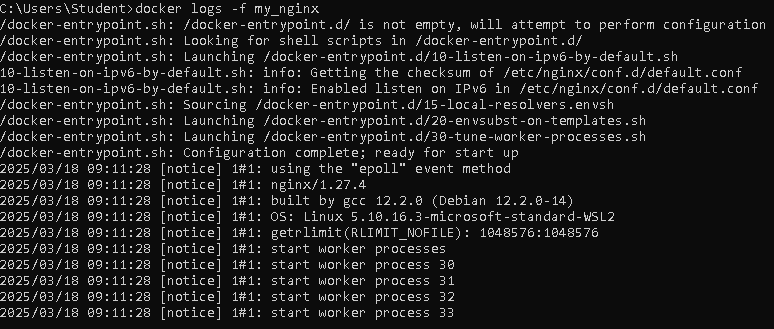
# 27. Chạy container với biến môi trường

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



# 28. Theo dõi logs realtime

docker logs -f my\_nginx

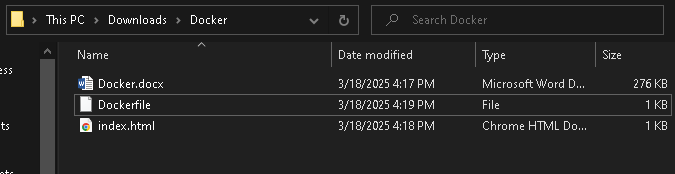


# 29. Tạo Dockerfile cho Nginx phục vụ trang web

# Dockerfile

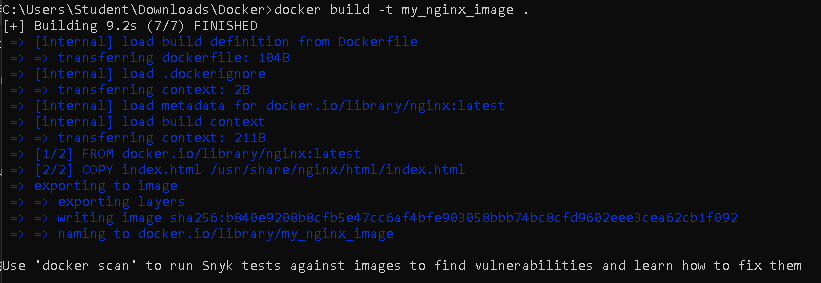
FROM nginx

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



# 30. Build image từ Dockerfile

docker build -t my\_nginx\_image .

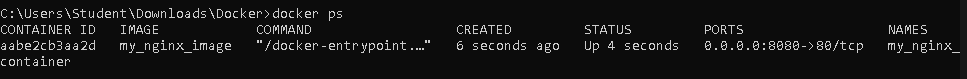


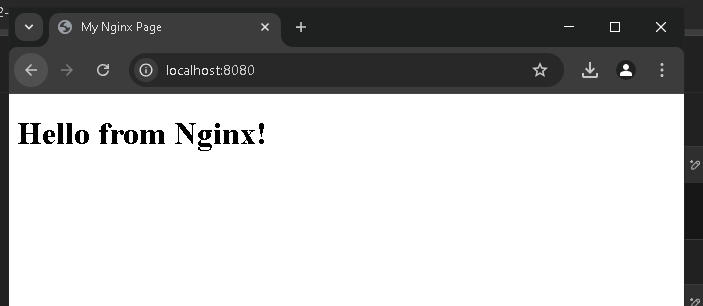
# 31. Chạy container từ image vừa build

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

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

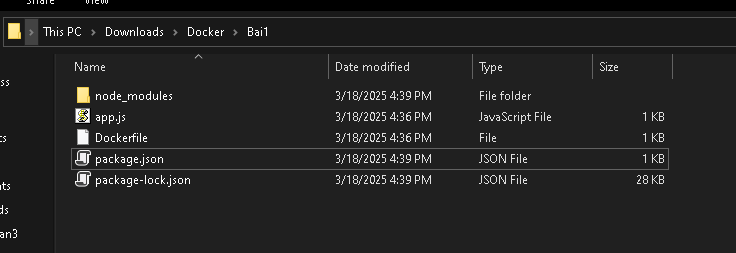


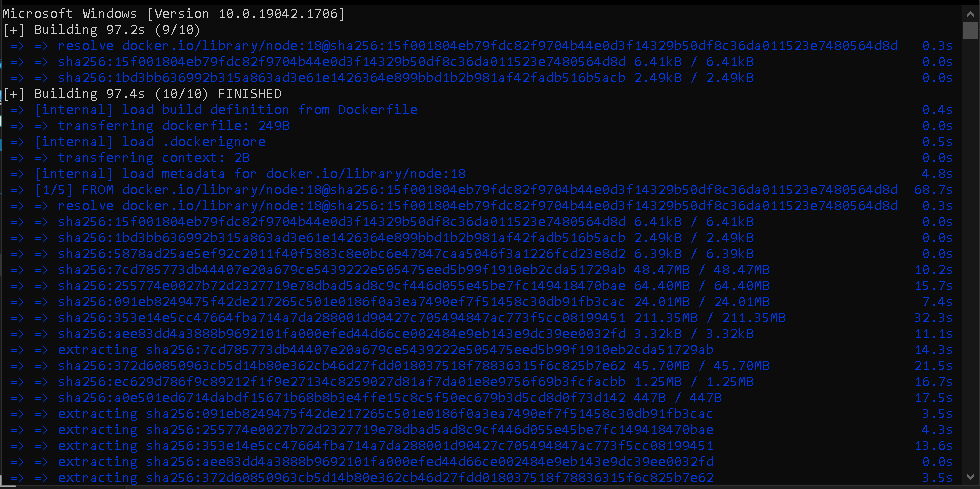


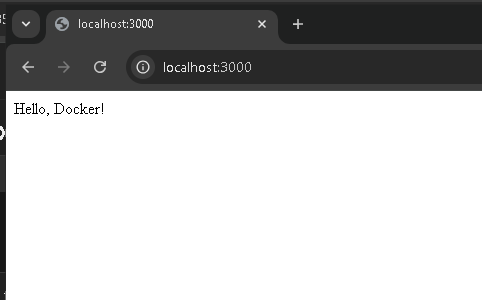


**Phần 2: Thao tác với Dockerfile**

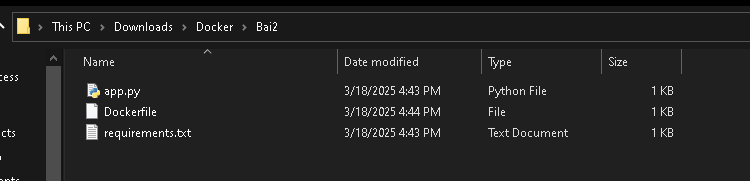
|  |  |  |  |
| --- | --- | --- | --- |
| Bài 1: Tạo Dockerfile chạy một ứng dụng Node.js đơn giản | | |  |
|  | ***Yêu cầu:*** |  |  |
|  | Viết Dockerfile để chạy một ứng dụng Node.js hiển thị "Hello, Docker!" trên cổng 3000. | | |
|  | Sử dụng node:18 làm base image. |  |  |



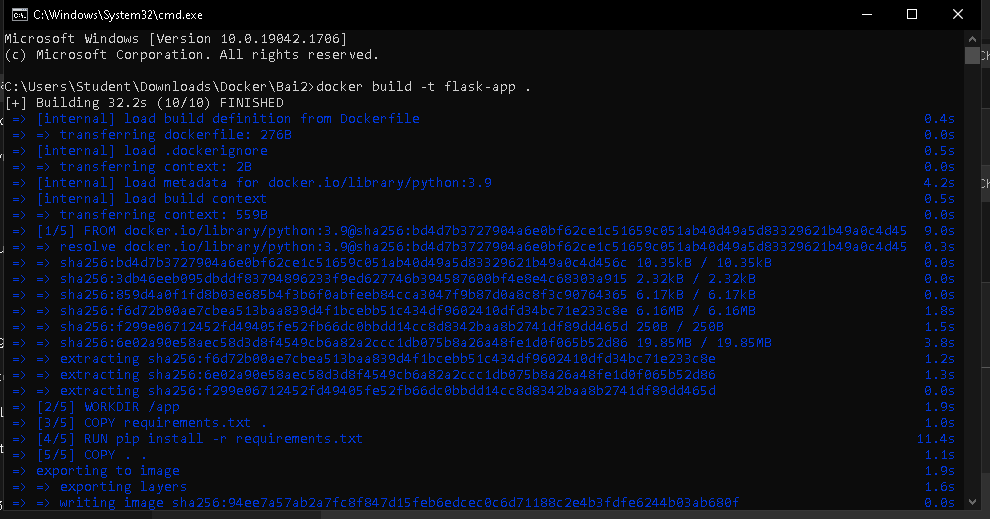




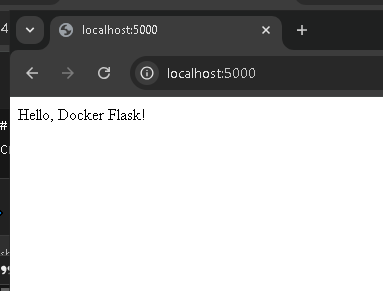
|  |  |  |  |
| --- | --- | --- | --- |
| Bài 2: Tạo Dockerfile chạy một ứng dụng Python Flask | | |  |
|  | ***Yêu cầu:*** |  |  |
|  | Viết Dockerfile để chạy một ứng dụng Flask hiển thị "Hello, Docker Flask!" trên cổng 5000. | | |
|  | Sử dụng python:3.9 làm base image. |  |  |



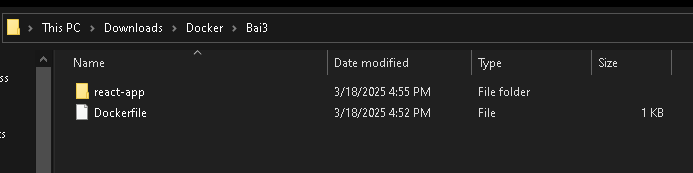
docker build -t flask-app .



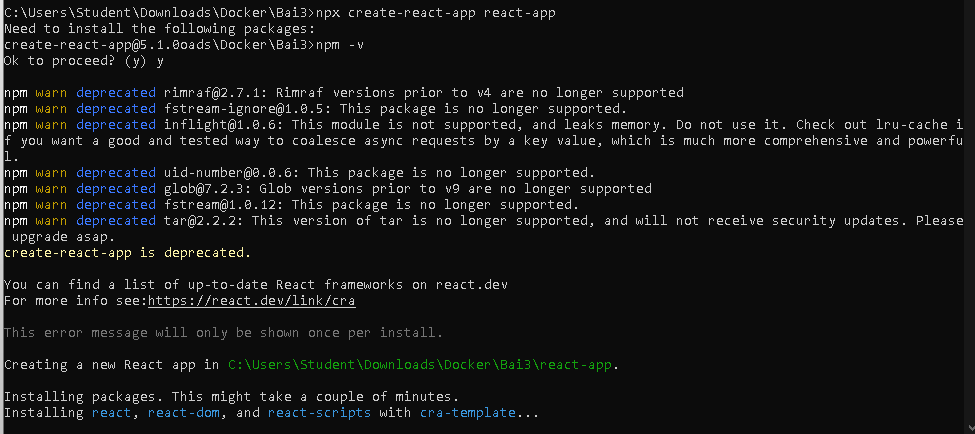
docker run -d -p 5000:5000 --name my\_flask\_app flask-app



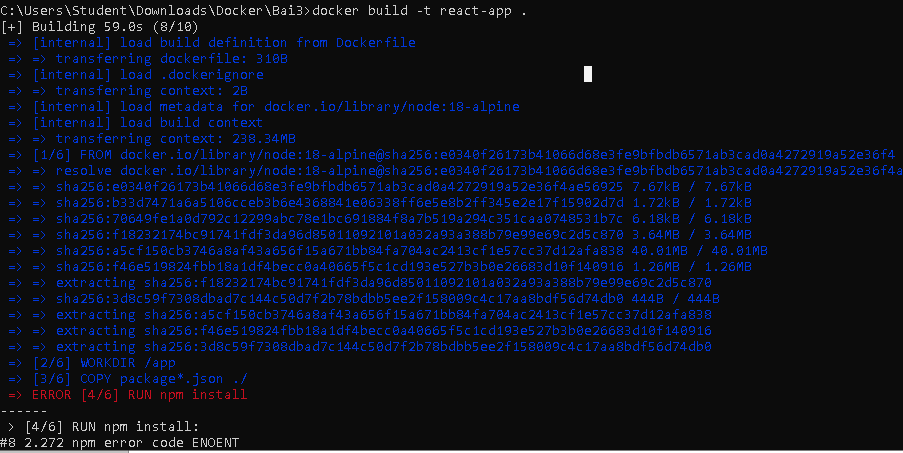
|  |  |  |
| --- | --- | --- |
| Bài 3: Tạo Dockerfile chạy một ứng dụng React | |  |
|  | ***Yêu cầu:*** |  |
|  | Viết Dockerfile để build và chạy một ứng dụng React. | |
|  | Sử dụng node:18-alpine làm base image. |  |
|  |  |  |

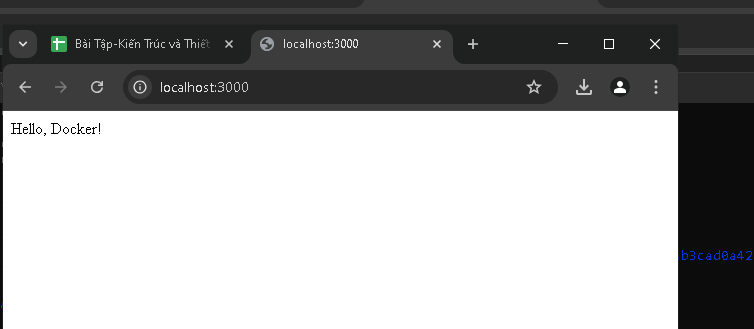


npx create-react-app react-app



docker build -t react-app .

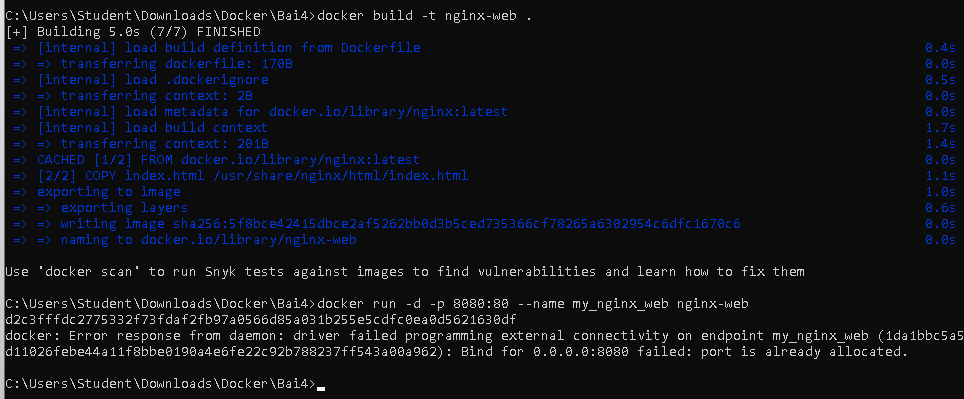


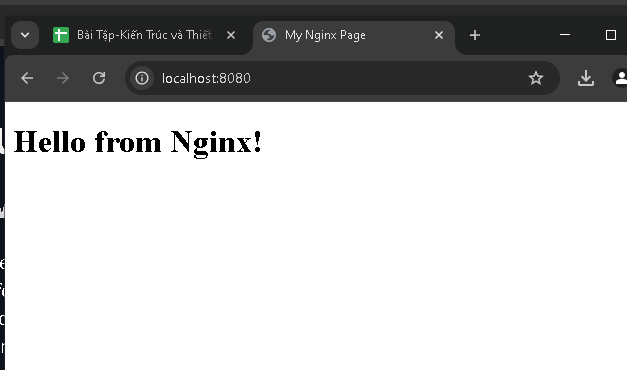


|  |  |  |
| --- | --- | --- |
| Bài 4: Tạo Dockerfile chạy một trang web tĩnh bằng Nginx | | |
|  |  |  |
| ***Yêu cầu:*** |  | |
| Tạo một file index.html đơn giản và sử dụng nginx:latest để phục vụ trang web. |  |  |
|  |  |  |

docker build -t nginx-web .

docker run -d -p 8080:80 --name my\_nginx\_web nginx-web





|  |  |  |
| --- | --- | --- |
| Bài 5: Tạo Dockerfile cho ứng dụng Go | |  |
|  | ***Yêu cầu:*** |  |
|  | Viết Dockerfile để build và chạy một ứng dụng Go đơn giản. | |

docker build -t go-app .

docker run -d -p 8080:8080 --name my\_go\_app go-app

http://localhost:8080