Họ và tên: Trần Minh Tiến

MSSV: 21010611

Tuần 6: Docker

**Phần 1: Các lệnh cơ bản thao tác với Docker**

1 docker –version

Phiên bản của docker

2 docker run hello-world

3 docker pull nginx

4 docker images

5 docker run -d nginx

Chạy container từ image nginx ở chế độ nền (detached mode).

6 docker ps

Xem các container đang chạy

7 docker ps -a

Liệt kê tất cả các container, bao gồm cả những container đã dừng.

8 docker logs <container\_id>

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

Truy cập vào container đang chạy với một shell (sh).

10 docker stop <container\_id>

Dừng container

11 docker restart <container\_id>

Khởi động lại container

12 docker rm <container\_id>

Xóa container

Thông báo lỗi khi container đang running

Stop -> Remove

13 docker container prune

Xóa các container đã dừng

* Quản lý image

14 docker rmi <image\_id>

Xóa image theo id

15 docker image prune -a

Xóa tất cả các image không được sử dụng

16 docker run -d -p 8080:80 nginx (<port host:port container>)

Chạy container nginx ở chế độ nền và ánh xạ cổng 8080 của máy host vào cổng 80 của container.

Chạy trên localhost:8080

17 docker inspect <container\_id>

Xem thông tin chi tiết của 1 container

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

Chạy container nginx và gắn volume mydata vào thư mục /data trong container. (cũng như là tạo folder data trong container nginx ở dạng volume)

19 docker volume ls

Liệt kê tất cả các volumes.

20 docker volume prune

Xóa tất cả volumes không được sử dụng.

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

Chạy container với tên my\_nginx

22 docker stats

Hiển thị thông tin thống kê tài nguyên của các container đang chạy.

* Quản lý mạng

23 docker network ls

Hiển thị tất cả networks trong Docker

24 docker network create <name\_network>

Tạo docker network

25 docker run -d --network <nam\_network> --name <nam\_new\_container> nginx

Chạy container nginx và kết nối vào network my\_network.

26 docker network connect my\_network my\_nginx

Chạy container nginx và kết nối nó vào network my\_network

* Thiết lập biến môi trường (ENV)

27 docker run -d -e <NAME\_VARIABLE>=<value> nginx

Chạy container nginx với biến môi trường MY\_ENV có giá trị hello\_world

28 docker logs -f <name\_container>

Xem logs của container với thoi gian thực

* Tạo Image từ Dockerfile

29 FROM nginx

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

Tạo một Dockerfile từ image nginx và sao chép file index.html vào thư mục chứa trang web của nginx.

File html

Dockerfile

30 docker build -t my\_nginx\_image .

Build image từ Dockerfile với tên my\_nginx\_image.

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

Chạy container từ image my\_nginx\_image và ánh xạ cổng 8080 của máy host vào cổng 80 của container.

**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.

Tạo thư mục và package.json project

Cấu trúc project

Cấu hình Dockerfile

Index.js

Lệnh chạy

Chạy trên web: docker run -d -p 3000:3000 exercise1\_nodejs\_images

Triển khai web

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.

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.

Tạo thư mục và package.json project

Cấu trúc project

Cấu hình Dockerfile

App.jsx

Lệnh build dockerimage

Chạy trên web: docker run -d -p 3000:3000 exercise1\_nodejs\_images

Container

Triển khai web

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.

Bài 5: Tạo Dockerfile cho ứng dụng Go

Yêu cầu:

Viết Dockerfile để build và chạy một ứng dụng Go đơn giản.

Bài 6: Sử dụng Multi-stage Build trong Dockerfile

Viết Dockerfile để build một ứng dụng Node.js với hai stage:

Stage 1: Dùng node:18 để build code.

Stage 2: Dùng node:18-alpine để chạy ứng dụng đã build.

Bài 7: Sử dụng biến môi trường trong Dockerfile

Yêu cầu:

Viết Dockerfile cho ứng dụng Python đọc biến môi trường APP\_ENV và in ra màn hình.

Sử dụng ENV APP\_ENV=development trong Dockerfile.

Bài 8: Tạo Dockerfile cho PostgreSQL tùy chỉnh

Yêu cầu:

Viết Dockerfile để chạy PostgreSQL (postgres:15).

Thêm file SQL để tự động tạo database khi container chạy lần đầu tiên.

Bài 9: Tạo Dockerfile chạy Redis với cấu hình tùy chỉnh

Yêu cầu:

Viết Dockerfile sử dụng redis:latest.

Thêm file redis.conf vào container.

Bài 10: Chạy ứng dụng PHP với Apache

Yêu cầu:

Viết Dockerfile để chạy một ứng dụng PHP đơn giản (php:8.2-apache).

Mount mã nguồn từ máy host vào container.