Docker

Docker là một nền tảng mã nguồn mở cho phép bạn đóng gói, triển khai và chạy các ứng dụng bên trong các container, giúp cho các ứng dụng chạy đồng nhất trên mọi môi trường.

**1. Docker là gì?**

* **Docker** là công cụ hỗ trợ cho DevOps để tạo, triển khai và chạy các ứng dụng dưới dạng container.
* Một **container** là một gói phần mềm nhỏ gọn, chứa tất cả các thành phần cần thiết của ứng dụng (như mã nguồn, thư viện, và các công cụ hệ thống), giúp chạy ứng dụng một cách đồng nhất, bất kể môi trường triển khai.

**2. Lợi ích của việc sử dụng Docker**

* **Tính nhất quán**: Container giúp ứng dụng chạy giống nhau trên mọi môi trường (local, staging, production), giảm thiểu lỗi do sự khác biệt của môi trường.
* **Độc lập và linh hoạt**: Các ứng dụng và các phụ thuộc được đóng gói trong container một cách độc lập, giúp bạn dễ dàng di chuyển hoặc chia sẻ.
* **Tiết kiệm tài nguyên**: Docker container nhẹ hơn nhiều so với máy ảo (VM), giúp tiết kiệm tài nguyên và tăng hiệu suất.
* **Triển khai nhanh chóng**: Dễ dàng đóng gói và triển khai một ứng dụng với đầy đủ các phụ thuộc.

**3. Đóng gói Image cho một dự án Spring Boot**

Giả sử bạn đã có một dự án Spring Boot, dưới đây là cách đóng gói nó thành Docker image.

**a. Tạo File Dockerfile**

Trong thư mục gốc của dự án Spring Boot, tạo một file có tên là **Dockerfile**. Nội dung có thể như sau:

Dockerfile

# Bước 1: Sử dụng hình ảnh JDK làm base

FROM openjdk:17-jdk-slim

# Bước 2: Đặt biến môi trường cho tên file JAR

ARG JAR\_FILE=target/\*.jar

# Bước 3: Sao chép file JAR từ bước build vào container

COPY ${JAR\_FILE} app.jar

# Bước 4: Định nghĩa lệnh khởi chạy

ENTRYPOINT ["java", "-jar", "/app.jar"]

**b. Build File JAR Cho Spring Boot**

Chạy lệnh Maven hoặc Gradle để build file JAR:

# Với Maven

mvn clean package -DskipTests

# Với Gradle

./gradlew build -x test

Lệnh này tạo file JAR trong thư mục target (với Maven) hoặc build/libs (với Gradle).

**c. Tạo Docker Image**

Chạy lệnh sau để build Docker image từ Dockerfile:

docker build -t my-spring-boot-app .

* -t my-spring-boot-app: Đặt tên image là my-spring-boot-app.
* .: Chỉ định thư mục hiện tại có chứa Dockerfile.

**d. Chạy Container từ Image**

Sau khi image được tạo thành công, bạn có thể chạy container:

docker run -p 8080:8080 my-spring-boot-app

* -p 8080:8080: Chuyển tiếp cổng 8080 trên máy host tới cổng 8080 của container.
* my-spring-boot-app: Tên của Docker image bạn đã tạo.

Lúc này, ứng dụng Spring Boot của bạn sẽ chạy bên trong container và có thể truy cập qua http://localhost:8080.

**4. Thực Hành Docker Hóa Dự Án**

**Bước 1: Kiểm tra Dockerfile**

Đảm bảo Dockerfile của bạn đã được cấu hình chính xác như trên.

**Bước 2: Build Image và Chạy Container**

Chạy các lệnh đã đề cập ở trên để build image và khởi động container.

**Bước 3: Kiểm Tra Container**

Dùng các lệnh sau để kiểm tra container:

* **Danh sách container đang chạy**:

docker ps

* **Xem logs của container**:

docker logs <container-id>

* **Dừng container**:

ndocker stop <container-id>

**Bước 4: Kiểm Tra Ứng Dụng**

Mở trình duyệt và vào http://localhost:8080 để kiểm tra ứng dụng của bạn đang chạy.