

# Chương 1: Setup môi trường

Chương này hướng dẫn chuẩn bị môi trường phát triển cho dự án, bao gồm cài đặt công cụ cần thiết, cấu hình biến môi trường, thiết lập IDE và kiểm tra kết nối đến các dịch vụ phụ trợ (DB, cache, message broker...). Sau khi hoàn thành, lập trình viên có thể chạy dự án ở môi trường local.

- [Hướng dẫn setup môi trường và start project](#)

# Hướng dẫn setup môi trường và start project

## 1. Mục tiêu

Hướng dẫn developer:

- Cài đặt môi trường chạy project
- Cấu hình biến môi trường
- Khởi động project Spring Boot
- Kiểm tra project chạy thành công

## 2. Yêu cầu hệ thống

### 2.1. Phần mềm cần cài đặt

Công cụ	Phiên bản khuyến nghị
Java JDK	<b>17 (bắt buộc)</b>
Maven	3.8+
Git	Mới nhất
IDE	IntelliJ IDEA 2023+
PostgreSQL	<b>14 hoặc 15</b>
DB Tool	DBeaver / PgAdmin

### 2.2. Kiểm tra cài đặt

Mở terminal:

```
java -version
```

Kết quả mong muốn:

```
java version "17.0.x"
```

Kiểm tra Maven:

```
mvn -v
```

---

## 3. Clone source code

```
link driver: https://drive.google.com/file/d/1Ytzo-\_uYqhhD8Pa4HUmCEjwPatoCnfPe/view?usp=sharing
```

---

## 4. Cấu hình biến môi trường

Project sử dụng file:

```
application.properties
```

---

### 4.1. Các biến môi trường cơ bản

Ví dụ file:

```
application.properties
```

```
server:
```

```
port: 8080
```

```
spring:
```

```
datasource:
```

```
url: jdbc:postgresql://localhost:5432/lifetex
```

```
username: lifetex
```

```
pass
```

```
#spring.application.name=demo
```

```
#server.port=8080
```

```
#
```

```
## ===== Oracle datasource =====  
#spring.datasource.url=jdbc:oracle:thin:@//192.168.0.111:1111/ORCLPDB1  
#spring.datasource.username=test  
#spring.datasource.password=test  
#spring.datasource.driver-class-name=oracle.jdbc.OracleDriver  
#  
## ===== JPA =====  
#spring.jpa.hibernate.ddl-auto=none  
#spring.jpa.show-sql=true  
#spring.jpa.properties.hibernate.format_sql=true  
#spring.jpa.database-platform=org.hibernate.dialect.OracleDialect  
#  
## ===== Hikari pool (optional) =====  
#spring.datasource.hikari.maximum-pool-size=10  
#spring.datasource.hikari.minimum-idle=2  
#spring.datasource.hikari.connection-timeout=30000  
#  
#management.endpoints.web.exposure.include=health,info
```

## 4.2. Cách chỉnh sửa biến môi trường

### Cách 1: Sửa trực tiếp trong file `application.properties`

Ví dụ đổi DB:

```
spring:  
  datasource:  
    url: jdbc:postgresql://localhost:5432/test_db  
    username: test  
    password: test123
```

# Cách 2: Dùng biến môi trường hệ điều hành

Ví dụ trong `application.properties`:

```
spring:  
  datasource:  
    url: ${DB_URL}  
    username: ${DB_USER}  
    password: ${DB_PASS}
```

---

## Trên Windows (PowerShell)

```
$env:DB_URL="jdbc:postgresql://localhost:5432/lifetex"  
$env:DB_USER="lifetex"  
$env:DB_PASS="123456"
```

---

## Trên Linux/Mac

```
export DB_URL=jdbc:postgresql://localhost:5432/lifetex  
export DB_USER=lifetex  
export DB_PASS=123456
```

---

# 5. Chọn profile chạy

Spring Boot thường dùng profile:

```
dev  
staging  
prod
```

Ví dụ chạy với profile `dev`.

---

## Cách cấu hình profile

### Cách 1: Trong biến môi trường

Windows:

```
$env:SPRING_PROFILES_ACTIVE="dev"
```

Linux/Mac:

```
export SPRING_PROFILES_ACTIVE=dev
```

---

## Cách 2: Trong IDE

VM options:

```
-Dspring.profiles.active=dev
```

---

# 6. Build project

Trong thư mục project:

```
mvn clean install
```

Nếu build thành công sẽ thấy:

```
BUILD SUCCESS
```

---

# 7. Start project

## Chạy trong IDE

Trong IntelliJ:

1. Mở project
2. Mở file:

```
DemoApplication.java
```

3. Nhấn nút **Run**
-

# 8. Kiểm tra project chạy thành công

Sau khi start, kiểm tra log:

```
Started DemoApplication in 5.123 seconds
```

Mở trình duyệt:

```
http://localhost:8080
```

Hoặc gọi API:

```
http://localhost:8080/api/v1/health
```