**Hands on 1**

**Spring Data JPA - Quick Example**

application.properties:

spring.datasource.url=jdbc:h2:mem:testdb

spring.datasource.driverClassName=org.h2.Driver

spring.datasource.username=sa

spring.datasource.password=

spring.jpa.database-platform=org.hibernate.dialect.H2Dialect

spring.h2.console.enabled=true

spring.jpa.hibernate.ddl-auto=update

schema.sql:

mysql> CREATE TABLE department (

-> id INT PRIMARY KEY,

-> name VARCHAR(100)

-> );

Department.java:

**package** com.cognizant.orm\_learn;

**import** javax.persistence.Entity;

**import** javax.persistence.Id;

@Entity

**public** **class** Department {

@Id

**private** **int** id;

**private** String name;

**public** **int** getId() {

**return** id;

}

**public** **void** setId(**int** id) {

**this**.id = id;

}

**public** String getName() {

**return** name;

}

**public** **void** setName(String name) {

**this**.name = name;

}

@Override

**public** String toString() {

**return** "Department [id=" + id + ", name=" + name + "]";

}

}

DepartmentRepository.java:

**package** com.cognizant.orm\_learn;

**import** org.springframework.data.jpa.repository.JpaRepository;

**public** **interface** DepartmentRepository **extends** JpaRepository<Department, Integer> {

}

OrmLearnApplication.java:

**package** com.cognizant.orm\_learn;

**import** java.util.List;

**import** org.springframework.beans.factory.annotation.Autowired;

**import** org.springframework.boot.CommandLineRunner;

**import** org.springframework.boot.SpringApplication;

**import** org.springframework.boot.autoconfigure.SpringBootApplication;

@SpringBootApplication

**public** **class** OrmLearnApplication **implements** CommandLineRunner {

@Autowired

**private** DepartmentRepository departmentRepository;

**public** **static** **void** main(String[] args) {

SpringApplication.*run*(OrmLearnApplication.**class**, args);

}

@Override

**public** **void** run(String... args) {

Department dept = **new** Department();

dept.setId(1);

dept.setName("HR");

departmentRepository.save(dept);

List<Department> departments = departmentRepository.findAll();

departments.forEach(System.***out***::println);

}

}

pom.xml:

<project xmlns="http://maven.apache.org/POM/4.0.0"

xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"

xsi:schemaLocation="http://maven.apache.org/POM/4.0.0

http://maven.apache.org/xsd/maven-4.0.0.xsd">

<modelVersion>4.0.0</modelVersion>

<groupId>com.example</groupId>

<artifactId>orm-learn</artifactId>

<version>1.0.0</version>

<parent>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-starter-parent</artifactId>

<version>2.7.18</version>

<relativePath/> <!-- lookup parent from repository -->

</parent>

<properties>

<java.version>11</java.version>

</properties>

<dependencies>

<dependency>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-starter-data-jpa</artifactId>

</dependency>

<dependency>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-starter</artifactId>

</dependency>

<dependency>

<groupId>com.h2database</groupId>

<artifactId>h2</artifactId>

<scope>runtime</scope>

</dependency>

</dependencies>

<build>

<plugins>

<plugin>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-maven-plugin</artifactId>

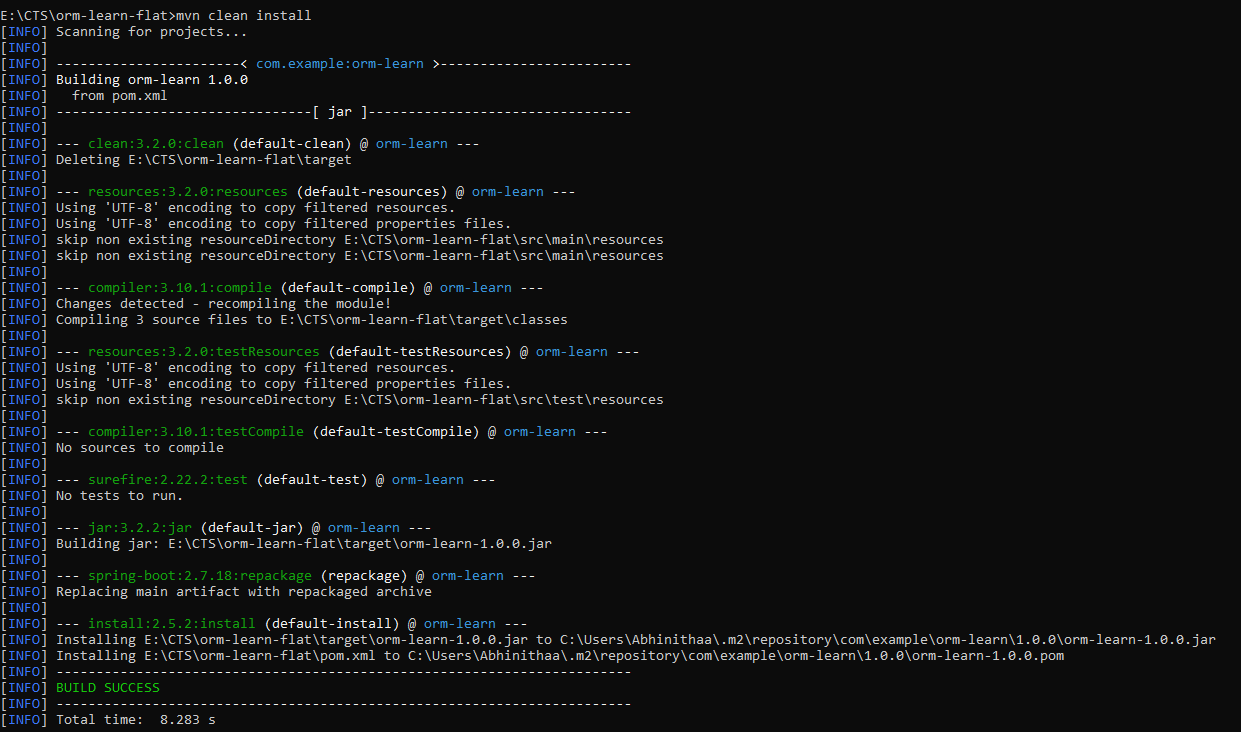
</plugin>

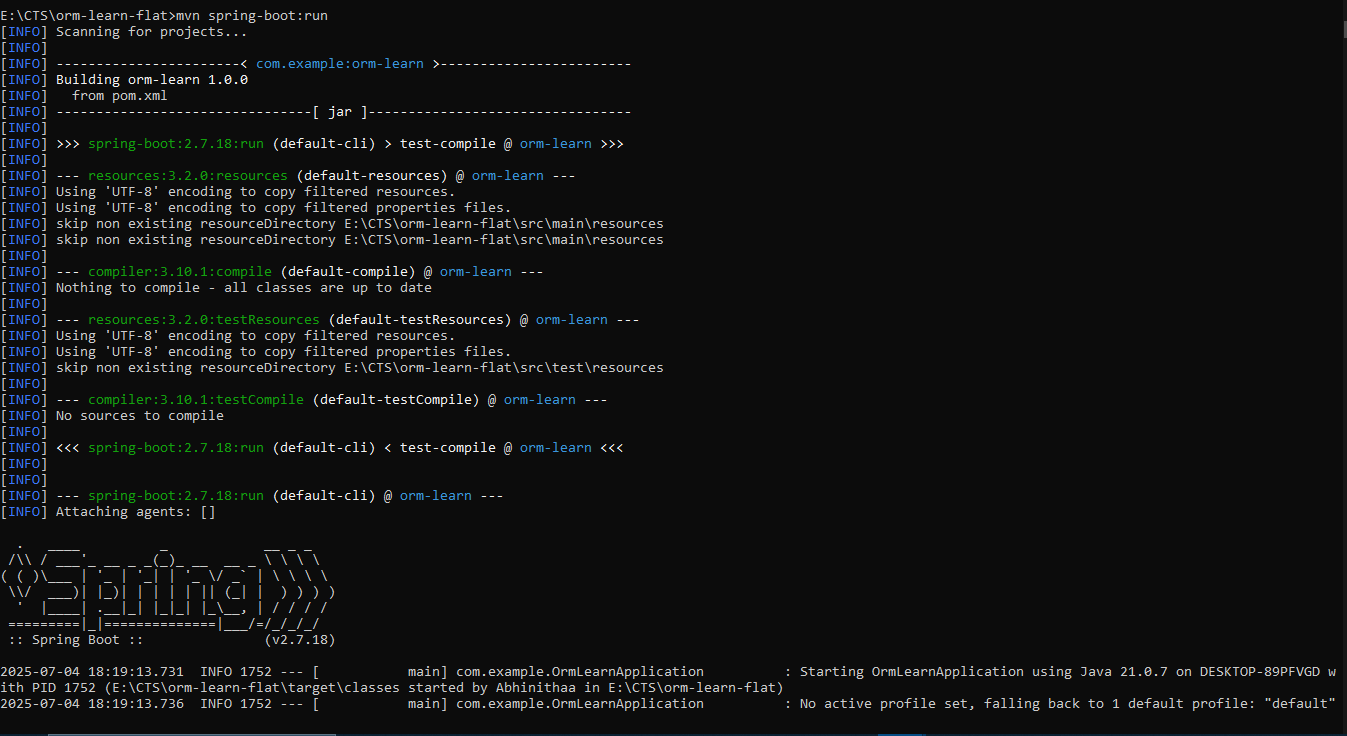
</plugins>

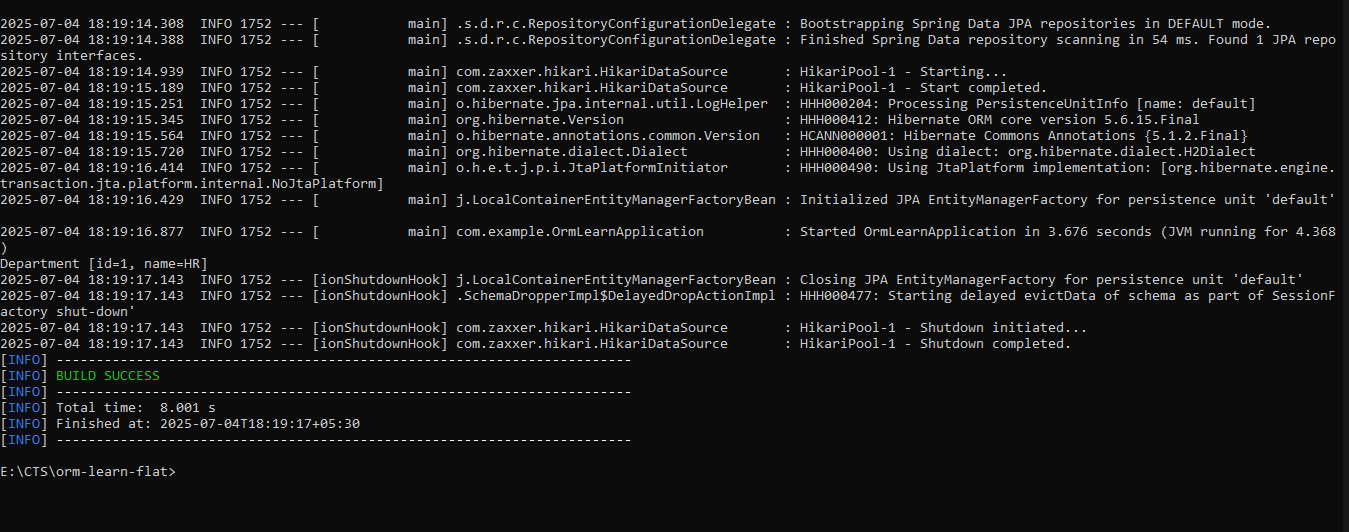
</build>

</project>

CommandPrompt Screenshot:

****

****

****

**Hands on 4**

**Difference between JPA, Hibernate and Spring Data JPA**

Pom.xml:

<project xmlns="http://maven.apache.org/POM/4.0.0"

xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"

xsi:schemaLocation="http://maven.apache.org/POM/4.0.0

http://maven.apache.org/xsd/maven-4.0.0.xsd">

<modelVersion>4.0.0</modelVersion>

<groupId>com.example</groupId>

<artifactId>orm-learn</artifactId>

<version>1.0.0</version>

<packaging>jar</packaging>

<properties>

<java.version>17</java.version>

</properties>

<dependencies>

<dependency>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-starter</artifactId>

<version>2.7.18</version>

</dependency>

<dependency>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-starter-data-jpa</artifactId>

<version>2.7.18</version>

</dependency>

<dependency>

<groupId>com.h2database</groupId>

<artifactId>h2</artifactId>

<version>2.1.214</version>

<scope>runtime</scope>

</dependency>

<!-- ✅ Add these for Java 9+ -->

<dependency>

<groupId>javax.xml.bind</groupId>

<artifactId>jaxb-api</artifactId>

<version>2.3.1</version>

</dependency>

<dependency>

<groupId>org.glassfish.jaxb</groupId>

<artifactId>jaxb-runtime</artifactId>

<version>2.3.1</version>

</dependency>

</dependencies>

<build>

<plugins>

<plugin>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-maven-plugin</artifactId>

<version>2.7.18</version>

<configuration>

<mainClass>com.example.OrmLearnApplication</mainClass> <!-- use full package here -->

</configuration>

</plugin>

</plugins>

</build>

</project>

Department.java:

**package** com.cognizant.orm\_learn;

**import** javax.persistence.Entity;

**import** javax.persistence.Id;

@Entity

**public** **class** Department {

@Id

**private** **int** id;

**private** String name;

**public** **int** getId() {

**return** id;

}

**public** **void** setId(**int** id) {

**this**.id = id;

}

**public** String getName() {

**return** name;

}

**public** **void** setName(String name) {

**this**.name = name;

}

@Override

**public** String toString() {

**return** "Department [id=" + id + ", name=" + name + "]";

}

}

DepartmentRepository.java:

**package** com.cognizant.orm\_learn;

**import** org.springframework.data.jpa.repository.JpaRepository;

**public** **interface** DepartmentRepository **extends** JpaRepository<Department, Integer> {

}

OrmLearnApplication.java:

**package** com.cognizant.orm\_learn;

**import** org.springframework.beans.factory.annotation.Autowired;

**import** org.springframework.boot.CommandLineRunner;

**import** org.springframework.boot.SpringApplication;

**import** org.springframework.boot.autoconfigure.SpringBootApplication;

@SpringBootApplication

**public** **class** OrmLearnApplication **implements** CommandLineRunner {

@Autowired

**private** StudentRepository studentRepository;

**public** **static** **void** main(String[] args) {

SpringApplication.*run*(OrmLearnApplication.**class**, args);

}

@Override

**public** **void** run(String... args) **throws** Exception {

studentRepository.save(**new** Student(1, "Alice"));

studentRepository.save(**new** Student(2, "Bob"));

System.***out***.println("All Students:");

studentRepository.findAll().forEach(System.***out***::println);

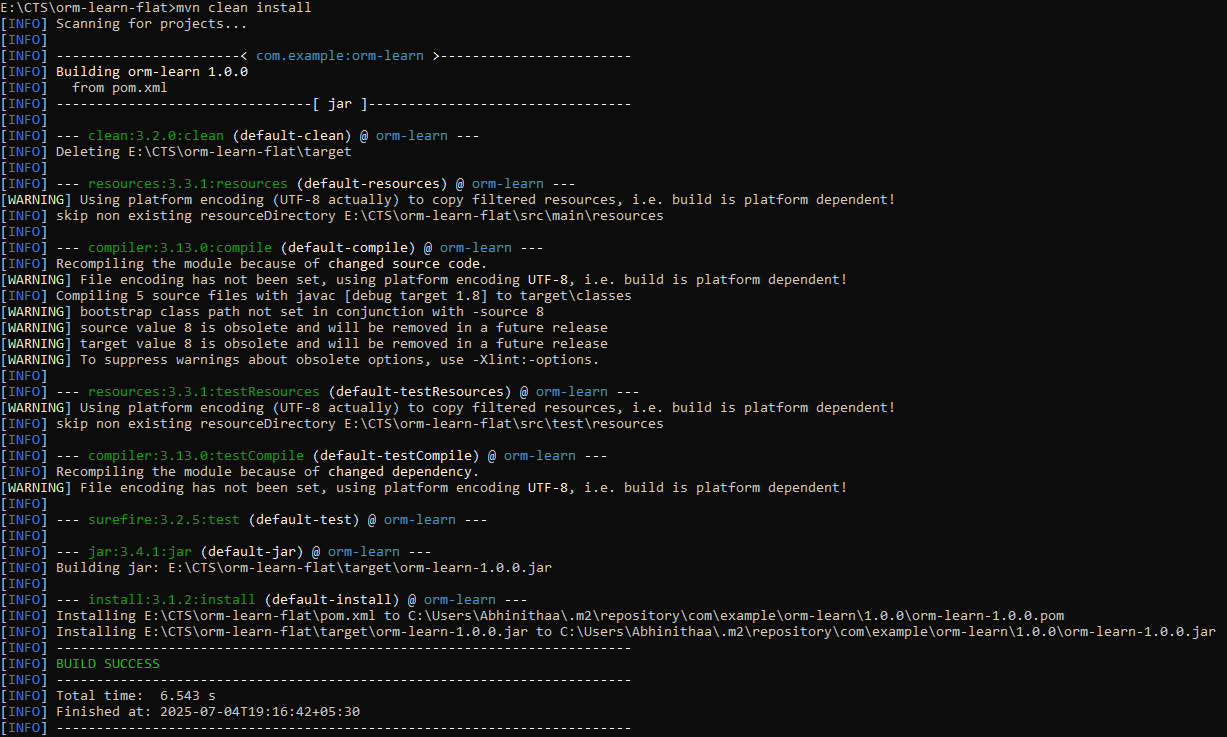
System.***out***.println("Students named Bob:");

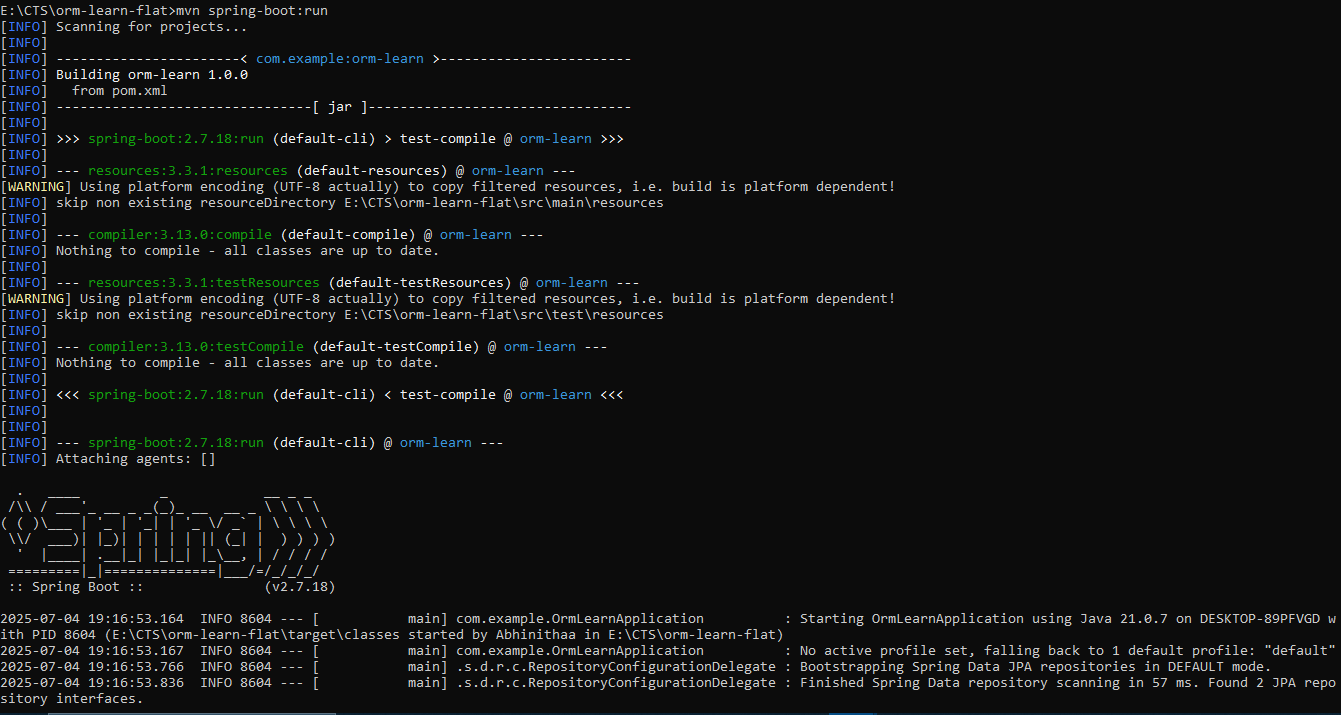
studentRepository.findByName("Bob").forEach(System.***out***::println);

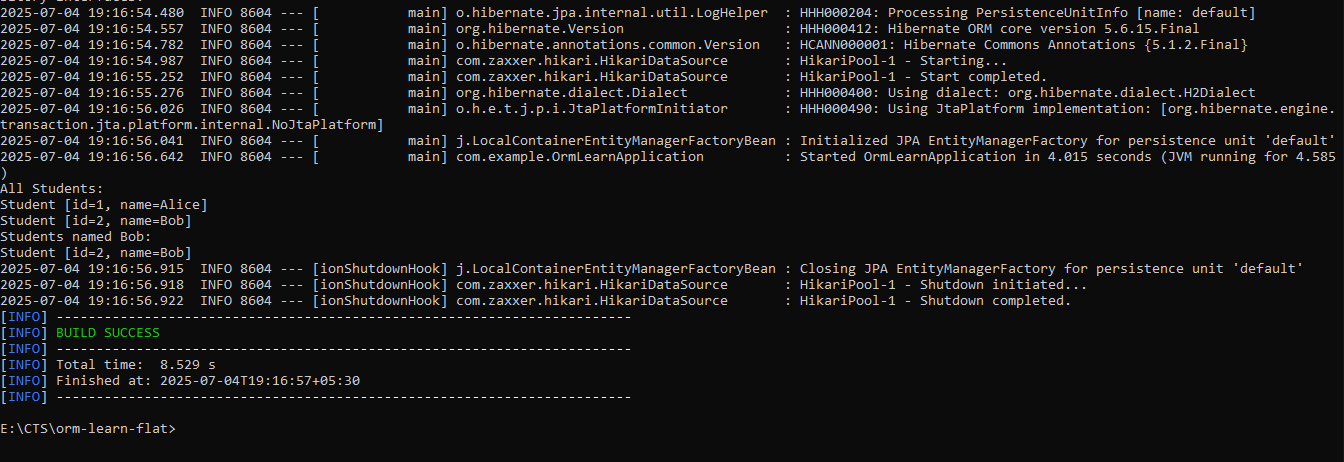
}

}

CommandPrompt Screenshot:

****

****

****