**Hands on 4**

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

**1. EmployeeEntityApplication.java**

package com.cognizant.employee.entity;

import com.cognizant.employee.entity.model.Employee;

import com.cognizant.employee.entity.service.EmployeeService;

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 EmployeeEntityApplication implements CommandLineRunner {

@Autowired

private EmployeeService employeeService;

public static void main(String[] args) {

SpringApplication.run(EmployeeEntityApplication.class, args);

}

@Override

public void run(String... args) {

Employee emp = new Employee();

emp.setName("Logesh");

emp.setDepartment("IT");

emp.setSalary(65000.0);

employeeService.addEmployee(emp);

System.out.println("All Employees:");

employeeService.getAllEmployees().forEach(System.out::println);

}

}

**2. model/Employee.java**

package com.cognizant.employee.entity.model;

import jakarta.persistence.\*;

@Entity

@Table(name = "employee")

public class Employee {

@Id

@GeneratedValue(strategy = GenerationType.IDENTITY)

private int ID;

private String name;

private String department;

private Double salary;

// Getters and Setters

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;

}

public String getDepartment() {

return department;

}

public void setDepartment(String department) {

this.department = department;

}

public Double getSalary() {

return salary;

}

public void setSalary(Double salary) {

this.salary = salary;

}

@Override

public String toString() {

return "Employee{" +

"ID=" + ID +

", name='" + name + '\'' +

", department='" + department + '\'' +

", salary=" + salary +

'}';

}

}

**3. repository/EmployeeRepository.java**

package com.cognizant.employee.entity.repository;

import com.cognizant.employee.entity.model.Employee;

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

public interface EmployeeRepository extends JpaRepository<Employee, Integer> {

}

**4. service/EmployeeService.java**

package com.cognizant.employee.entity.service;

import com.cognizant.employee.entity.model.Employee;

import com.cognizant.employee.entity.repository.EmployeeRepository;

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

import org.springframework.stereotype.Service;

import java.util.List;

import jakarta.transaction.Transactional;

@Service

public class EmployeeService {

@Autowired

private EmployeeRepository employeeRepository;

@Transactional

public void addEmployee(Employee employee) {

employeeRepository.save(employee);

}

public List<Employee> getAllEmployees() {

return employeeRepository.findAll();

}

}

**5. application.properties**

# Database Configuration

spring.datasource.url=jdbc:mysql://localhost:3306/employeeentity

spring.datasource.username=root

spring.datasource.password=Loges@2005

spring.datasource.driver-class-name=com.mysql.cj.jdbc.Driver

# JPA / Hibernate Configuration

spring.jpa.hibernate.ddl-auto=update

spring.jpa.show-sql=true

spring.jpa.properties.hibernate.dialect=org.hibernate.dialect.MySQLDialect

**6.EmployeeController.java:**

package com.cognizant.employee.entity.Controller;

import com.cognizant.employee.entity.model.Employee;

import com.cognizant.employee.entity.service.EmployeeService;

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

import org.springframework.web.bind.annotation.\*;

import java.util.List;

*@RestController*

*@RequestMapping*("/employees")

public class EmployeeController {

*@Autowired*

private EmployeeService employeeService;

// Get all employees

*@GetMapping*

public List<Employee> getAllEmployees() {

return employeeService.getAllEmployees();

}

// Add one employee

*@PostMapping*

public Employee addEmployee(*@RequestBody* Employee employee) {

return employeeService.addEmployee(employee);

}

// Add multiple employees

*@PostMapping*("/bulk")

public List<Employee> addEmployees(*@RequestBody* List<Employee> employees) {

return employeeService.addEmployees(employees);

}

}

**7. pom.xml**

<?xml version="1.0" encoding="UTF-8"?>

<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>

<parent>

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

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

<version>3.5.3</version>

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

</parent>

<groupId>com.cognizant</groupId>

<artifactId>employee-entity</artifactId>

<version>0.0.1-SNAPSHOT</version>

<name>employee-entity</name>

<description>Spring Boot app with JPA and MySQL</description>

<properties>

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

</properties>

<dependencies>

<dependency>

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

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

</dependency>

<dependency>

<groupId>com.mysql</groupId>

<artifactId>mysql-connector-j</artifactId>

<scope>runtime</scope>

</dependency>

<dependency>

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

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

</dependency>

</dependencies>

<build>

<plugins>

<plugin>

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

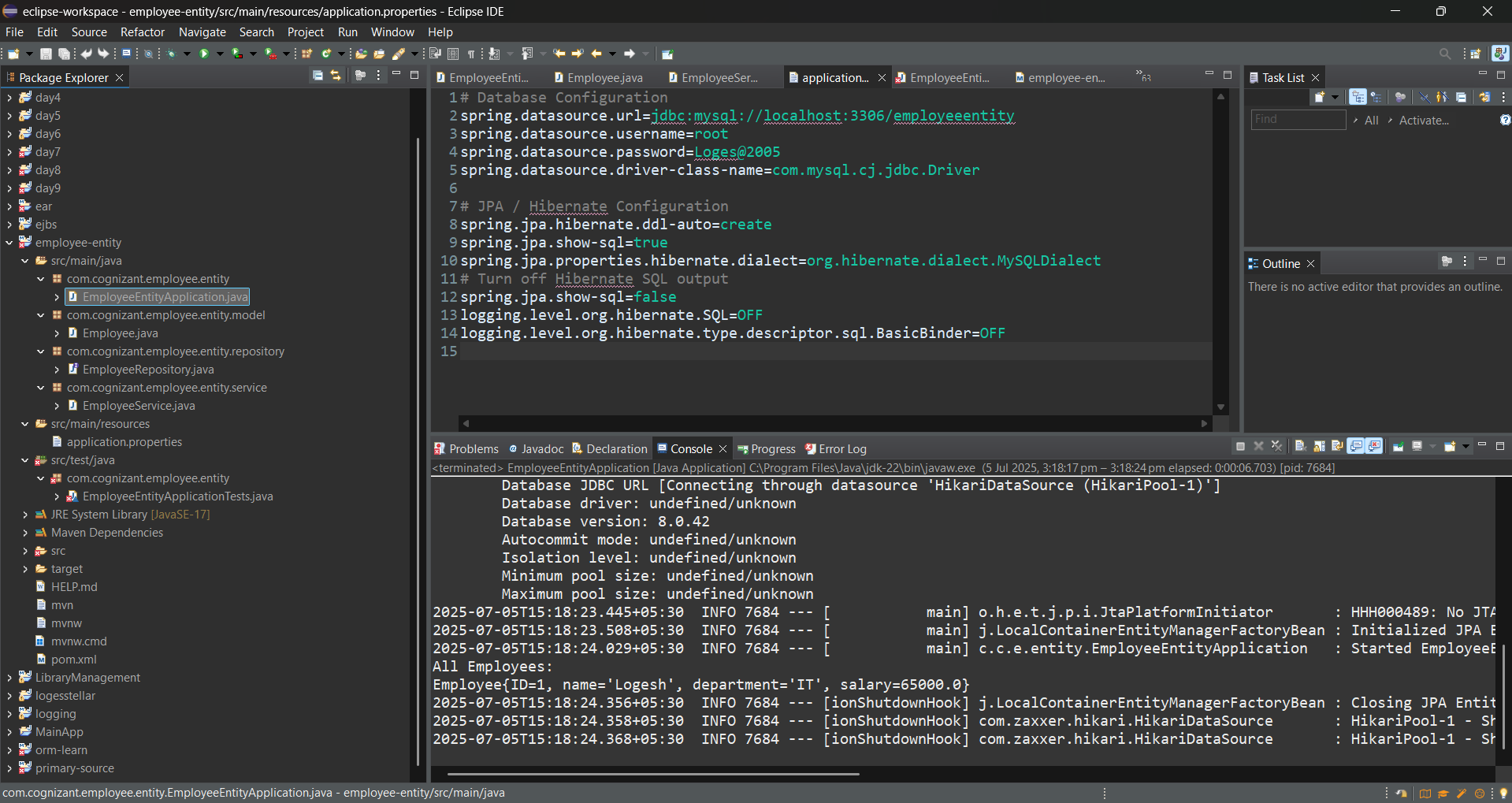
<artifactId>spring-boot-maven-plugin</artifactId>

</plugin>

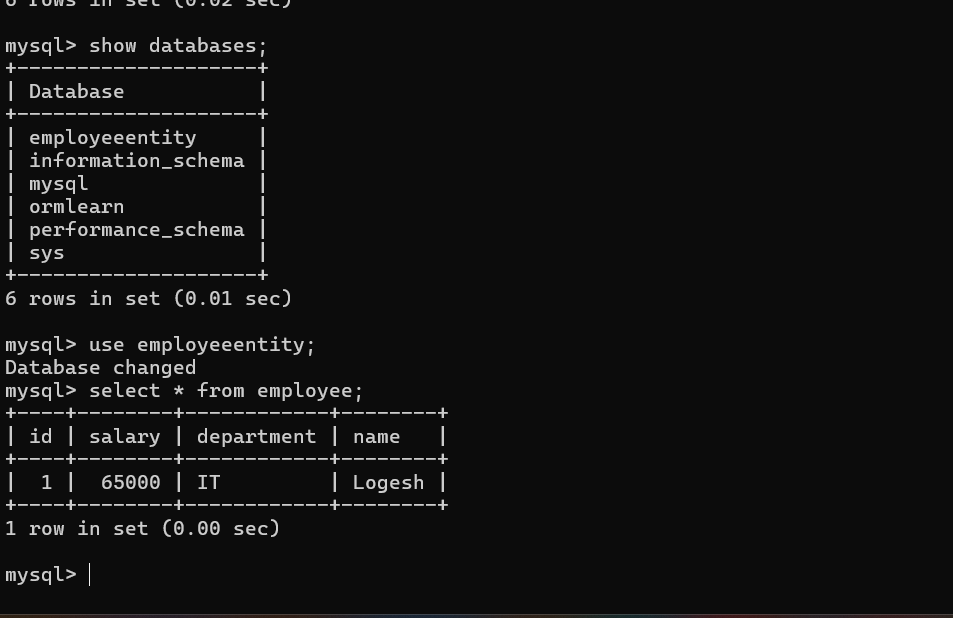
</plugins>

</build>

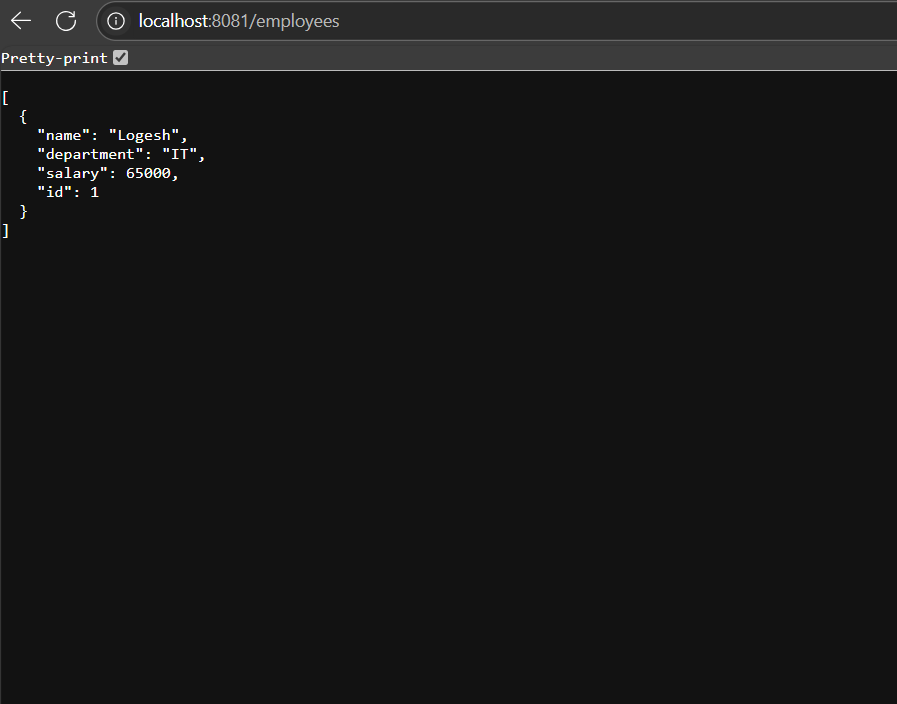
</project>

**Console Output:**

**MySQL Output:**

****

**Browser Output:**

****