**WEEK-3 1.Spring Data JPA handson**

**Hands on 1**

**Spring Data JPA - Quick Example**

**orm\_learn.sql:**

CREATE DATABASE orm\_learn;

USE orm\_learn;

CREATE TABLE country (

code VARCHAR(5) PRIMARY KEY,

name VARCHAR(100)

);

INSERT INTO country (code, name) VALUES ('IN', 'India');

INSERT INTO country (code, name) VALUES ('US', 'United States');

SELECT \* FROM country;

**pom.xml:**

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

<modelVersion>4.0.0</modelVersion>

<groupId>com.cognizant</groupId>

<artifactId>orm-learn</artifactId>

<version>0.0.1-SNAPSHOT</version>

<name>orm-learn</name>

<description>Spring Boot JPA Project</description>

<properties>

<java.version>21</java.version>

</properties>

<dependencies>

<!-- Spring Boot Starter -->

<dependency>

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

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

</dependency>

<dependency>

<groupId>com.mysql</groupId>

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

</dependency>

<dependency>

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

<artifactId>spring-boot-devtools</artifactId>

<scope>runtime</scope>

</dependency>

<!-- Spring Boot Starter Test -->

<dependency>

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

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

<scope>test</scope>

</dependency>

</dependencies>

<build>

<plugins>

<!-- Spring Boot Plugin -->

<plugin>

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

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

</plugin>

</plugins>

</build>

</project>

**application.properties:**

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

spring.datasource.username=root

spring.datasource.password=your\_password

spring.jpa.hibernate.ddl-auto=update

spring.jpa.show-sql=true

**Country.java:**

package com.cognizant.ormlearn.model;

import jakarta.persistence.Entity;

import jakarta.persistence.Id;

import jakarta.persistence.Table;

@Entity

@Table(name = "country")

public class Country {

@Id

private String code;

private String name;

// Getters and setters

public String getCode() {

return code;

}

public void setCode(String code) {

this.code = code;

}

public String getName() {

return name;

}

public void setName(String name) {

this.name = name;

}

}

**CountryRepository.java**

package com.cognizant.ormlearn.repository;

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

import com.cognizant.ormlearn.model.Country;

public interface CountryRepository extends JpaRepository<Country, String> {

}

**CountryService.java**

package com.cognizant.ormlearn.service;

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

import org.springframework.stereotype.Service;

import com.cognizant.ormlearn.model.Country;

import com.cognizant.ormlearn.repository.CountryRepository;

import java.util.Optional;

@Service

public class CountryService {

@Autowired

private CountryRepository countryRepository;

public void addCountry(Country country) {

countryRepository.save(country);

}

public Country findCountryByCode(String code) {

Optional<Country> country = countryRepository.findById(code);

return country.orElse(null);

}

}

**OrmLearnApplication.java:**

package com.cognizant.ormlearn;

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

import org.springframework.boot.CommandLineRunner;

import org.springframework.boot.SpringApplication;

import org.springframework.boot.autoconfigure.SpringBootApplication;

import com.cognizant.ormlearn.model.Country;

import com.cognizant.ormlearn.service.CountryService;

@SpringBootApplication

public class OrmLearnApplication implements CommandLineRunner {

@Autowired

private CountryService countryService;

public static void main(String[] args) {

SpringApplication.run(OrmLearnApplication.class, args);

}

@Override

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

// Create and save new country

Country country = new Country();

country.setCode("IN");

country.setName("India");

countryService.addCountry(country);

System.out.println("✅ Country inserted successfully!");

// Fetch the inserted country

Country retrieved = countryService.findCountryByCode("IN");

if (retrieved != null) {

System.out.println("📌 Country Details:");

System.out.println("Code: " + retrieved.getCode());

System.out.println("Name: " + retrieved.getName());

} else {

System.out.println("❌ Country not found!");

}

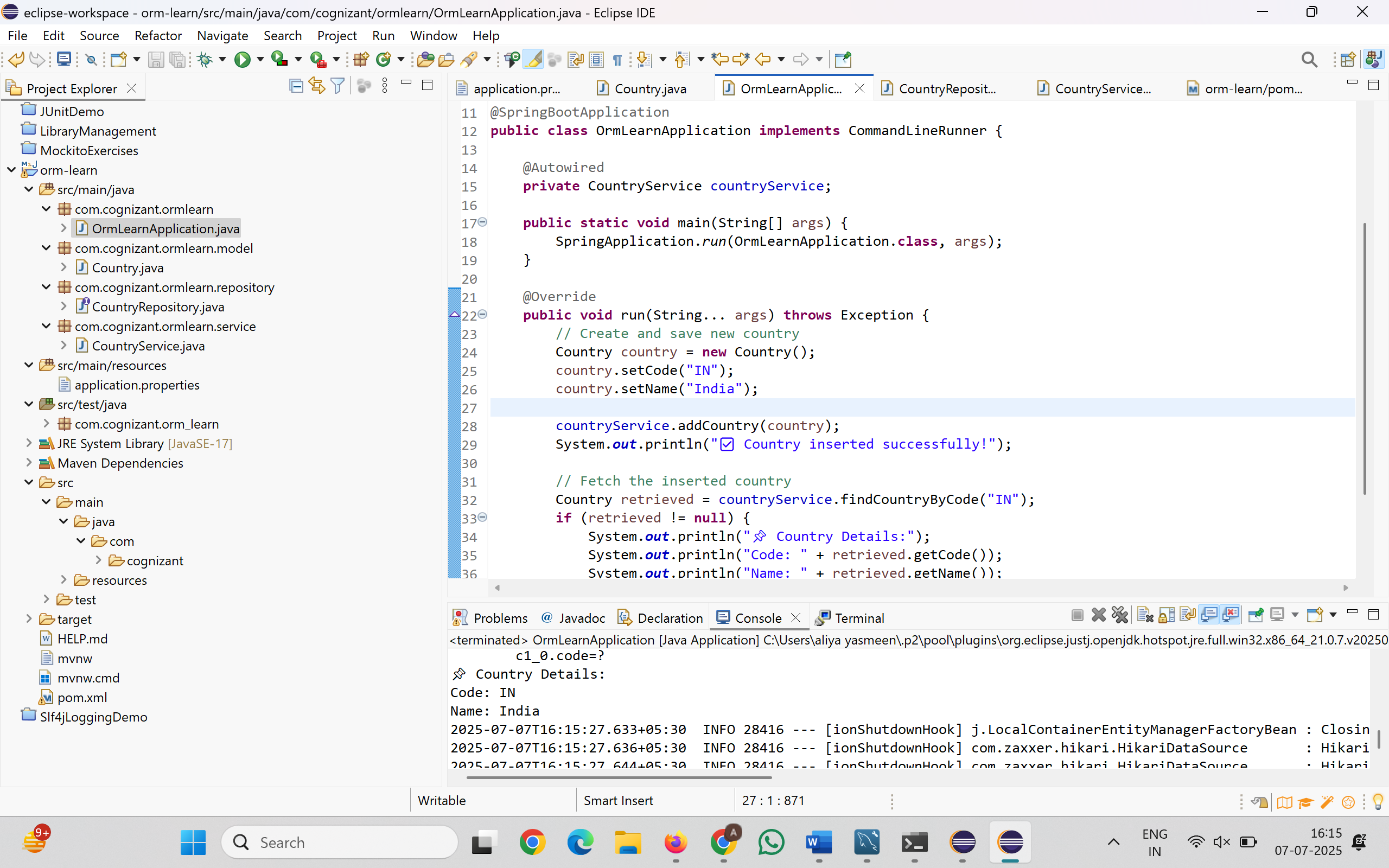
}

}**Output:**

**📌 Country Details:**

**Code: IN**

**Name: India**



**Hands on 4**

**Implement services for managing Country**

1. **Java Persistence API (JPA)**

* JPA is a Java specification (JSR 338) for object-relational mapping (ORM).
* It provides rules and annotations to map Java objects to database tables.
* JPA does not provide actual implementation; it requires a provider like Hibernate.
* Common annotations: @Entity, @Id, @Table, @Column.

1. **Hibernate**

* Hibernate is a popular implementation of the JPA specification.
* It is a full-featured ORM tool used to interact with databases using Java objects.
* Offers features like automatic SQL generation, transaction management, and caching.
* More boilerplate code is required compared to Spring Data JPA.
* Used with or without Spring Framework.

1. **Spring Data JPA**

* Spring Data JPA is not a JPA implementation but an abstraction built on top of JPA and Hibernate.
* It simplifies database operations and eliminates boilerplate code using repository interfaces.
* Automatically provides CRUD methods like save(), findAll(), deleteById() etc.
* Integrates smoothly with Spring Boot and handles transactions, sessions, and queries internally.

**Orm\_learn.sql:**

USE orm\_learn;

DROP TABLE IF EXISTS employee;

CREATE TABLE employee (

id INT PRIMARY KEY,

name VARCHAR(100),

salary DOUBLE,

department VARCHAR(100)

);

**OrmLearnApplication.java:**

package com.cognizant.ormlearn;

import com.cognizant.ormlearn.entity.Employee;

import com.cognizant.ormlearn.repository.EmployeeRepository;

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

import org.springframework.boot.\*;

import org.springframework.boot.autoconfigure.SpringBootApplication;

@SpringBootApplication

public class OrmLearnApplication implements CommandLineRunner {

@Autowired

private EmployeeRepository employeeRepository;

public static void main(String[] args) {

SpringApplication.run(OrmLearnApplication.class, args);

}

@Override

public void run(String... args) {

// Insert an employee

Employee emp = new Employee(6, "Aliya Yasmeen", 75000, "Data Science");

employeeRepository.save(emp);

// Display all employees

System.out.println("Employee list:");

employeeRepository.findAll().forEach(e ->

System.out.println(e.getId() + " | " + e.getName() + " | " + e.getSalary() + " | " + e.getDepartment()));

}

}

**Employee.java:**

**package** com.cognizant.ormlearn.entity;

**import** jakarta.persistence.\*;

@Entity

@Table(name = "employee")

**public** **class** Employee {

@Id

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

**private** String name;

**private** **double** salary;

**private** String department;

**public** Employee() {}

**public** Employee(**int** id, String name, **double** salary, String department) {

**this**.id = id;

**this**.name = name;

**this**.salary = salary;

**this**.department = department;

}

// 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** **double** getSalary() { **return** salary; }

**public** **void** setSalary(**double** salary) { **this**.salary = salary; }

**public** String getDepartment() { **return** department; }

**public** **void** setDepartment(String department) { **this**.department = department; }

}

**EmployeeRepository.java:**

**package** com.cognizant.ormlearn.repository;

**import** com.cognizant.ormlearn.entity.Employee;

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

**public** **interface** EmployeeRepository **extends** JpaRepository<Employee, Integer> {}

**EmployeeService.java:**

**package** com.cognizant.ormlearn.service;

**import** com.cognizant.ormlearn.entity.Employee;

**public** **interface** EmployeeService {

**void** addEmployee(Employee employee);

}

