**WEEK-3**

**Spring-data-jpa-handson**

**Spring Data JPA – Quick Example**

**Software Pre-requisites**

* MySQL Server 8.0
* MySQL Workbench 8
* Eclipse IDE for Enterprise Java Developers (2019-03 R)
* Maven 3.6.2

**Creating a Spring Boot Project using Spring Initializr**

1. Navigate to [https://start.spring.io](https://start.spring.io/)
2. Set the following fields:
   * **Group**: com.cognizant
   * **Artifact**: orm-learn
   * **Description**: Demo project for Spring Data JPA and Hibernate
3. Add the following dependencies:
   * Spring Boot DevTools
   * Spring Data JPA
   * MySQL Driver
4. Click **Generate** to download the project zip file.
5. Extract the zip file and place the folder in the Eclipse workspace.
6. Open Eclipse and import the project:
   * Go to **File > Import > Maven > Existing Maven Projects**
   * Click **Browse**, select the extracted folder, and click **Finish**.

**Database Setup**

1. Open MySQL client using:
2. mysql -u root -p
3. Create the schema:
4. create schema ormlearn;

**Configure application.properties**

**File Path**: src/main/resources/application.properties

# Logging Configuration

logging.level.org.springframework=info

logging.level.com.cognizant=debug

logging.level.org.hibernate.SQL=trace

logging.level.org.hibernate.type.descriptor.sql=trace

logging.pattern.console=%d{dd-MM-yy} %d{HH:mm:ss.SSS} %-20.20thread %5p %-25.25logger{25} %25M %4L %m%n

# Database Configuration

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

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

spring.datasource.username=root

spring.datasource.password=root

# Hibernate Configuration

spring.jpa.hibernate.ddl-auto=validate

spring.jpa.properties.hibernate.dialect=org.hibernate.dialect.MySQL5Dialect

**Build the Project**

Open terminal or command prompt and run the following Maven command:

mvn clean package -Dhttp.proxyHost=proxy.cognizant.com -Dhttp.proxyPort=6050 -Dhttps.proxyHost=proxy.cognizant.com -Dhttps.proxyPort=6050 -Dhttp.proxyUser=123456

**Logging Check in OrmLearnApplication.java**

Add the following code inside the main() class:

import org.slf4j.Logger;

import org.slf4j.LoggerFactory;

@SpringBootApplication

public class OrmLearnApplication {

private static final Logger LOGGER = LoggerFactory.getLogger(OrmLearnApplication.class);

public static void main(String[] args) {

SpringApplication.run(OrmLearnApplication.class, args);

LOGGER.info("Inside main");

}

}

**Create Country Table in MySQL**

create table country (

co\_code varchar(2) primary key,

co\_name varchar(50)

);

insert into country values ('IN', 'India');

insert into country values ('US', 'United States of America');

**Model Class - Country.java**

**Package**: com.cognizant.ormlearn.model

import javax.persistence.\*;

@Entity

@Table(name = "country")

public class Country {

@Id

@Column(name = "co\_code")

private String code;

@Column(name = "co\_name")

private String name;

// Getters and Setters

@Override

public String toString() {

return "Country [code=" + code + ", name=" + name + "]";

}

}

**Repository Interface - CountryRepository.java**

**Package**: com.cognizant.ormlearn.repository

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

import org.springframework.stereotype.Repository;

import com.cognizant.ormlearn.model.Country;

@Repository

public interface CountryRepository extends JpaRepository<Country, String> {

}

**Service Class - CountryService.java**

**Package**: com.cognizant.ormlearn.service

import java.util.List;

import javax.transaction.Transactional;

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

import org.springframework.stereotype.Service;

import com.cognizant.ormlearn.model.Country;

import com.cognizant.ormlearn.repository.CountryRepository;

@Service

public class CountryService {

@Autowired

private CountryRepository countryRepository;

@Transactional

public List<Country> getAllCountries() {

return countryRepository.findAll();

}

}

**Testing the Application**

Update OrmLearnApplication.java:

import java.util.List;

import org.springframework.context.ApplicationContext;

import org.springframework.boot.SpringApplication;

private static CountryService countryService;

public static void main(String[] args) {

ApplicationContext context = SpringApplication.run(OrmLearnApplication.class, args);

countryService = context.getBean(CountryService.class);

testGetAllCountries();

}

private static void testGetAllCountries() {

LOGGER.info("Start");

List<Country> countries = countryService.getAllCountries();

LOGGER.debug("countries={}", countries);

LOGGER.info("End");

}

**SME Walkthrough Checklist**

| **Component** | **Description** |
| --- | --- |
| src/main/java | Contains application source code |
| src/main/resources | Application config (e.g., application.properties) |
| src/test/java | Unit and integration tests (optional for now) |
| OrmLearnApplication.java | Entry point of Spring Boot app |
| @SpringBootApplication | Combines @Configuration, @EnableAutoConfiguration, and @ComponentScan |
| pom.xml | Maven dependency configuration |
| Dependency Tree | Can be viewed via Eclipse “Dependency Hierarchy” tab |