**Spring Data JPA - Quick Example**

**application.properties**

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

spring.datasource.username=root

spring.datasource.password=Sajitha\_$2004

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

spring.jpa.hibernate.ddl-auto=update

spring.jpa.show-sql=true

spring.jpa.properties.hibernate.format\_sql=true

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

logging.level.org.hibernate.SQL=debug

logging.level.org.hibernate.orm.jdbc.bind=trace

**Application.java**

package com.example.countryapp;

import com.example.countryapp.model.Country;

import com.example.countryapp.service.CountryService;

import org.springframework.boot.SpringApplication;

import org.springframework.boot.autoconfigure.SpringBootApplication;

import org.springframework.context.ApplicationContext;

import org.slf4j.Logger;

import org.slf4j.LoggerFactory;

import java.util.List;

@SpringBootApplication

public class CountryAppApplication {

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

private static CountryService countryService;

public static void main(String[] args) {

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

LOGGER.info("Inside main");

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");

}

}

**CountryServiceImpl.java**

package com.example.countryapp.service;

import com.example.countryapp.model.Country;

import com.example.countryapp.repository.CountryRepository;

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

import org.springframework.stereotype.Service;

import java.util.List;

@Service

public class CountryServiceImpl extends CountryService {

@Autowired

private CountryRepository countryRepository;

@Override

public List<Country> searchCountriesByName(String name) {

return countryRepository.findByNameContainingIgnoreCase(name);

}

@Override

public List<Country> searchCountriesByNameSorted(String name) {

return countryRepository.findByNameContainingIgnoreCaseOrderByNameAsc(name);

}

@Override

public List<Country> searchCountriesByStartingLetter(String prefix) {

return countryRepository.findByNameStartingWith(prefix);

}

}

**CountryService.java**

package com.example.countryapp.service;

import com.example.countryapp.model.Country;

import com.example.countryapp.repository.CountryRepository;

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

import org.springframework.stereotype.Service;

import org.springframework.transaction.annotation.Transactional;

import java.util.List;

@Service

public abstract class CountryService {

@Autowired

private CountryRepository countryRepository;

@Transactional

public List<Country> getAllCountries() {

return countryRepository.findAll();

}

public abstract List<Country> searchCountriesByName(String name);

public abstract List<Country> searchCountriesByNameSorted(String name);

public abstract List<Country> searchCountriesByStartingLetter(String prefix);

}

**Country.java**

package com.example.countryapp.model;

import jakarta.persistence.Entity;

import jakarta.persistence.Id;

import jakarta.persistence.Table;

import jakarta.persistence.Column;

@Entity

@Table(name = "country")

public class Country {

@Id

@Column(name = "co\_code")

private String code;

@Column(name = "co\_name")

private String name;

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;

}

@Override

public String toString() {

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

}

}

**CountryRepository.java**

package com.example.countryapp.repository;

import com.example.countryapp.model.Country;

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

import org.springframework.stereotype.Repository;

import java.util.List;

@Repository

public interface CountryRepository extends JpaRepository<Country, String> {

List<Country> findByNameContainingIgnoreCase(String name);

List<Country> findByNameContainingIgnoreCaseOrderByNameAsc(String name);

List<Country> findByNameStartingWith(String prefix);

}

**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>countryapp</artifactId>

<version>1.0-SNAPSHOT</version>

<properties>

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

<spring.boot.version>3.2.5</spring.boot.version>

</properties>

<dependencies>

<dependency>

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

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

<version>${spring.boot.version}</version>

</dependency>

<dependency>

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

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

<version>${spring.boot.version}</version>

</dependency>

<dependency>

<groupId>com.mysql</groupId>

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

<version>8.3.0</version>

</dependency>

<dependency>

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

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

<version>${spring.boot.version}</version>

<scope>runtime</scope>

</dependency>

<dependency>

<groupId>org.projectlombok</groupId>

<artifactId>lombok</artifactId>

<version>1.18.30</version>

<scope>provided</scope>

</dependency>

</dependencies>

<build>

<plugins>

<plugin>

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

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

<version>${spring.boot.version}</version>

</plugin>

</plugins>

</build>

</project>



















