**Spring Data JPA - Quick Example**

**Spring Data JPA with Spring Boot, Hibernate**

Vaishnavi

4/7/2025

This project demonstrates the integration of **Spring Data JPA** with **Hibernate** to perform Object-Relational Mapping (ORM) in a Java application. It is built using **Spring Boot** and connects to a **MySQL** database. The project follows industry practices using layered architecture, separating the model, repository, service, and application logic.

**Objective:**

To understand how **JPA**, **Hibernate**, and **Spring Data JPA** work together.

To demonstrate CRUD operations using **Spring Boot** and **JPA Repository**.

To reduce boilerplate database code with **Spring Data JPA abstraction**.

To connect and query a **MySQL** database using **Spring Boot** configuration.

**Implementation:**

### Setup a Project Initialization

### ****Open Spring Initializer****

Group: com.cognizant

Artifact: orm-learn

Dependencies:

Spring Boot DevTools

Spring Data JPA

MySQL Driver

### ****Open a Existing Maven Project****:

File > Import > Maven > Existing Maven Projects

**MySQL Schema**

CREATE SCHEMA ormlearn;

CREATE TABLE country (

co\_code VARCHAR(2) PRIMARY KEY,

co\_name VARCHAR(50)

);

INSERT INTO country VALUES ('IN', 'India'), ('US', 'United States of America');

## Add Application Properties

# Logging

logging.level.org.springframework=info

logging.level.com.cognizant=debug

logging.level.org.hibernate.SQL=trace

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

# Console Log Pattern

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

# DB 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=my\_passoword

# Hibernate

spring.jpa.hibernate.ddl-auto=validate

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

**Entity Class: Country.java**

@Entity

@Table(name = "country")

public class Country {

@Id

@Column(name = "co\_code")

private String code;

@Column(name = "co\_name")

private String name;

// Getters, Setters, toString()

}

**Repository: CountryRepository.java**

@Repository

public interface CountryRepository extends JpaRepository<Country, String> {

}

**Service: CountryService.java**

@Service

public class CountryService {

@Autowired

private CountryRepository countryRepository;

@Transactional

public List<Country> getAllCountries() {

return countryRepository.findAll();

}

}

**MainApp**

@SpringBootApplication

public class Application {

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

private static CountryService countryService;

public static void main(String[] args) {

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

LOGGER.info("Inside main");

countryService = context.getBean(CountryService.class);

testGetAllCountries();

}

private static void testGetAllCountries() {

LOGGER.info("Start");

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

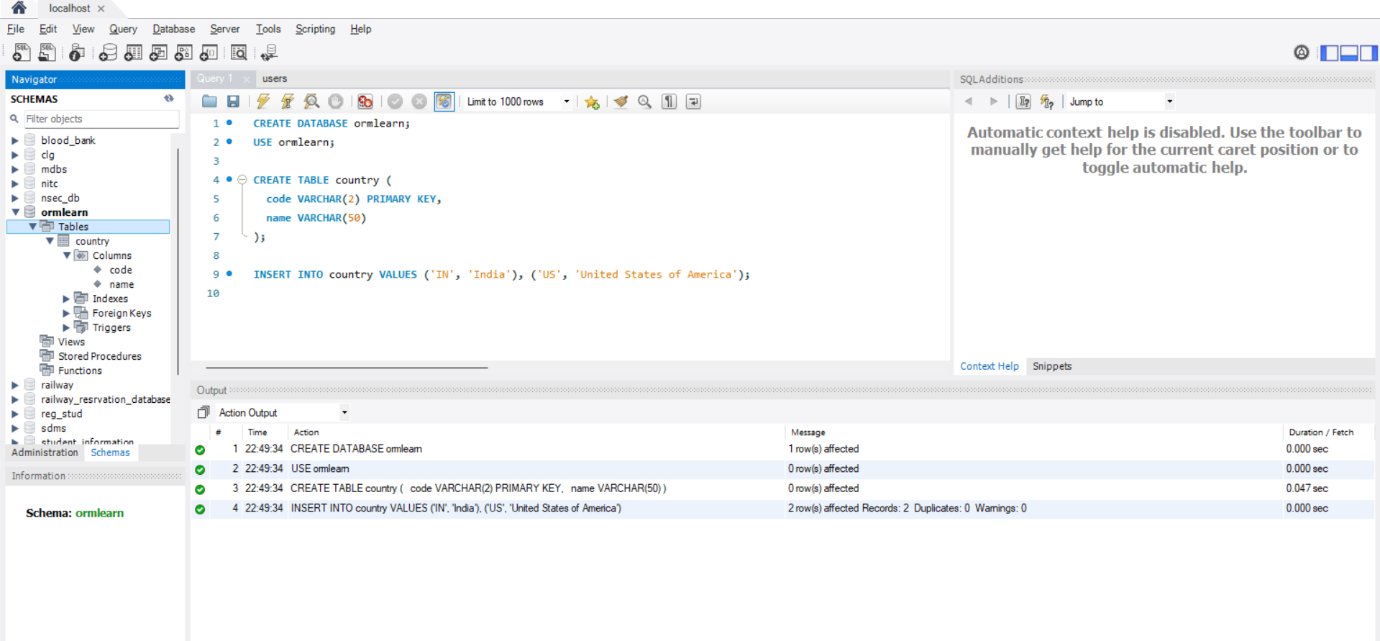
LOGGER.debug("countries={}", countries);

LOGGER.info("End");

}

}

**Output:**

****