**Hands on 1**

**Spring Data JPA - Quick Example** 

Directory:

orm-learn/

├── pom.xml

└── src/

├── main/

│ ├── java/

│ │ └── com/

│ │ └── cognizant/

│ │ └── ormlearn/

│ │ ├── OrmLearnApplication.java

│ │ ├── model/

│ │ │ └── Country.java

│ │ ├── repository/

│ │ │ └── CountryRepository.java

│ │ └── service/

│ │ └── CountryService.java

│ └── resources/

│ ├── application.properties

│

│ └── data.sql

**Code for Country.java:**

package com.cognizant.ormlearn.model;

import javax.persistence.Column;

import javax.persistence.Entity;

import javax.persistence.Id;

import javax.persistence.Table;

@Entity

@Table(name = "country")

public class Country {

    @Id

    @Column(name = "co\_code", length = 2, nullable = false)

    private String code;

    @Column(name = "co\_name", length = 50, nullable = false)

    private String name;

    public Country() {

    }

    public Country(String code, String name) {

        this.code = code;

        this.name = 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 + "]";

    }

}

**Code for Repo.java:**

package com.cognizant.ormlearn.repository;

import com.cognizant.ormlearn.model.Country;

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

import org.springframework.stereotype.Repository;

@Repository

public interface Repo extends JpaRepository<Country, String> {

}

**Code for CountryService.java:**

package com.cognizant.ormlearn.service;

import com.cognizant.ormlearn.model.Country;

import com.cognizant.ormlearn.repository.Repo;

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

import org.springframework.stereotype.Service;

import java.util.List;

import java.util.Optional;

@Service

public class CountryService {

    @Autowired

    private Repo repo;

    public List<Country> getAllCountries() {

        return repo.findAll();

    }

}

**Code for Applicataion.java:**

package com.cognizant.ormlearn;

import com.cognizant.ormlearn.service.CountryService;

import com.cognizant.ormlearn.model.Country;

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;

import com.sun.imageio.plugins.common.I18N;

@SpringBootApplication

public class Application {

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

    private static CountryService countryService;

    public static void main(String[] args) {

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

        countryService = ctx.getBean(CountryService.class);

        testGetAllCountries();

    }

    private static void testGetAllCountries() {

        LOGGER.info("Start test");

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

        System.out.println("Finding All Countries:");

                System.out.println("Countries"+countryService.getAllCountries());

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

        LOGGER.info("End test");

    }

}

**Code for data.sql:**

CREATE TABLE country (

    co\_code VARCHAR(2)  PRIMARY KEY,

    co\_name VARCHAR(50) NOT NULL

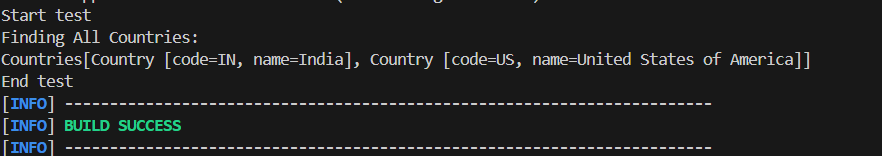
);

INSERT INTO country (co\_code, co\_name) VALUES

  ('IN', 'India'),

  ('US', 'United States of America');

Output:

 **Hands on 4**

**Difference between JPA, Hibernate and Spring Data JPA**   
1. **JPA** (Java Persistence API)

Use annotations like @Entity,@Id

Using pseudo code:

import javax.persistence.\*;

@Entity

@Table(name = "country")

public class Country {

@Id

private String code;

@Column(name = "name")

private String name;

}

// Using EntityManager (JPA standard)

EntityManagerFactory emf = Persistence.createEntityManagerFactory("my-unit");

EntityManager em = emf.createEntityManager();

Country c = em.find(Country.class, "IN");

System.out.println(c.getName());

**Hibernate:**

It provides,Lazy Loading ,caching etc

Pseudo code:

import org.hibernate.Session;

import org.hibernate.SessionFactory;

import org.hibernate.cfg.Configuration;

SessionFactory factory = new Configuration().configure().buildSessionFactory();

Session session = factory.openSession();

Country country = session.get(Country.class, "IN");

System.out.println(country.getName());

**Spring Data JPA**

// Entity

@Entity

public class Country {

@Id

private String code;

private String name;

}

// Repository

@Repository

public interface CountryRepository extends JpaRepository<Country, String> {

List<Country> findByNameContaining(String keyword); // custom query method

}

// Service

@Service

public class CountryService {

@Autowired

private CountryRepository repo;

public List<Country> getAllCountries() {

return repo.findAll();

}

}

// In Main class

ApplicationContext ctx = SpringApplication.run(App.class, args);

CountryService service = ctx.getBean(CountryService.class);

service.getAllCountries().forEach(System.out::println);

**Hands on 6**

**Find a country based on country code** 

Directory:

orm-learn/

├── pom.xml

└── src/

├── main/

│ ├── java/

│ │ └── com/

│ │ └── cognizant/

│ │ └── ormlearn/

│ │ ├── OrmLearnApplication.java

│ │ ├── model/

│ │ │ └── Country.java

│ │ ├── repository/

│ │ │ └── CountryRepository.java

│ │ └── service/

│ │ └── CountryService.java

│ └── resources/

│ ├── application.properties

│

│ └── data.sql

**Code for Country.java:**

package com.cognizant.ormlearn.model;

import javax.persistence.Column;

import javax.persistence.Entity;

import javax.persistence.Id;

import javax.persistence.Table;

@Entity

@Table(name = "country")

public class Country {

    @Id

    @Column(name = "co\_code", length = 2, nullable = false)

    private String code;

    @Column(name = "co\_name", length = 50, nullable = false)

    private String name;

    public Country() {

    }

    public Country(String code, String name) {

        this.code = code;

        this.name = 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 + "]";

    }

}

**Code for Repo.java:**

package com.cognizant.ormlearn.repository;

import com.cognizant.ormlearn.model.Country;

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

import org.springframework.stereotype.Repository;

@Repository

public interface Repo extends JpaRepository<Country, String> {

}

**Code for CountryService.java:**

package com.cognizant.ormlearn.service;

import com.cognizant.ormlearn.model.Country;

import com.cognizant.ormlearn.repository.Repo;

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

import org.springframework.stereotype.Service;

import java.util.List;

import java.util.Optional;

@Service

public class CountryService {

    @Autowired

    private Repo repo;

 public Optional<Country> getCountryByCode(String code) {

        return repo.findById(code);

    }

}

**Code for Applicataion.java:**

package com.cognizant.ormlearn;

import com.cognizant.ormlearn.service.CountryService;

import com.cognizant.ormlearn.model.Country;

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;

import com.sun.imageio.plugins.common.I18N;

@SpringBootApplication

public class Application {

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

    private static CountryService countryService;

    public static void main(String[] args) {

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

        countryService = ctx.getBean(CountryService.class);

        testGetAllCountries();

    }

    private static void testGetAllCountries() {

        LOGGER.info("Start test");

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

        System.out.println("Getting Country by code:");

                System.out.println("Countries"+countryService.getCountryByCode (“IN”));

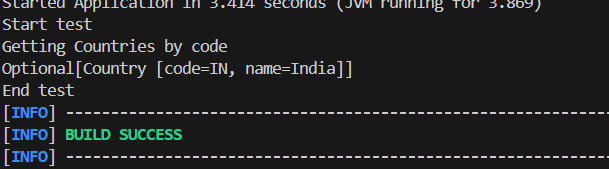
        LOGGER.debug("countries={}", countries);

        LOGGER.info("End test");

    }

}

**Ouput:**

****

**Hands on 7**

**Add a new country**

Directory:

orm-learn/

├── pom.xml

└── src/

├── main/

│ ├── java/

│ │ └── com/

│ │ └── cognizant/

│ │ └── ormlearn/

│ │ ├── OrmLearnApplication.java

│ │ ├── model/

│ │ │ └── Country.java

│ │ ├── repository/

│ │ │ └── CountryRepository.java

│ │ └── service/

│ │ └── CountryService.java

│ └── resources/

│ ├── application.properties

│

│ └── data.sql

**Code for Country.java:**

package com.cognizant.ormlearn.model;

import javax.persistence.Column;

import javax.persistence.Entity;

import javax.persistence.Id;

import javax.persistence.Table;

@Entity

@Table(name = "country")

public class Country {

    @Id

    @Column(name = "co\_code", length = 2, nullable = false)

    private String code;

    @Column(name = "co\_name", length = 50, nullable = false)

    private String name;

    public Country() {

    }

    public Country(String code, String name) {

        this.code = code;

        this.name = 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 + "]";

    }

}

**Code for Repo.java:**

package com.cognizant.ormlearn.repository;

import com.cognizant.ormlearn.model.Country;

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

import org.springframework.stereotype.Repository;

@Repository

public interface Repo extends JpaRepository<Country, String> {

}

**Code for CountryService.java:**

package com.cognizant.ormlearn.service;

import com.cognizant.ormlearn.model.Country;

import com.cognizant.ormlearn.repository.Repo;

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

import org.springframework.stereotype.Service;

import java.util.List;

import java.util.Optional;

@Service

public class CountryService {

    @Autowired

    private Repo repo;

public Country addCountry(Country country) {

        return repo.save(country);

    }

}

**Code for Applicataion.java:**

package com.cognizant.ormlearn;

import com.cognizant.ormlearn.service.CountryService;

import com.cognizant.ormlearn.model.Country;

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;

import com.sun.imageio.plugins.common.I18N;

@SpringBootApplication

public class Application {

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

    private static CountryService countryService;

    public static void main(String[] args) {

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

        countryService = ctx.getBean(CountryService.class);

        testGetAllCountries();

    }

    private static void testGetAllCountries() {

        LOGGER.info("Start test");

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

 System.out.println("ADDING COUNTRY:");

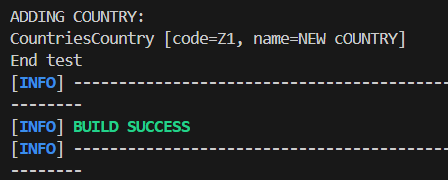
                System.out.println("Countries"+countryService.addCountry(new Country("Z1","NEWcOUNTRY")));

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

        LOGGER.info("End test");

    }

**OUTPUT:**

****