File:

package com.cognizant.orm\_learn.model;

import jakarta.persistence.Column;

import jakarta.persistence.Entity;

import jakarta.persistence.Id;

import jakarta.persistence.Table;

@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 + "]";

    }

}

package com.cognizant.orm\_learn.repository;

import java.util.List;

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

import com.cognizant.orm\_learn.model.Country;

public interface CountryRepository extends JpaRepository<Country, String> {

List<Country> findByNameContainingIgnoreCase(String keyword);

}

package com.cognizant.orm\_learn.service;

import java.util.List;

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

import org.springframework.stereotype.Service;

import org.springframework.transaction.annotation.Transactional;

import com.cognizant.orm\_learn.model.Country;

import com.cognizant.orm\_learn.repository.CountryRepository;

@Service

public class CountryService {

    @Autowired

    private CountryRepository countryRepository;

    @Transactional

    public List<Country> getAllCountries() {

        return countryRepository.findAll();

    }

    @Transactional

    public Country findCountryByCode(String code) throws Exception {

        return countryRepository.findById(code)

            .orElseThrow(() -> new Exception("Country not found: " + code));

    }

    @Transactional

    public void addCountry(Country country) {

        countryRepository.save(country);

    }

    @Transactional

    public void updateCountry(String code, String name) throws Exception {

        Country country = findCountryByCode(code);

        country.setName(name);

        countryRepository.save(country);

    }

    @Transactional

    public void deleteCountry(String code) {

        countryRepository.deleteById(code);

    }

    @Transactional

    public List<Country> findCountriesByName(String keyword) {

        return countryRepository.findByNameContainingIgnoreCase(keyword);

    }

}

package com.cognizant.orm\_learn;

import org.springframework.boot.SpringApplication;

import org.springframework.boot.autoconfigure.SpringBootApplication;

import org.springframework.context.ApplicationContext;

import com.cognizant.orm\_learn.model.Country;

import com.cognizant.orm\_learn.service.CountryService;

@SpringBootApplication

public class OrmLearnApplication {

    public static void main(String[] args) throws Exception {

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

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

        System.out.println("== All Countries ==");

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

        System.out.println("== Add Country ==");

        Country newC = new Country();

        newC.setCode("ZZ");

        newC.setName("Zanadia");

        service.addCountry(newC);

        System.out.println("== Find ZZ ==");

        System.out.println(service.findCountryByCode("ZZ"));

        System.out.println("== Update ZZ ==");

        service.updateCountry("ZZ", "Zanzibar");

        System.out.println(service.findCountryByCode("ZZ"));

        System.out.println("== Search name contains 'land' ==");

        service.findCountriesByName("land").forEach(System.out::println);

        System.out.println("== Delete ZZ ==");

        service.deleteCountry("ZZ");

    }

}

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

spring.datasource.username=root

spring.datasource.password=root

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

spring.jpa.hibernate.ddl-auto=validate

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

logging.level.org.hibernate.SQL=debug

Output:

A screenshot of a computer program

AI-generated content may be incorrect.