**Implement services for managing Country** :

**CountryManagementApplication.java:**

package com.example;

import org.springframework.boot.SpringApplication;

import org.springframework.boot.autoconfigure.SpringBootApplication;

@SpringBootApplication

public class CountryManagementApplication {

public static void main(String[] args) {

SpringApplication.run(CountryManagementApplication.class, args);

}

@Override

public void run(String[] args) throws Exception {

countryService.printAllCountriesToConsole();

}

}

**Country.java:**

package com.example.entity;

import jakarta.persistence.Entity;

import jakarta.persistence.Id;

import jakarta.persistence.Column;

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;

}

}

**CountryController.java:**

package com.example.controller;

import com.example.entity.Country;

import com.example.service.CountryService;

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

import org.springframework.web.bind.annotation.\*;

import java.util.List;

*@RestController*

*@RequestMapping*("/countries")

public class CountryController {

*@Autowired*

private CountryService service;

*@GetMapping*("/{code}")

public Country getCountry(*@PathVariable* String code) {

return service.getCountryByCode(code);

}

*@PostMapping*

public Country addCountry(*@RequestBody* Country country) {

return service.addCountry(country);

}

*@PutMapping*

public Country updateCountry(*@RequestBody* Country country) {

return service.updateCountry(country);

}

*@DeleteMapping*("/{code}")

public void deleteCountry(*@PathVariable* String code) {

service.deleteCountry(code);

}

*@GetMapping*("/search")

public List<Country> searchCountries(*@RequestParam* String name) {

return service.searchCountriesByName(name);

}

*@GetMapping*("/console-print")

public void printCountriesToConsole() {

service.printAllCountriesToConsole();

}

}

**CountryRepository.java:**

package com.example.repository;

import com.example.entity.Country;

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

import java.util.List;

public interface CountryRepository extends JpaRepository<Country, String> {

List<Country> findByNameContainingIgnoreCase(String name);

}

**CountryService.java:**

package com.example.service;

import com.example.entity.Country;

import java.util.List;

public interface CountryService {

Country getCountryByCode(String code);

Country addCountry(Country country);

Country updateCountry(Country country);

void deleteCountry(String code);

List<Country> searchCountriesByName(String name);

void printAllCountriesToConsole();

}

**CountryServiceImpl.java:**

package com.example.service;

import com.example.entity.Country;

import com.example.repository.CountryRepository;

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

import org.springframework.stereotype.Service;

import java.util.List;

*@Service*

public class CountryServiceImpl implements CountryService {

*@Autowired*

private CountryRepository repository;

*@Override*

public void printAllCountriesToConsole() {

List<Country> countries = repository.findAll();

System.***out***.println("All countries:");

countries.forEach(c -> System.***out***.println(c.getName()));

}

*@Override*

public Country getCountryByCode(String code) {

return repository.findById(code).orElse(null);

}

*@Override*

public Country addCountry(Country country) {

return repository.save(country);

}

*@Override*

public Country updateCountry(Country country) {

return repository.save(country);

}

*@Override*

public void deleteCountry(String code) {

repository.deleteById(code);

}

*@Override*

public List<Country> searchCountriesByName(String name) {

return repository.findByCoNameContainingIgnoreCase(name);

}

}

**Application.properties:**

spring.datasource.url=jdbc:h2:mem:countrydb

spring.datasource.driverClassName=org.h2.Driver

spring.datasource.username=sa

spring.datasource.password=

spring.jpa.database-platform=org.hibernate.dialect.H2Dialect

spring.jpa.hibernate.ddl-auto=update

spring.h2.console.enabled=true

**data.sql:**

insert into country (co\_code, co\_name) values ('AF', 'Afghanistan');

insert into country (co\_code, co\_name) values ('AL', 'Albania');

insert into country (co\_code, co\_name) values ('DZ', 'Algeria');

insert into country (co\_code, co\_name) values ('AS', 'American Samoa');

insert into country (co\_code, co\_name) values ('AD', 'Andorra');

insert into country (co\_code, co\_name) values ('AO', 'Angola');

insert into country (co\_code, co\_name) values ('AI', 'Anguilla');

insert into country (co\_code, co\_name) values ('AQ', 'Antarctica');

insert into country (co\_code, co\_name) values ('AG', 'Antigua and Barbuda');

**pom.xml:**

<dependencies>

<dependency>

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

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

</dependency>

<dependency>

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

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

</dependency>

<dependency>

<groupId>com.h2database</groupId>

<artifactId>h2</artifactId>

<scope>runtime</scope>

</dependency>

</dependencies>



