**WEEK-3**

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

**Hands On-5: Implement services for managing Country**

**1.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>CountryServiceSpringBoot</artifactId>

    <version>1.0.0</version>

    <parent>

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

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

        <version>3.2.5</version>

        <relativePath/>

    </parent>

    <properties>

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

    </properties>

    <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>

    <build>

        <plugins>

            <plugin>

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

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

            </plugin>

        </plugins>

    </build>

</project>

**2.Country.java:**

package com.example.country;

import jakarta.persistence.Entity;

import jakarta.persistence.Id;

@Entity

public class Country {

    @Id

    private String coCode;

    private String coName;

    public String getCoCode() { return coCode; }

    public void setCoCode(String coCode) { this.coCode = coCode; }

    public String getCoName() { return coName; }

    public void setCoName(String coName) { this.coName = coName; }

}

**3.CountryController.java:**

package com.example.country;

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

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

import java.util.List;

import java.util.Optional;

@RestController

@RequestMapping("/countries")

public class CountryController {

    @Autowired

    private CountryService service;

    @GetMapping("/{code}")

    public Optional<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/{name}")

    public List<Country> searchCountry(@PathVariable String name) {

        return service.searchByPartialName(name);

    }

}

**4.CountryRepository.java:**

package com.example.country;

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

import java.util.List;

public interface CountryRepository extends JpaRepository<Country, String> {

    List<Country> findByCoNameContainingIgnoreCase(String partialName);

}

**5.CountryService.java:**

package com.example.country;

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 CountryRepository repository;

    public Optional<Country> getCountryByCode(String code) {

        return repository.findById(code);

    }

    public Country addCountry(Country country) {

        return repository.save(country);

    }

    public Country updateCountry(Country country) {

        return repository.save(country);

    }

    public void deleteCountry(String code) {

        repository.deleteById(code);

    }

    public List<Country> searchByPartialName(String name) {

        return repository.findByCoNameContainingIgnoreCase(name);

    }

}

**6.CountryServiceSpringBootApplication.java:**

package com.example.country;

import org.springframework.boot.SpringApplication;

import org.springframework.boot.autoconfigure.SpringBootApplication;

@SpringBootApplication

public class CountryServiceSpringBootApplication {

    public static void main(String[] args) {

        SpringApplication.run(CountryServiceSpringBootApplication.class, args);

    }

}

**7.application.properties:**

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

spring.datasource.driverClassName=org.h2.Driver

spring.datasource.username=sa

spring.datasource.password=

spring.jpa.hibernate.ddl-auto=none

spring.sql.init.mode=always

spring.h2.console.enabled=true

**8.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');

**9.schema.sql:**

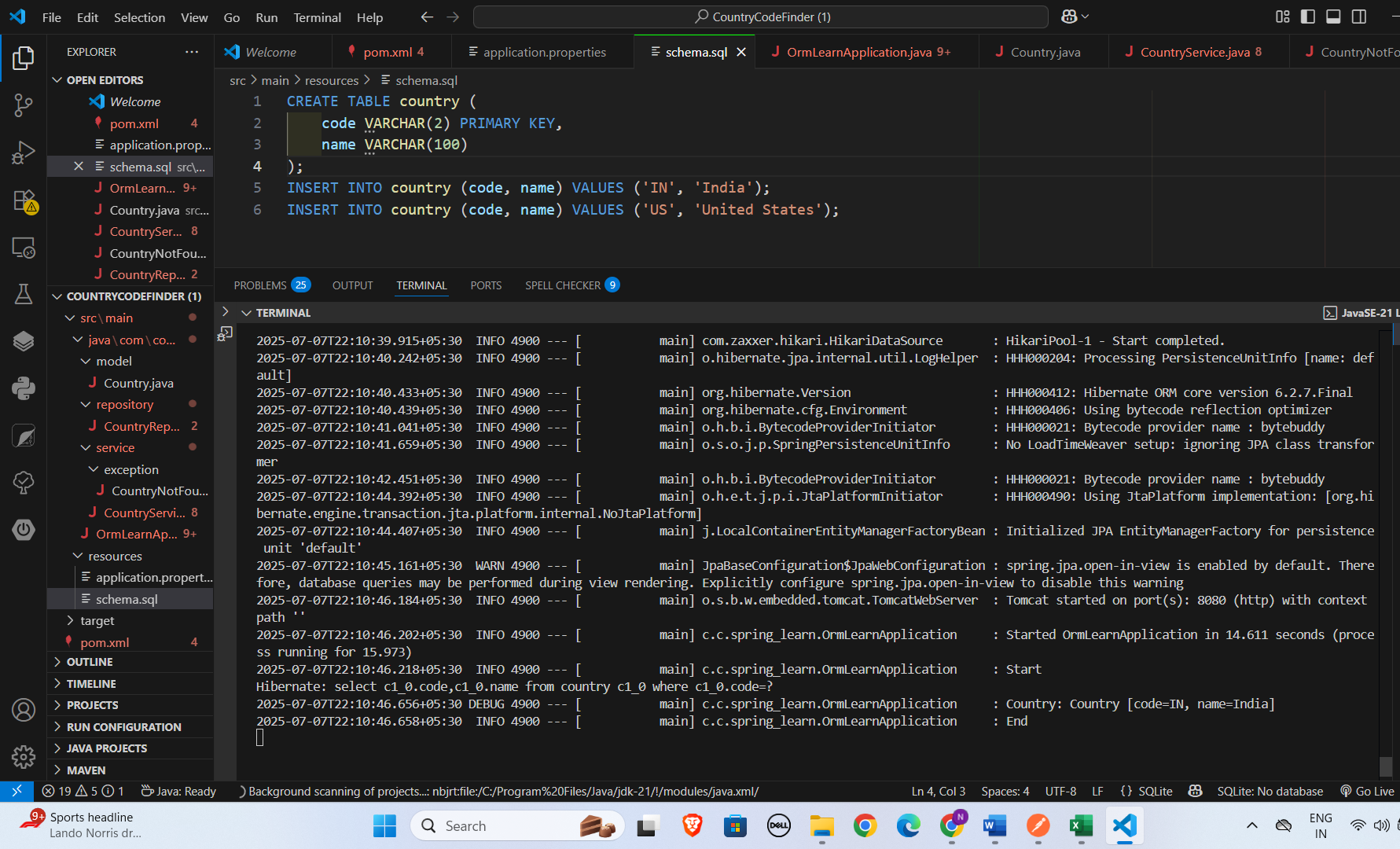
CREATE TABLE country (

    co\_code VARCHAR(5) PRIMARY KEY,

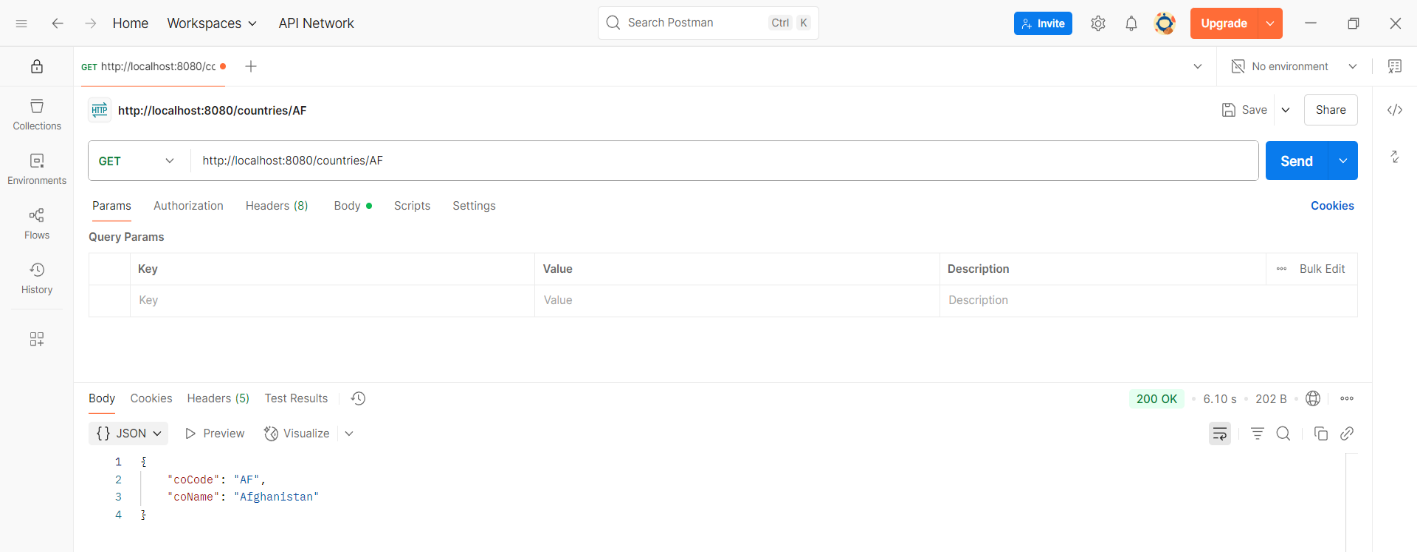
    co\_name VARCHAR(255)

);

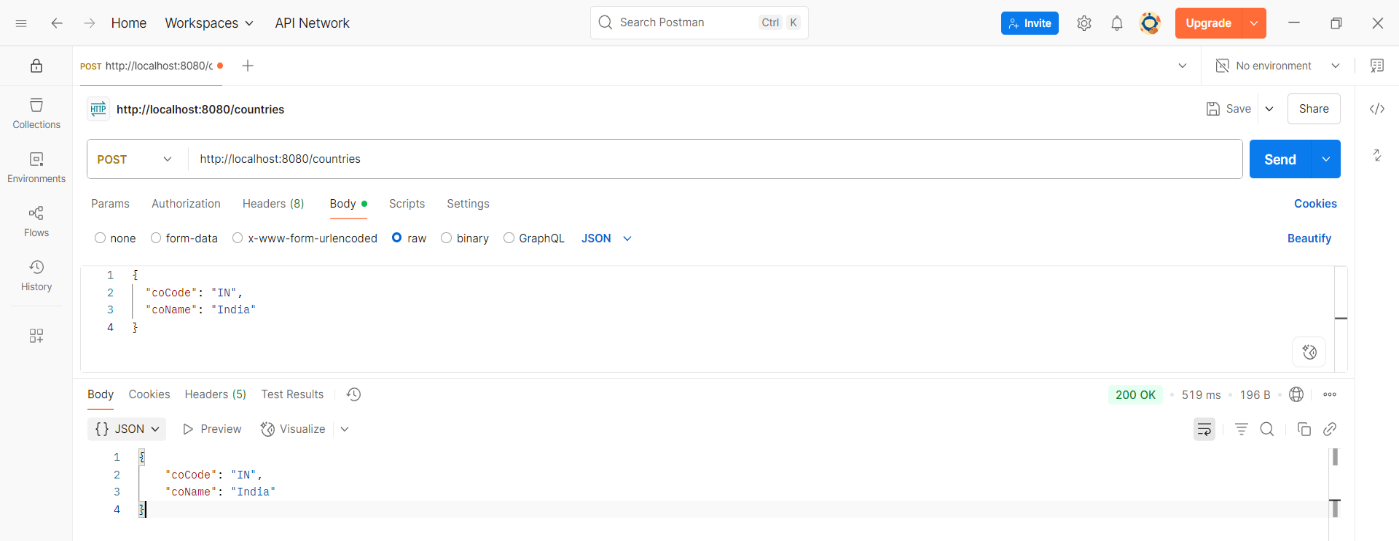
**Output:**



GET /countries/AF:



POST /countries:



PUT /countries:

