com.example.countrysystem

│

├── controller // Handles API requests

│ └── CountryController.java

│

── service // Contains service interfaces and logic

│ ── CountryService.java

│ ── CountryServiceImpl.java

│

── repository // JPA repository interface

│ └── CountryRepository.java

│── entity // JPA Entity

│ └── Country.java

│ ountrysystemApplication.java

Pom.xml

<?xml version="1.0" encoding="UTF-8"?>

<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 https://maven.apache.org/xsd/maven-4.0.0.xsd">

<modelVersion>4.0.0</modelVersion>

<parent>

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

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

<version>3.5.3</version>

<relativePath/> <!-- lookup parent from repository -->

</parent>

<groupId>com.example</groupId>

<artifactId>countrysystem</artifactId>

<version>0.0.1-SNAPSHOT</version>

<name>countrysystem</name>

<description>Demo project for Spring Boot</description>

<url/>

<licenses>

<license/>

</licenses>

<developers>

<developer/>

</developers>

<scm>

<connection/>

<developerConnection/>

<tag/>

<url/>

</scm>

<properties>

<java.version>21</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>org.springframework.boot</groupId>

<artifactId>spring-boot-devtools</artifactId>

<scope>runtime</scope>

<optional>true</optional>

</dependency>

<dependency>

<groupId>com.mysql</groupId>

<artifactId>mysql-connector-j</artifactId>

<scope>runtime</scope>

</dependency>

<dependency>

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

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

<scope>test</scope>

</dependency>

</dependencies>

<build>

<plugins>

<plugin>

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

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

</plugin>

</plugins>

</build>

</project>

Country.java

package com.example.countrysystem.entity;

import jakarta.persistence.Entity;

import jakarta.persistence.Id;

import jakarta.persistence.Table;

*@Entity*

*@Table*(name = "country")

public class Country {

*@Id*

private String coCode;

private String coName;

// Getters and Setters

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; }

}

CountryService.java

package com.example.countrysystem.service;

import com.example.countrysystem.entity.Country;

import java.util.List;

import java.util.Optional;

public interface CountryService {

Country addCountry(Country country);

List<Country> getAllCountries();

Optional<Country> getCountryByCode(String code);

Country updateCountry(String code, Country country);

void deleteCountry(String code);

List<Country> searchByPartialName(String name);

}

CountryController.java

package com.example.countrysystem.controller;

import com.example.countrysystem.entity.Country;

import com.example.countrysystem.service.CountryService;

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 countryService;

*@PostMapping*

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

return countryService.addCountry(country);

}

*@GetMapping*

public List<Country> getAllCountries() {

return countryService.getAllCountries();

}

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

public Optional<Country> getCountryByCode(*@PathVariable* String code) {

return countryService.getCountryByCode(code);

}

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

public Country updateCountry(*@PathVariable* String code, *@RequestBody* Country country) {

return countryService.updateCountry(code, country);

}

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

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

countryService.deleteCountry(code);

}

*@GetMapping*("/search/{name}")

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

return countryService.searchByPartialName(name);

}

}

CountrysystemApplication.java

package com.example.countrysystem;

import org.springframework.boot.SpringApplication;

import org.springframework.boot.autoconfigure.SpringBootApplication;

*@SpringBootApplication*

public class CountrysystemApplication {

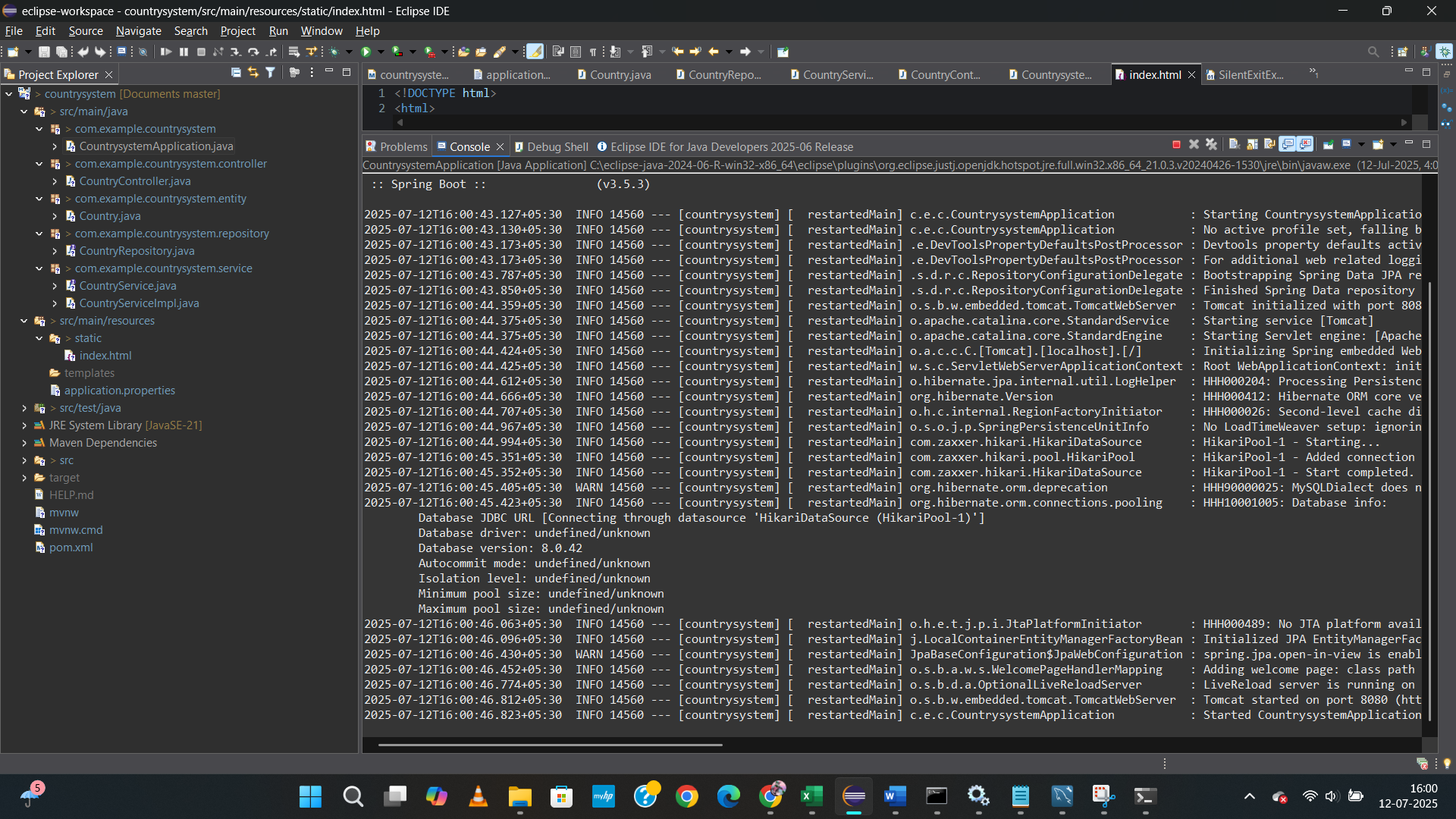
public static void main(String[] args) {

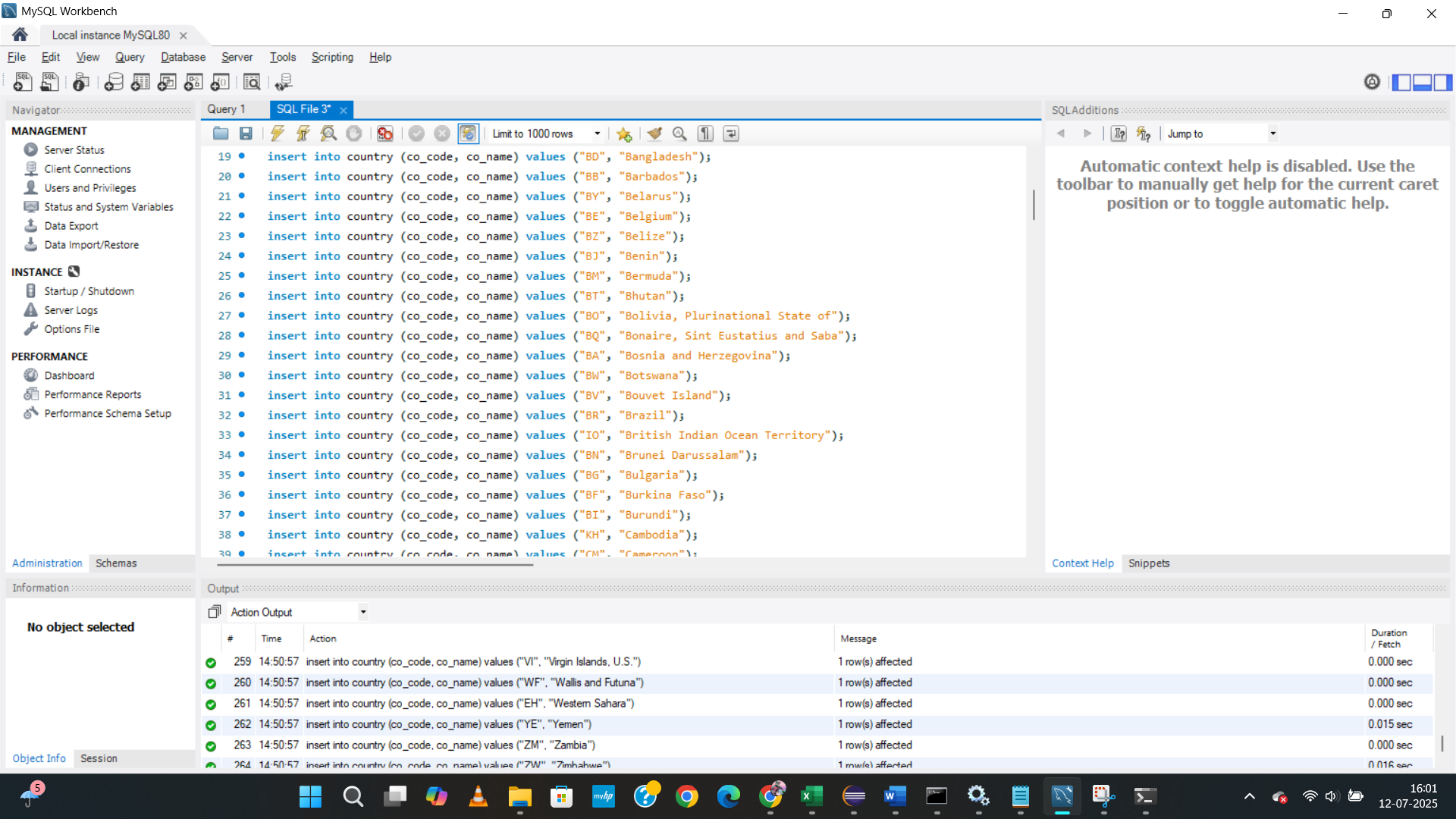
SpringApplication.*run*(CountrysystemApplication.class, args);

}

}

OUTPUTS:





A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer screen

AI-generated content may be incorrect.

SEARCHING using get method

A screenshot of a computer

AI-generated content may be incorrect.

Search by code

A screenshot of a computer

AI-generated content may be incorrect.