**Exercise 2**

**TASK:**

A simple Spring Boot REST API that:

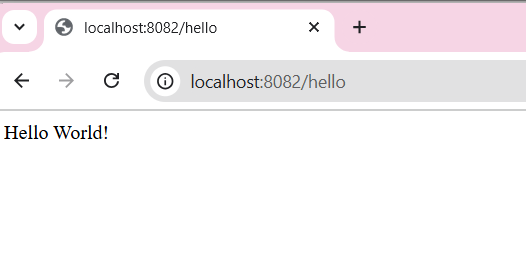
* Returns a basic "Hello World"
* Returns a list of countries
* Returns details of a specific country by its country code

**Task 1: Setup Hello World API**

Goal: Create a simple REST endpoint /hello that returns "Hello World"

**What to Do:**

* Create a HelloController class
* Use @RestController and @GetMapping("/hello")
* Return "Hello World!" as plain text



**Task 2: Create a Country Model Class**

Goal: Define a simple class to represent a country

**What to Do:**

* Create Country.java with 2 fields: code (e.g., "IN") and name (e.g., "India")
* Add constructor, getters, setters

**Task 3: Build Country Service (In-Memory Data)**

Goal: Store a few countries in memory and return them

**What to Do:**

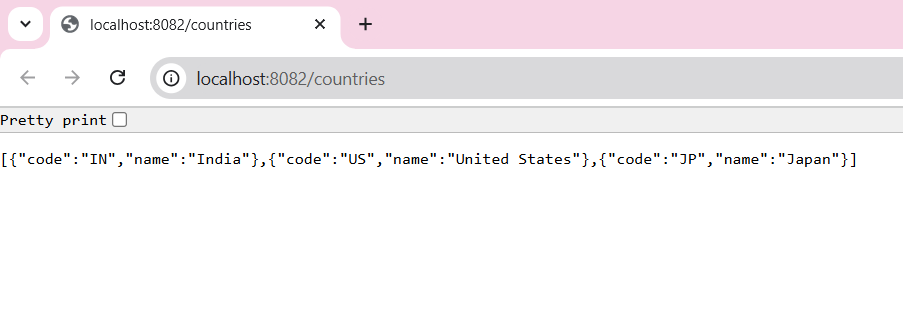
* Create CountryService.java
* Annotate with @Service
* Add 3 countries to a list in the constructor
* Write methods:
  + List<Country> getAllCountries()
  + Country getCountryByCode(String code)

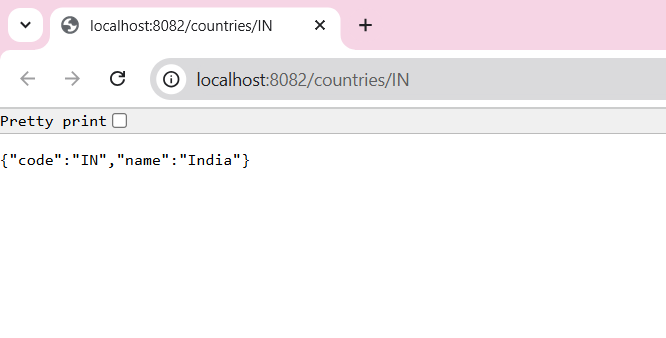
**Task 4: Create CountryController**

Goal: Expose 2 REST APIs

**What to Do:**

* Create CountryController.java
* Annotate with @RestController and @RequestMapping("/countries")
* Inject CountryService using @Autowired
* Add 2 endpoints:
  + GET /countries → returns all countries
  + GET /countries/{code} → returns country by code



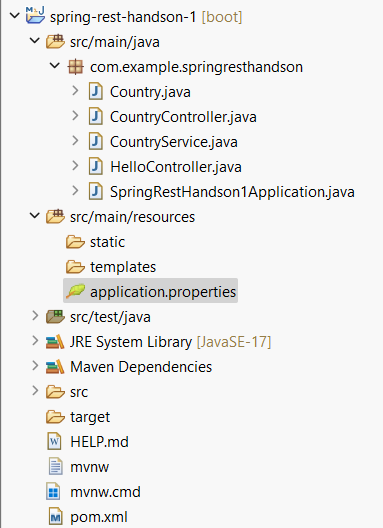


STEP 1: Create Spring Boot Project in Eclipse and add Dependencies to it

**Add Dependencies**

* Search and add: **Spring Web**

STEP 2:Create a structured class



**CODING:**

**SpringRestHandson1Application.java(MAIN CLASS)**

**package** com.example.springresthandson;

**import** org.springframework.boot.SpringApplication;

**import** org.springframework.boot.autoconfigure.SpringBootApplication;

@SpringBootApplication

**public** **class** SpringRestHandson1Application {

**public** **static** **void** main(String[] args) {

SpringApplication.*run*(SpringRestHandson1Application.**class**, args);

}

}

**Country.java**

package com.example.springresthandson;

public class Country {

private String code;

private String name;

public Country() {}

public Country(String code, String name) {

this.code = code;

this.name = name;

}

public String getCode() {

return code;

}

public String getName() {

return name;

}

public void setCode(String code) {

this.code = code;

}

public void setName(String name) {

this.name = name;

}

}

**CountryController.java**

**package** com.example.springresthandson;

**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 countryService;

@GetMapping

**public** List<Country> getAllCountries() {

**return** countryService.getAllCountries();

}

@GetMapping("/{code}")

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

**return** countryService.getCountryByCode(code);

}

}

**CountryService.java**

**package** com.example.springresthandson;

**import** org.springframework.stereotype.Service;

**import** java.util.ArrayList;

**import** java.util.List;

@Service

**public** **class** CountryService {

**private** List<Country> countries = **new** ArrayList<>();

**public** CountryService() {

countries.add(**new** Country("IN", "India"));

countries.add(**new** Country("US", "United States"));

countries.add(**new** Country("JP", "Japan"));

}

**public** List<Country> getAllCountries() {

**return** countries;

}

**public** Country getCountryByCode(String code) {

**for** (Country country : countries) {

**if** (country.getCode().equalsIgnoreCase(code)) {

**return** country;

}

}

**return** **null**;

}

}

**HelloController.java**

**package** com.example.springresthandson;

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

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

@RestController

**public** **class** HelloController {

@GetMapping("/hello")

**public** String sayHello() {

**return** "Hello World!";

}

}

**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>spring-rest-handson-1</artifactId>

<version>0.0.1-SNAPSHOT</version>

<name>spring-rest-handson-1</name>

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

<url/>

<licenses>

<license/>

</licenses>

<developers>

<developer/>

</developers>

<scm>

<connection/>

<developerConnection/>

<tag/>

<url/>

</scm>

<properties>

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

</properties>

<dependencies>

<dependency>

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

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

</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>

**Application.properties**

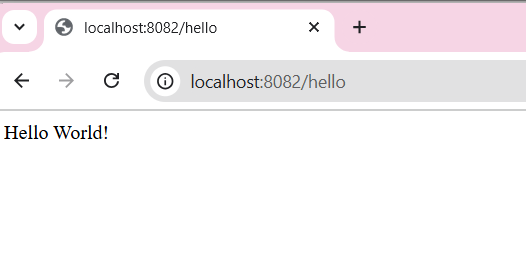
spring.application.name=spring-rest-handson-1

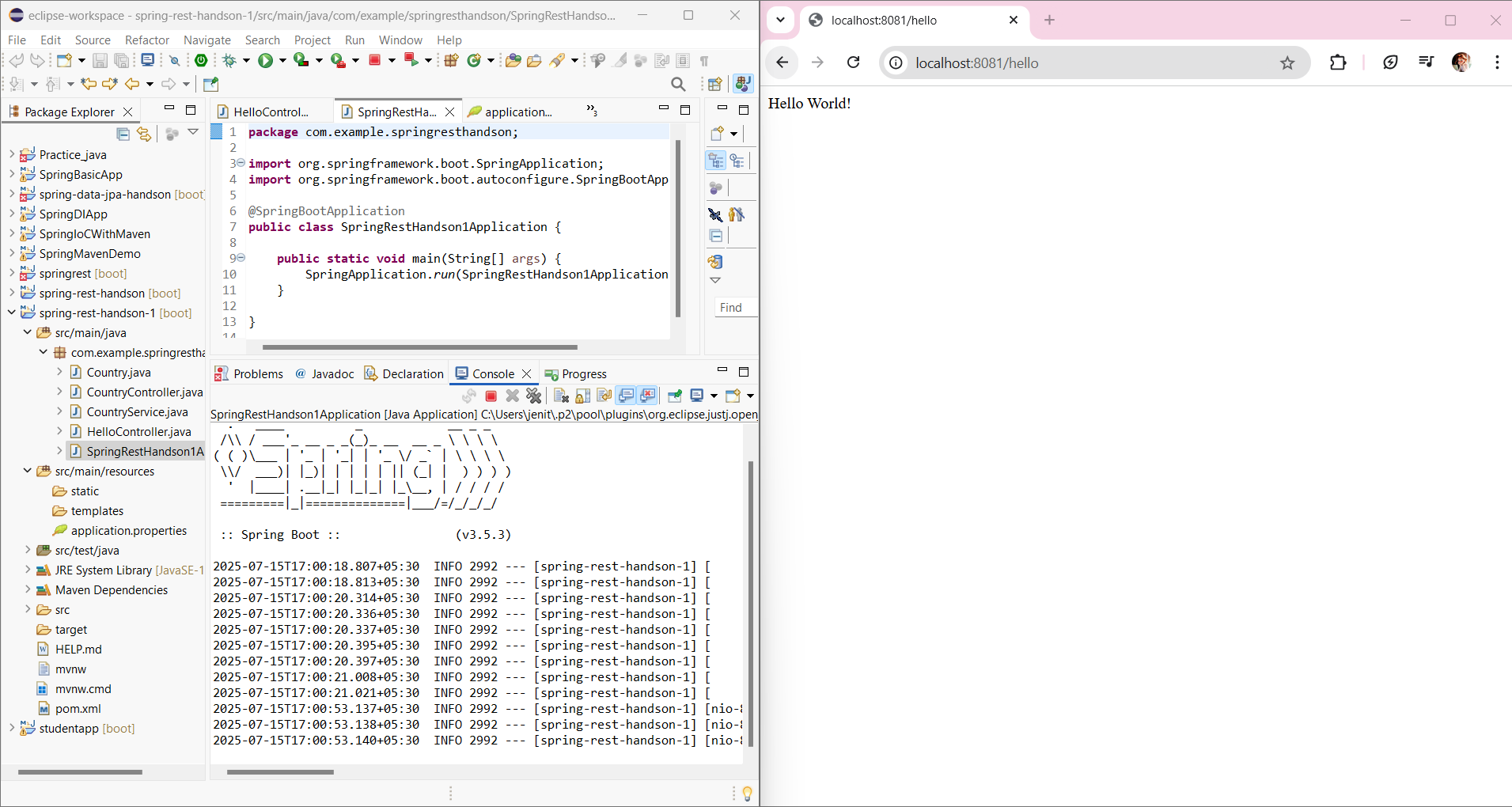
server.port=8082

**OUTPUTS:**

**TASK 1: SHOWING “Hello World”**

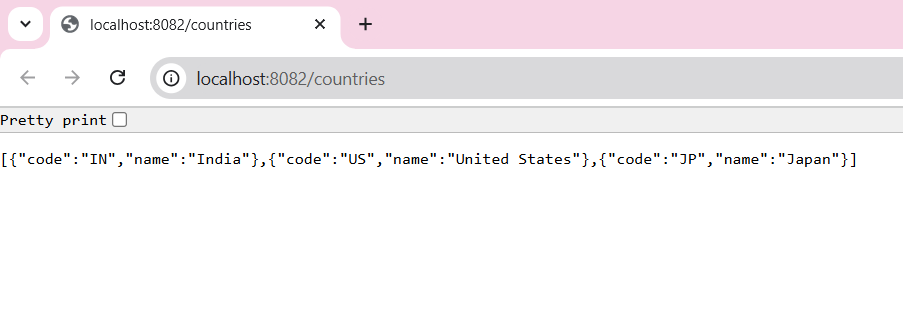
**search url:** <http://localhost:8082/hello>

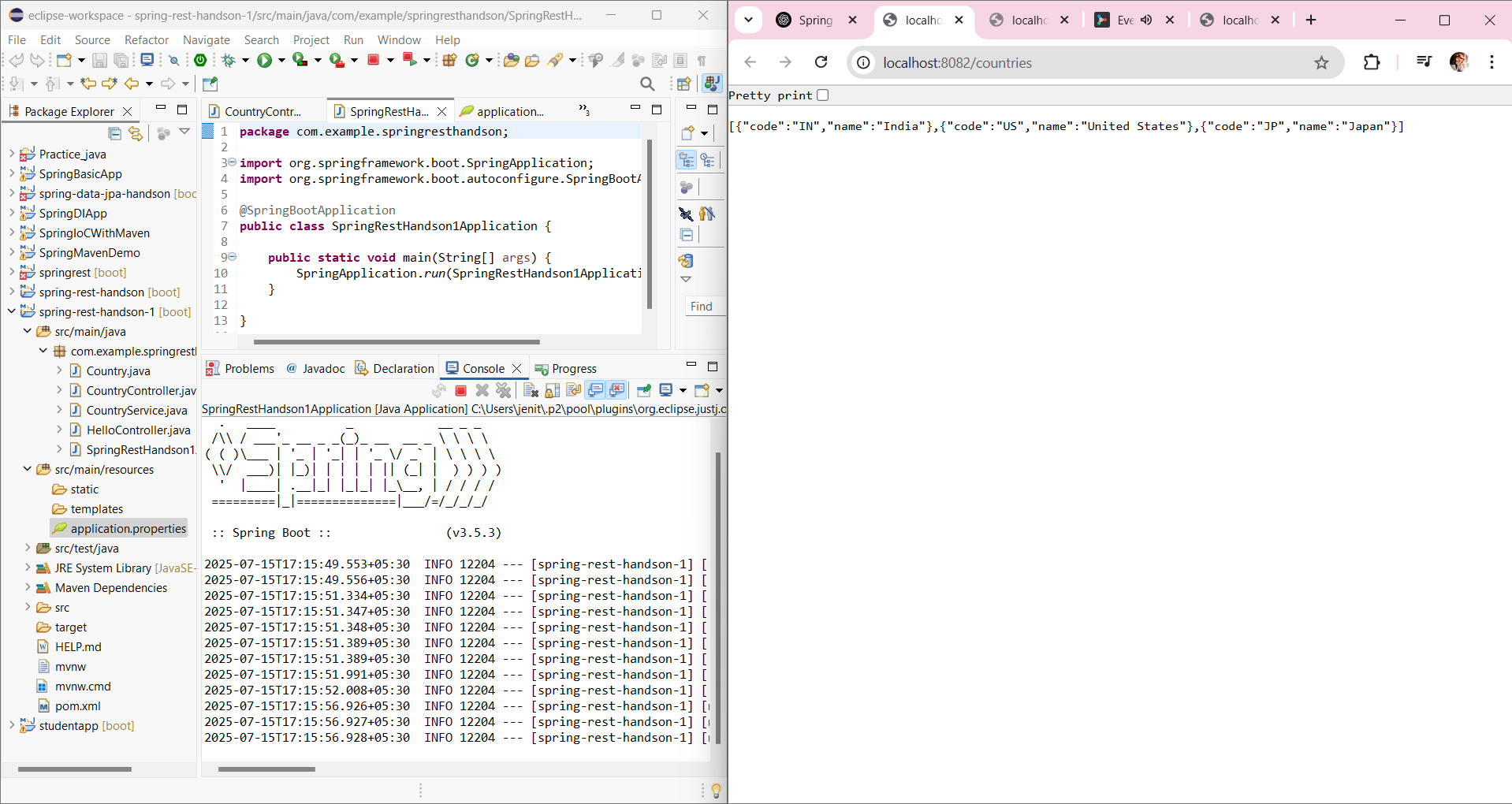


****

**TASK 2:SHOWING “All Countries”**

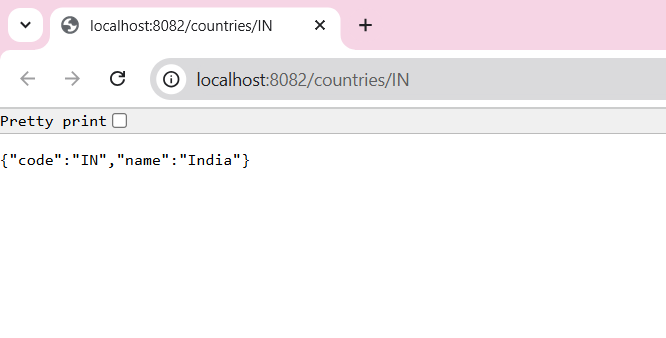
**search url:** <http://localhost:8082/countries>

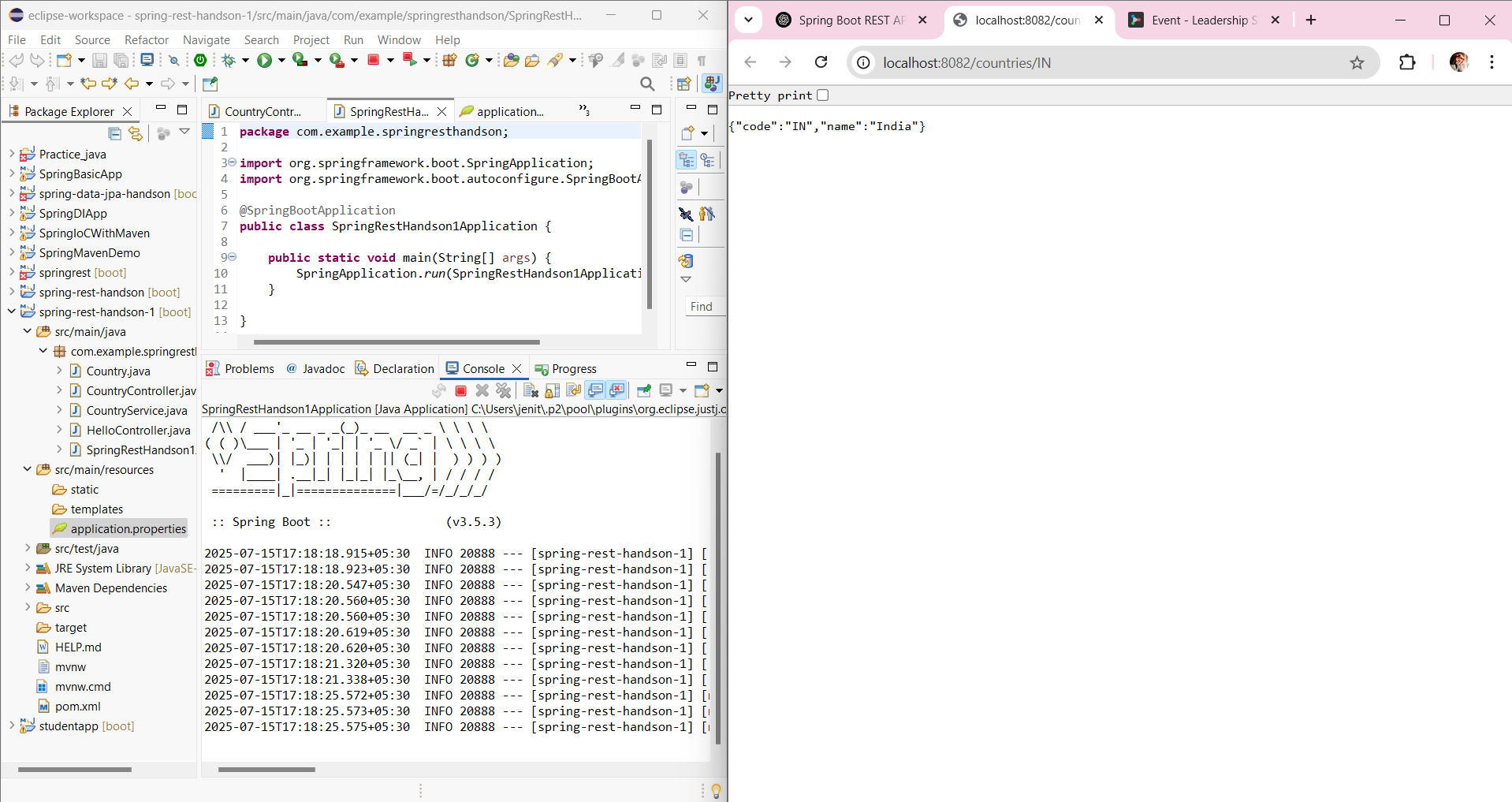


****

**TASK 3:SHOWING “Specific Country by code”**

**search url:** <http://localhost:8082/countries/IN>



****