**spring-rest-handson**

**Exercise 1:** **Create a Spring Web Project using Maven**

**CODING:**

**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.jenitta</groupId>

<artifactId>springrest</artifactId>

<version>0.0.1-SNAPSHOT</version>

<packaging>jar</packaging>

<parent>

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

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

<version>3.1.2</version> <!-- or latest available version -->

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

</parent>

<dependencies>

<!-- Spring Web for REST API -->

<dependency>

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

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

</dependency>

</dependencies>

<build>

<plugins>

<!-- Spring Boot Plugin -->

<plugin>

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

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

</plugin>

</plugins>

</build>

</project>

**SpringrestApplication.java**

**package** com.jenitta.springrest;

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

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

@SpringBootApplication

**public** **class** SpringrestApplication {

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

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

}

}

**HelloController.java**

**package** com.jenitta.springrest.controller;

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

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

@RestController

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

@GetMapping("/hello")

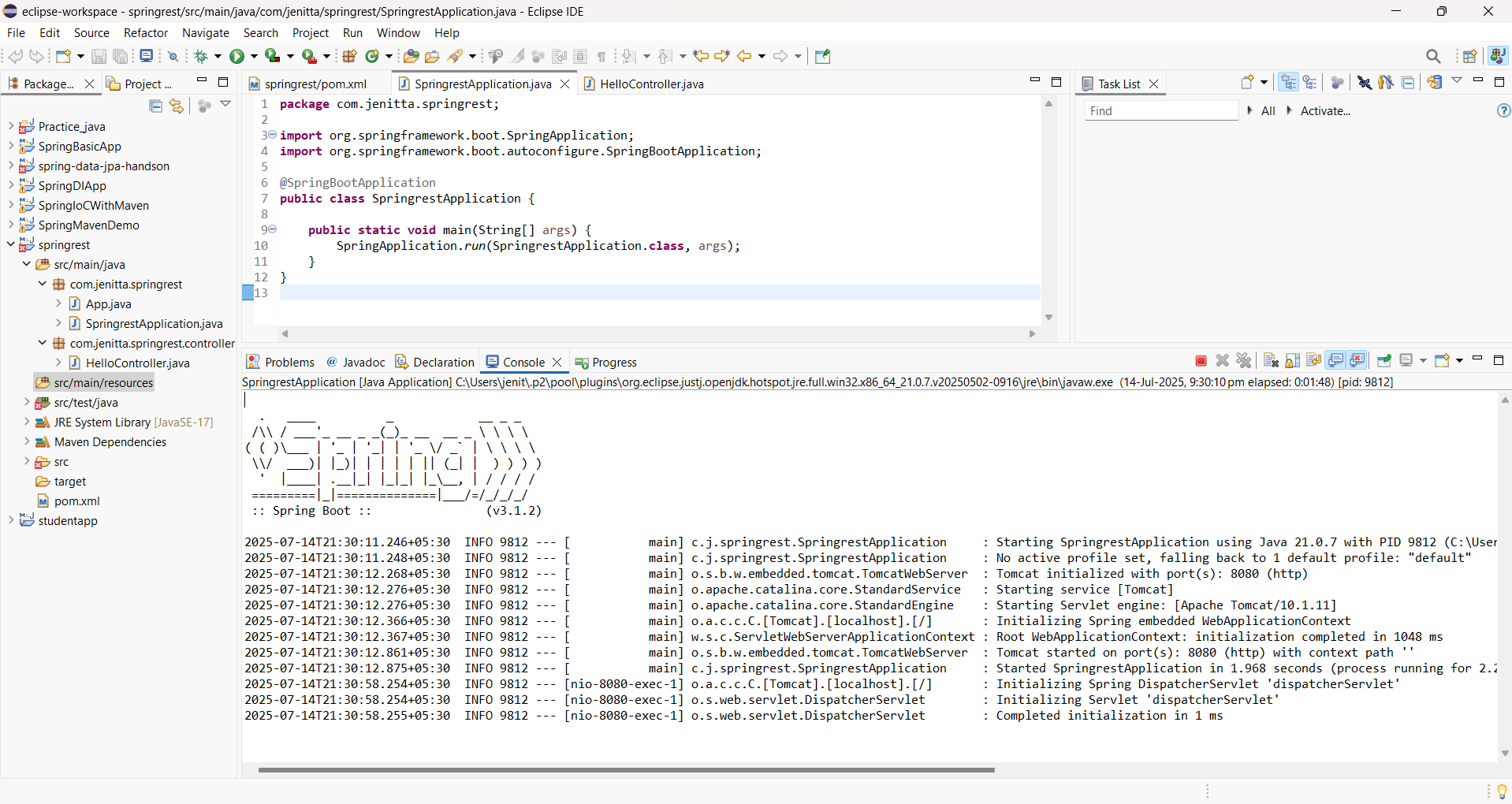
**public** String sayHello() {

**return** "Hello, Jenitta! Welcome to Spring REST!";

}

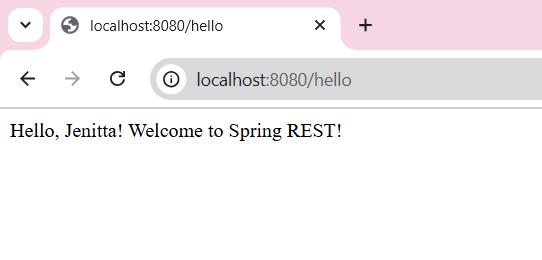
}

**OUTPUT:1**

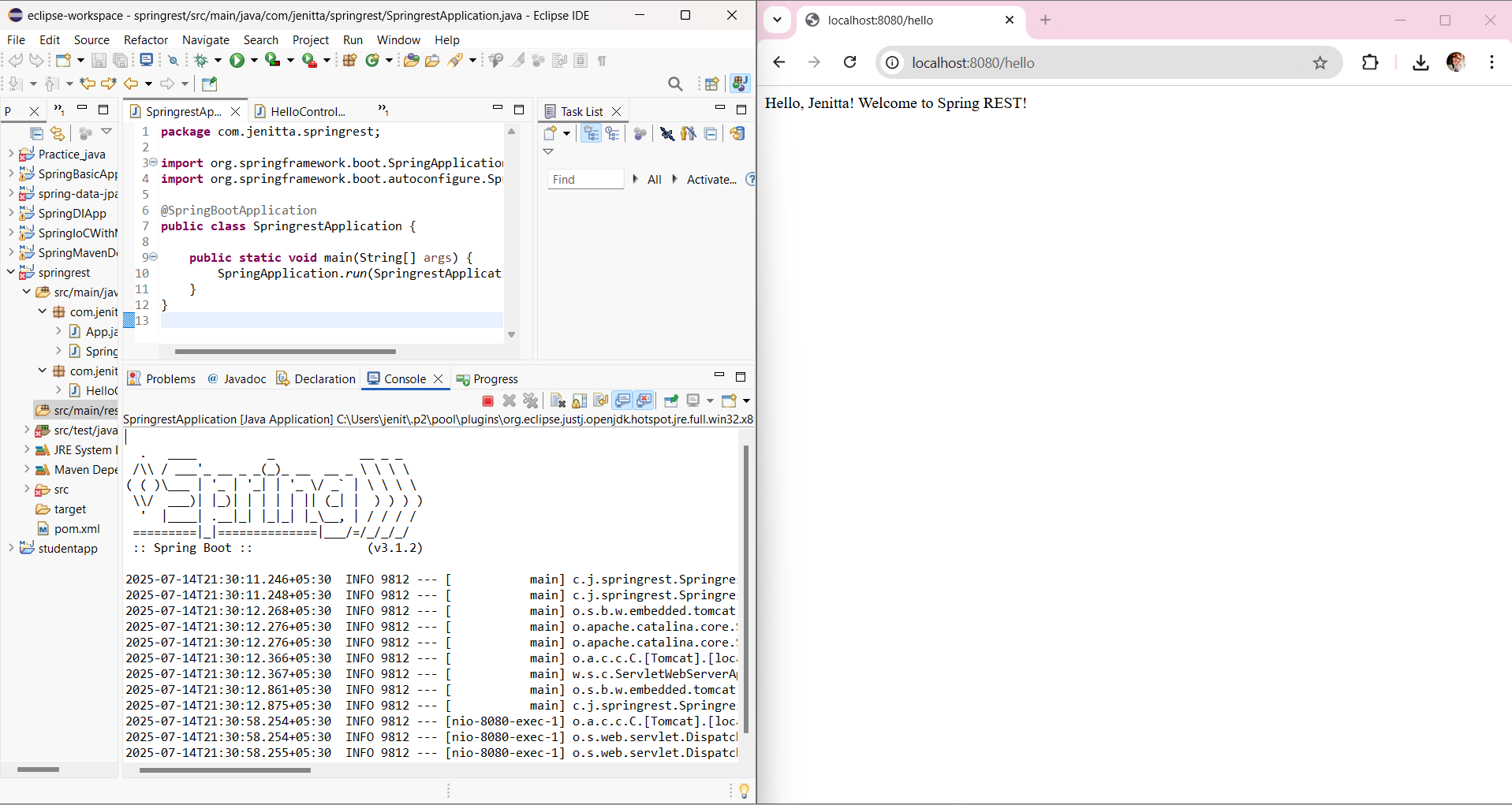
****

**OUTPUT:2**

Open Browser and search the wehosting url **--🡪** [**http://localhost:8080/hello**](http://localhost:8080/hello)

****

**OVERALL OUTPUT:**

****

**2.Spring Core – Load Country from Spring Configuration XML**

**TASK:**

Build a Spring Boot 3.2.1 REST API in Eclipse that:

* Loads a Country bean from an XML config file
* Returns it in a REST endpoint: GET /country

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

**Dependencies to add:**

* Spring Web
* Spring Context

STEP 2: Project Structure Created

* Create the XML Config File(src/main/resources/country-config.xml)

**country-config.xml**

<!-- country-config.xml -->

<beans xmlns="http://www.springframework.org/schema/beans"

xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"

xsi:schemaLocation="http://www.springframework.org/schema/beans

http://www.springframework.org/schema/beans/spring-beans.xsd">

<bean id="country" class="com.example.demo.Country">

<property name="code" value="IN"/>

<property name="name" value="India"/>

</bean>

</beans>

* Create the Class files(com/example/demo)

**Country.java**

package com.example.demo;

public class Country {

private String code;

private String name;

// Getters and Setters

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 + "]";

}

}

**CountryController.java**

package com.example.demo;

import org.springframework.context.ApplicationContext;

import org.springframework.context.support.ClassPathXmlApplicationContext;

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

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

@RestController

public class CountryController {

@GetMapping("/country")

public Country getCountry() {

ApplicationContext context = new ClassPathXmlApplicationContext("country-config.xml");

return (Country) context.getBean("country");

}

}

**SpringRestHandsonApplication.java**

package com.example.demo;

import org.springframework.boot.SpringApplication;

import org.springframework.boot.autoconfigure.SpringBootApplication;

@SpringBootApplication

public class SpringRestHandsonApplication {

public static void main(String[] args) {

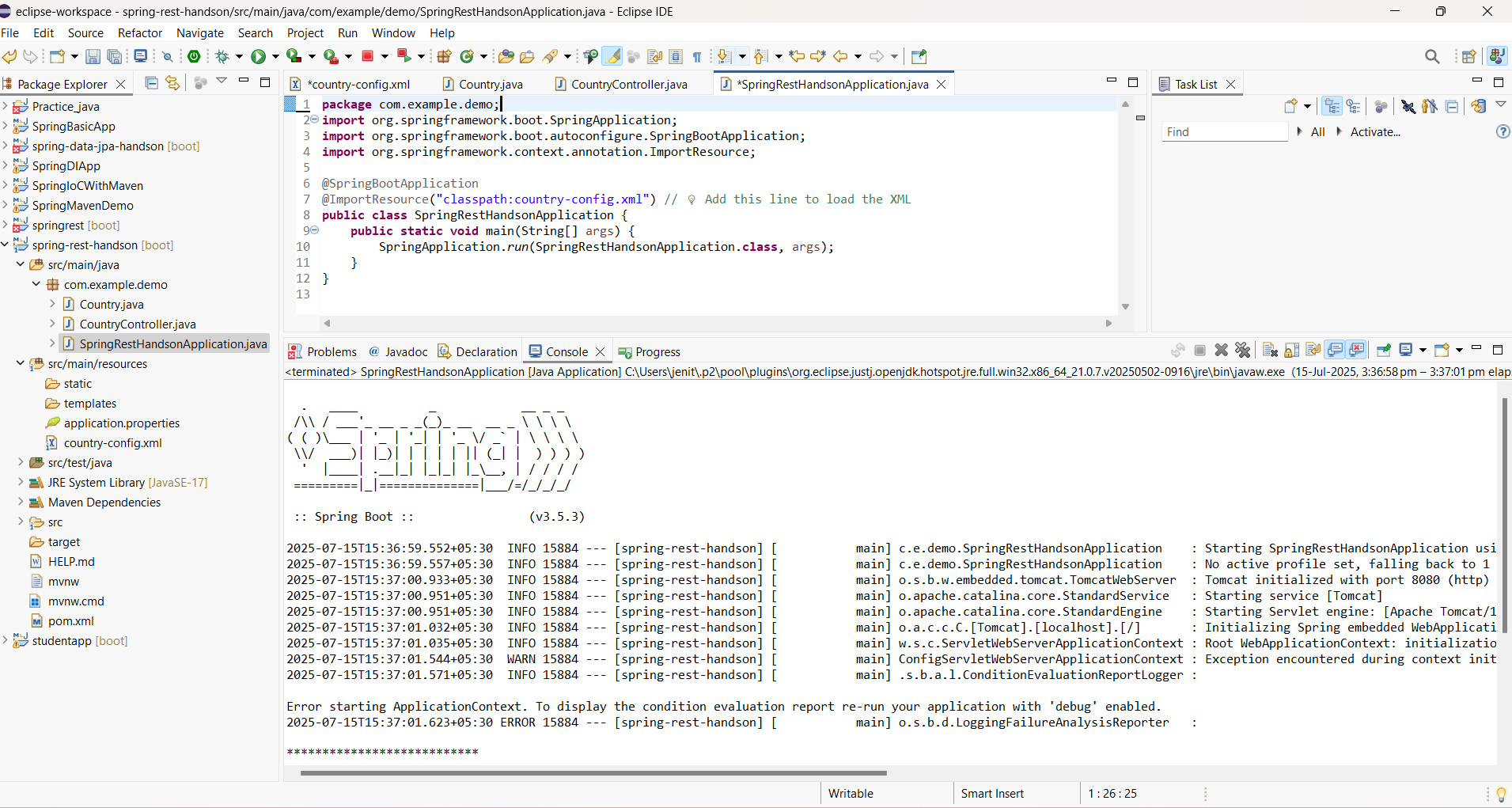
SpringApplication.run(SpringRestHandsonApplication.class, args);

}

}

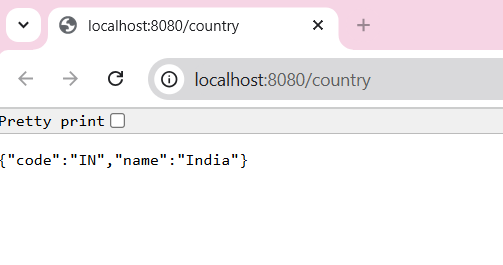
STEP 3: Run the App (SpringRestHandsonApplication.java

)in Eclipse



STEP 8: Test the REST API

1.  Open your browser or Postman
2.  Go to: [**http://localhost:8080/country**](http://localhost:8080/country)



**OVERALL OUTPUT:**

