**WEEK 4 – SPRING REST USING SPRING BOOT 3**

**EXERCISE 1: CREATE A SPRING WEB PROJECT USING MAVEN**

**STEP 1:** Generate and Set Up the Project using Spring Initializer.

**STEP 2:** Add Dependencies. Import the project into eclipse.

SpringLearnApplication.java

package com.cognizant.spring\_learn;

import org.slf4j.Logger;

import org.slf4j.LoggerFactory;

import org.springframework.boot.SpringApplication;

import org.springframework.boot.autoconfigure.SpringBootApplication;

*@SpringBootApplication*

public class SpringLearnApplication {

private static final Logger ***LOGGER*** = LoggerFactory.*getLogger*(SpringLearnApplication.class);

public static void main(String[] args) {

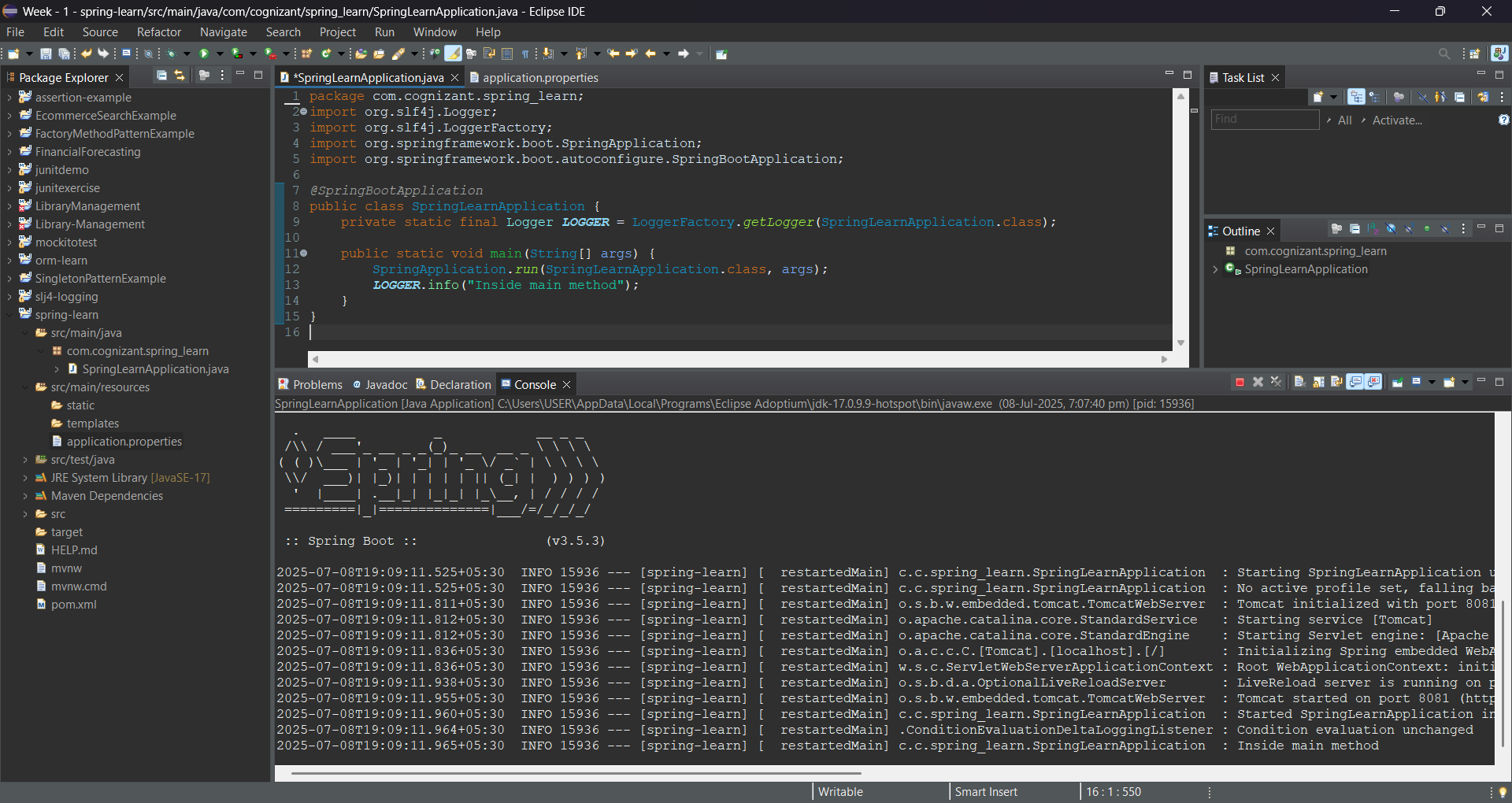
SpringApplication.*run*(SpringLearnApplication.class, args);

***LOGGER***.info("Inside main method");

}

}

**OUTPUT:**



**EXERCISE 2: SPRING CORE – LOAD COUNTRY FROM SPRING CONFIGURATION XML**

**STEP 1:** Generate and Set Up the Project using Spring Initializer.

**STEP 2:** Add Dependencies. Import the project into eclipse.

**SpringLearnApplication.java**

package com.cognizant.spring\_learn;

import org.slf4j.Logger;

import org.slf4j.LoggerFactory;

import org.springframework.context.ApplicationContext;

import org.springframework.context.support.ClassPathXmlApplicationContext;

import org.springframework.boot.SpringApplication;

import org.springframework.boot.autoconfigure.SpringBootApplication;

@SpringBootApplication

public class SpringLearnApplication {

private static final Logger LOGGER = LoggerFactory.getLogger(SpringLearnApplication.class);

public static void main(String[] args) {

SpringApplication.run(SpringLearnApplication.class, args);

displayCountry();

}

@SuppressWarnings("resource")

public static void displayCountry() {

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

Country country = (Country) context.getBean("country", Country.class);

LOGGER.debug("Country : {}", country.toString());

}

}

Country.java:

package com.cognizant.spring\_learn;

import org.slf4j.Logger;

import org.slf4j.LoggerFactory;

public class Country {

private static final Logger ***LOGGER*** = LoggerFactory.*getLogger*(Country.class);

private String code;

private String name;

public Country() {

***LOGGER***.debug("Inside Country Constructor.");

}

public String getCode() {

***LOGGER***.debug("Inside getCode.");

return code;

}

public void setCode(String code) {

***LOGGER***.debug("Inside setCode.");

this.code = code;

}

public String getName() {

***LOGGER***.debug("Inside getName.");

return name;

}

public void setName(String name) {

***LOGGER***.debug("Inside setName.");

this.name = name;

}

*@Override*

public String toString() {

return "Country [code=" + code + ", name=" + name + "]";

}

}

Country.xml:

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

<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.cognizant.spring\_learn.Country">

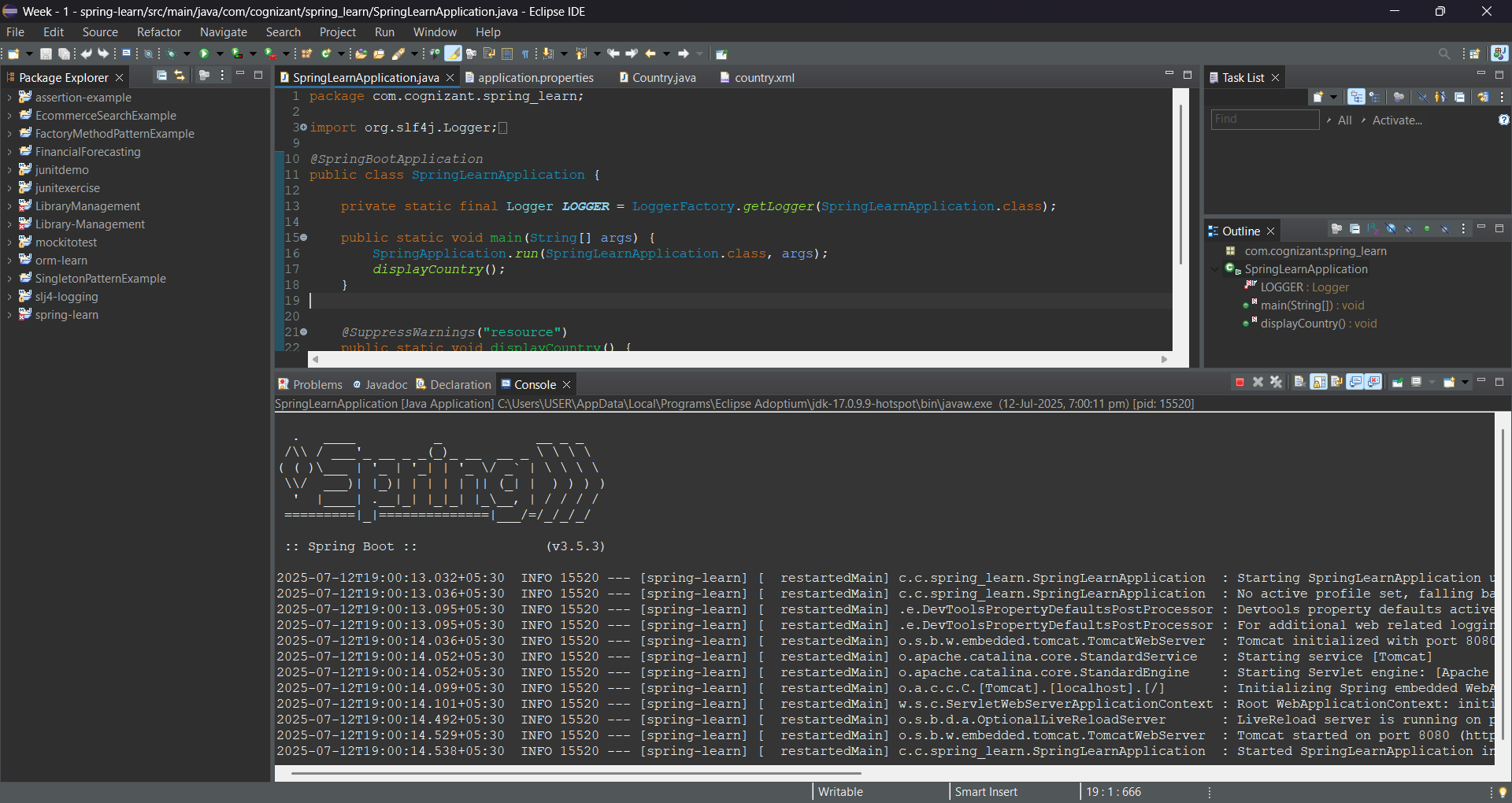
<property name="code" value="IN"/>

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

</bean>

</beans>

Output:



**EXERCISE 3: HELLO WORLD RESTFUL WEB SERVICE**

**STEP 1:** Generate and Set Up the Project using Spring Initializer.

**STEP 2:** Add Dependencies. Import the project into eclipse.

**SpringLearnApplication.java**

package com.cognizant.spring\_learn;

import org.springframework.boot.SpringApplication;

import org.springframework.boot.autoconfigure.SpringBootApplication;

*@SpringBootApplication*

public class SpringLearnApplication {

public static void main(String[] args) {

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

}

}

HelloController.java:

package com.cognizant.spring\_learn.controller;

import org.slf4j.Logger;

import org.slf4j.LoggerFactory;

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

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

*@RestController*

public class HelloController {

private static final Logger ***LOGGER*** = LoggerFactory.*getLogger*(HelloController.class);

*@GetMapping*("/hello")

public String sayHello() {

***LOGGER***.info("START: sayHello()");

String response = "Hello World!!";

***LOGGER***.info("END: sayHello()");

return response;

}

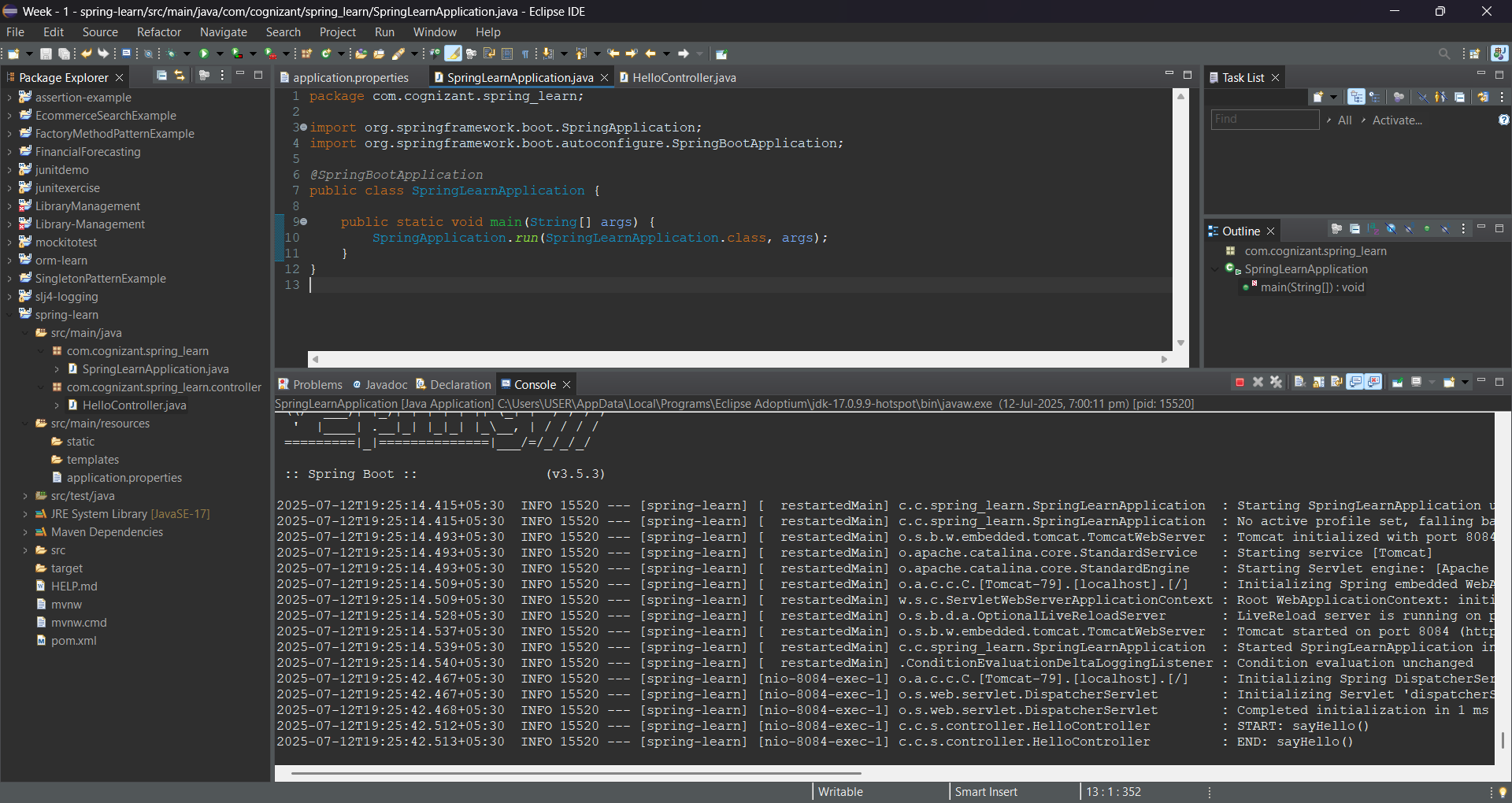
}

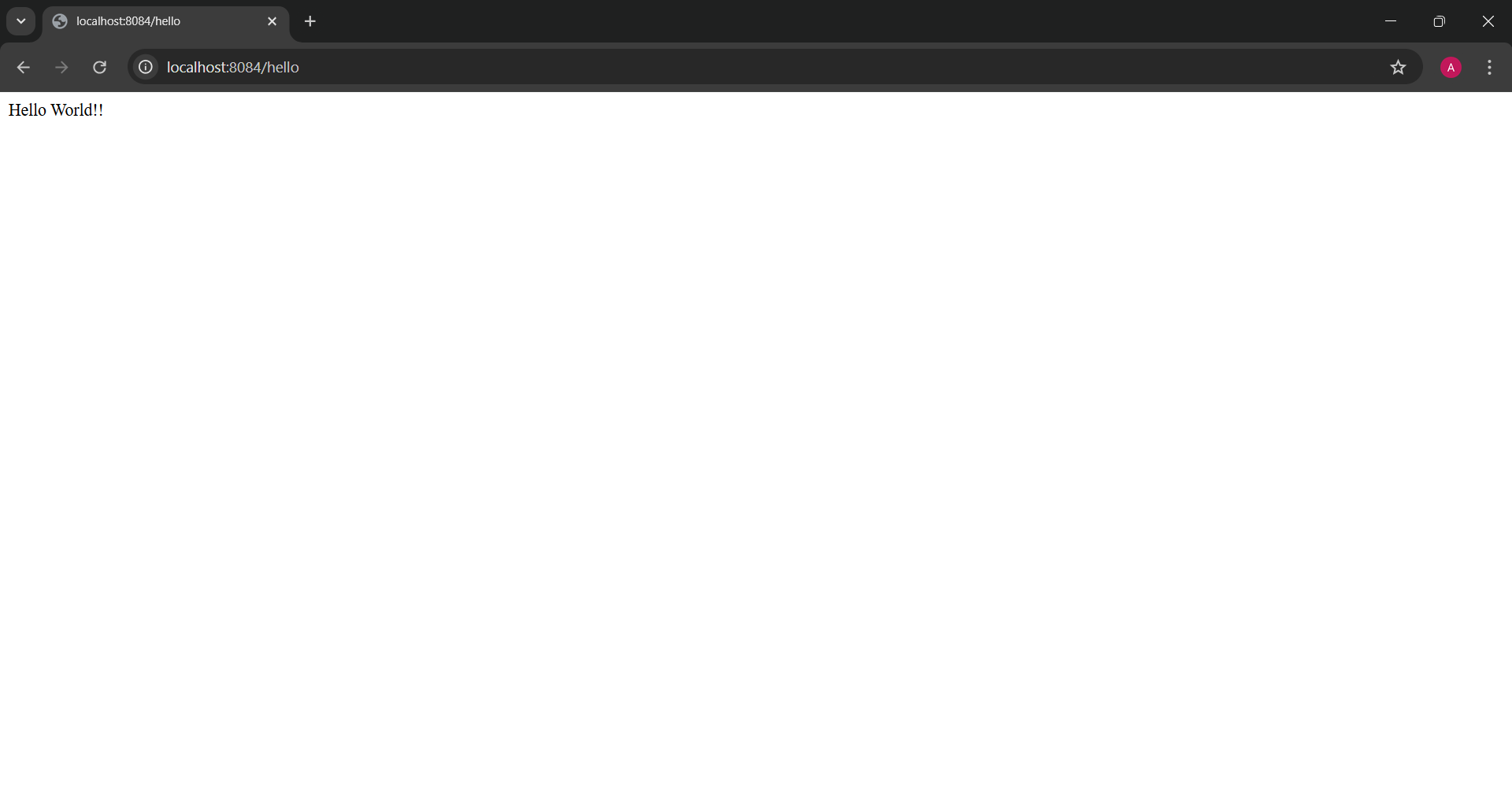
application.properties:

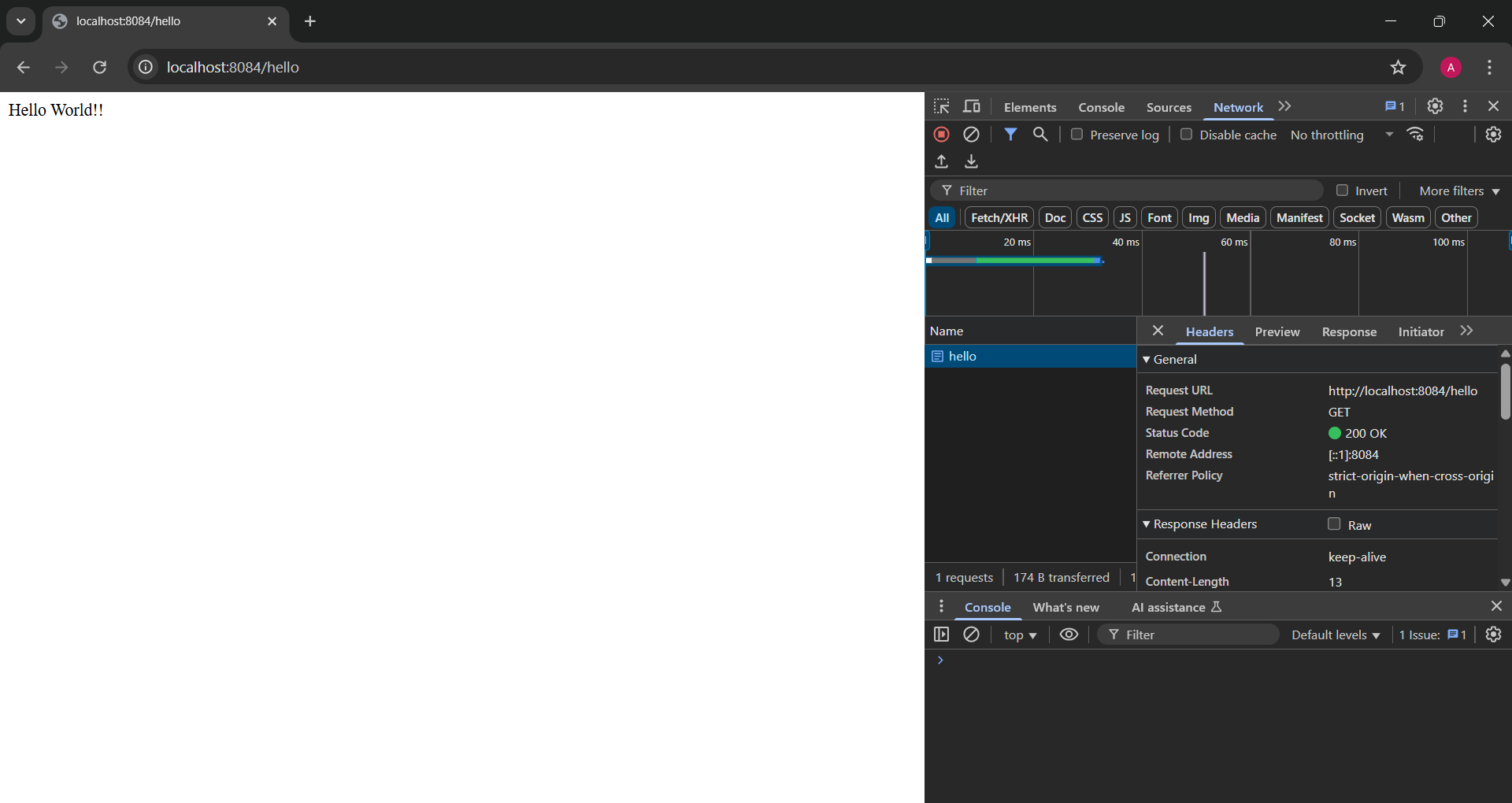
spring.application.name=spring-learn

server.port = 8084

OUTPUT:







**EXERCISE 4: REST COUNTRY WEB SERVICE**

**STEP 1:** Generate and Set Up the Project using Spring Initializer.

**STEP 2:** Add Dependencies. Import the project into eclipse.

SpringLearnApplication.java

package com.cognizant.spring\_learn;

import org.springframework.boot.SpringApplication;

import org.springframework.boot.autoconfigure.SpringBootApplication;

*@SpringBootApplication*

public class SpringLearnApplication {

public static void main(String[] args) {

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

}

}

Country.xml

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

<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.cognizant.spring\_learn.Country">

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

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

</bean>

</beans>

CountryController.java

package com.cognizant.spring\_learn.controller;

import com.cognizant.spring\_learn.Country;

import org.slf4j.Logger;

import org.slf4j.LoggerFactory;

import org.springframework.context.support.ClassPathXmlApplicationContext;

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

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

*@RestController*

public class CountryController {

private static final Logger ***LOGGER*** = LoggerFactory.*getLogger*(CountryController.class);

*@RequestMapping*("/country")

public Country getCountryIndia() {

***LOGGER***.info("START: getCountryIndia()");

try (ClassPathXmlApplicationContext context = new ClassPathXmlApplicationContext("country.xml")) {

Country country = context.getBean("country", Country.class);

***LOGGER***.info("END: getCountryIndia()");

return country;

}

}

}

Country.java:

package com.cognizant.spring\_learn;

public class Country {

private String code;

private String name;

public Country() {

System.***out***.println("Inside Country Constructor.");

}

public String getCode() {

System.***out***.println("Getting code");

return code;

}

public void setCode(String code) {

System.***out***.println("Setting code");

this.code = code;

}

public String getName() {

System.***out***.println("Getting name");

return name;

}

public void setName(String name) {

System.***out***.println("Setting name");

this.name = name;

}

*@Override*

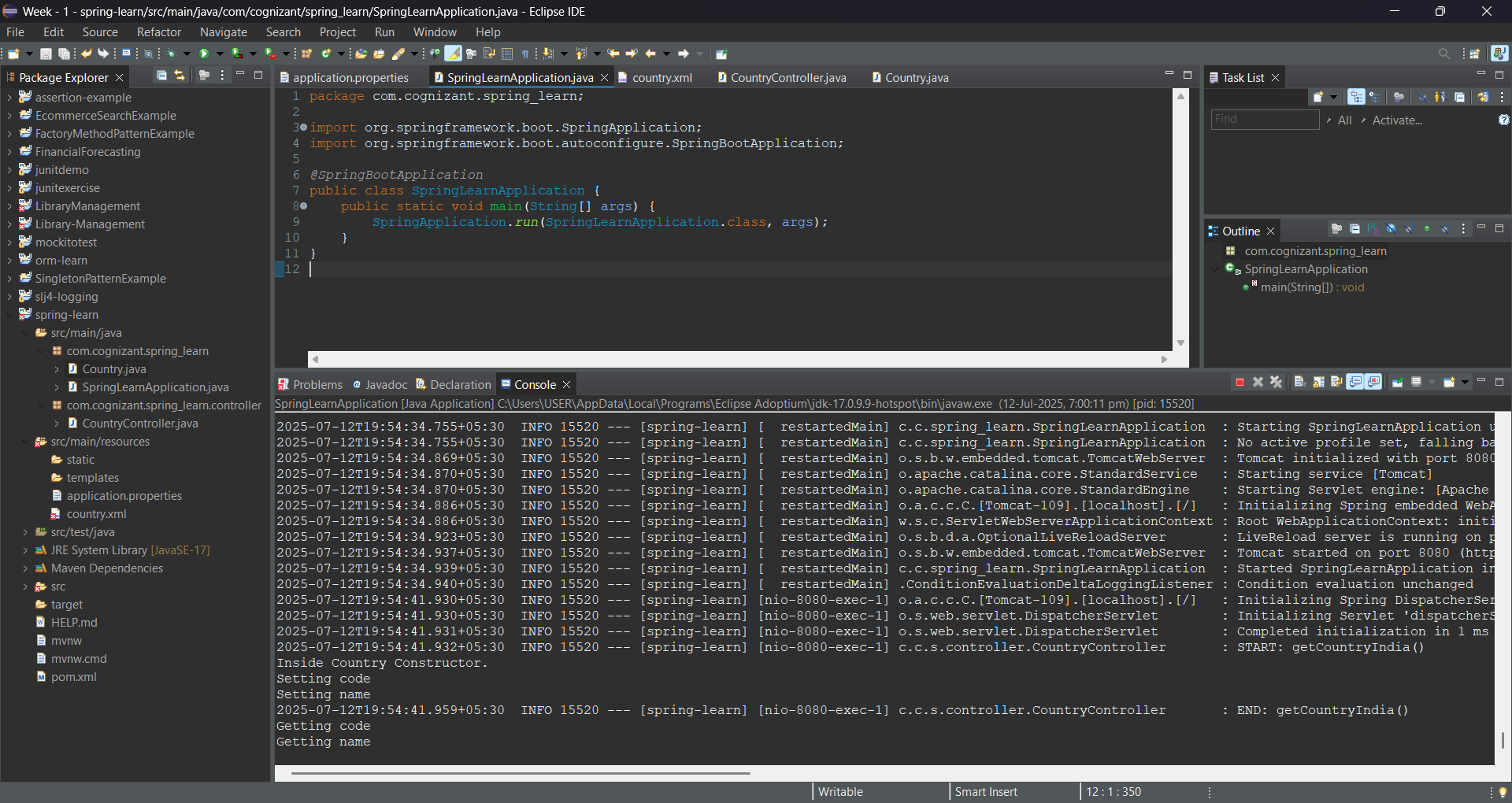
public String toString() {

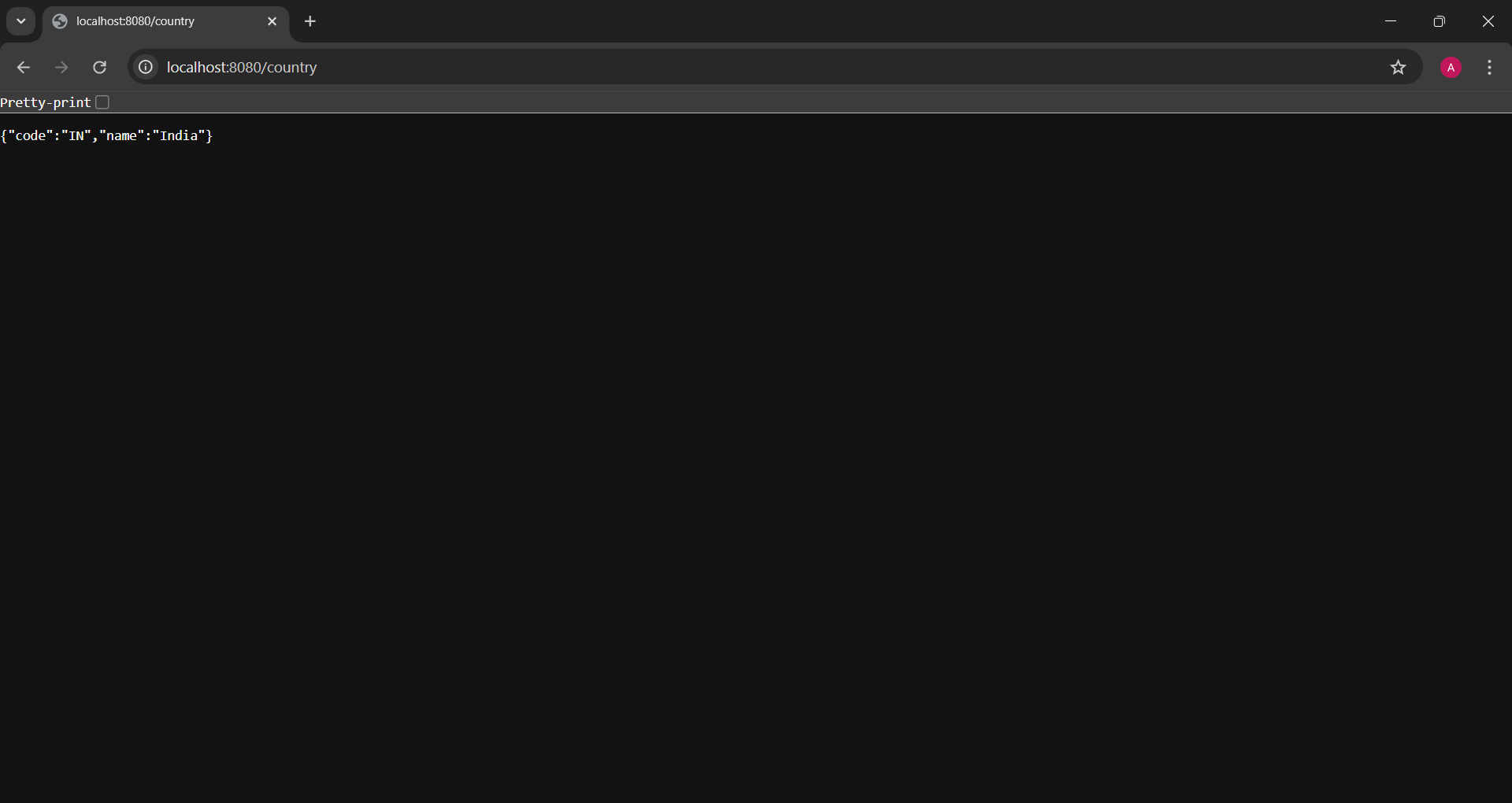
return "Country [code=" + code + ", name=" + name + "]";

}

}

OUTPUT:





**EXERCISE 5: REST – GET COUNTRY BASED ON COUNTRY CODE**

**STEP 1:** Generate and Set Up the Project using Spring Initializer.

**STEP 2:** Add Dependencies. Import the project into eclipse.

SpringLearnApplication.java

package com.cognizant.spring\_learn;

import org.springframework.boot.SpringApplication;

import org.springframework.boot.autoconfigure.SpringBootApplication;

*@SpringBootApplication*

public class SpringLearnApplication {

public static void main(String[] args) {

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

}

}

country.xml

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

<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">

<!-- Country Beans -->

<bean id="countryIN" class="com.cognizant.spring\_learn.Country">

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

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

</bean>

<bean id="countryUS" class="com.cognizant.spring\_learn.Country">

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

<property name="name" value="United States"/>

</bean>

<bean id="countryJP" class="com.cognizant.spring\_learn.Country">

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

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

</bean>

<bean id="countryDE" class="com.cognizant.spring\_learn.Country">

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

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

</bean>

<bean id="countryFR" class="com.cognizant.spring\_learn.Country">

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

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

</bean>

<bean id="countryAU" class="com.cognizant.spring\_learn.Country">

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

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

</bean>

<bean id="countryBR" class="com.cognizant.spring\_learn.Country">

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

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

</bean>

<bean id="countryCA" class="com.cognizant.spring\_learn.Country">

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

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

</bean>

<bean id="countryCN" class="com.cognizant.spring\_learn.Country">

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

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

</bean>

<bean id="countryGB" class="com.cognizant.spring\_learn.Country">

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

<property name="name" value="United Kingdom"/>

</bean>

<!-- Country List -->

<bean id="countryList" class="java.util.ArrayList">

<constructor-arg>

<list>

<ref bean="countryIN"/>

<ref bean="countryUS"/>

<ref bean="countryJP"/>

<ref bean="countryDE"/>

<ref bean="countryFR"/>

<ref bean="countryAU"/>

<ref bean="countryBR"/>

<ref bean="countryCA"/>

<ref bean="countryCN"/>

<ref bean="countryGB"/>

</list>

</constructor-arg>

</bean>

</beans>

CountryController.java:

package com.cognizant.spring\_learn.controller;

import com.cognizant.spring\_learn.Country;

import com.cognizant.spring\_learn.service.CountryService;

import org.slf4j.Logger;

import org.slf4j.LoggerFactory;

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

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

*@RestController*

public class CountryController {

private static final Logger ***LOGGER*** = LoggerFactory.*getLogger*(CountryController.class);

*@Autowired*

private CountryService countryService;

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

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

***LOGGER***.info("Start getCountry()");

Country country = countryService.getCountry(code);

***LOGGER***.info("End getCountry()");

return country;

}

}

Country.java:

package com.cognizant.spring\_learn;

public class Country {

private String code;

private String name;

public Country() {

System.***out***.println("Inside Country Constructor.");

}

public String getCode() {

System.***out***.println("Inside getCode");

return code;

}

public void setCode(String code) {

System.***out***.println("Inside setCode");

this.code = code;

}

public String getName() {

System.***out***.println("Inside getName");

return name;

}

public void setName(String name) {

System.***out***.println("Inside setName");

this.name = name;

}

*@Override*

public String toString() {

return "Country [code=" + code + ", name=" + name + "]";

}

}

CountryService.java:

package com.cognizant.spring\_learn.service;

import com.cognizant.spring\_learn.Country;

import org.springframework.context.support.ClassPathXmlApplicationContext;

import org.springframework.stereotype.Service;

import java.util.List;

*@Service*

public class CountryService {

*@SuppressWarnings*("unchecked")

public Country getCountry(String code) {

try (ClassPathXmlApplicationContext context = new ClassPathXmlApplicationContext("country.xml")) {

List<Country> countryList = (List<Country>) context.getBean("countryList");

return countryList.stream()

.filter(c -> c.getCode().equalsIgnoreCase(code))

.findFirst()

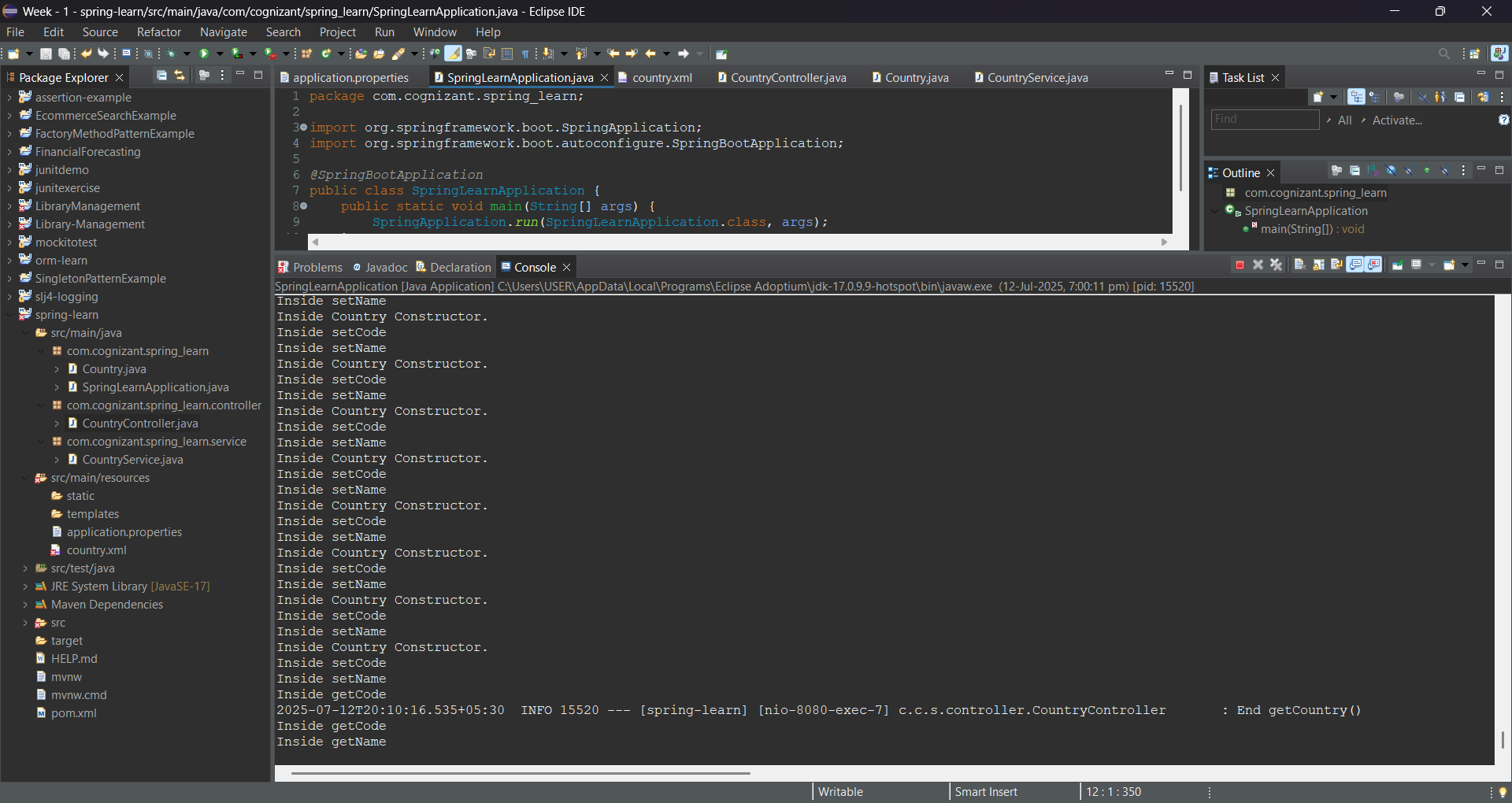
.orElse(null); // You may throw a custom exception here if needed

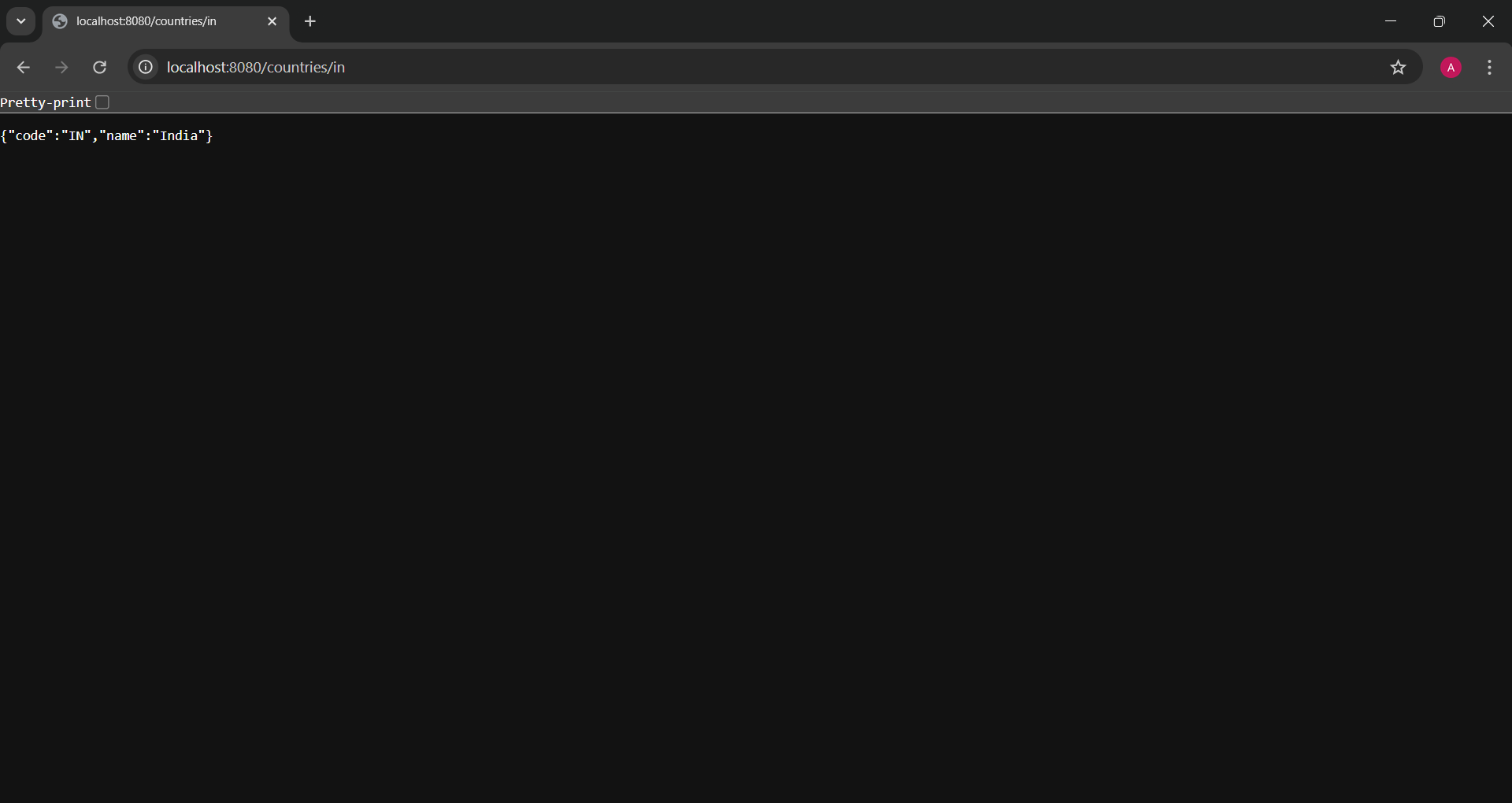
}

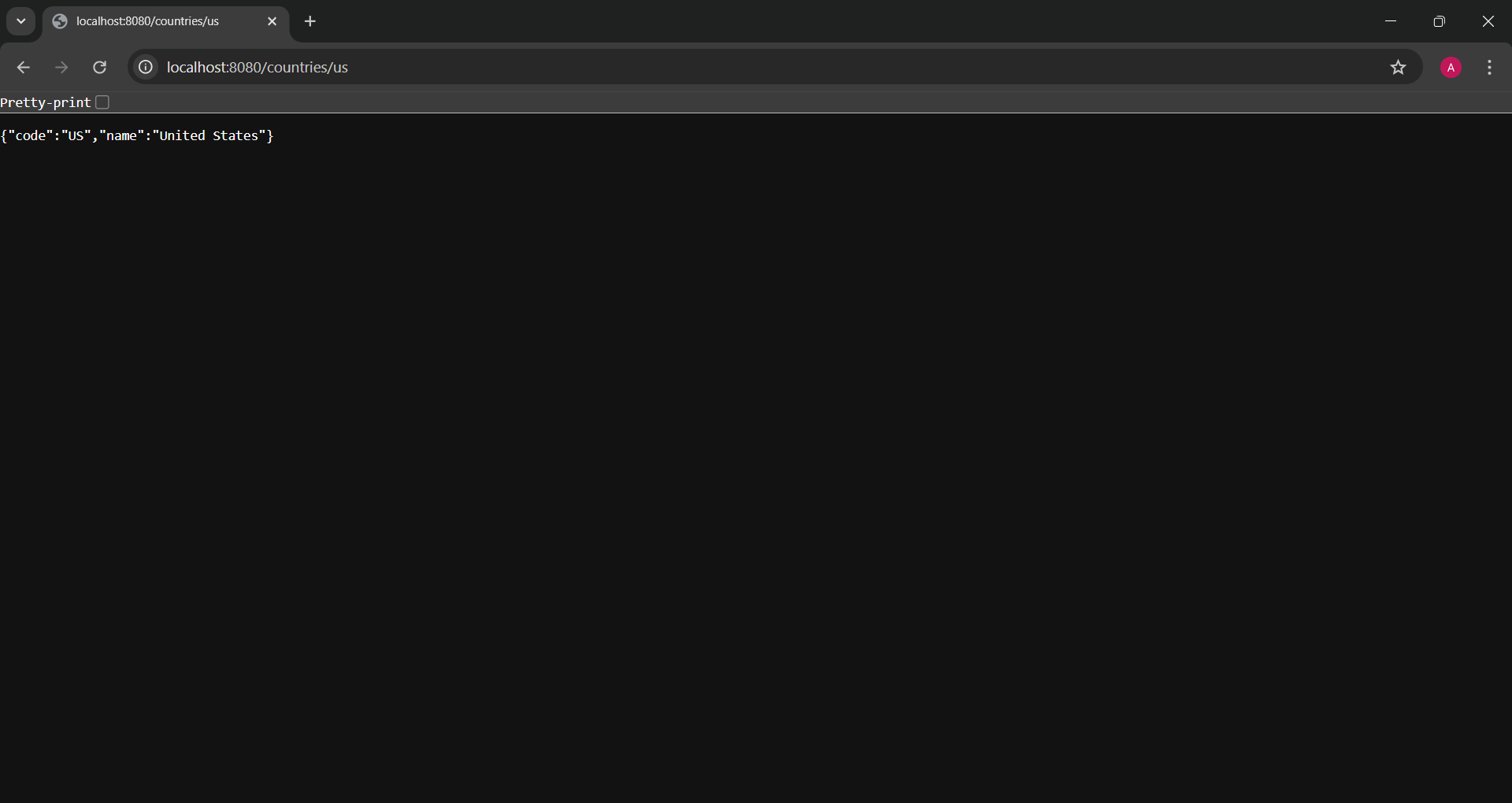
}

}

OUTPUT:







**EXERCISE 6: CREATE AUTHENTICATION SERVICE THAT RETURNS JWT**

**STEP 1:** Generate and Set Up the Project using Spring Initializer.

**STEP 2:** Add Dependencies. Import the project into eclipse.

SpringLearnApplication.java

package com.cognizant.spring\_learn;

import org.springframework.boot.SpringApplication;

import org.springframework.boot.autoconfigure.SpringBootApplication;

*@SpringBootApplication*

public class SpringLearnApplication {

public static void main(String[] args) {

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

}

}

SecurityConfig.java:

package com.cognizant.spring\_learn.config;

import org.springframework.context.annotation.Bean;

import org.springframework.context.annotation.Configuration;

import org.springframework.security.config.Customizer;

import org.springframework.security.config.annotation.web.builders.HttpSecurity;

import org.springframework.security.web.SecurityFilterChain;

*@Configuration*

public class SecurityConfig {

*@Bean*

public SecurityFilterChain filterChain(HttpSecurity http) throws Exception {

http

.csrf(csrf -> csrf.disable())

.authorizeHttpRequests(auth -> auth

.requestMatchers("/authenticate").permitAll()

.anyRequest().authenticated()

)

.httpBasic(Customizer.*withDefaults*()); // Enables HTTP Basic auth for /authenticate

return http.build();

}

}

AuthenticationController.java:

package com.cognizant.spring\_learn.controller;

import com.cognizant.spring\_learn.model.AuthResponse;

import com.cognizant.spring\_learn.util.JwtUtil;

import jakarta.servlet.http.HttpServletRequest;

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

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

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

import java.nio.charset.StandardCharsets;

import java.util.Base64;

*@RestController*

public class AuthenticationController {

*@Autowired*

private JwtUtil jwtUtil;

*@GetMapping*("/authenticate")

public AuthResponse authenticate(HttpServletRequest request) {

String header = request.getHeader("Authorization");

if (header == null || !header.startsWith("Basic ")) {

throw new RuntimeException("Missing or invalid Authorization header");

}

String[] credentials = extractCredentials(header);

String username = credentials[0];

String token = jwtUtil.generateToken(username);

return new AuthResponse(token);

}

private String[] extractCredentials(String header) {

String base64Credentials = header.substring("Basic ".length());

byte[] credDecoded = Base64.*getDecoder*().decode(base64Credentials);

String credentials = new String(credDecoded, StandardCharsets.***UTF\_8***);

return credentials.split(":", 2);

}

}

AuthResponse.java

package com.cognizant.spring\_learn.model;

public class AuthResponse {

private String token;

public AuthResponse() {}

public AuthResponse(String token) {

this.token = token;

}

public String getToken() {

return token;

}

public void setToken(String token) {

this.token = token;

}

}

JwtUtil.java:

package com.cognizant.spring\_learn.util;

import io.jsonwebtoken.Jwts;

import io.jsonwebtoken.SignatureAlgorithm;

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

import org.springframework.stereotype.Component;

import java.util.Date;

*@Component*

public class JwtUtil {

*@Value*("${jwt.secret}")

private String secret;

*@Value*("${jwt.expiration}")

private long expiration;

public String generateToken(String username) {

return Jwts.*builder*()

.setSubject(username)

.setIssuedAt(new Date())

.setExpiration(new Date(System.*currentTimeMillis*() + expiration))

.signWith(*SignatureAlgorithm*.***HS256***, secret)

.compact();

}

}

OUTPUT:

