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

**File: 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;

import org.springframework.context.ConfigurableApplicationContext;

*@SpringBootApplication*

public class SpringLearnApplication {

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

public static void main(String[] args) {

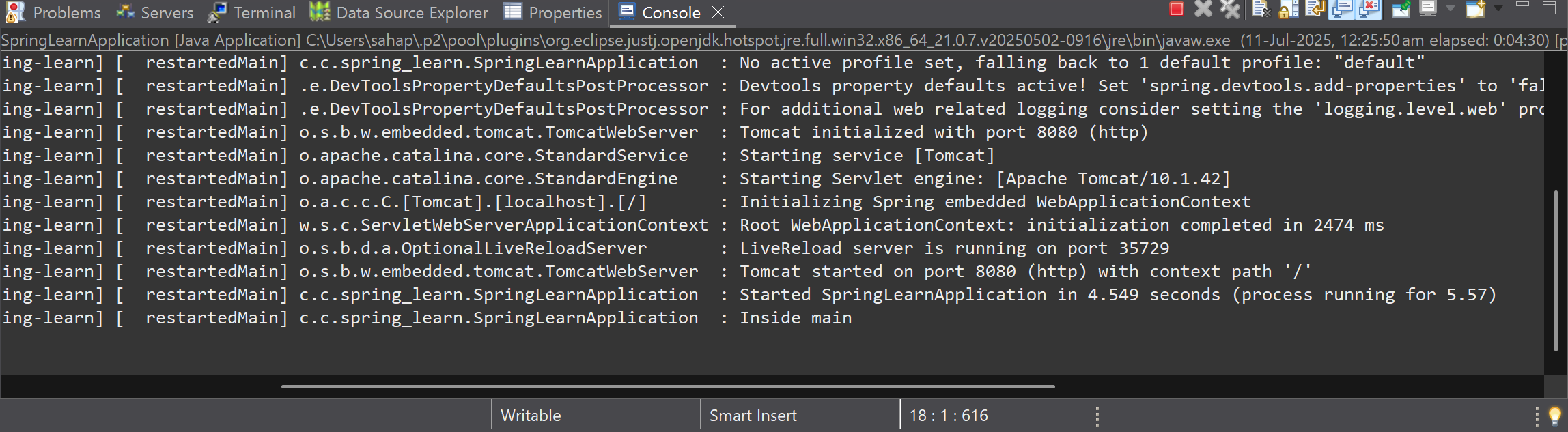
ConfigurableApplicationContext context = SpringApplication.*run*(SpringLearnApplication.class, args);

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

}

}

**OUTPUT:**



**EXERCISE 2:Spring Core – Load Country from Spring Configuration XML**

**File: SpringLearnApplication.java**

package com.cognizant.spring\_learn;

import org.slf4j.Logger;

import org.slf4j.LoggerFactory;

import org.springframework.boot.autoconfigure.SpringBootApplication;

import org.springframework.context.ApplicationContext;

import org.springframework.context.support.ClassPathXmlApplicationContext;

*@SpringBootApplication*

public class SpringLearnApplication {

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

public static void main(String[] args) {

*displayCountry*();

}

public static void displayCountry() {

***LOGGER***.info("Inside displayCountry()");

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

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

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

}

}

**File: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***.info("Inside Country Constructor");

}

public String getCode() {

***LOGGER***.info("Inside Code Getter");

return Code;

}

public void setCode(String code) {

Code = code;

***LOGGER***.info("Inside Code Setter");

}

public String getName() {

***LOGGER***.info("INSIDE NAME GETTER");

return name;

}

public void setName(String name) {

this.name = name;

***LOGGER***.info("Inside Name setter");

}

*@Override*

public String toString() {

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

}

}

**File: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**

**File: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*.debug("START: sayHello()");

String response = "Hello World!!";

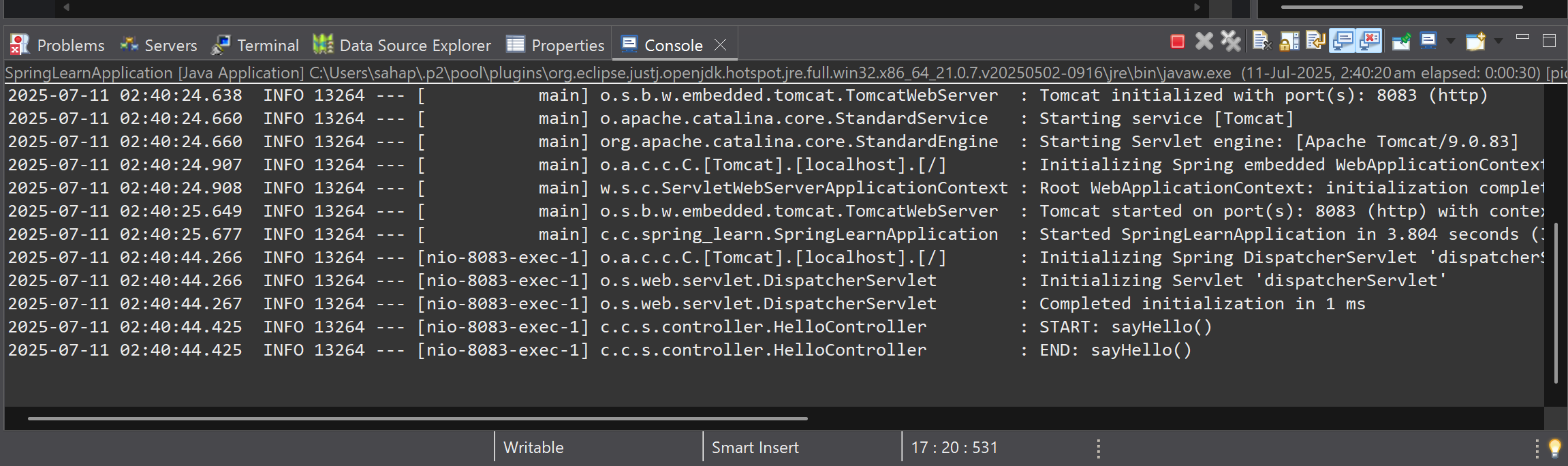
*LOGGER*.debug("END: sayHello()");

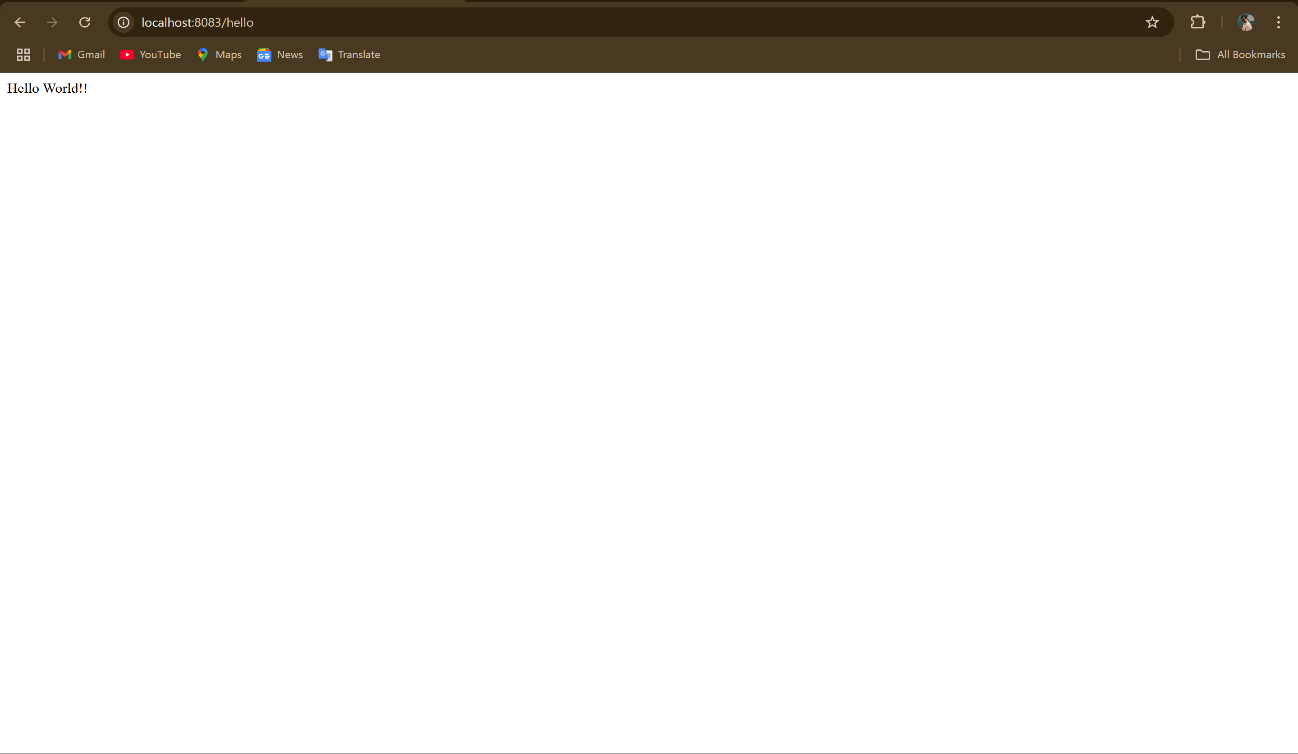
return response;

}

}

**OUTPUT:**

****

****

**EXERCISE 4: REST-Country Web Service**

**File:CountryController.java**

package com.cognizant.spring\_learn.controller;

import org.slf4j.LoggerFactory;

import org.springframework.context.ApplicationContext;

import org.springframework.context.support.ClassPathXmlApplicationContext;

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

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

import org.slf4j.Logger;

import com.cognizant.spring\_learn.Country;

*@RestController*

public class CountryController {

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

*@RequestMapping*("/country")

public Country getCountryIndia() {

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

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

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

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

return country;

}

}

**File:SpringLearnApplication.java**

package com.cognizant.spring\_learn;

import org.springframework.boot.SpringApplication;

import org.springframework.boot.autoconfigure.SpringBootApplication;

*@SpringBootApplication*

public class SpringLearnApplication {

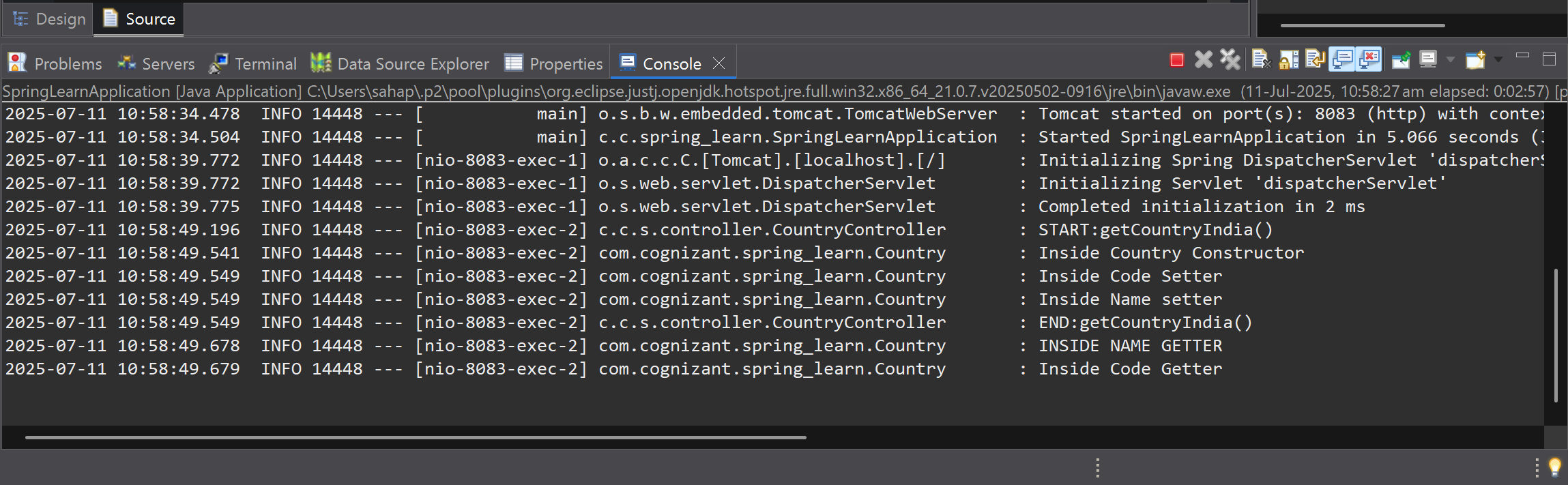
public static void main(String[] args) {

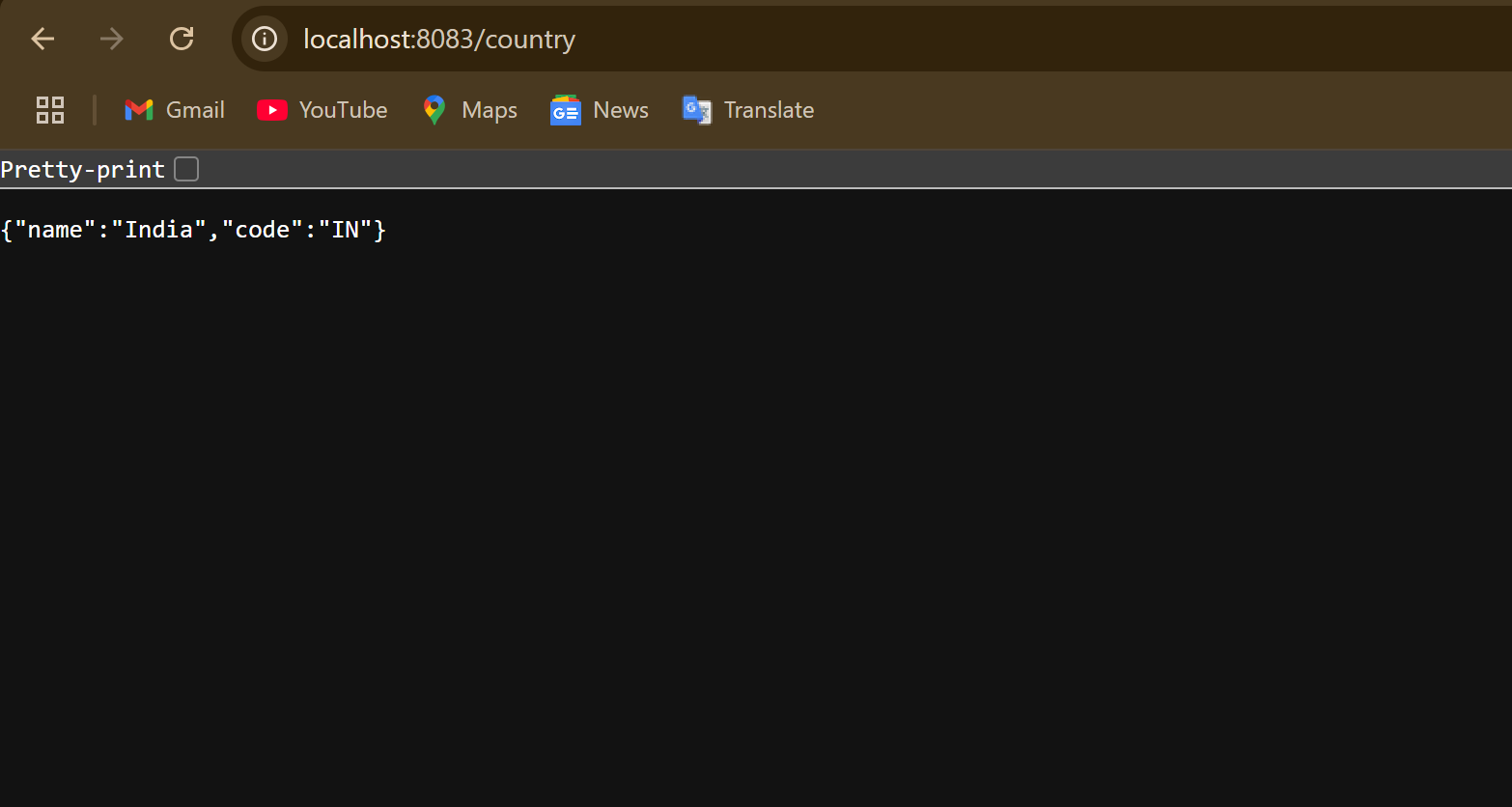
SpringApplication.*run*(SpringLearnApplication.class, args);

}

}

**OUTPUT:**

****

****

**EXERCISE 5:REST – Get Country Based on Country Code**

**File:CountryService.java**

package com.cognizant.spring\_learn.service;

import java.util.List;

import org.springframework.context.ApplicationContext;

import org.springframework.context.support.ClassPathXmlApplicationContext;

import org.springframework.stereotype.Service;

import org.slf4j.Logger;

import org.slf4j.LoggerFactory;

import com.cognizant.spring\_learn.Country;

*@Service*

public class CountryService {

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

public Country getCountry(String code) {

***LOGGER***.debug("START: getCountry({})", code);

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

*@SuppressWarnings*("unchecked")

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

return countries.stream()

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

.findFirst()

.orElse(null);

}

}

**File:CountryController.java**

package com.cognizant.spring\_learn.controller;

import org.slf4j.LoggerFactory;

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

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

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

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

import org.slf4j.Logger;

import com.cognizant.spring\_learn.Country;

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

*@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;

}

}

**File: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=*"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=*"countryDE"* class=*"com.cognizant.spring\_learn.Country"*>

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

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

</**bean**>

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

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

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

</**bean**>

<!-- List Bean -->

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

<**constructor-arg**>

<**list**>

<**ref** bean=*"countryIN"*/>

<**ref** bean=*"countryUS"*/>

<**ref** bean=*"countryDE"*/>

<**ref** bean=*"countryJP"*/>

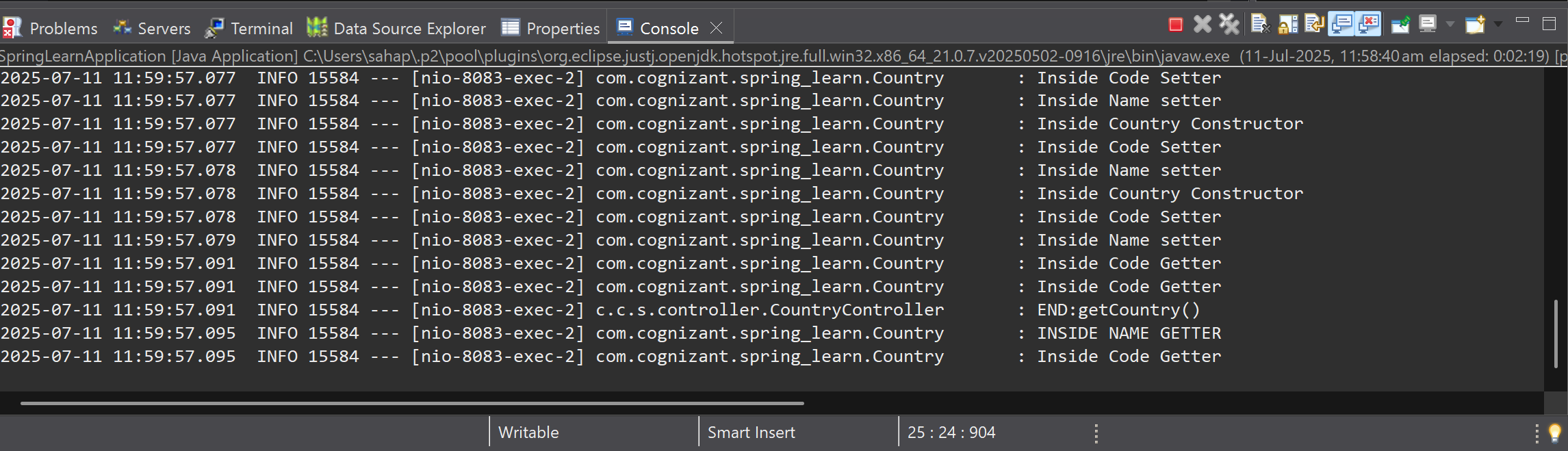
</**list**>

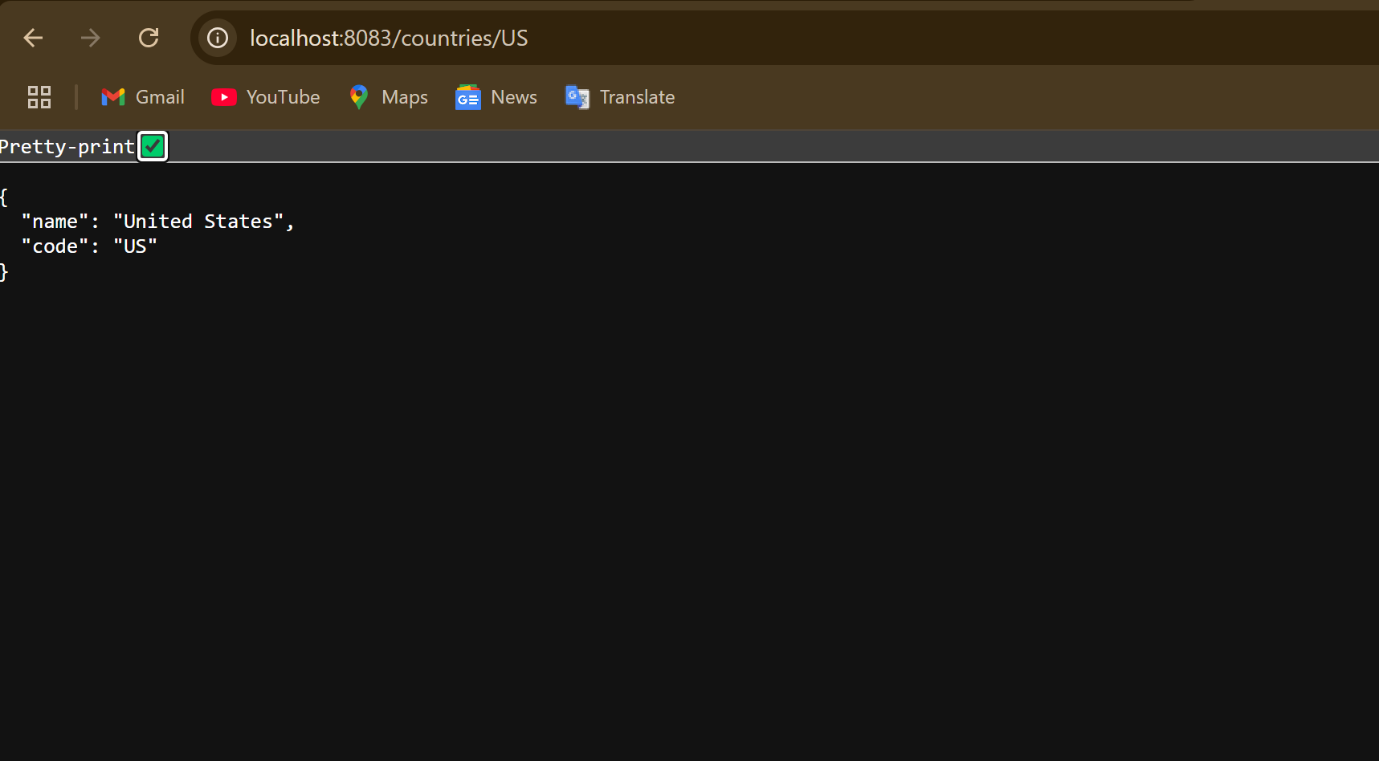
</**constructor-arg**>

</**bean**>

</**beans**>

**OUTPUT:**

****



**Exercise 6: Create Authentication Service that returns JWT**

**File: AuthenticationController.java**

package com.cognizant.spring\_learn.controller;

import java.util.Base64;

import javax.servlet.http.HttpServletRequest;

import org.springframework.http.ResponseEntity;

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

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

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

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

*@RestController*

public class AuthenticationController {

private final JwtUtil jwtUtil;

public AuthenticationController(JwtUtil jwtUtil) {

this.jwtUtil = jwtUtil;

}

*@RequestMapping*(value = "/authenticate", method = *RequestMethod*.***GET***)

public ResponseEntity<?> generateToken(HttpServletRequest request) {

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

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

return ResponseEntity.*status*(401).body("Missing or invalid Authorization header");

}

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

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

String credentials = new String(decoded);

String[] values = credentials.split(":", 2);

if ("user".equals(values[0]) && "pwd".equals(values[1])) {

String token = jwtUtil.generateToken(values[0]);

return ResponseEntity.*ok*().body("{\"token\":\"" + token + "\"}");

} else {

return ResponseEntity.*status*(403).body("Invalid credentials");

}

}

}

**File:SecurityConfig.js**

package com.cognizant.spring\_learn.security;

import org.springframework.context.annotation.Configuration;

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

import org.springframework.security.config.annotation.web.configuration.WebSecurityConfigurerAdapter;

*@SuppressWarnings*("deprecation")

*@Configuration*

public class SecurityConfig extends WebSecurityConfigurerAdapter{

*@Override*

protected void configure(HttpSecurity http) throws Exception {

http

.csrf().disable()

.authorizeRequests()

.antMatchers("/authenticate").permitAll()

.anyRequest().authenticated()

.and()

.httpBasic(); // enables -u username:password for testing

}

}

**File:JwtUtil.java**

package com.cognizant.spring\_learn.util;

import java.util.Date;

import org.springframework.stereotype.Component;

import io.jsonwebtoken.Jwts;

import io.jsonwebtoken.SignatureAlgorithm;

*@Component*

public class JwtUtil {

private String secret = "mySecretKey"; // ideally move to application.properties

public String generateToken(String username) {

return Jwts.*builder*()

.setSubject(username)

.setIssuedAt(new Date(System.*currentTimeMillis*()))

.setExpiration(new Date(System.*currentTimeMillis*() + 1000 \* 60 \* 60)) // 1 hour expiry

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

.compact();

}

}

**OUTPUT:**

