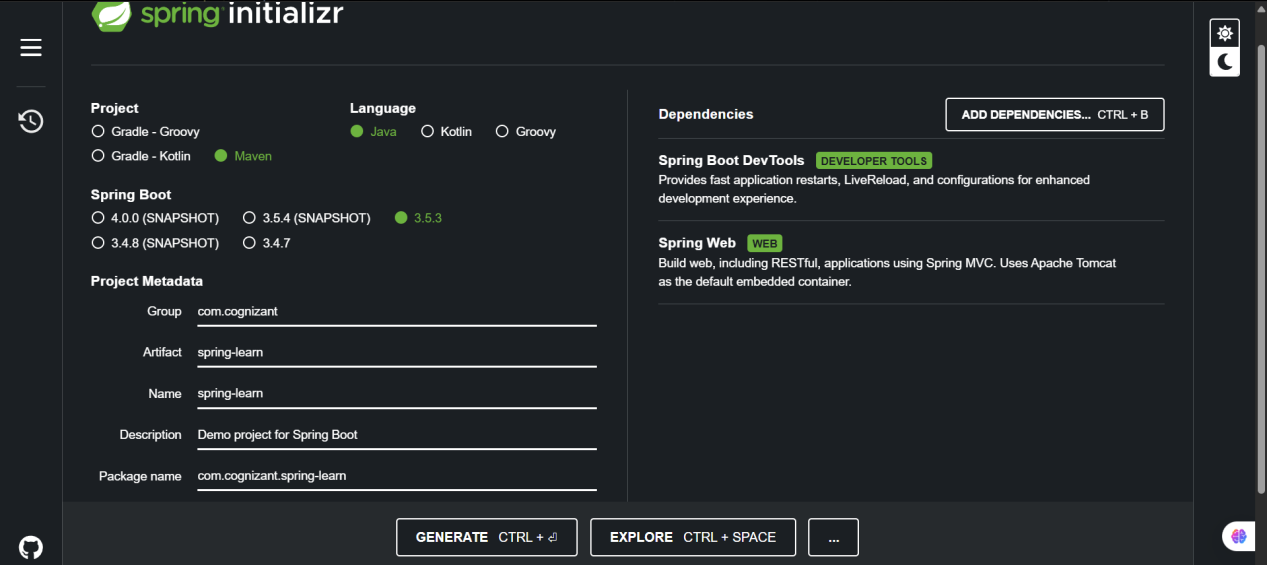
**Spring REST using Spring Boot 3**

**1.spring-rest-handson**

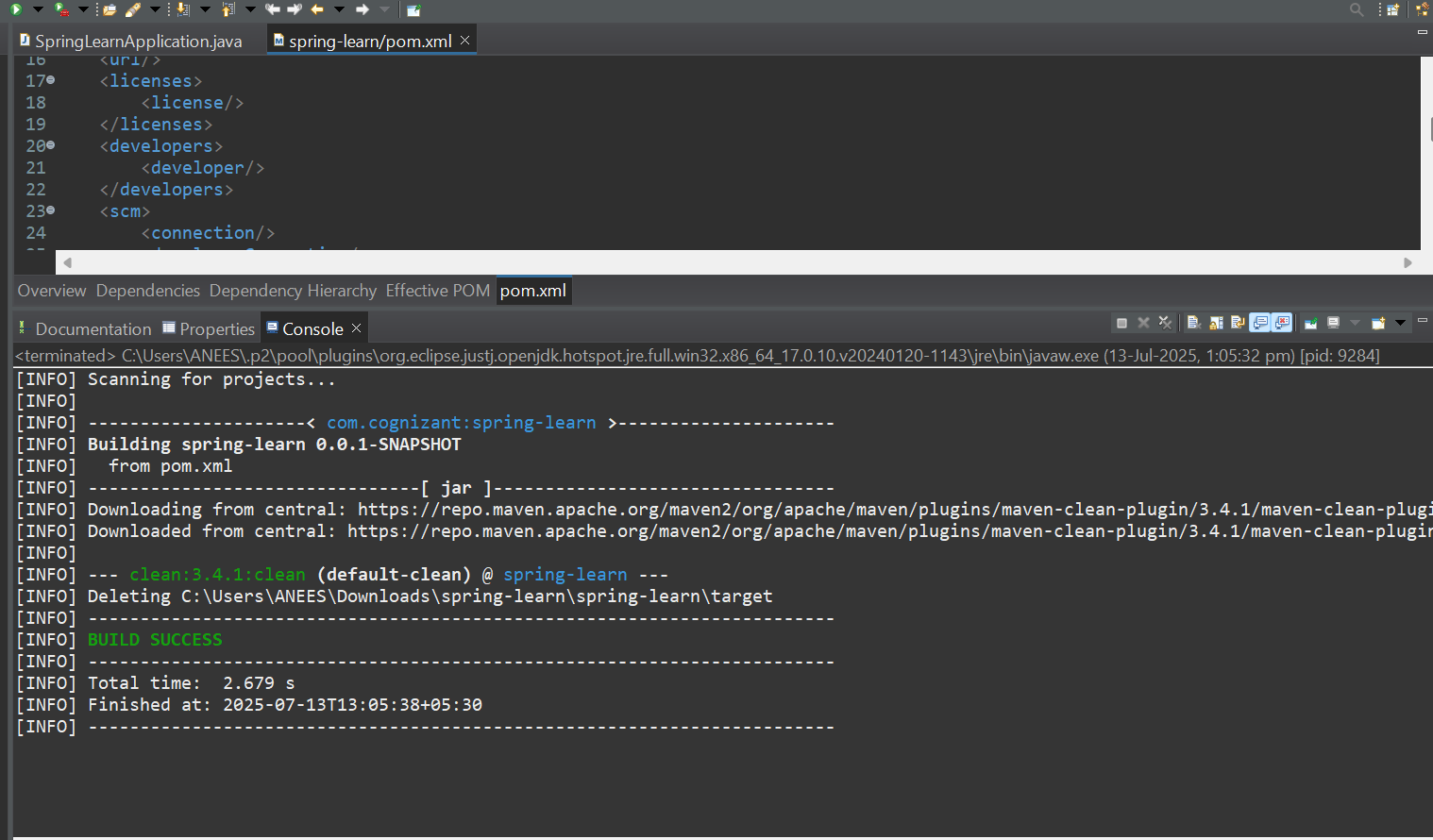
**Hands on 1**

**Create a Spring Web Project using Maven**

1. Go to <https://start.spring.io/>
2. Change Group as “com.cognizant”
3. Change Artifact Id as “spring-learn”
4. Select Spring Boot DevTools and Spring Web
5. Create and download the project as zip



1. Extract the zip in root folder to Eclipse Workspace
2. Build the project using ‘mvn clean package -Dhttp.proxyHost=proxy.cognizant.com -Dhttp.proxyPort=6050 -Dhttps.proxyHost=proxy.cognizant.com -Dhttps.proxyPort=6050 -Dhttp.proxyUser=123456’ command in command line.
3. Import the project in Eclipse



1. Include logs to verify if main() method of SpringLearnApplication.

**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) {

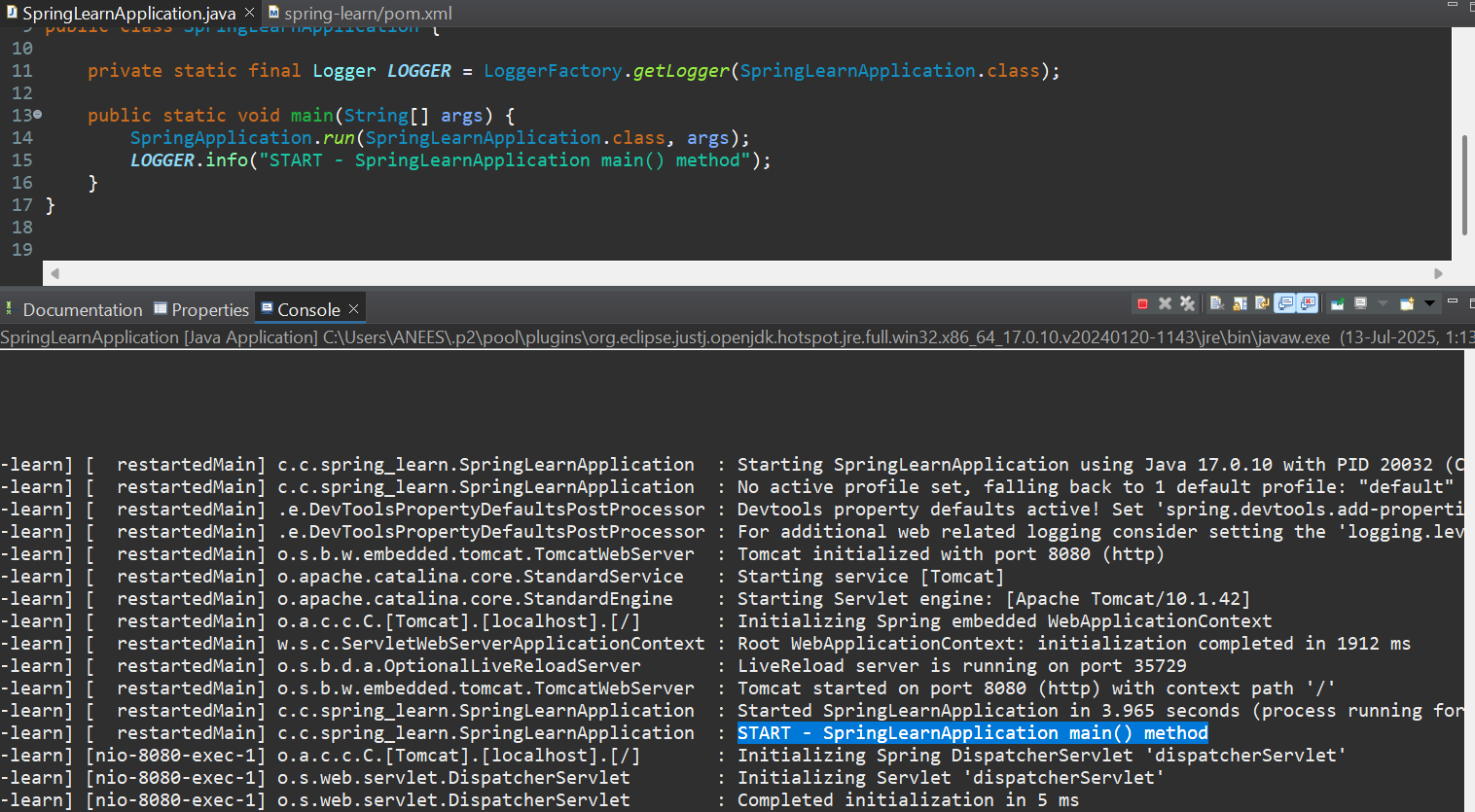
SpringApplication.*run*(SpringLearnApplication.class, args);

***LOGGER***.info("START - SpringLearnApplication main() method");

}

}

10.Run the SpringLearnApplication class.



**Hands on 4**

**Spring Core – Load Country from Spring Configuration XML**   
  
An airlines website is going to support booking on four countries. There will be a drop down on the home page of this website to select the respective country. It is also important to store the two-character ISO code of each country. 

|  |  |
| --- | --- |
| **Code** | **Name** |
| US | United States |
| DE | Germany |
| IN | India |
| JP | Japan |

**Steps to implement**

**Create 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*

*https://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**>

</**beans**>

**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 + "]";

}

}

**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;

public class SpringLearnApplication {

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

public static void main(String[] args) {

***LOGGER***.info("START - SpringLearnApplication");

*displayCountryIndia*();

***LOGGER***.info("END - SpringLearnApplication");

}

public static void displayCountryIndia() {

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

// Load the bean with ID 'countryIN'

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

System.***out***.println("Code: " + country.getCode());

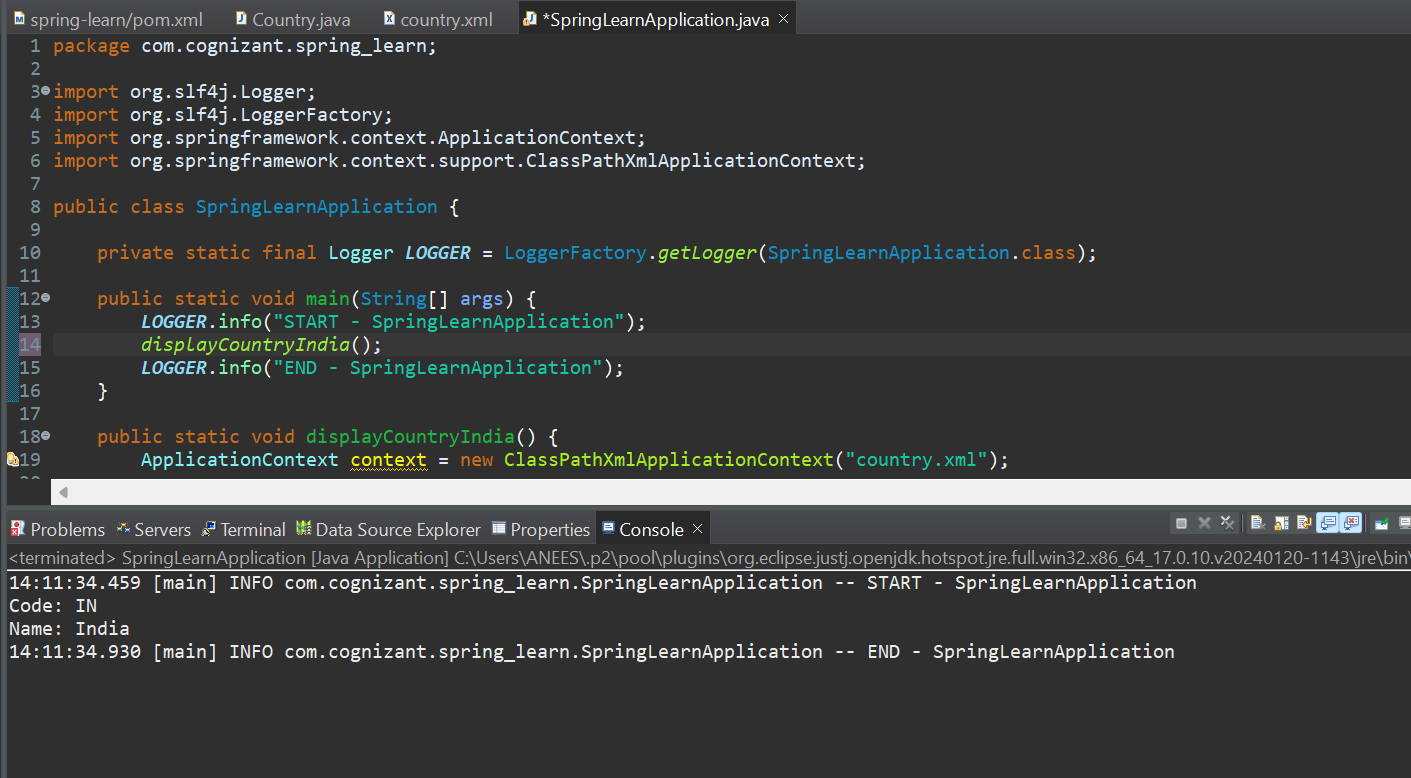
System.***out***.println("Name: " + country.getName());

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

}

}

**Output:**

****

**2. spring-rest-handson**

**HandsOn-1**

**Hello World RESTful Web Service**

Write a REST service in the spring learn application created earlier, that returns the text "Hello World!!" using Spring Web Framework. Refer details below:  
Method: GET  
URL: /hello  
Controller: com.cognizant.spring-learn.controller.HelloController  
Method Signature: public String sayHello()  
Method Implementation: return hard coded string "Hello World!!"  
Sample Request: http://localhost:8083/hello  
Sample Response: Hello World!! 

**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 - package com.cognizant.spring\_learn;

**SpringLearnApplication.java**

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) {

***LOGGER***.info("START - SpringLearnApplication");

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

***LOGGER***.info("END - SpringLearnApplication");

}

}

sayHello()");

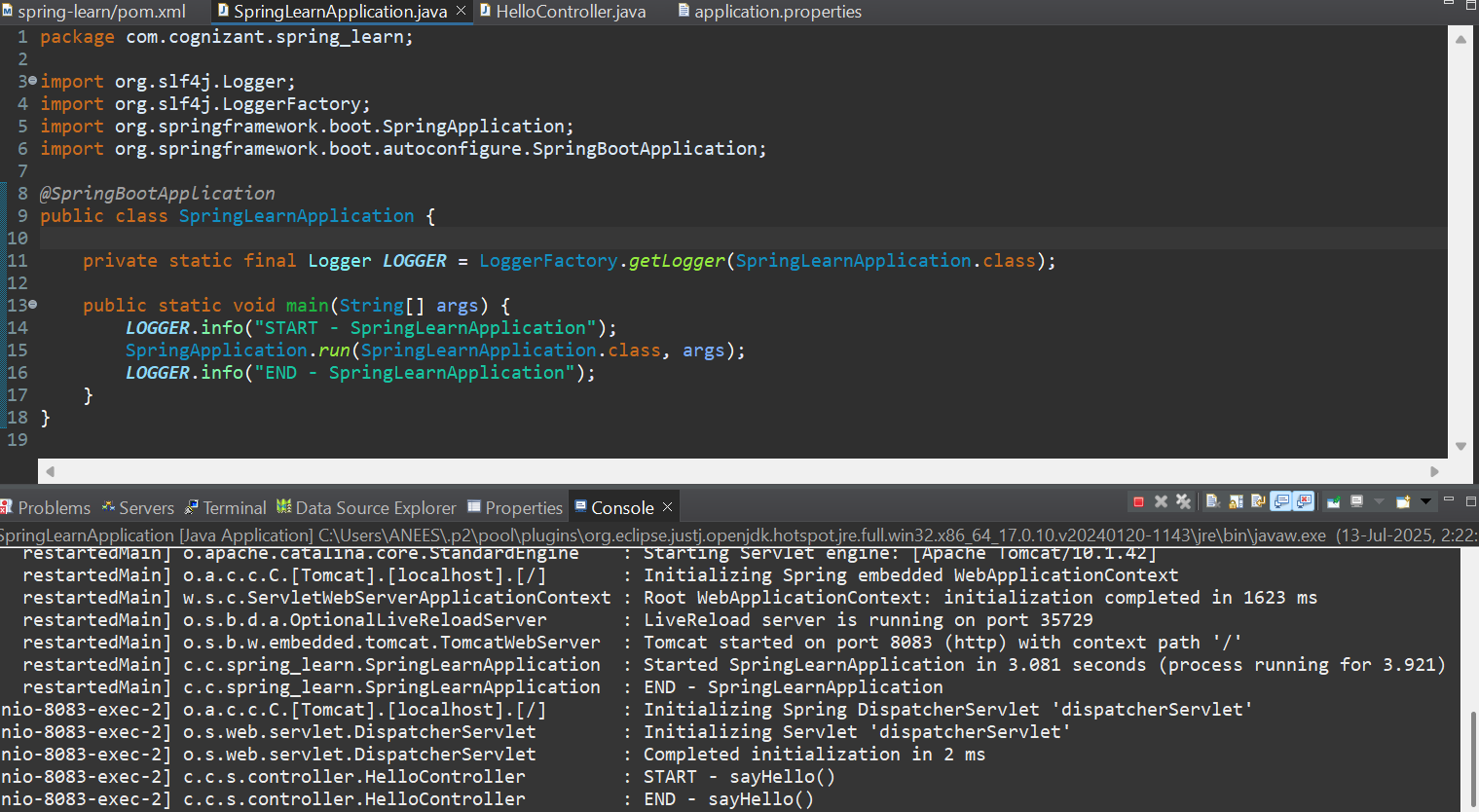
String response = "Hello World!!";

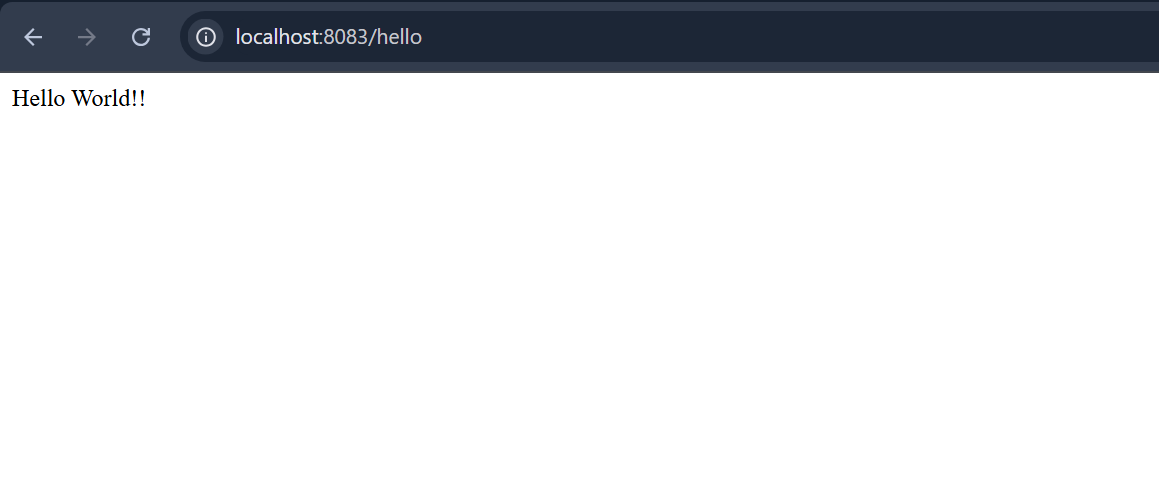
*LOGGER*.info("END - sayHello()");

return response;

}

}

**Output  
**

****

**HandsOn2**

**REST - Country Web Service**Write a REST service that returns India country details in the earlier created spring learn application.  
  
URL: /country  
Controller: com.cognizant.spring-learn.controller.CountryController  
Method Annotation: @RequestMapping  
Method Name: getCountryIndia()  
Method Implementation: Load India bean from spring xml configuration and return  
Sample Request: <http://localhost:8083/country>

**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

https://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> **Create Country POJO:**

**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("Getting code");

return code;

}

public void setCode(String code) {

LOGGER.debug("Setting code: {}", code);

this.code = code;

}

public String getName() {

LOGGER.debug("Getting name");

return name;

}

public void setName(String name) {

LOGGER.debug("Setting name: {}", name);

this.name = name;

}

@Override

public String toString() {

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

}

}

**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.ApplicationContext;

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()");

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

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

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

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

return country;

}

}

**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) {

***LOGGER***.info("START - SpringLearnApplication");

// Start Spring Boot application

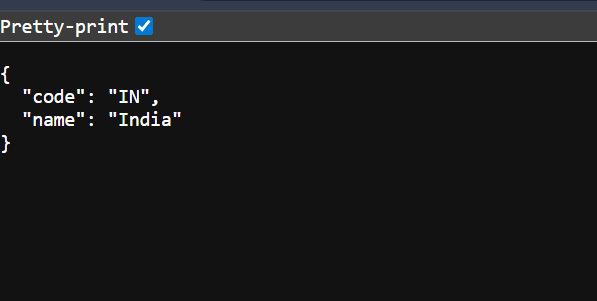
SpringApplication.*run*(SpringLearnApplication.class, args);

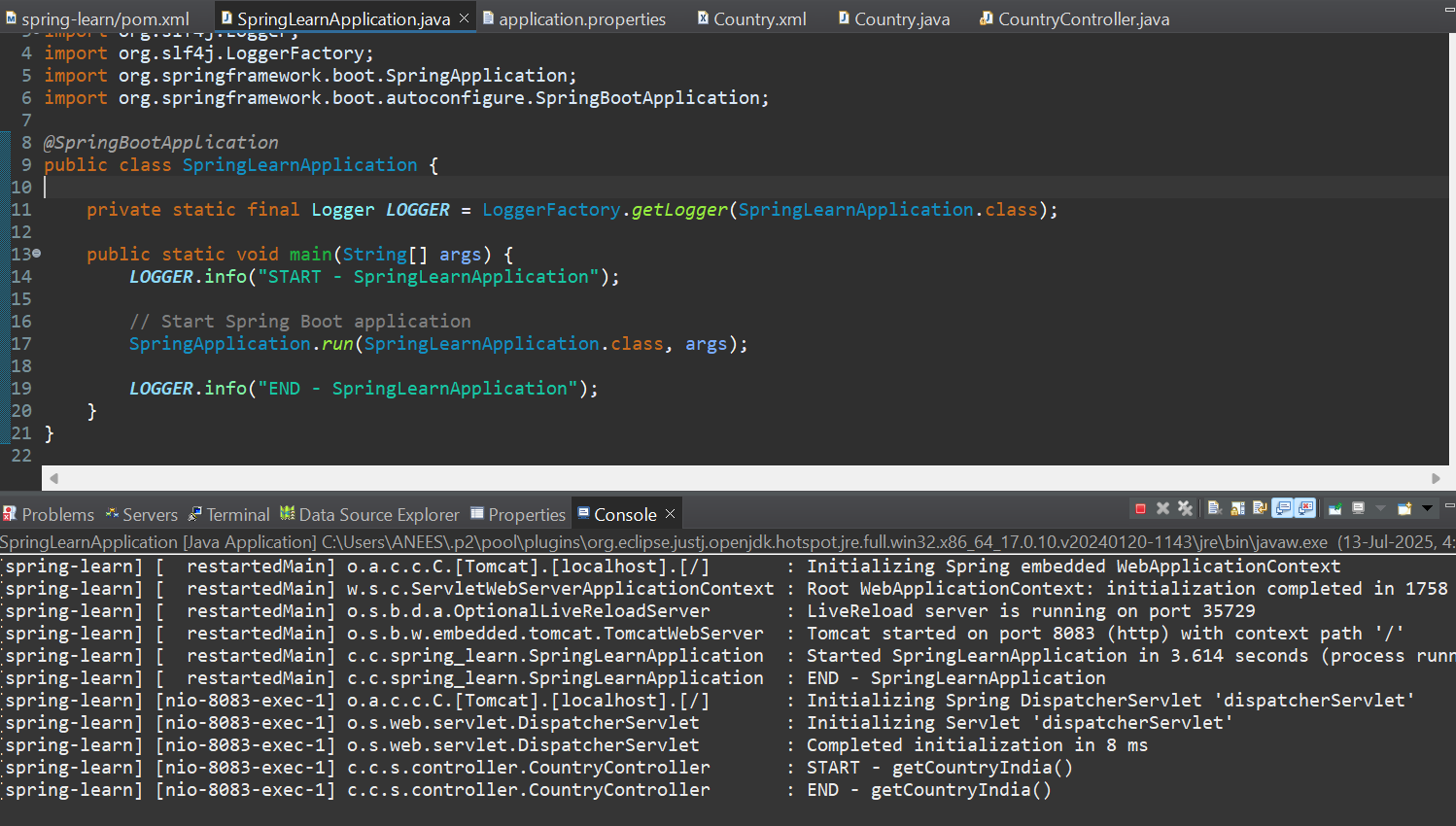
***LOGGER***.info("END - SpringLearnApplication");

}

}

**Output:**





**REST - Get country based on country code**   
  
Write a REST service that returns a specific country based on country code. The country code should be case-insensitive.  
  
**Controller**: com.cognizant.spring-learn.controller.CountryController  
**Method Annotation:** @GetMapping("/countries/{code}")  
**Method Name**: getCountry(String code)  
**Method Implemetation**: Invoke countryService.getCountry(code)   
**Service Method:**com.cognizant.spring-learn.service.CountryService.getCountry(String code)

**Sample Request**: <http://localhost:8083/country/in>

**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 + "]";

}

}

**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*

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

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

<**constructor-arg**>

<**list**>

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

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

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

</**bean**>

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

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

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

</**bean**>

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

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

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

</**bean**>

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

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

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

</**bean**>

</**list**>

</**constructor-arg**>

</**bean**>

</**beans**>

**CountryService.java**

package com.cognizant.spring\_learn.services;

import com.cognizant.spring\_learn.Country;

import org.springframework.context.ApplicationContext;

import org.springframework.context.support.ClassPathXmlApplicationContext;

import org.springframework.stereotype.Service;

import java.util.List;

@Service

public class CountryService {

public Country getCountry(String code) {

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

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

// Case-insensitive match using lambda

return countries.stream()

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

.findFirst()

.orElse(null);

}

}

**CountryController .java**

package com.cognizant.spring\_learn.controller;

import com.cognizant.spring\_learn.Country;

import com.cognizant.spring\_learn.services.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;

}

}

**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) {

***LOGGER***.info("START - SpringLearnApplication");

// Start Spring Boot application

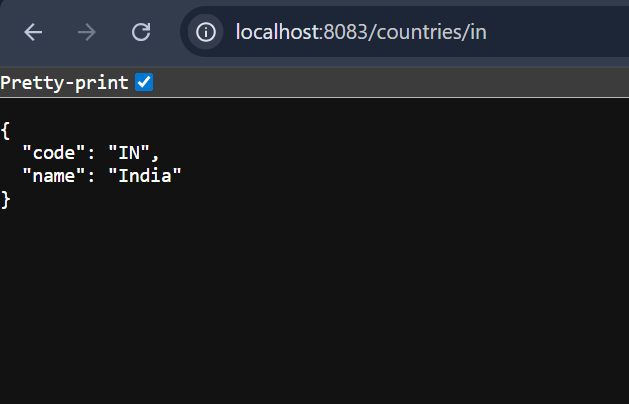
SpringApplication.*run*(SpringLearnApplication.class, args);

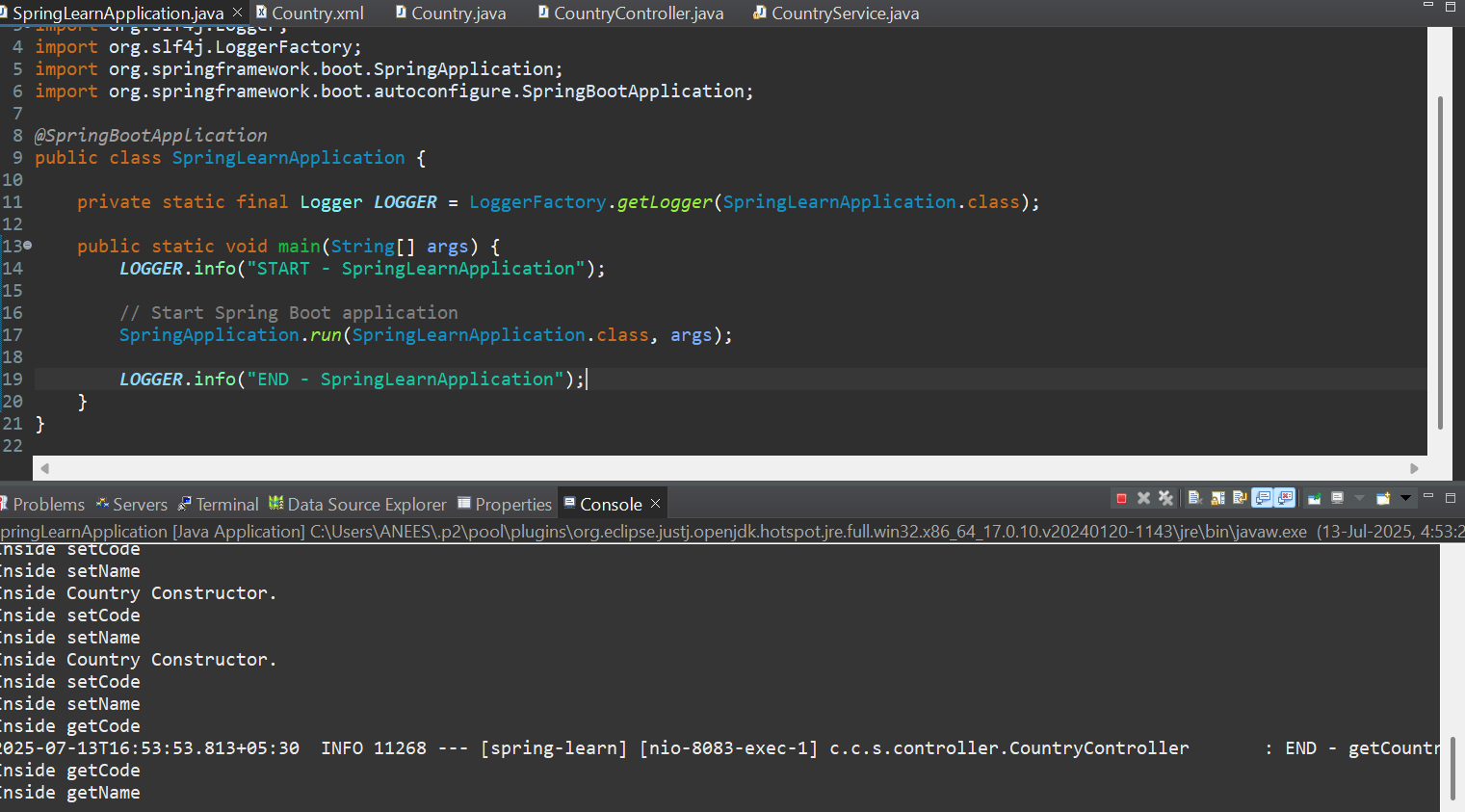
***LOGGER***.info("END - SpringLearnApplication");

}

}

**Output**





**5.JWT HandsOn**

**Create an authentication service that returns JWT**

**Code:**

**UserRequest.java**

package com.cognizant.jwt\_auth\_service.model;

import jakarta.validation.constraints.NotBlank;

import jakarta.validation.constraints.Size;

public class UserRequest {

*@NotBlank*(message = "Username is required")

*@Size*(min = 3, max = 20, message = "Username must be 3 to 20 characters")

private String username;

*@NotBlank*(message = "Password is required")

*@Size*(min = 5, message = "Password must be at least 5 characters")

private String password;

// Getters and Setters

public String getUsername() {

return username;

}

public void setUsername(String username) {

this.username = username;

}

public String getPassword() {

return password;

}

public void setPassword(String password) {

this.password = password;

}

}

**AuthController.java**

package com.cognizant.jwt\_auth\_service.controller;

import com.cognizant.jwt\_auth\_service.model.UserRequest;

import org.slf4j.Logger;

import org.slf4j.LoggerFactory;

import org.springframework.http.ResponseEntity;

import org.springframework.validation.annotation.Validated;

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

import jakarta.validation.Valid;

*@RestController*

*@RequestMapping*("/api")

*@Validated*

public class AuthController {

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

*@PostMapping*("/validate")

public ResponseEntity<String> validateUser(*@Valid* *@RequestBody* UserRequest userRequest) {

***LOGGER***.info("Received validation request for user: {}", userRequest.getUsername());

return ResponseEntity.*ok*("Valid input!");

}

}

**JwtAuthServiceApplication.java**

package com.cognizant.jwt\_auth\_service;

import org.springframework.boot.SpringApplication;

import org.springframework.boot.autoconfigure.SpringBootApplication;

@SpringBootApplication

public class JwtAuthServiceApplication {

public static void main(String[] args) {

SpringApplication.run(JwtAuthServiceApplication.class, args);

}

}

**Output:**

