**Create a Spring Web Project using Maven**mvn clean package -Dhttp.proxyHost=proxy.cognizant.com -Dhttp.proxyPort=6050 -Dhttps.proxyHost=proxy.cognizant.com -Dhttps.proxyPort=6050 -Dhttp.proxyUser=6410114

**Program:**

**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)

{

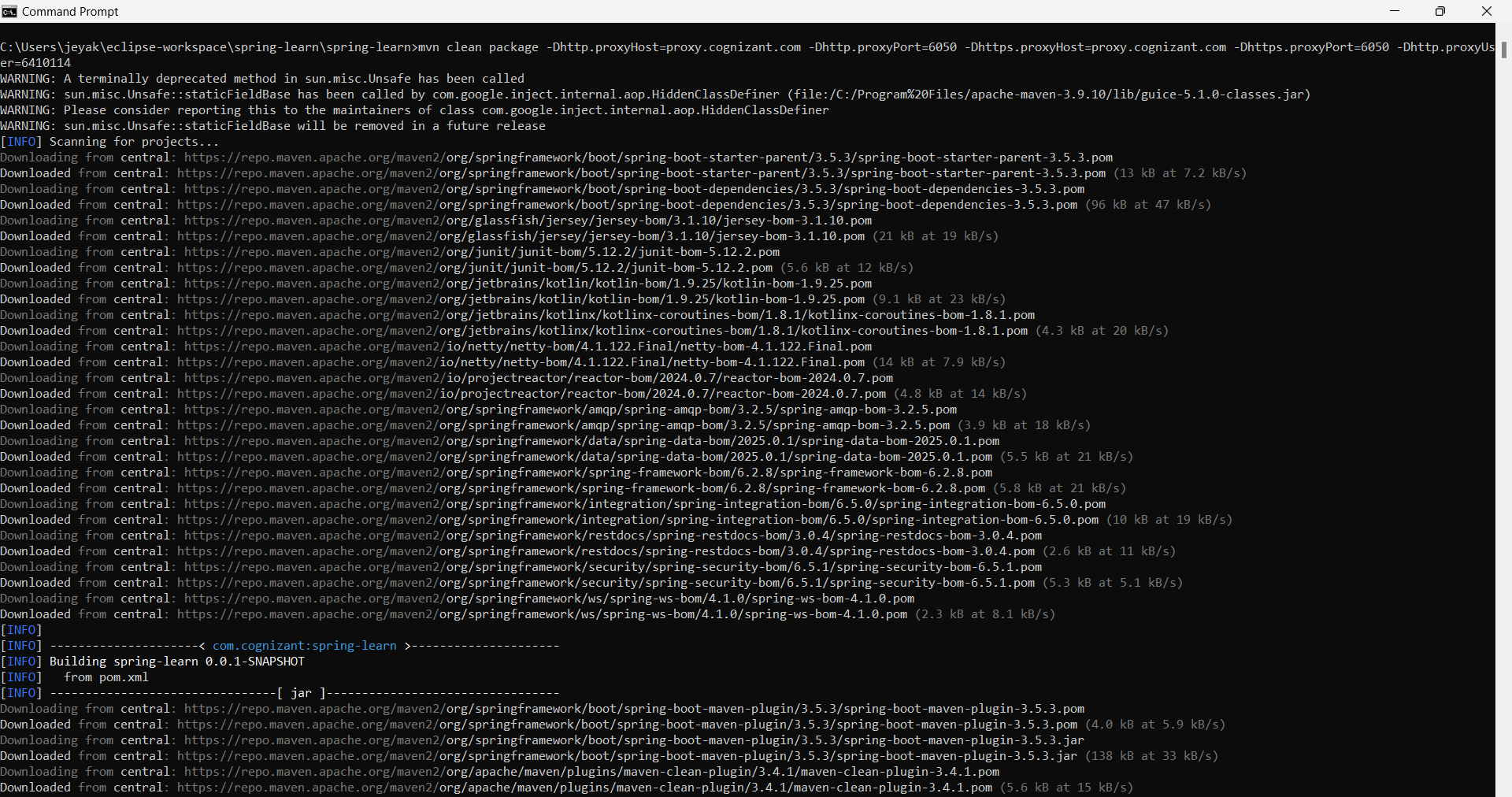
***LOGGER***.info("Inside main method...");

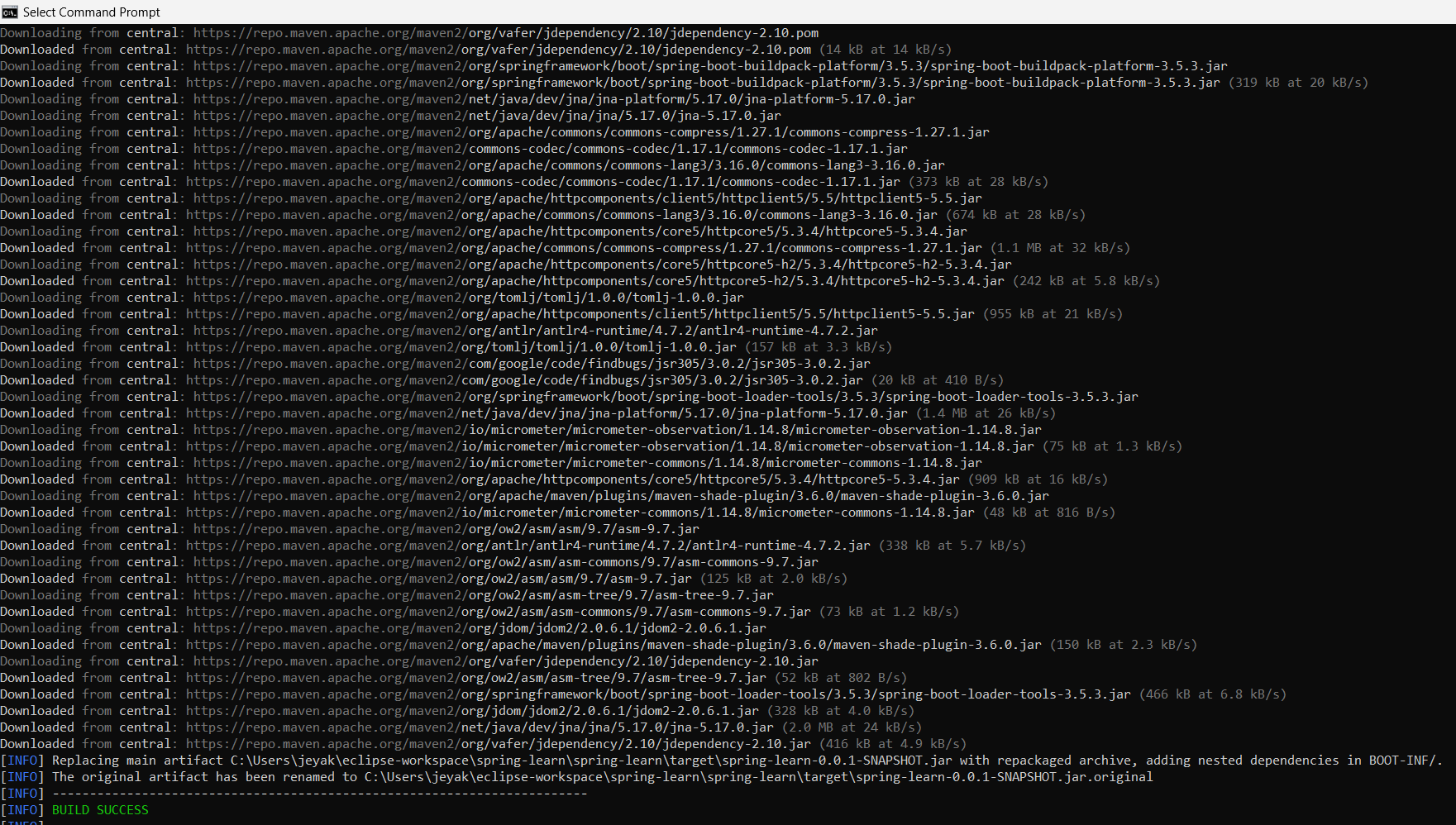
SpringApplication.*run*(SpringLearnApplication.**class**,args);

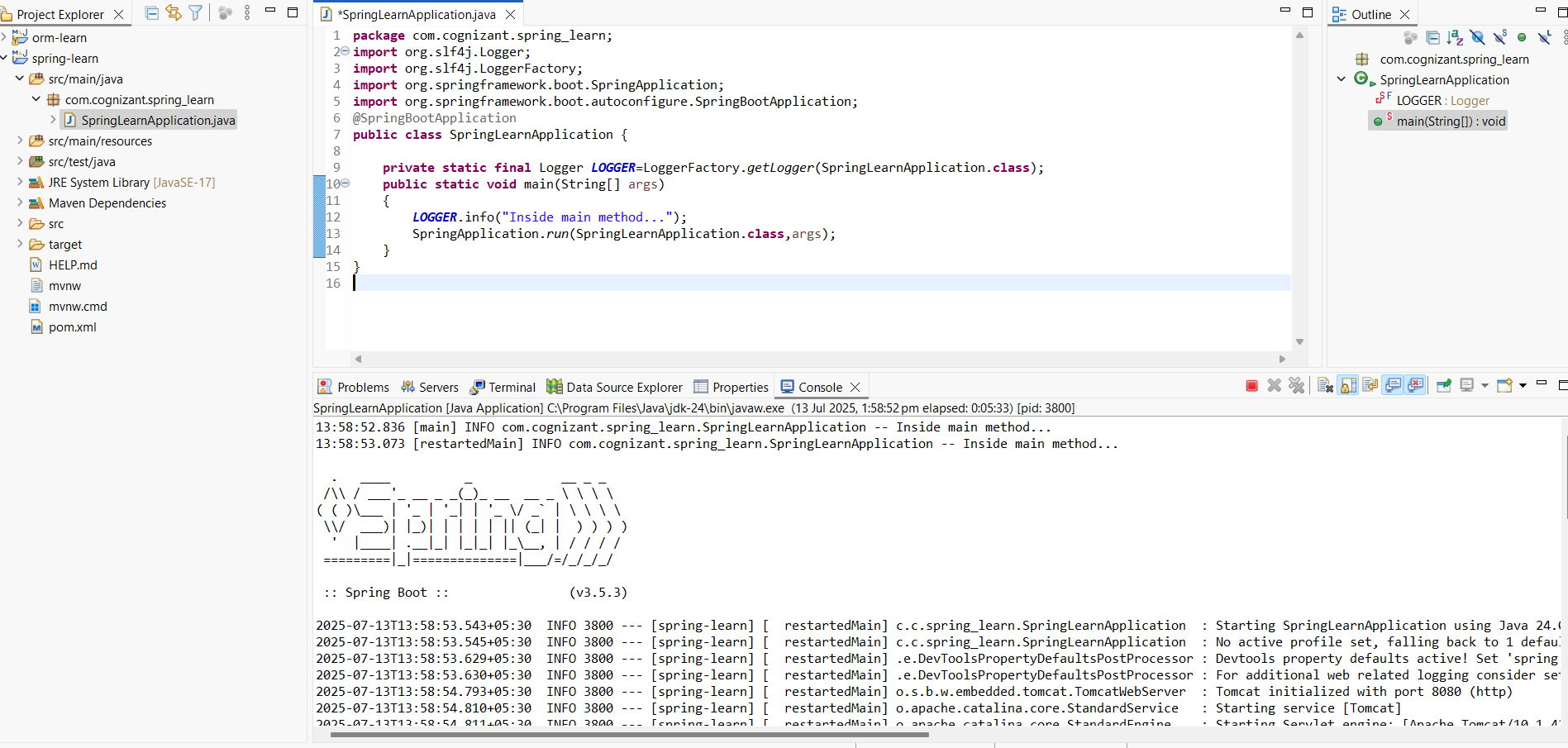
}

}

**Output:**







**Spring Core – Load Country from Spring Configuration XML   
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>

**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");

*displayCountry*();

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

}

**public** **static** **void** displayCountry()

{

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

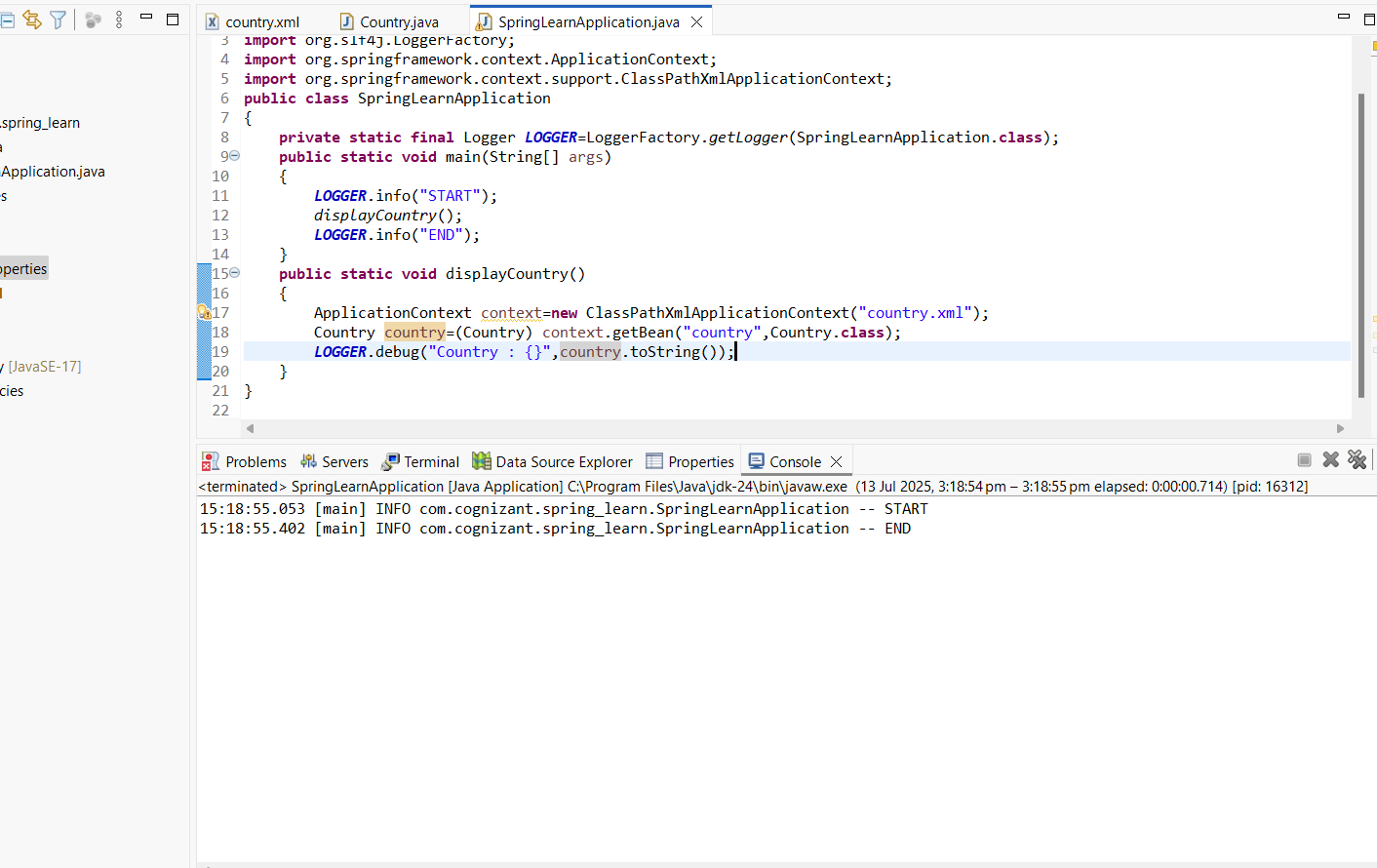
Country country=(Country) context.getBean("country",Country.**class**);

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

}

}

**Output**

****

**Hello World RESTful Web Service**

**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.cognizant</groupId>

<artifactId>spring-learn</artifactId>

<version>0.0.1-SNAPSHOT</version>

<name>spring-learn</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-devtools</artifactId>

<scope>runtime</scope>

<optional>true</optional>

</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-learn

logging.level.root=DEBUG

server.port=8083

**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");

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

**return** "Hello World!!";

}

}

**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)

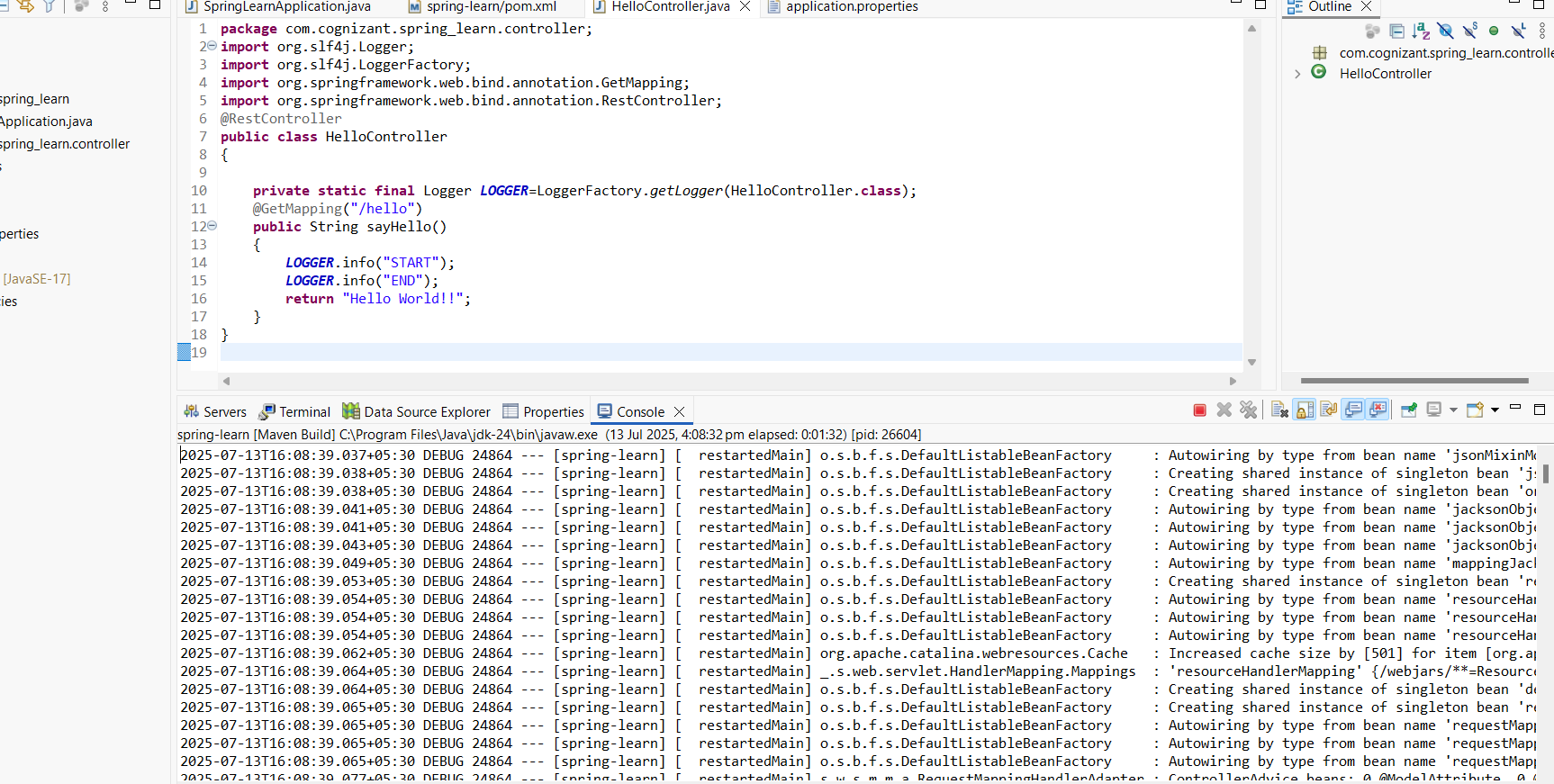
{

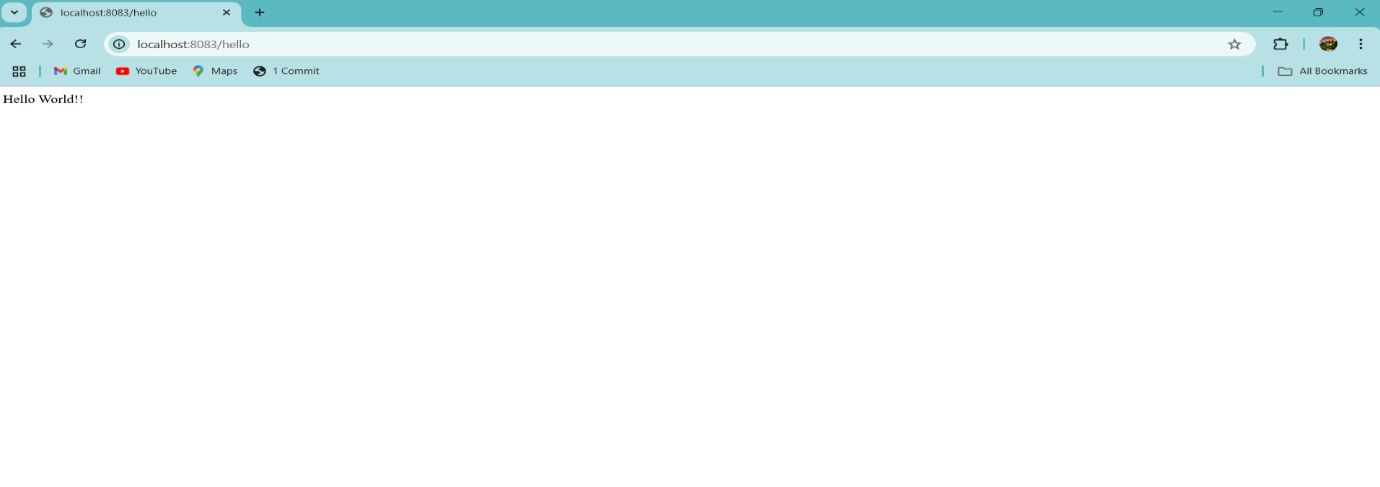
SpringApplication.*run*(SpringLearnApplication.**class**,args);

}

}

**Output**





**REST - Country Web Service   
application.properties**

spring.application.name=spring-learn

logging.level.root=DEBUG

server.port=8083

**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=*"in"* class=*"com.cognizant.spring\_learn.model.Country"*>

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

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

</bean>

</beans>

**Country.java**

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

**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** **void** setCode(String code)

{

**this**.code=code;

}

**public** String getName()

{

**return** name;

}

**public** **void** setName(String name)

{

**this**.name=name;

}

}

**CountryController.java**

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

**import** com.cognizant.spring\_learn.model.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");

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

Country country = (Country) context.getBean("in");

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

**return** country;

}

}

**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)

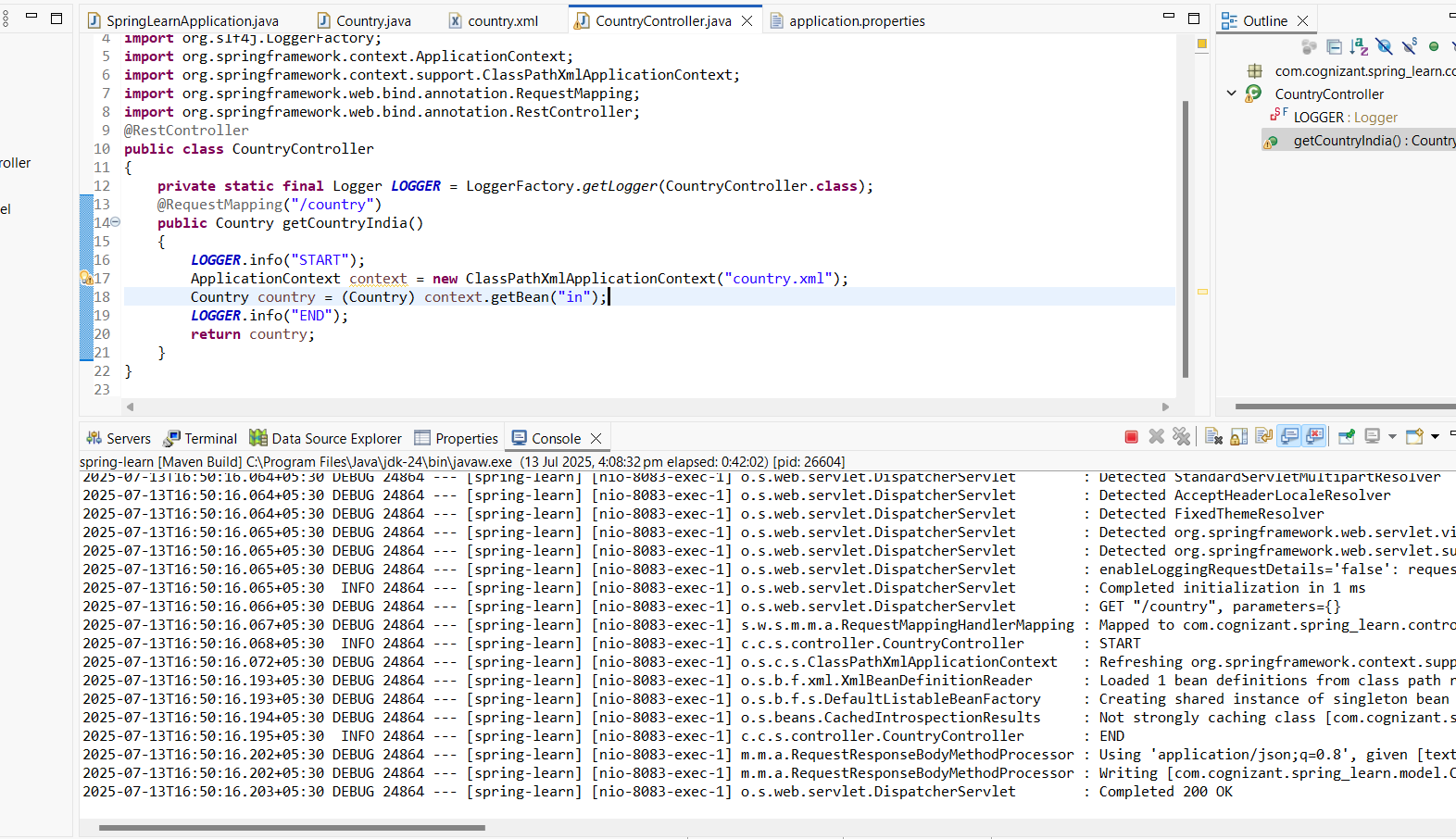
{

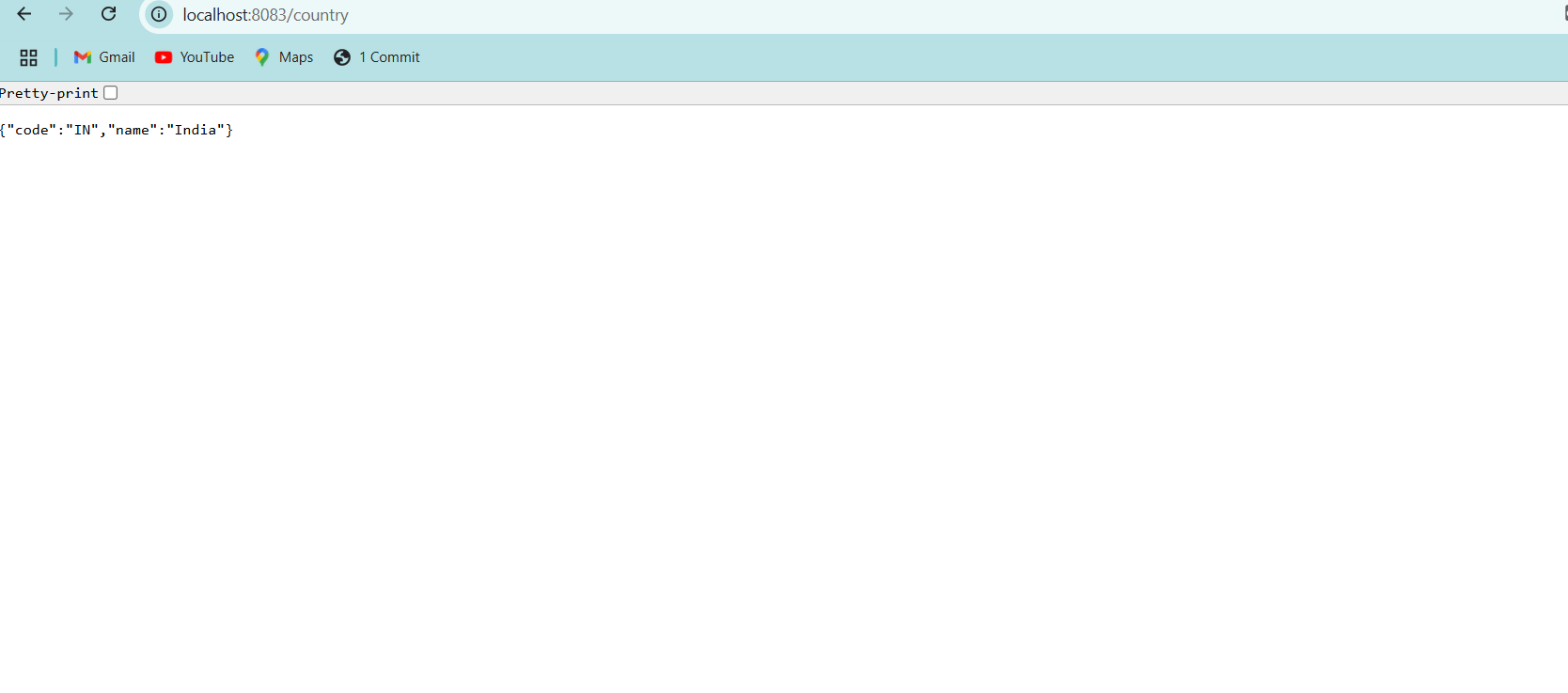
SpringApplication.*run*(SpringLearnApplication.**class**,args);

}

}

**Output**

****

****

**REST - Get country based on country code**

**application.properties**

spring.application.name=spring-learn

logging.level.root=DEBUG

server.port=8083

**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=*"in"* class=*"com.cognizant.spring\_learn.model.Country"*>

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

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

</bean>

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

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

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

</bean>

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

<constructor-arg>

<list>

<ref bean=*"in"*/>

<ref bean=*"us"*/>

</list>

</constructor-arg>

</bean>

</beans>

**Country.java**

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

**public** **class** Country

{

**private** String code;

**private** String name;

**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;

}

}

**CountryService.java**

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

**import** com.cognizant.spring\_learn.model.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> countryList=(List<Country>) context.getBean("countryList");

**return** countryList.stream()

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

.findFirst()

.orElse(**null**);

}

}

**CountryController.java**

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

**import** com.cognizant.spring\_learn.model.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("/country/{code}")

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

{

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

Country country = countryService.getCountry(code);

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

**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");

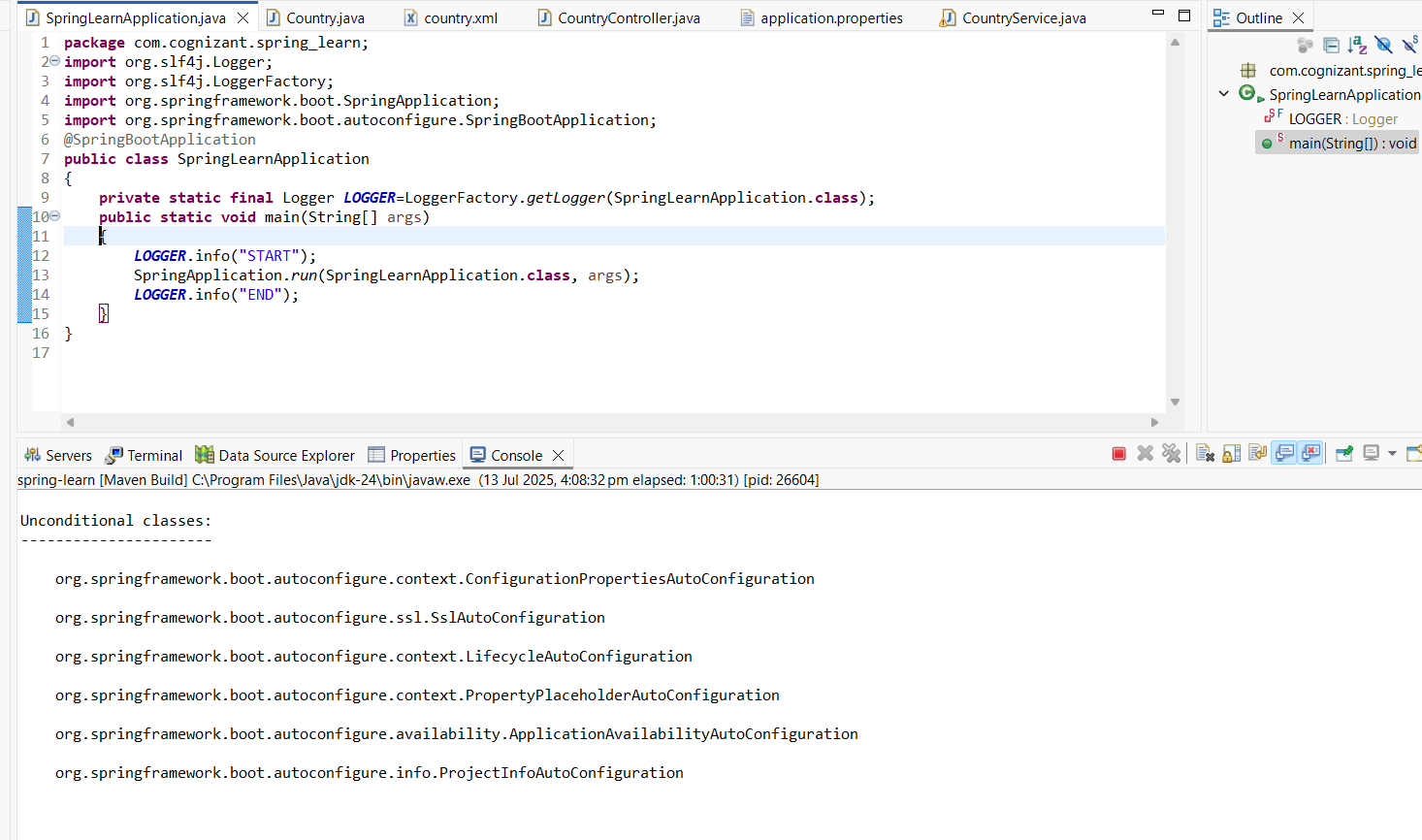
SpringApplication.*run*(SpringLearnApplication.**class**, args);

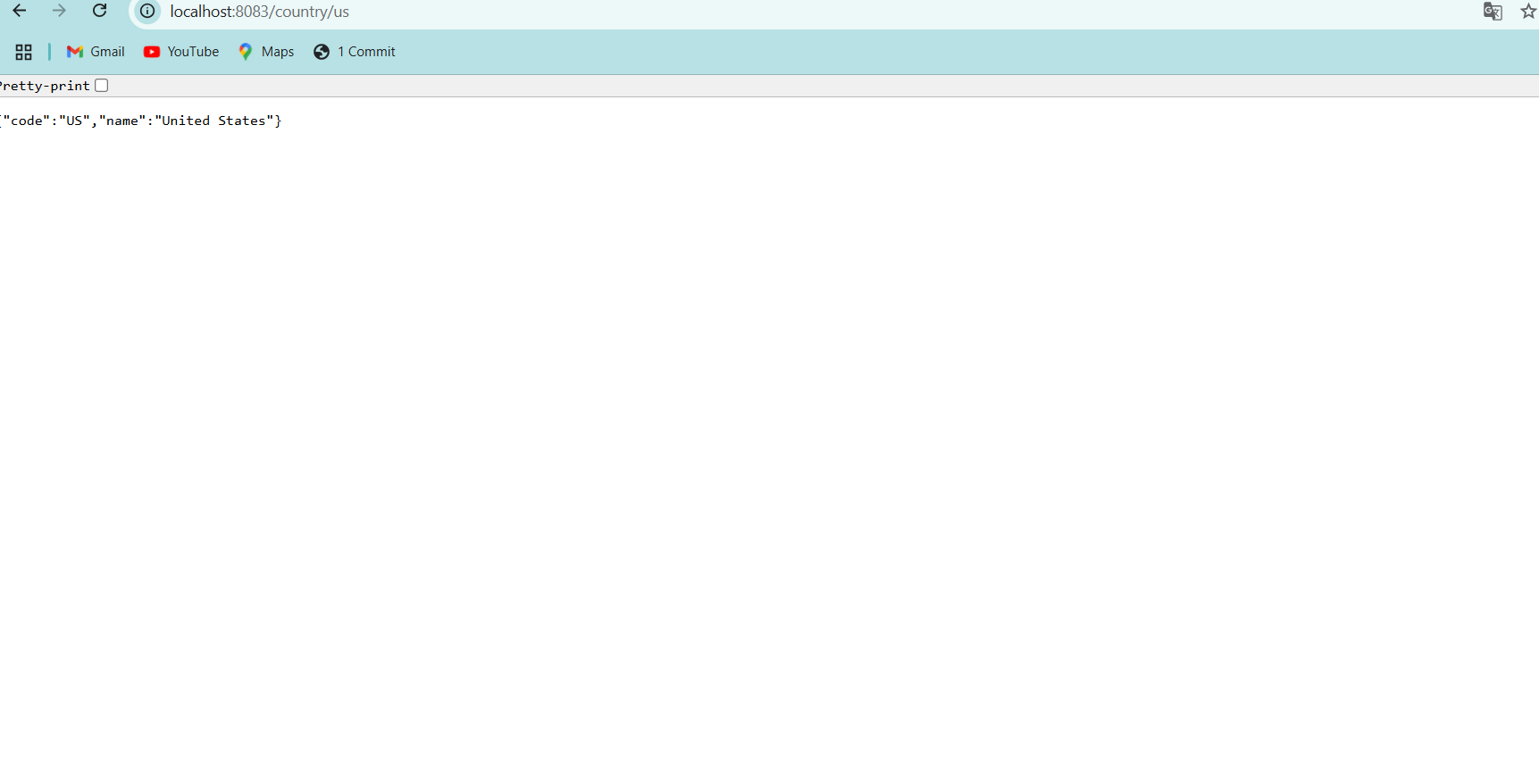
***LOGGER***.info("END");

}

}

**Output**

****

****

**Create authentication service that returns JWT**

**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.cognizant</groupId>

<artifactId>spring-learn</artifactId>

<version>0.0.1-SNAPSHOT</version>

<name>spring-learn</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-devtools</artifactId>

<scope>runtime</scope>

<optional>true</optional>

</dependency>

<dependency>

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

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

<scope>test</scope>

</dependency>

<dependency>

<groupId>io.jsonwebtoken</groupId>

<artifactId>jjwt</artifactId>

<version>0.9.1</version>

</dependency>

<dependency>

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

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

</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-learn

logging.level.root=DEBUG

server.port=8085

**SecurityConfig.java**

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

**import** org.springframework.context.annotation.Configuration;

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

**import** org.springframework.security.config.annotation.web.configuration.EnableWebSecurity;

**import** org.springframework.security.web.SecurityFilterChain;

**import** org.springframework.context.annotation.Bean;

@Configuration

@EnableWebSecurity

**public** **class** SecurityConfig

{

@Bean

**public** SecurityFilterChain filterChain(HttpSecurity http) **throws** Exception

{

http.csrf().disable()

.authorizeHttpRequests(authz -> authz

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

.anyRequest().authenticated()

)

.httpBasic();

**return** http.build();

}

}

**JwtUtil.java**

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

**import** io.jsonwebtoken.Jwts;

**import** io.jsonwebtoken.SignatureAlgorithm;

**import** org.springframework.stereotype.Component;

**import** java.util.Date;

@Component

**public** **class** JwtUtil

{

**private** **final** String SECRET\_KEY = "secret";

**public** String generateToken(String username)

{

**return** Jwts.*builder*()

.setSubject(username)

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

.setExpiration(**new** Date(System.*currentTimeMillis*() +1000\*60\*10))

.signWith(SignatureAlgorithm.***HS256***, SECRET\_KEY)

.compact();

}

}

**AuthenticationController.java**

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

**import** org.springframework.http.ResponseEntity;

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

**import** jakarta.servlet.http.HttpServletRequest;

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

**import** java.util.Map;

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

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

**import** java.util.Base64;

**import** java.util.HashMap;

@RestController

**public** **class** AuthenticationController

{

@Autowired

**private** JwtUtil jwtUtil;

@GetMapping("/authenticate")

**public** Map<String, String> authenticate(@RequestHeader("Authorization") String authHeader)

{

String[] tokens=extractUsernameAndPassword(authHeader);

String username=tokens[0];

String password=tokens[1];

**if** (username.equals("user")&&password.equals("pwd"))

{

String token=jwtUtil.generateToken(username);

**return** Map.*of*("token",token);

}

**else**

{

**throw** **new** RuntimeException("Invalid Credentials");

}

}

**private** String[] extractUsernameAndPassword(String header)

{

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

{

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

}

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

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

String credentials=**new** String(decoded);

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

}

}

**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)

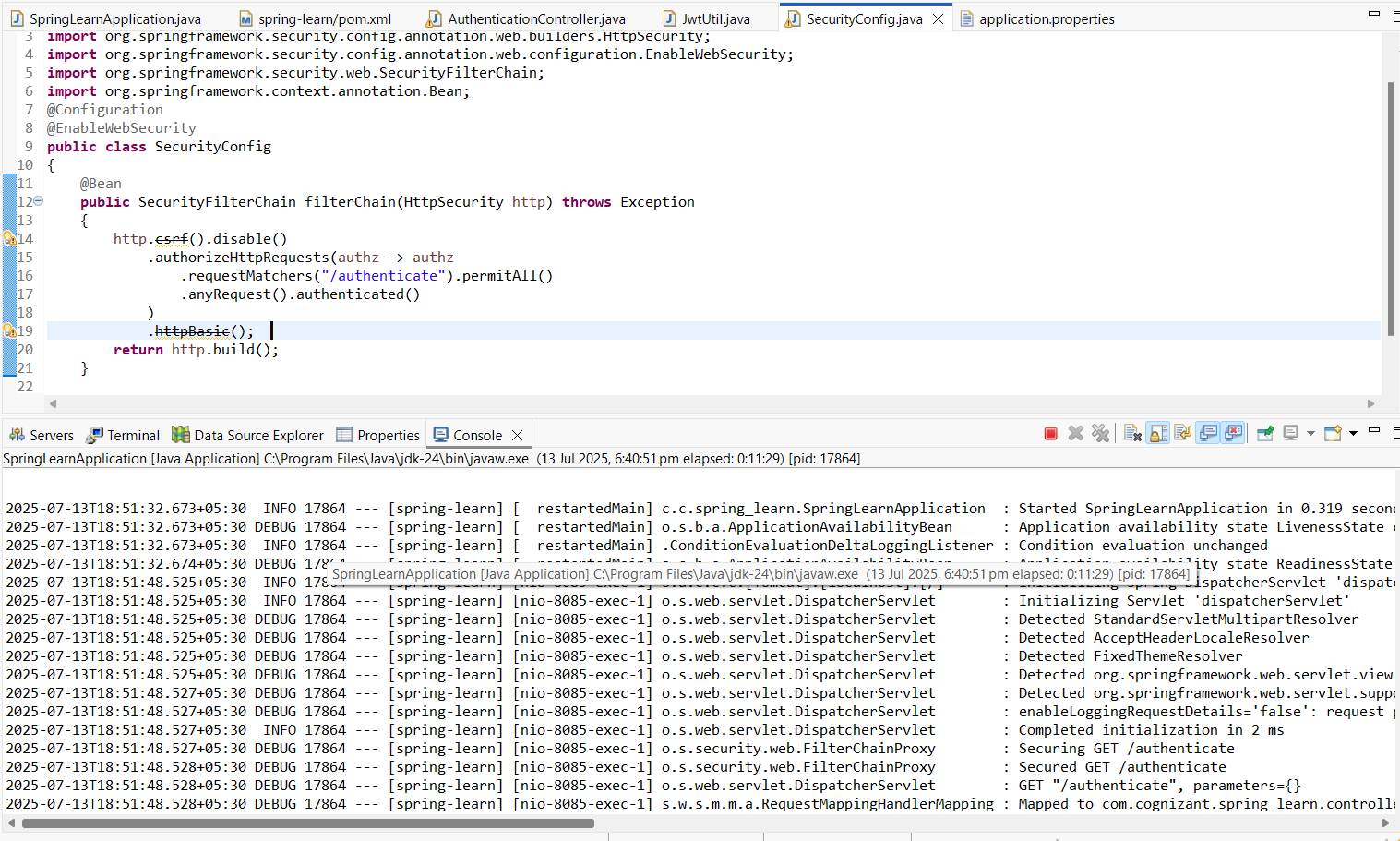
{

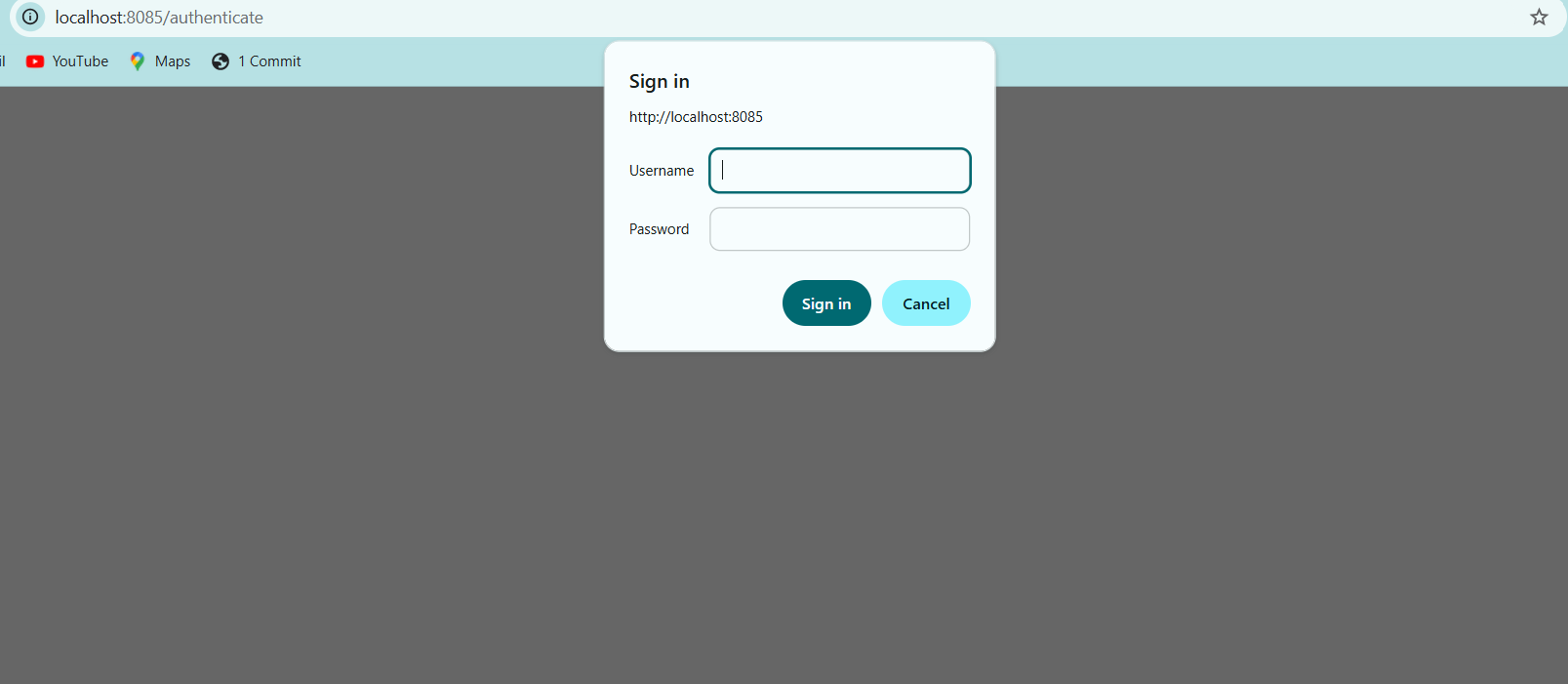
SpringApplication.*run*(SpringLearnApplication.**class**, args);

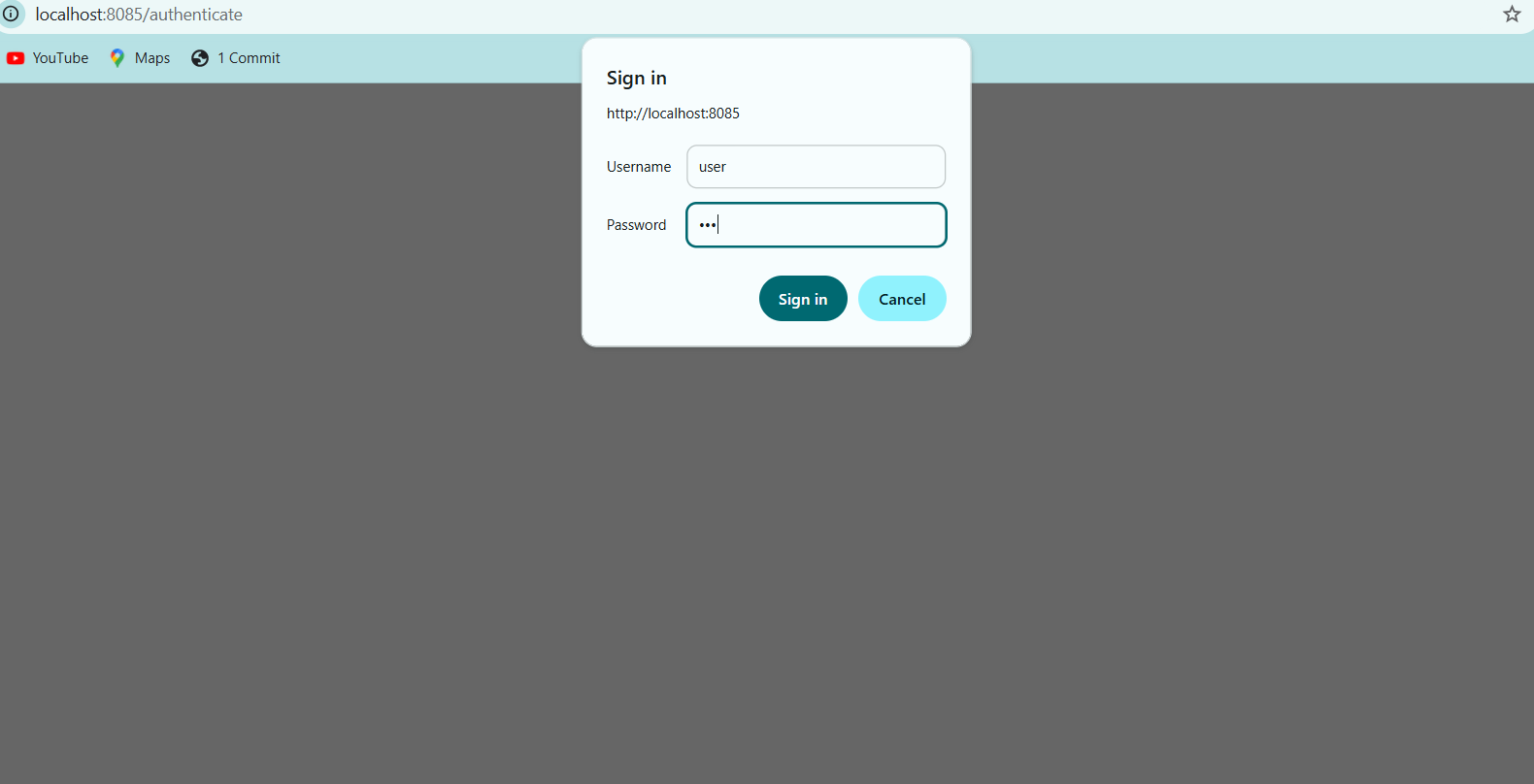
}

}

**Output**

****

****

****