**SPRING REST USING SPRING BOOT 3**

**Create a Spring Web Project using Maven**

**spring-learn\spring-learn\src\main\java\com\cognizant\spring\_learn\SpringLearnApplication.java**

package com.cognizant.spring\_learn;

import org.springframework.boot.SpringApplication;

import org.springframework.boot.autoconfigure.SpringBootApplication;

@SpringBootApplication

public class SpringLearnApplication {

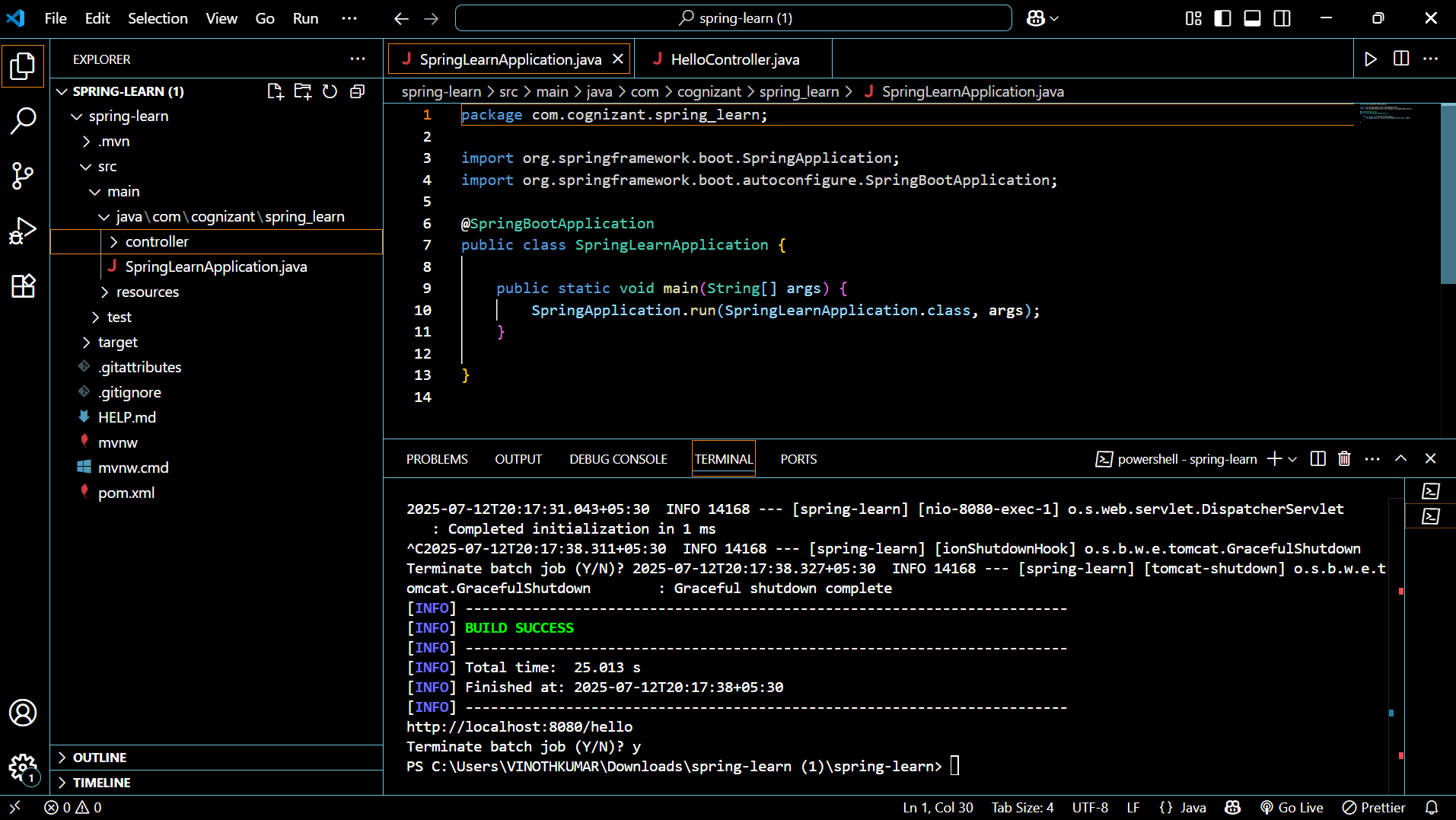
    public static void main(String[] args) {

        SpringApplication.run(SpringLearnApplication.class, args);

    }

}

**OUTPUT :**

****

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

**spring-learn\spring-learn\src\main\java\com\cognizant\spring\_learn\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.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) {

        SpringApplication.run(SpringLearnApplication.class, args);

        displayCountry();

    }

    public static void displayCountry() {

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

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

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

    }

}

**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"/>

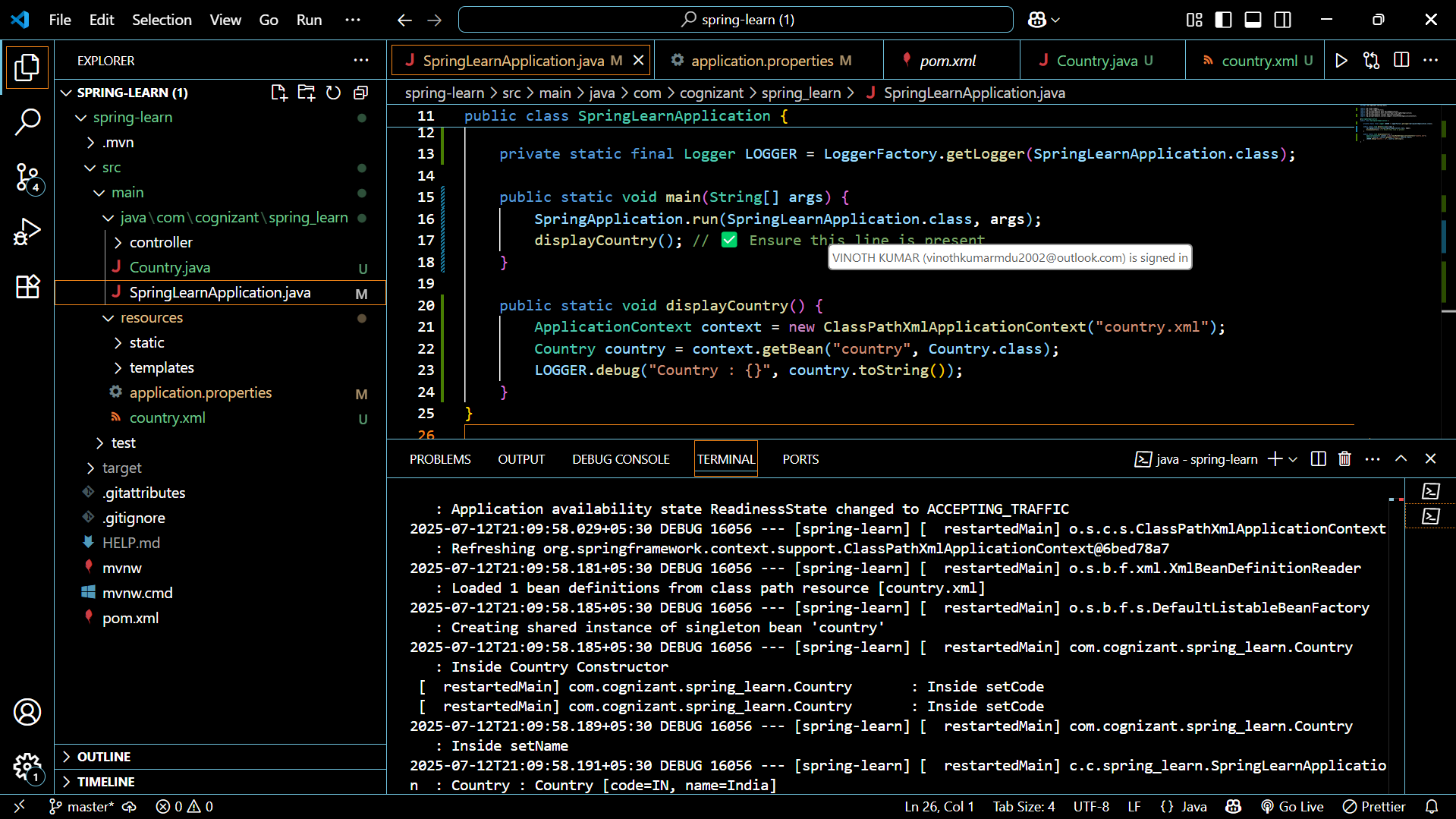
        <property name="name" value="India"/>



    </bean>

</beans>

**OUTPUT :**

****



**Hello World RESTful Web Service**

**spring-learn\spring-learn\src\main\java\com\cognizant\spring\_learn\controller\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 message = "Hello World!!";

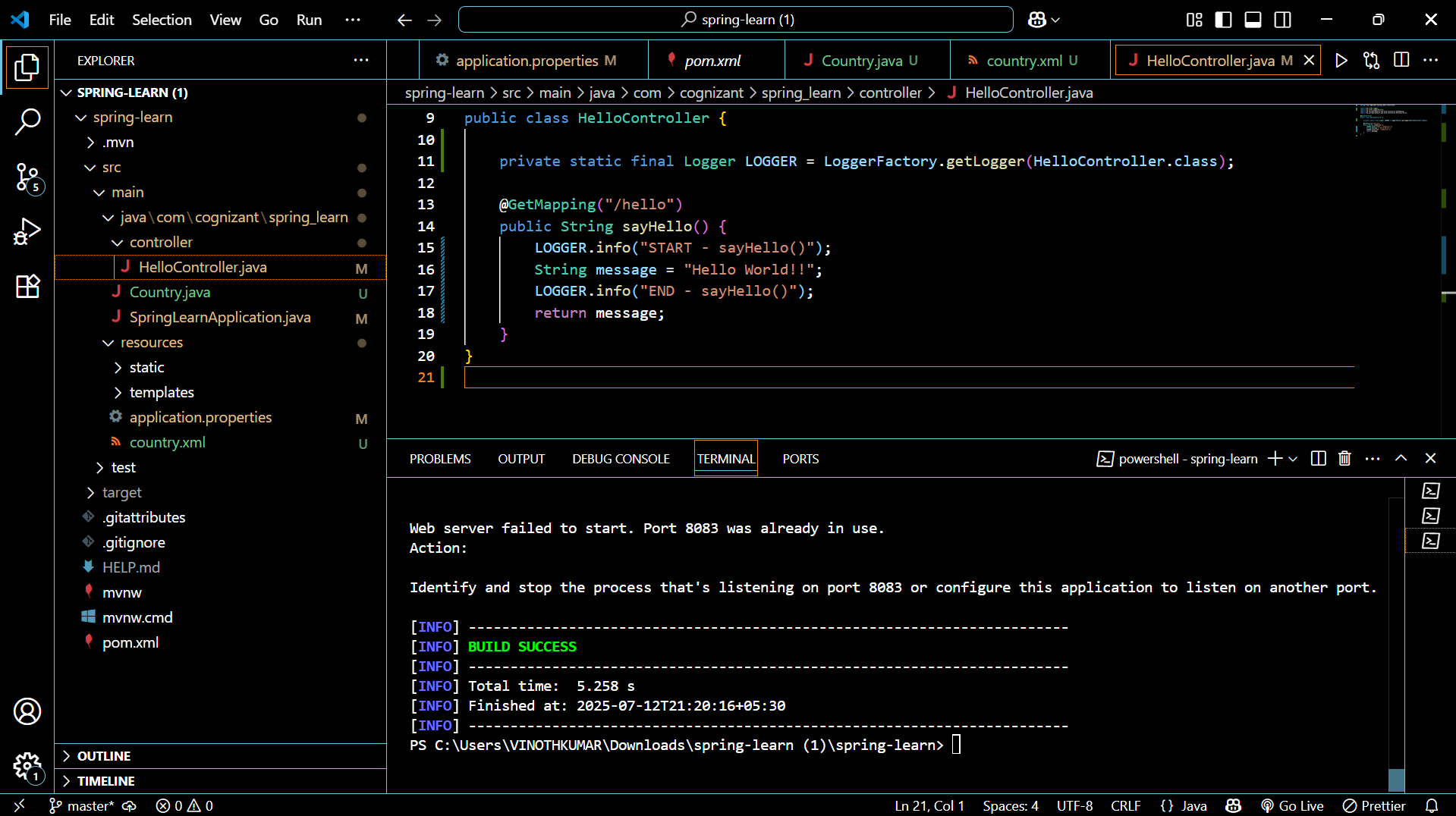
        LOGGER.info("END - sayHello()");

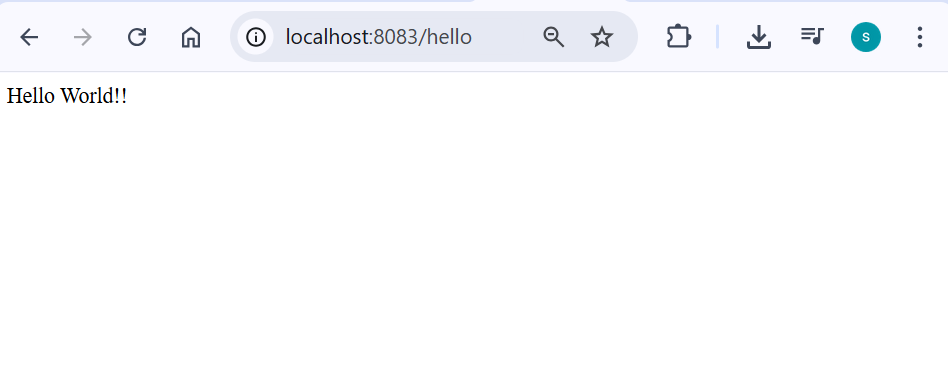
        return message;

    }

}

**OUTPUT :**

****

****

**REST - Country Web Service**

**spring-learn\spring-learn\src\main\java\com\cognizant\spring\_learn\controller\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.info("END - getCountryIndia()");

        return country;

    }

}

**spring-learn\spring-learn\src\main\java\com\cognizant\spring\_learn\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

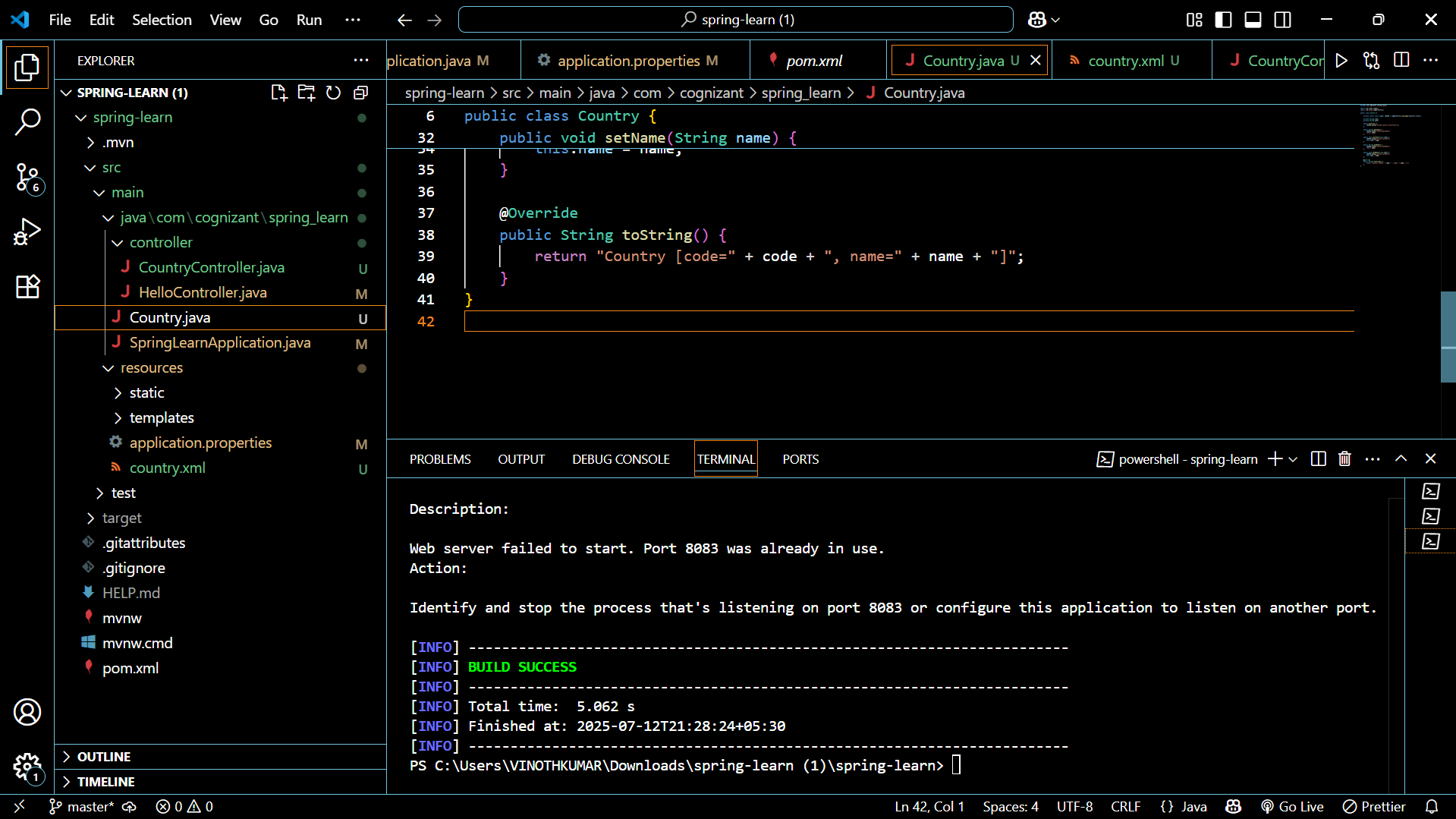
    public String toString() {

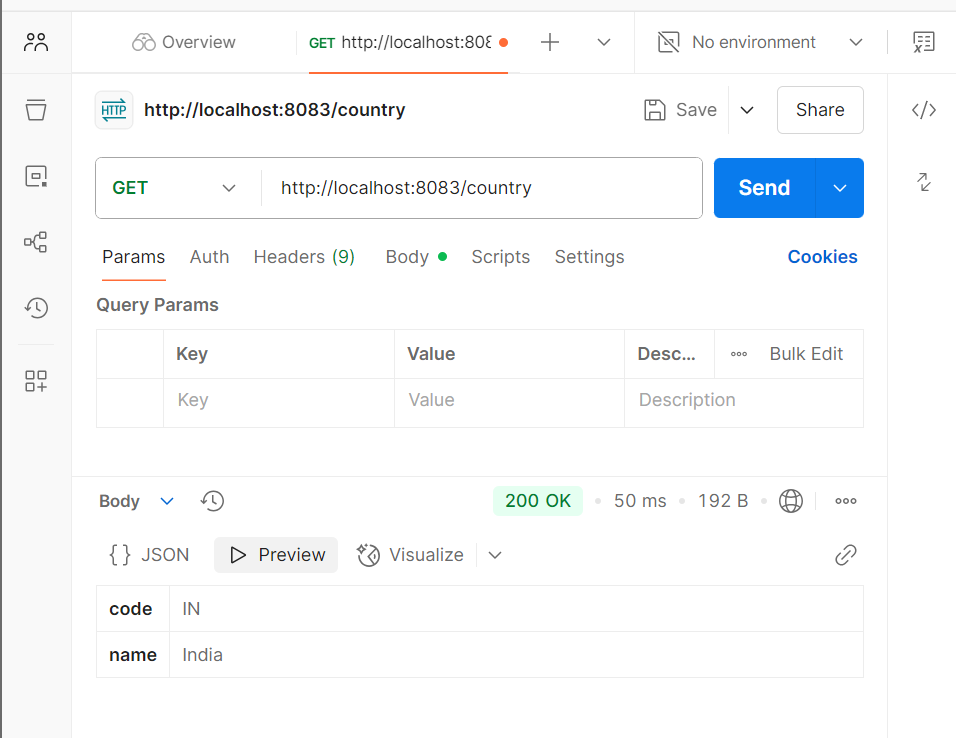
        return "Country [code=" + code + ", name=" + name + "]";

    }

}

**OUTPUT :**

****

****

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

**spring-learn\spring-learn\src\main\java\com\cognizant\spring\_learn\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.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) {

        SpringApplication.run(SpringLearnApplication.class, args);

        displayCountry();

    }

    public static void displayCountry() {

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

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

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

    }

}

**spring-learn\spring-learn\src\main\java\com\cognizant\spring\_learn\service\CountryService.java**

package com.cognizant.spring\_learn.service;

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

        return countryList.stream()

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

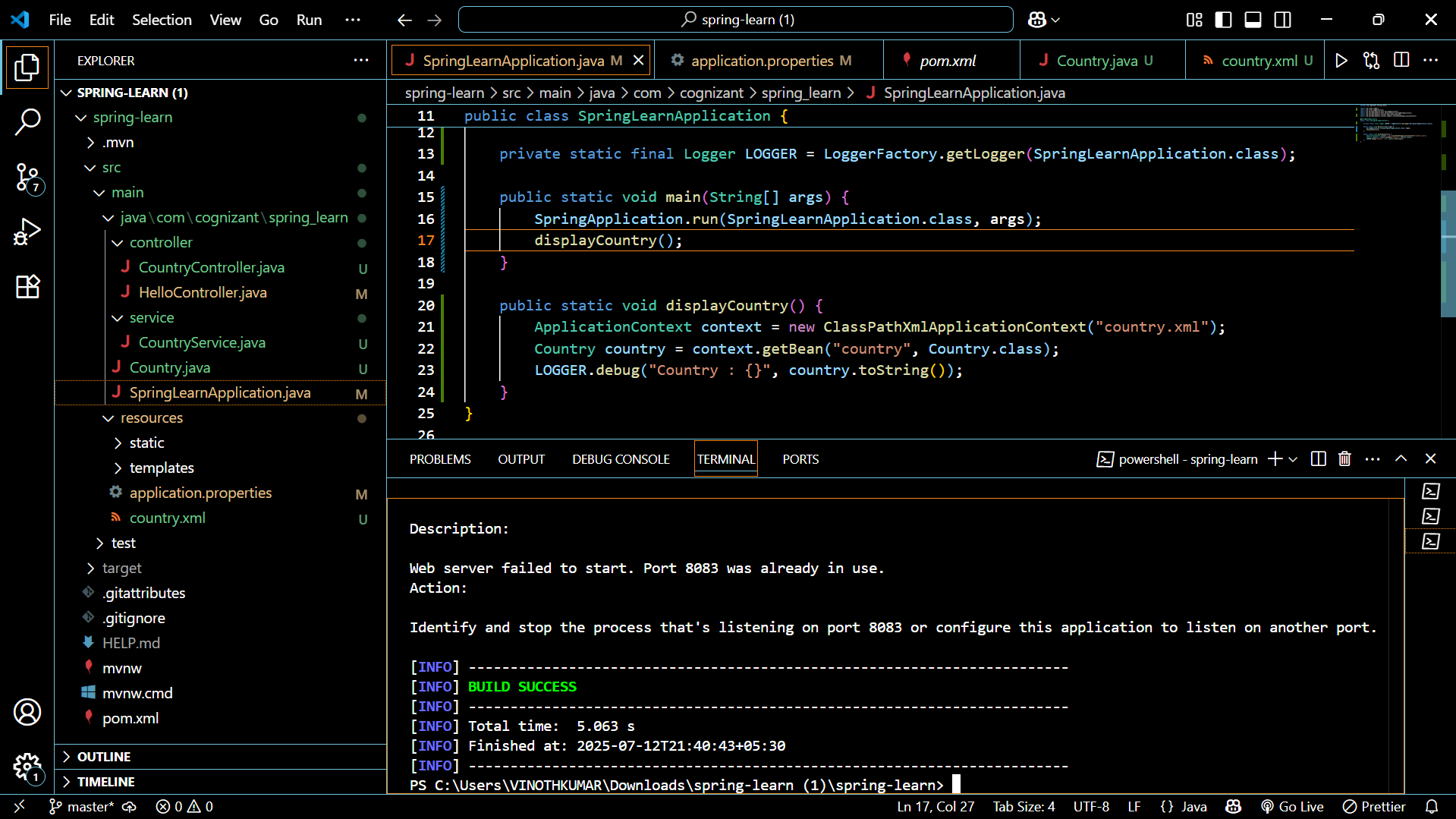
                .findFirst()

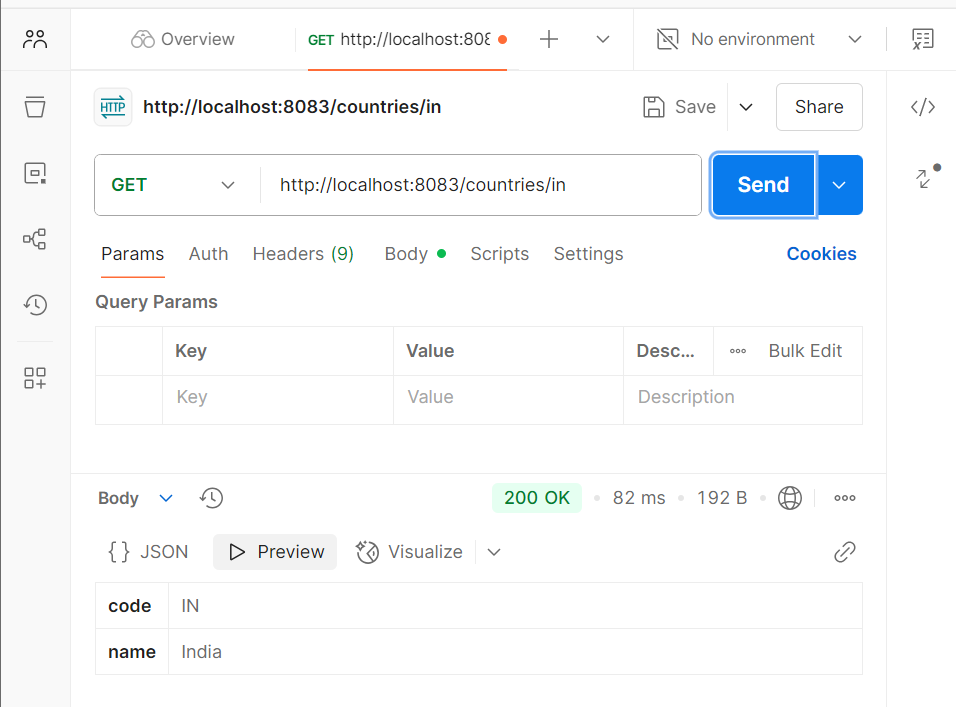
                .orElse(null);

    }

}

**OUTPUT :**

****

****

**Create authentication service that returns JWT**

**spring-learn\spring-learn\src\main\java\com\cognizant\spring\_learn\config\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.annotation.web.configuration.EnableWebSecurity;

import org.springframework.security.web.SecurityFilterChain;

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

@Configuration

@EnableWebSecurity

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();

        return http.build();

    }

}

**spring-learn\spring-learn\src\main\java\com\cognizant\spring\_learn\controller\AuthenticationController.java**

package com.cognizant.spring\_learn.controller;

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

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

import org.springframework.http.ResponseEntity;

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

import java.util.Base64;

@RestController

public class AuthenticationController {

    @Autowired

    private JwtUtil jwtUtil;

    @GetMapping("/authenticate")

    public ResponseEntity<?> authenticate(@RequestHeader("Authorization") String authHeader) {

        // Decode Basic Auth

        String[] parts = authHeader.split(" ");

        if (parts.length != 2 || !parts[0].equals("Basic")) {

            return ResponseEntity.badRequest().body("Invalid Authorization Header");

        }

        String decoded = new String(Base64.getDecoder().decode(parts[1]));

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

        if (credentials.length != 2 || !credentials[0].equals("user") || !credentials[1].equals("pwd")) {

            return ResponseEntity.status(401).body("Invalid Credentials");

        }

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

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

    }

}

**spring-learn\spring-learn\src\main\java\com\cognizant\spring\_learn\util\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 String secretKey = "mySecretKey";

    public String generateToken(String username) {

        return Jwts.builder()

                .setSubject(username)

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

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

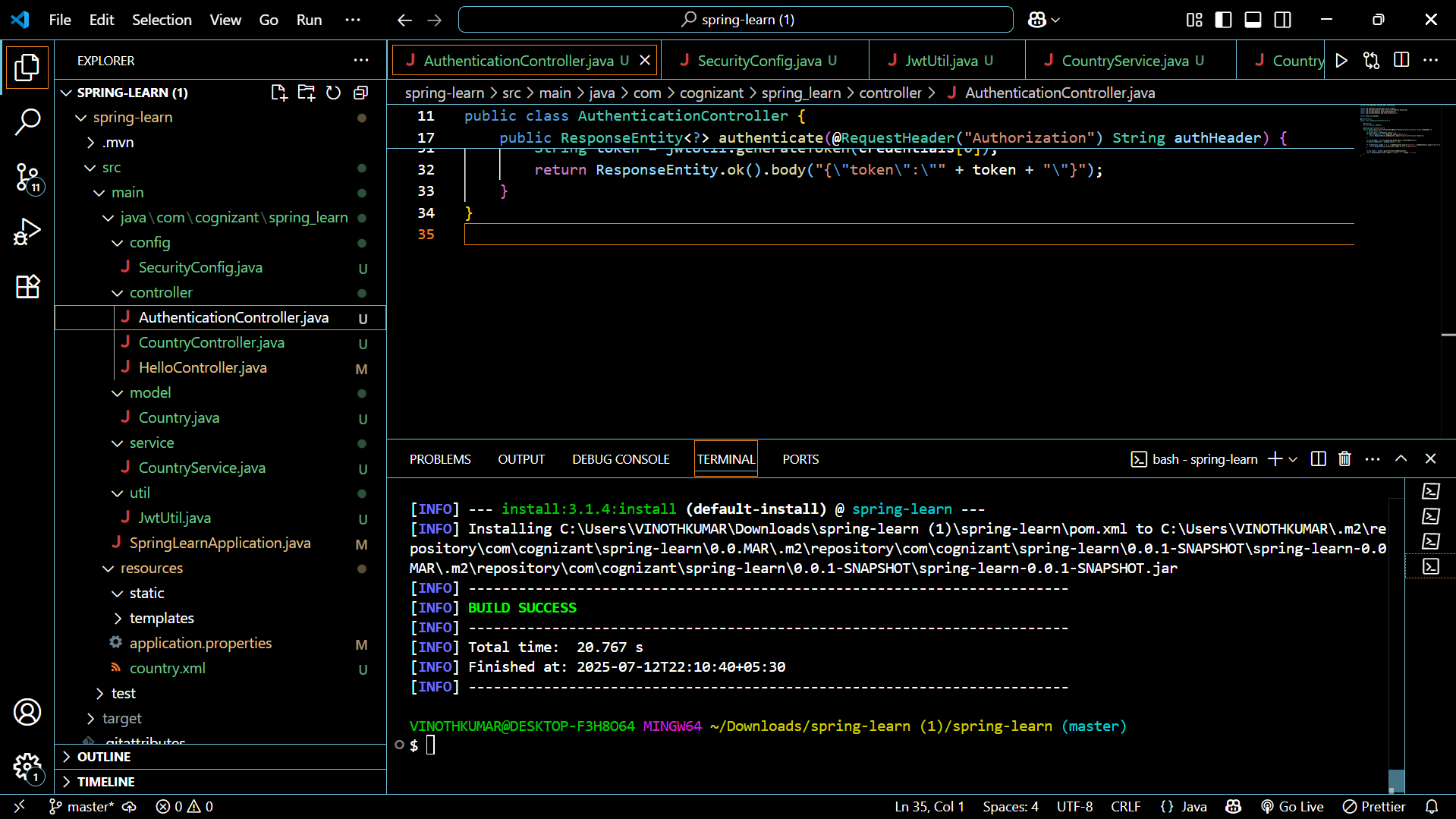
                .signWith(SignatureAlgorithm.HS256, secretKey)

                .compact();

    }

}

**OUTPUT :**

****