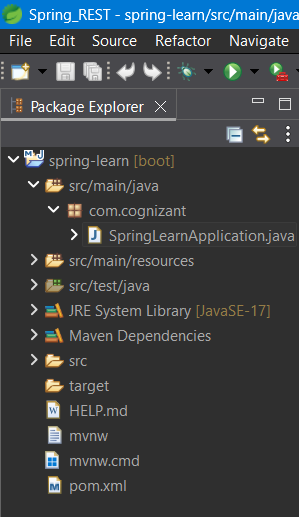
Spring REST

# Assignment Filename: 1. spring-rest-handson.docx

**Hands on 1**

**Create a Spring Web Project using Maven**  ****

**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-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>

**SpringLearnApplication.java**

package com.cognizant;

import org.springframework.boot.SpringApplication;

import org.springframework.boot.autoconfigure.SpringBootApplication;

*@SpringBootApplication*

public class SpringLearnApplication {

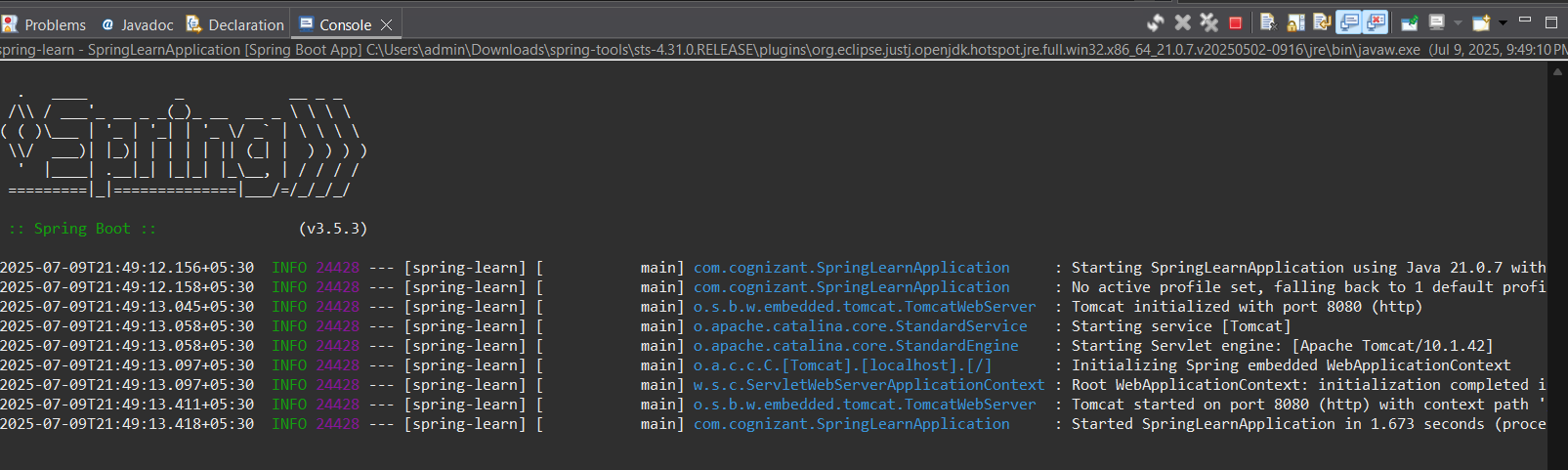
public static void main(String[] args) {

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

}

}

**Output:**

****

**Hands on 2**

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

**date-format.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="dateFormat" class="java.text.SimpleDateFormat">

<constructor-arg value="dd/MM/yyyy"></constructor-arg>

</bean>

</beans>

**SpringLearnApplication.java**

package com.cognizant;

import java.text.SimpleDateFormat;

import java.util.Date;

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 {

public static void main(String[] args) throws Exception {

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

*displayDate*();

}

public static void displayDate() throws Exception{

ApplicationContext context=new ClassPathXmlApplicationContext("date-format.xml");

SimpleDateFormat format=context.getBean(SimpleDateFormat.class);

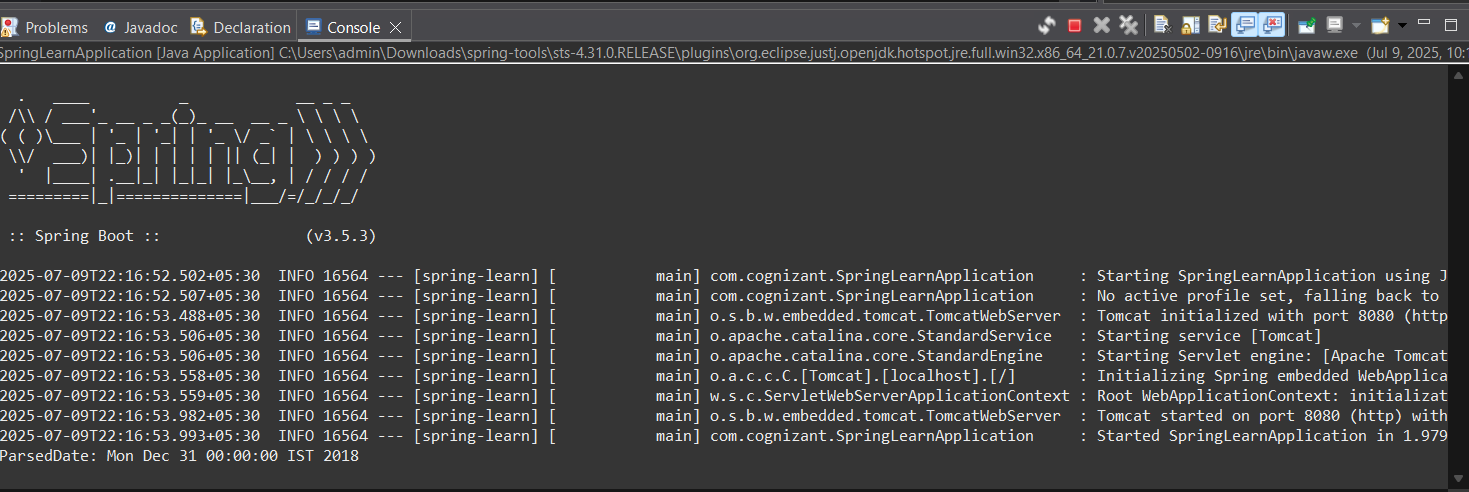
Date parsedDate=format.parse("31/12/2018");

System.***out***.println("ParsedDate: "+parsedDate);

}

}

**Output:**



**Hands on 3**

**Spring Core - Incorporate Logging**

application.properties

spring.application.name=spring-learn

logging.level.org.springframework=info

logging.level.com.cognizant=debug

logging.pattern.console=%d{yyMMdd}|%d{HH:mm:ss.SSS}|%-20.20thread|%5p|%-25.25logger{25}|%25M|%m%n

SpringLearnApplication.java

package com.cognizant;

import java.text.SimpleDateFormat;

import java.util.Date;

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) throws Exception {

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

*displayDate*();

}

public static void displayDate() throws Exception{

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

ApplicationContext context=new ClassPathXmlApplicationContext("date-format.xml");

SimpleDateFormat format=context.getBean(SimpleDateFormat.class);

Date parsedDate=format.parse("31/12/2018");

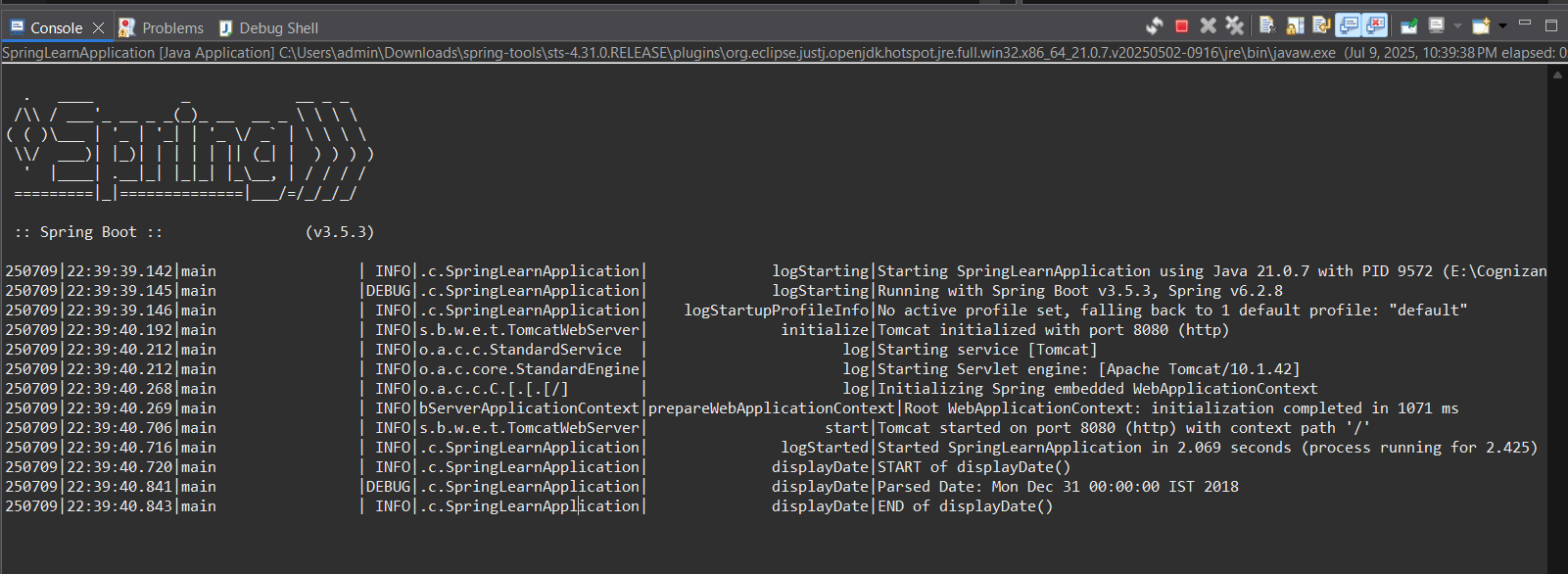
***LOGGER***.debug("Parsed Date: {}", parsedDate);

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

}

}

Output:



**Hands on 4**

**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.Country">

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

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

</bean>

</beans>

Country.java

package com.cognizant;

import org.slf4j.Logger;

import org.slf4j.LoggerFactory;

public class Country {

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

private String code;

private String name;

public Country() {

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

}

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;

}

*@Override*

public String toString() {

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

}

}

# SpringLearnApplication.java

package com.cognizant;

import java.text.SimpleDateFormat;

import java.util.Date;

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) throws Exception {

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

*displayDate*();

*displayCountry*();

}

public static void displayDate() throws Exception{

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

ApplicationContext context=new ClassPathXmlApplicationContext("date-format.xml");

SimpleDateFormat format=context.getBean(SimpleDateFormat.class);

Date parsedDate=format.parse("31/12/2018");

***LOGGER***.debug("Parsed Date: {}", parsedDate);

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

}

public static void displayCountry() throws Exception{

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

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

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

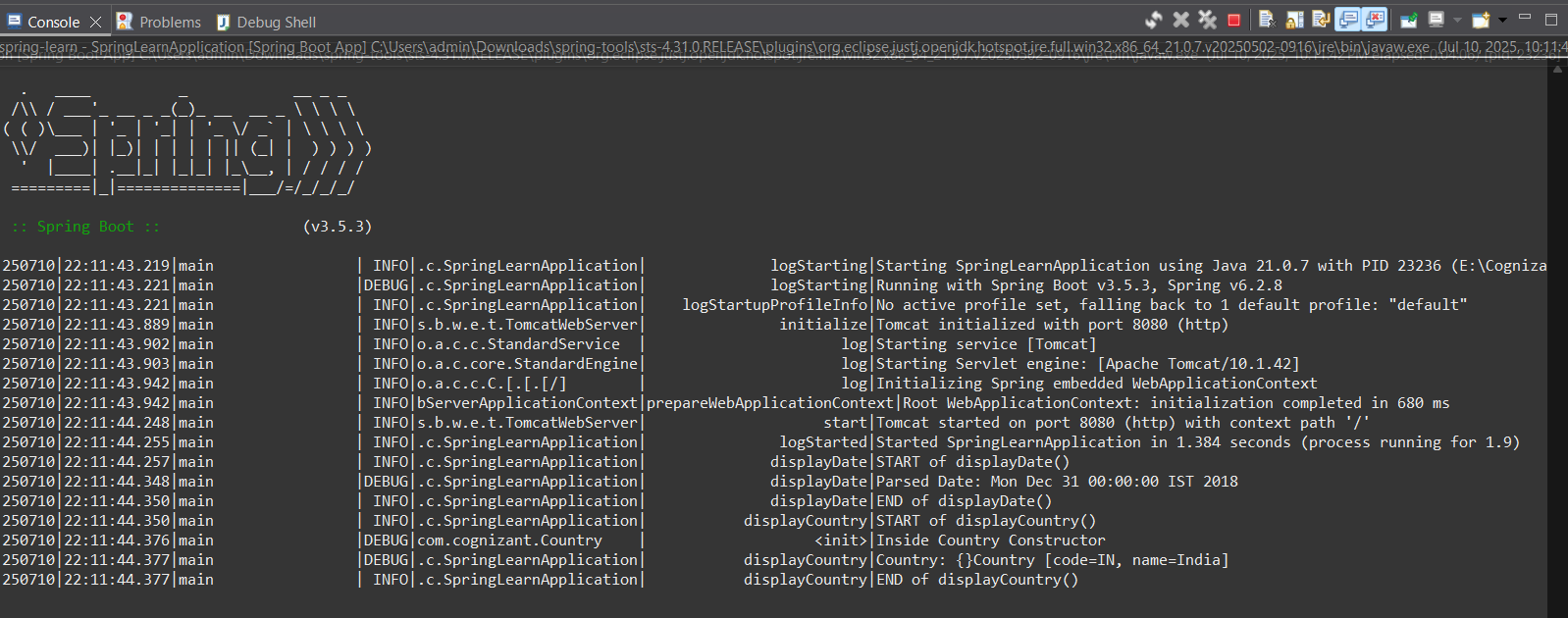
***LOGGER***.debug("Country: {}"+country);

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

}

}

# Output:



**Hands on 5**

**Spring Core – Demonstration of Singleton Scope and Prototype Scope**

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.Country" scope="prototype">

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

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

</bean>

</beans>

SpringLearnApplication.java

package com.cognizant;

import java.text.SimpleDateFormat;

import java.util.Date;

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) throws Exception {

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

*displayDate*();

*displayCountry*();

}

public static void displayDate() throws Exception{

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

ApplicationContext context=new ClassPathXmlApplicationContext("date-format.xml");

SimpleDateFormat format=context.getBean(SimpleDateFormat.class);

Date parsedDate=format.parse("31/12/2018");

***LOGGER***.debug("Parsed Date: {}", parsedDate);

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

}

public static void displayCountry() throws Exception{

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

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

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

Country anothercountry=container.getBean("country",Country.class);

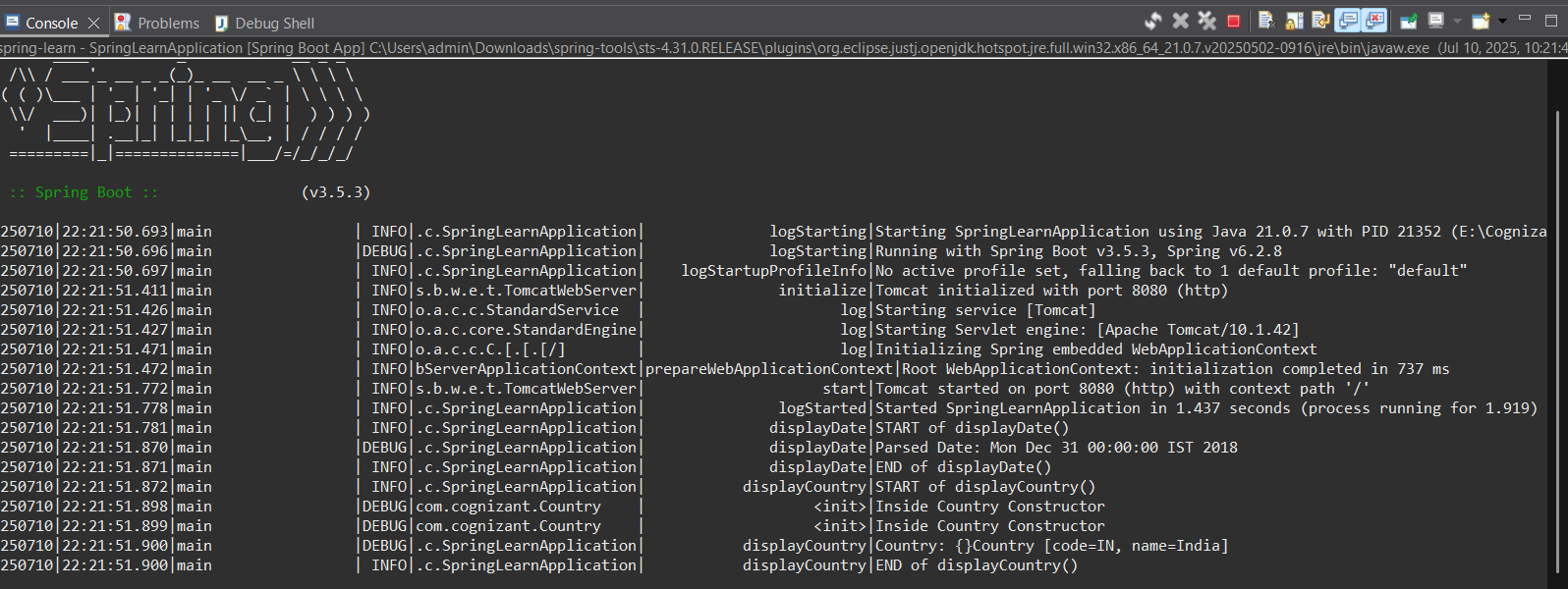
***LOGGER***.debug("Country: {}"+country);

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

}

}

Output:



**Hands on 6**

**Spring Core – Load list of countries 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="in" class="com.cognizant.Country">

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

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

</bean>

<bean id="us" class="com.cognizant.Country">

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

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

</bean>

<bean id="de" class="com.cognizant.Country">

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

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

</bean>

<bean id="jp" class="com.cognizant.Country">

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

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

</bean>

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

<constructor-arg>

<list>

<ref bean="in"/>

<ref bean="us"/>

<ref bean="de"/>

<ref bean="jp"/>

</list>

</constructor-arg>

</bean>

</beans>

SpringLearnApplication.java

package com.cognizant;

import java.text.SimpleDateFormat;

import java.util.Date;

import java.util.List;

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) throws Exception {

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

*displayDate*();

*displayCountries*();

}

public static void displayDate() throws Exception{

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

ApplicationContext context=new ClassPathXmlApplicationContext("date-format.xml");

SimpleDateFormat format=context.getBean(SimpleDateFormat.class);

Date parsedDate=format.parse("31/12/2018");

***LOGGER***.debug("Parsed Date: {}", parsedDate);

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

}

public static void displayCountries() {

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

ApplicationContext context1 = new ClassPathXmlApplicationContext("country.xml");

*@SuppressWarnings*("unchecked")

List<Country> countries = (List<Country>) context1.getBean("countryList");

for (Country country : countries) {

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

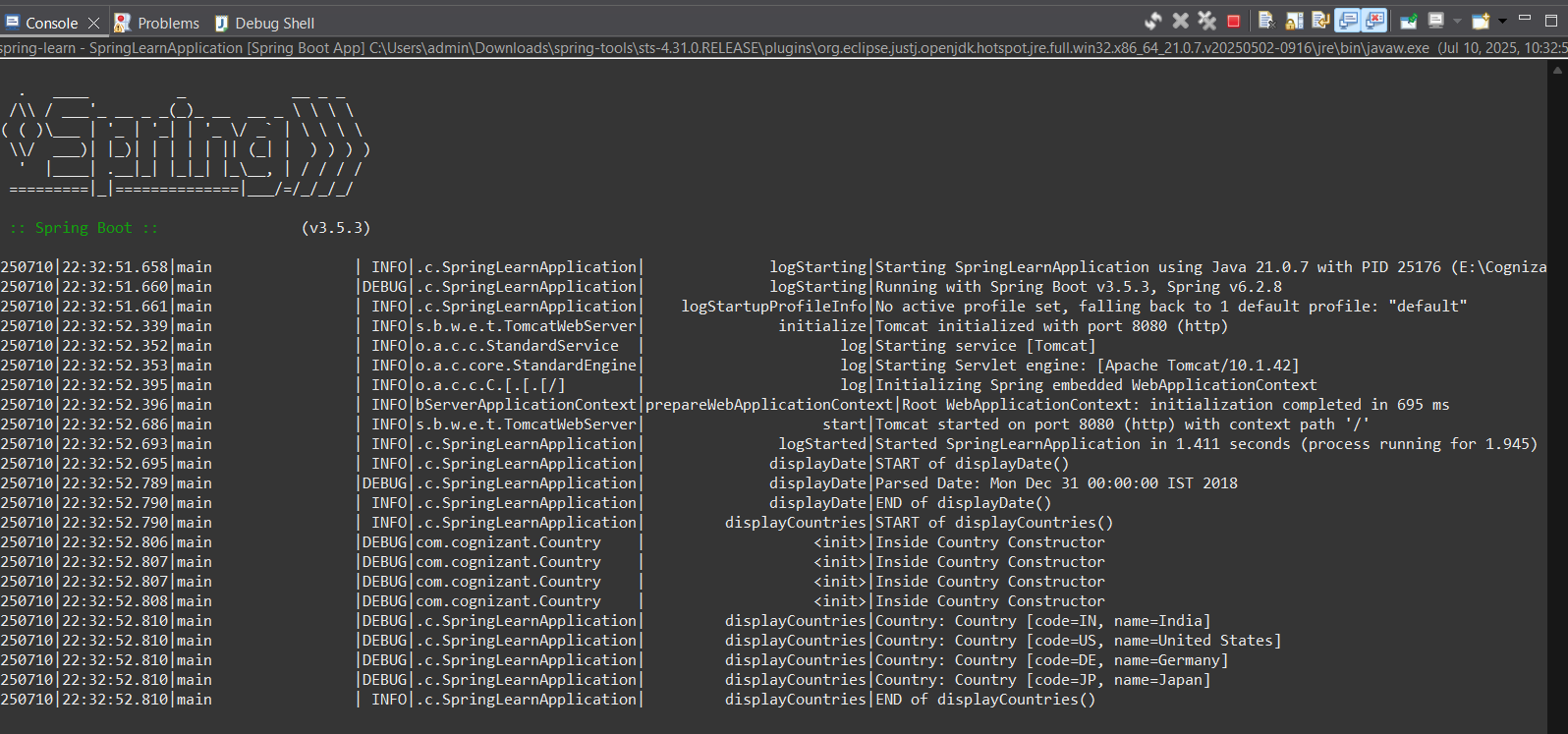
}

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

}

}

Output:



# Assignment Filename: 2. spring-rest-handson.docx

**Hello World RESTful Web Service**

com.cognizant.controllers/HelloController.java

package com.cognizant.controllers;

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;

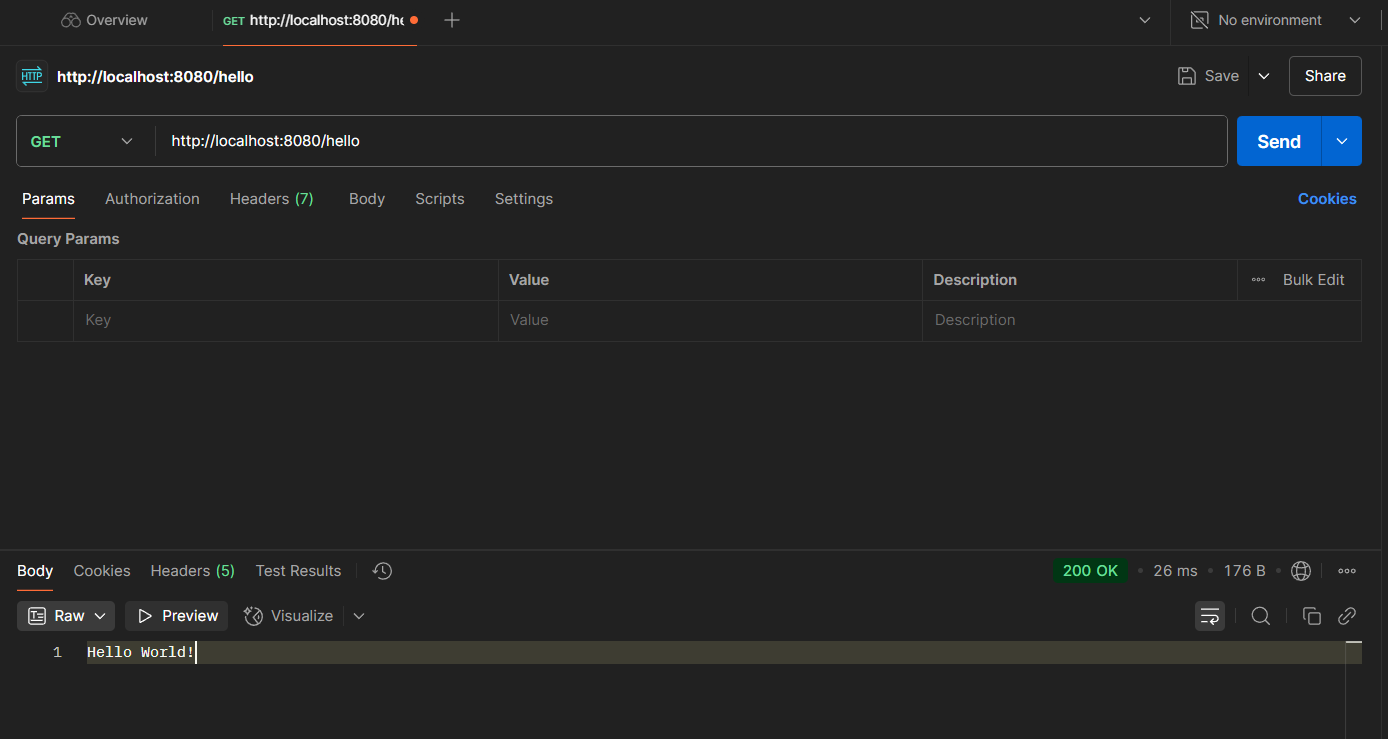
}

}

Browser output



Postman Output



**REST - Country Web Service**

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.Country">

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

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

</bean>

<bean id="us" class="com.cognizant.Country">

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

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

</bean>

<bean id="de" class="com.cognizant.Country">

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

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

</bean>

<bean id="jp" class="com.cognizant.Country">

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

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

</bean>

</beans>

com.cognizant.controllers.CountryController.java

package com.cognizant.controllers;

import org.slf4j.Logger;

import org.slf4j.LoggerFactory;

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

import org.springframework.context.ApplicationContext;

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

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

import com.cognizant.Country;

*@RestController*

public class CountryController {

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

*@Autowired*

private ApplicationContext context;

*@RequestMapping*("/country")

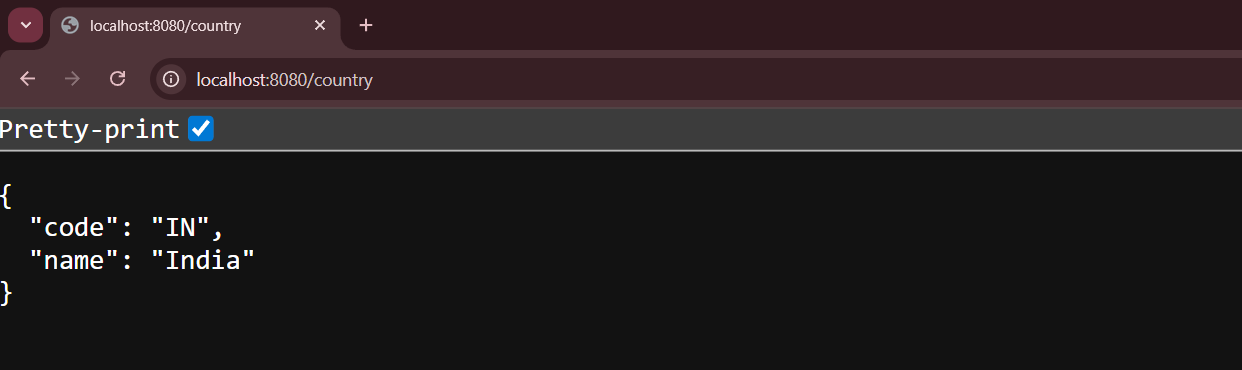
public Country getCountry() {

return (Country) context.getBean("in");

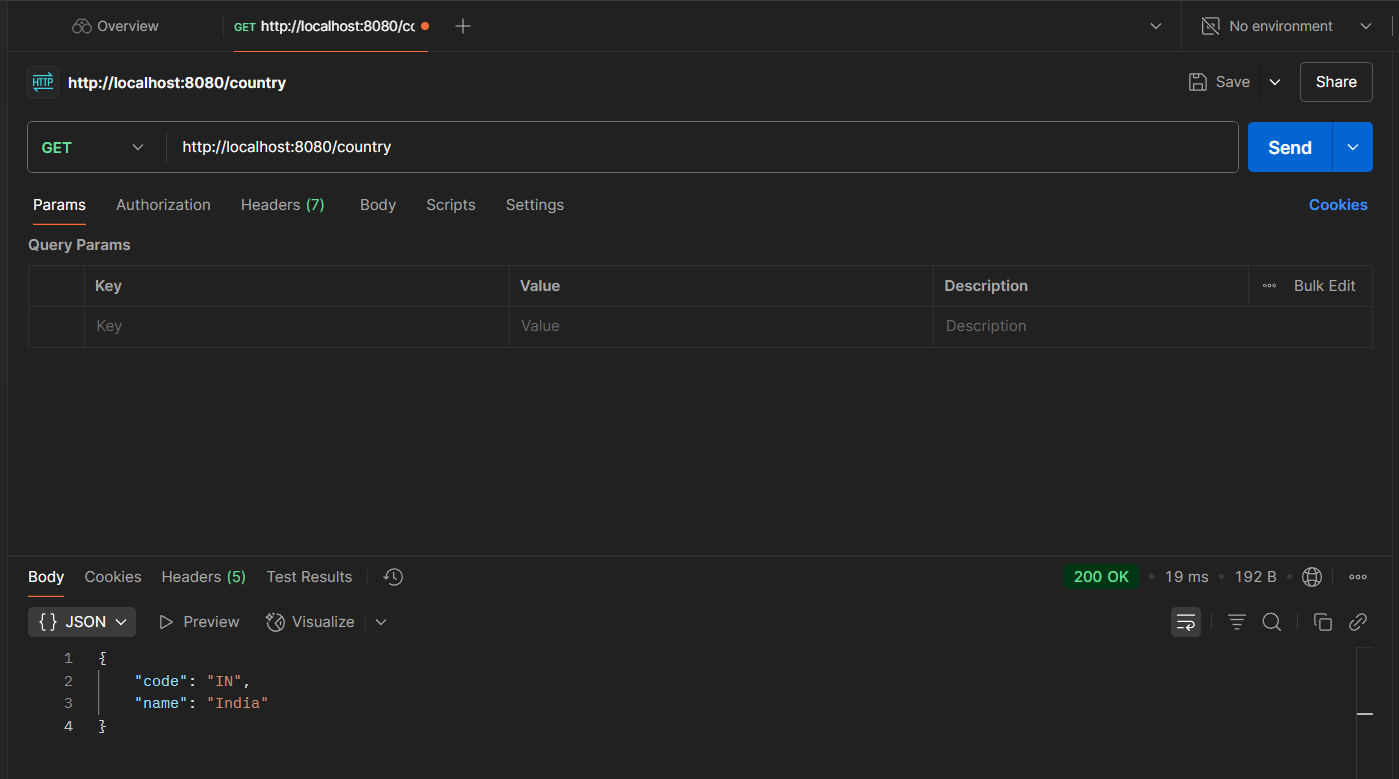
}

}

Browser output:



Postman Output:



**REST - Get all countries**

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.Country">

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

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

</bean>

<bean id="us" class="com.cognizant.Country">

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

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

</bean>

<bean id="de" class="com.cognizant.Country">

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

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

</bean>

<bean id="jp" class="com.cognizant.Country">

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

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

</bean>

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

<constructor-arg>

<list>

<ref bean="in"/>

<ref bean="us"/>

<ref bean="de"/>

<ref bean="jp"/>

</list>

</constructor-arg>

</bean>

</beans>

com.cognizant.controllers.Country.java

package com.cognizant.controllers;

import java.util.List;

import org.slf4j.Logger;

import org.slf4j.LoggerFactory;

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

import org.springframework.context.ApplicationContext;

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

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

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

import com.cognizant.Country;

*@RestController*

public class CountryController {

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

*@Autowired*

private ApplicationContext context;

*@RequestMapping*("/country")

public Country getCountry() {

return (Country) context.getBean("in");

}

*@GetMapping*("/countries")

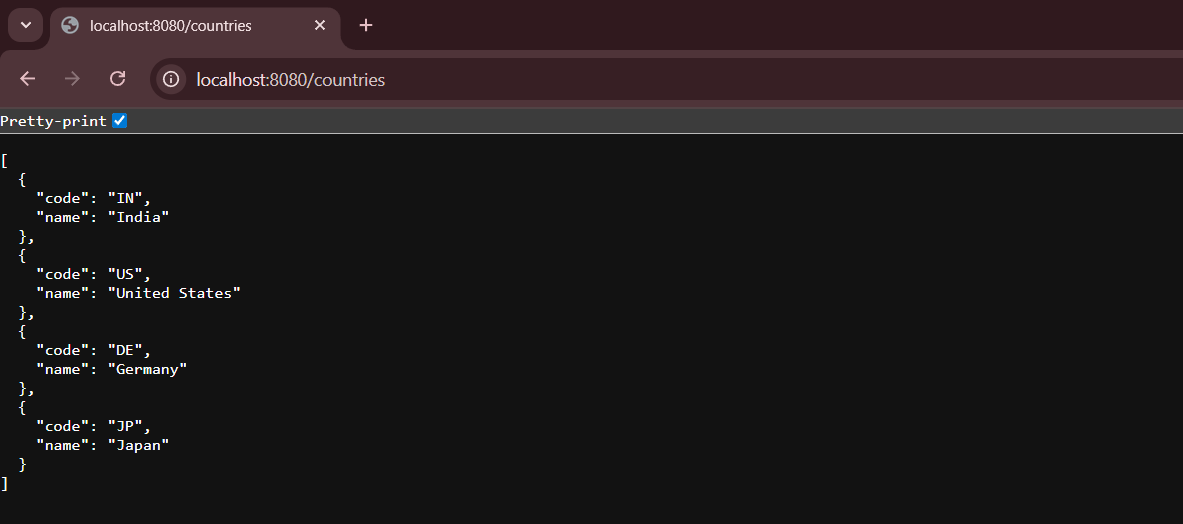
public List<Country> getCountries(){

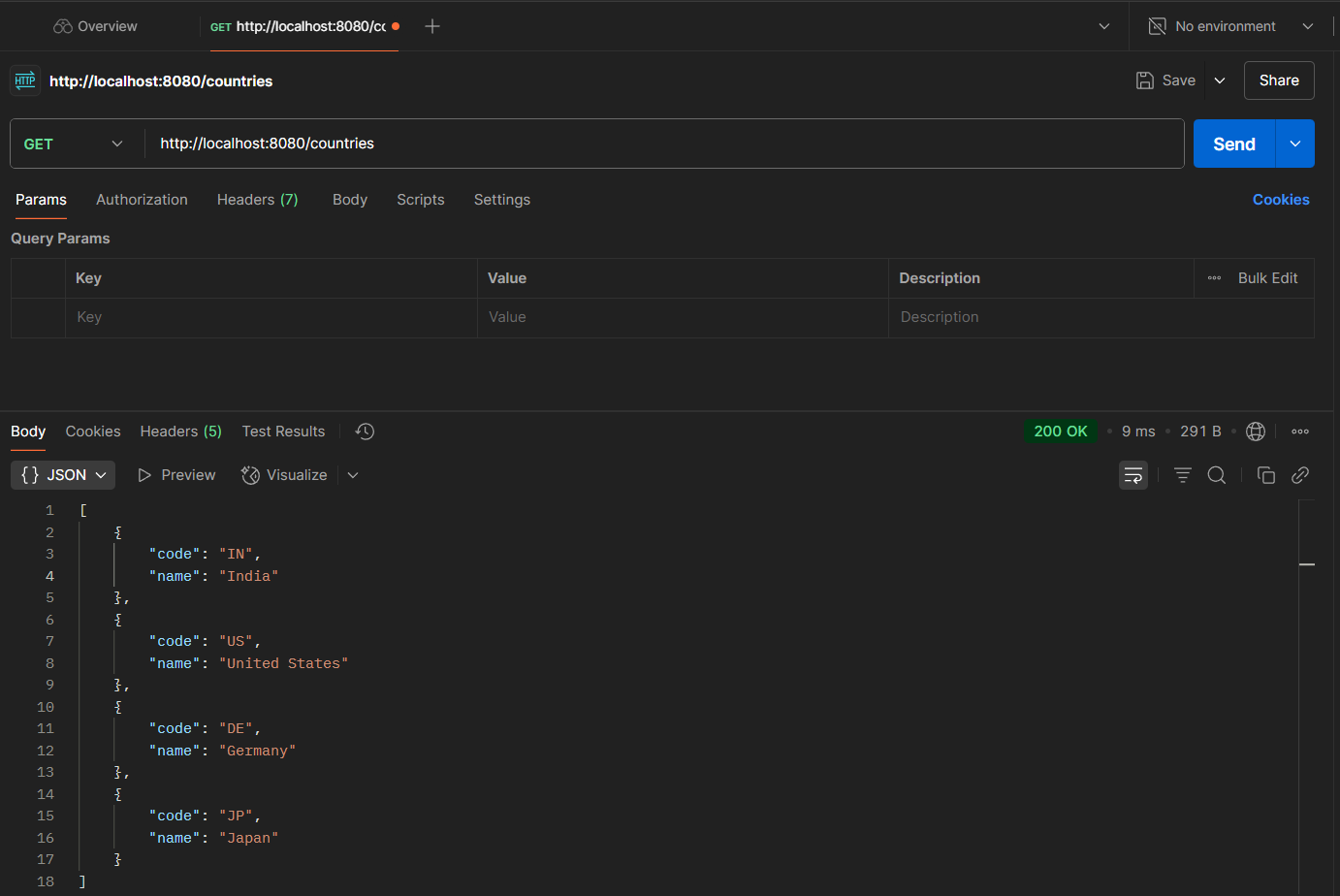
return (List<Country>) context.getBean("countryList");

}

}

Browser Output:



Postman Output: 

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

CountryController.java

package controllers;

import java.util.List;

import org.slf4j.Logger;

import org.slf4j.LoggerFactory;

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

import org.springframework.context.ApplicationContext;

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

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

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

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

import com.cognizant.Country;

import com.cognizant.service.CountryService;

*@RestController*

public class CountryController {

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

*@Autowired*

private ApplicationContext context;

*@Autowired*

private CountryService countryService;

*@RequestMapping*("/country")

public Country getCountry() {

return (Country) context.getBean("in");

}

*@GetMapping*("/countries")

public List<Country> getCountries(){

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

}

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

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

***LOGGER***.info("START /countries/ "+code);

Country country=countryService.getCountry(code);

***LOGGER***.info("END /countries/ "+code);

return country;

}

}

CountryService.java

package com.cognizant.service;

import com.cognizant.Country;

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

import org.springframework.context.ApplicationContext;

import org.springframework.stereotype.Service;

import java.util.List;

*@Service*

public class CountryService {

*@Autowired*

private ApplicationContext context;

public Country getCountry(String code) {

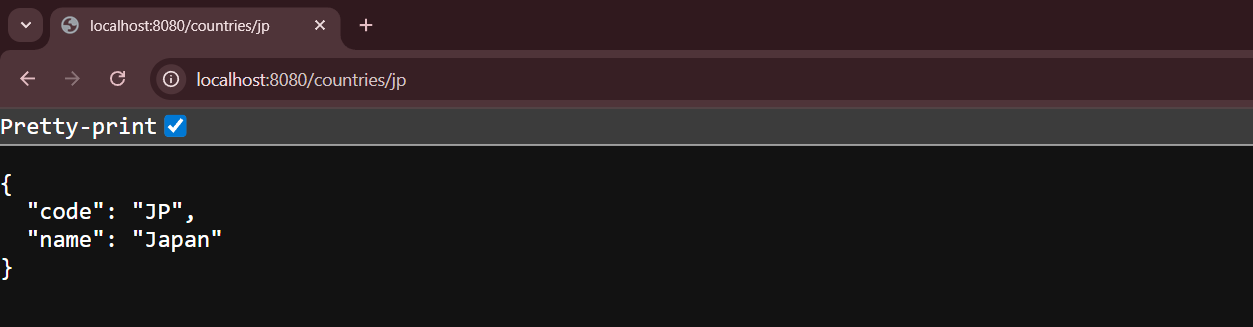
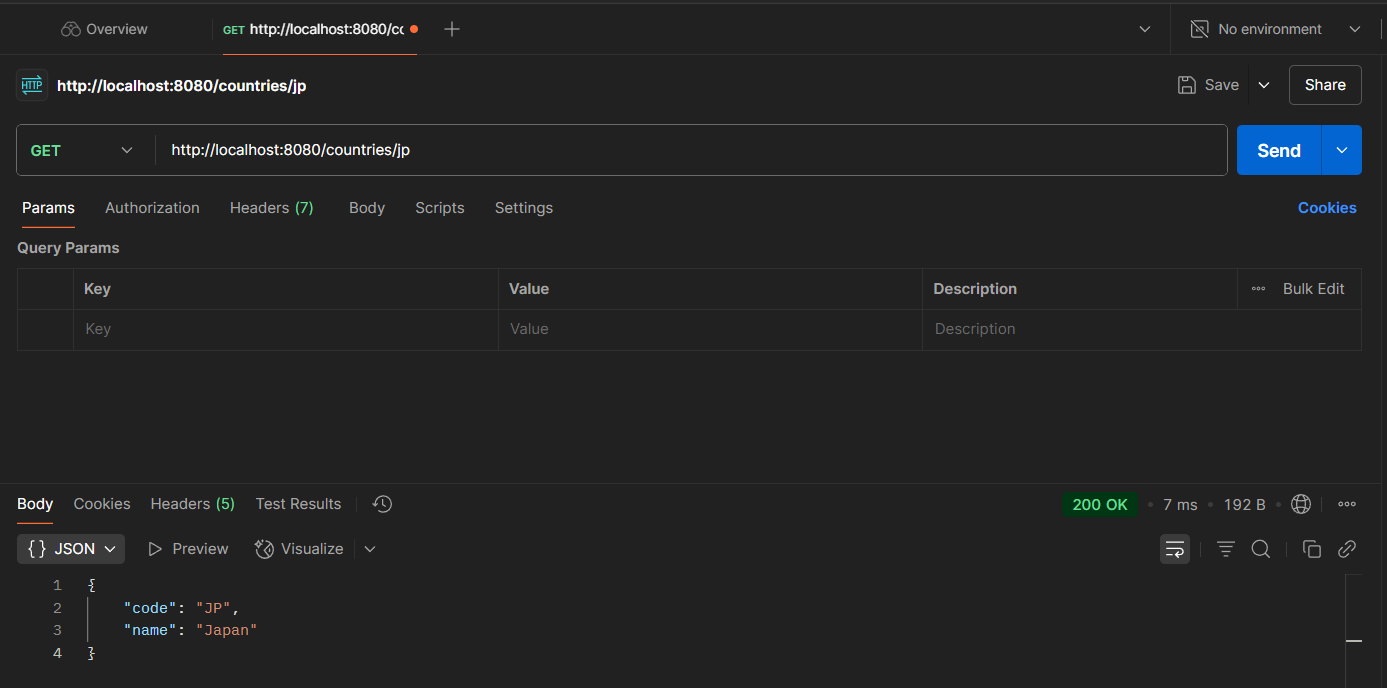
List<Country> countries = (List<Country>) context.getBean("countryList");

return (Country) countries.stream()

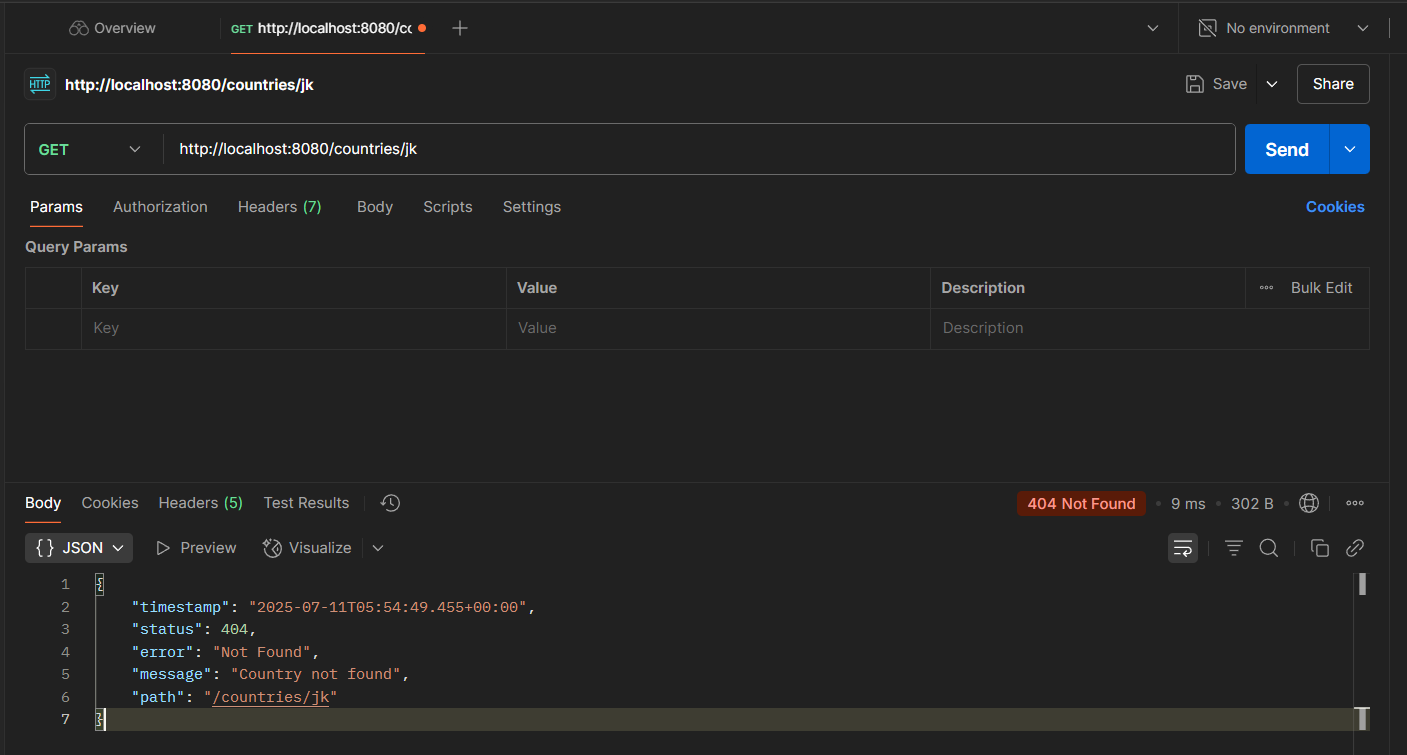
.filter(country -> country.getCode().equalsIgnoreCase(code));

}

}

Browser Output:Postman Output

Exception



**MockMVC - Test get country service**

SpringLearnApplicationTests.java

package com.cognizant;

import static org.junit.jupiter.api.Assertions.*assertNotNull*;

import static org.springframework.test.web.servlet.request.MockMvcRequestBuilders.*get*;

import static org.springframework.test.web.servlet.result.MockMvcResultMatchers.*jsonPath*;

import static org.springframework.test.web.servlet.result.MockMvcResultMatchers.*status*;

import org.junit.jupiter.api.Test;

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

import org.springframework.boot.test.autoconfigure.web.servlet.AutoConfigureMockMvc;

import org.springframework.boot.test.context.SpringBootTest;

import org.springframework.test.web.servlet.MockMvc;

import org.springframework.test.web.servlet.ResultActions;

import controllers.CountryController;

*@SpringBootTest*

*@AutoConfigureMockMvc*

class SpringLearnApplicationTests {

*@Autowired*

private CountryController countryController;

*@Autowired*

private MockMvc mvc;

*@Test*

void contextLoads() {

*assertNotNull*(countryController);

}

*@Test*

public void testGetCountry() throws Exception {

ResultActions actions = mvc.perform(*get*("/country"));

actions.andExpect(*status*().isOk());

actions.andExpect(*jsonPath*("$.code").exists());

actions.andExpect(*jsonPath*("$.code").value("IN"));

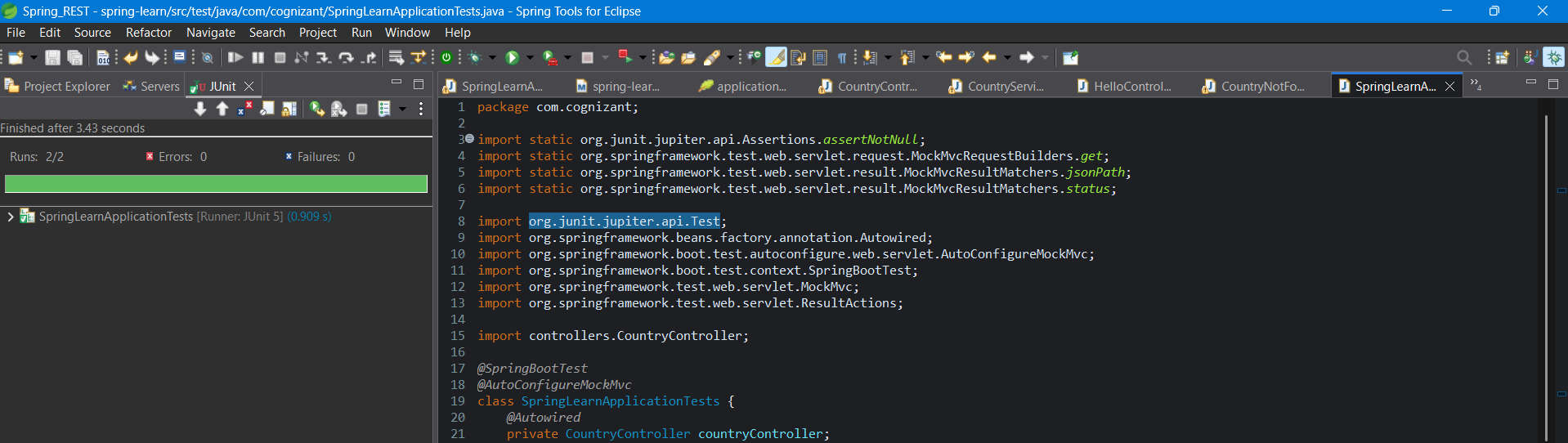
actions.andExpect(*jsonPath*("$.name").exists());

actions.andExpect(*jsonPath*("$.name").value("India"));

}

}

Output



**MockMVC - Test get country service for exceptional scenario**

package com.cognizant;

import static org.junit.jupiter.api.Assertions.*assertNotNull*;

import static org.springframework.test.web.servlet.request.MockMvcRequestBuilders.*get*;

import static org.springframework.test.web.servlet.result.MockMvcResultMatchers.*jsonPath*;

import static org.springframework.test.web.servlet.result.MockMvcResultMatchers.*status*;

import org.junit.jupiter.api.Test;

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

import org.springframework.boot.test.autoconfigure.web.servlet.AutoConfigureMockMvc;

import org.springframework.boot.test.context.SpringBootTest;

import org.springframework.test.web.servlet.MockMvc;

import org.springframework.test.web.servlet.ResultActions;

import controllers.CountryController;

*@SpringBootTest*

*@AutoConfigureMockMvc*

class SpringLearnApplicationTests {

*@Autowired*

private CountryController countryController;

*@Autowired*

private MockMvc mvc;

*@Test*

void contextLoads() {

*assertNotNull*(countryController);

}

*@Test*

public void testGetCountry() throws Exception {

ResultActions actions = mvc.perform(*get*("/country"));

actions.andExpect(*status*().isOk());

actions.andExpect(*jsonPath*("$.code").exists());

actions.andExpect(*jsonPath*("$.code").value("IN"));

actions.andExpect(*jsonPath*("$.name").exists());

actions.andExpect(*jsonPath*("$.name").value("India"));

}

*@Test*

public void testGetCountryException() throws Exception {

ResultActions actions = mvc.perform(*get*("/countries/xyz"));

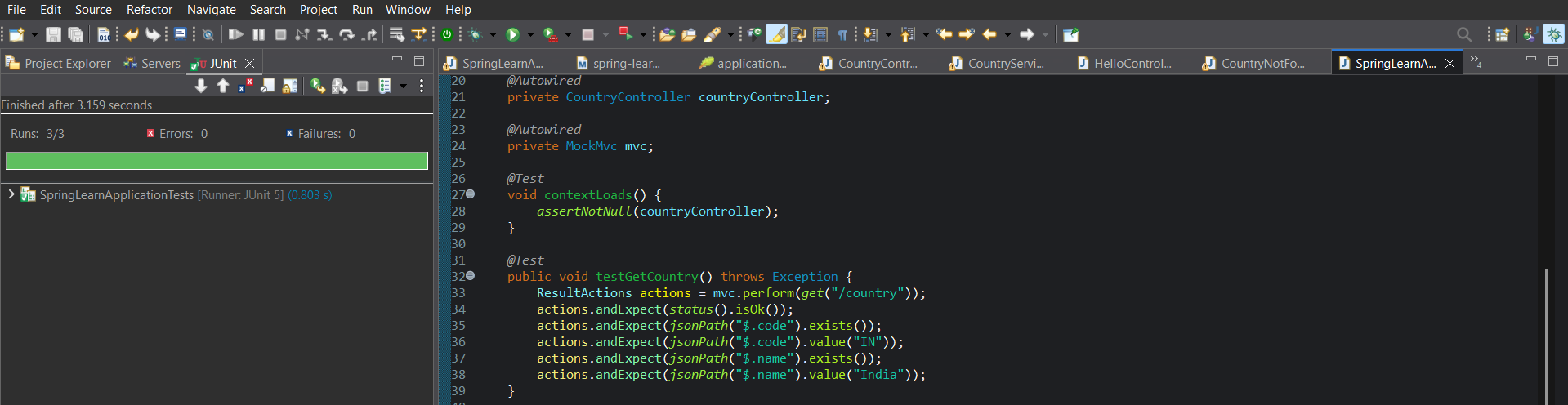
actions.andExpect(*status*().isNotFound());

actions.andExpect(*status*().reason("Country not found"));

}

}

Output



# JWT HandsOn

Pom.xml

<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>

<groupId>com.cognizant</groupId>

<artifactId>spring-learn</artifactId>

<version>1.0-SNAPSHOT</version>

<parent>

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

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

<version>3.5.3</version>

<relativePath/> <!-- lookup parent from repository -->

</parent>

<properties>

<java.version>17</java.version>

</properties>

<dependencies>

<!-- Spring Boot Web -->

<dependency>

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

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

</dependency>

<!-- Spring Boot Security -->

<dependency>

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

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

</dependency>

<!-- Logging -->

<dependency>

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

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

</dependency>

<!-- JJWT 0.11.5 -->

<dependency>

<groupId>io.jsonwebtoken</groupId>

<artifactId>jjwt-api</artifactId>

<version>0.11.5</version>

</dependency>

<dependency>

<groupId>io.jsonwebtoken</groupId>

<artifactId>jjwt-impl</artifactId>

<version>0.11.5</version>

<scope>runtime</scope>

</dependency>

<dependency>

<groupId>io.jsonwebtoken</groupId>

<artifactId>jjwt-jackson</artifactId>

<version>0.11.5</version>

<scope>runtime</scope>

</dependency>

<dependency>

<groupId>org.projectlombok</groupId>

<artifactId>lombok</artifactId>

<optional>true</optional>

</dependency>

<!-- Spring Boot Test -->

<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>

SecurityConfig.java

package com.cognizant.config;

import org.springframework.context.annotation.Bean;

import org.springframework.context.annotation.Configuration;

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

import org.springframework.security.core.userdetails.User;

import org.springframework.security.core.userdetails.UserDetailsService;

import org.springframework.security.crypto.password.~~NoOpPasswordEncoder~~;

import org.springframework.security.crypto.password.PasswordEncoder;

import org.springframework.security.provisioning.InMemoryUserDetailsManager;

import org.springframework.security.web.SecurityFilterChain;

*@Configuration*

public class SecurityConfig {

*@Bean*

public UserDetailsService userDetailsService() {

var uds = new InMemoryUserDetailsManager();

uds.createUser(User.*withUsername*("user")

.password("pwd")

.roles("USER")

.build());

uds.createUser(User.*withUsername*("admin")

.password("admin")

.roles("ADMIN")

.build());

return uds;

}

*@Bean*

public PasswordEncoder passwordEncoder() {

// Only for testing, do not use NoOpPasswordEncoder in production

return ~~NoOpPasswordEncoder~~.~~getInstance~~();

}

*@Bean*

public SecurityFilterChain securityFilterChain(HttpSecurity http) throws Exception {

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

.authorizeHttpRequests(authz -> authz

.requestMatchers("/countries").hasRole("USER")

.requestMatchers("/authenticate").hasAnyRole("USER", "ADMIN")

.anyRequest().authenticated()

)

.~~httpBasic~~();

return http.build();

}

}

AuthenticationController.java

package controllers;

import com.cognizant.JwtUtil;

import org.slf4j.Logger;

import org.slf4j.LoggerFactory;

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

import java.util.Base64;

import java.util.HashMap;

import java.util.Map;

*@RestController*

public class AuthenticationController {

private static final Logger ***logger*** = LoggerFactory.*getLogger*(AuthenticationController.class);

private final JwtUtil jwtUtil;

public AuthenticationController(JwtUtil jwtUtil) {

this.jwtUtil = jwtUtil;

}

*@GetMapping*("/authenticate")

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

***logger***.info("START - /authenticate");

***logger***.debug("Authorization Header: {}", authHeader);

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

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

String credentials = new String(decodedBytes);

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

String username = values[0];

String password = values[1];

***logger***.debug("Decoded Username: {}", username);

***logger***.debug("Decoded Password: {}", password);

if (!("user".equals(username) && "pwd".equals(password))) {

throw new RuntimeException("Invalid credentials");

}

String token = jwtUtil.generateToken(username);

Map<String, String> map = new HashMap<>();

map.put("token", token);

***logger***.info("END - /authenticate");

return map;

}

}

JwtUtil.java

package com.cognizant;

import io.jsonwebtoken.Jwts;

import io.jsonwebtoken.SignatureAlgorithm;

import io.jsonwebtoken.security.Keys;

import org.springframework.stereotype.Component;

import java.security.Key;

import java.util.Date;

*@Component*

public class JwtUtil {

private final Key key = Keys.*secretKeyFor*(*SignatureAlgorithm*.***HS256***);

public String generateToken(String username) {

return Jwts.*builder*()

.setSubject(username)

.setIssuedAt(new Date())

.setExpiration(new Date(System.*currentTimeMillis*() + 24 \* 60 \* 60 \* 1000))

.signWith(key)

.compact();

}

}

SpringLearnApplication.java

package com.cognizant;

import java.text.SimpleDateFormat;

import java.util.Date;

import java.util.List;

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.annotation.ComponentScan;

import org.springframework.context.annotation.ImportResource;

import org.springframework.context.support.ClassPathXmlApplicationContext;

import com.cognizant.service.exception.CountryNotFoundException;

*@SpringBootApplication*

*@ImportResource*("classpath:country.xml")

*@ComponentScan*(basePackages={"com.cognizant","controllers","service"})

public class SpringLearnApplication {

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

public static void main(String[] args) throws Exception {

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

*displayDate*();

*displayCountries*();

}

public static void displayDate() throws Exception{

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

ApplicationContext context=new ClassPathXmlApplicationContext("date-format.xml");

SimpleDateFormat format=context.getBean(SimpleDateFormat.class);

Date parsedDate=format.parse("31/12/2018");

***LOGGER***.debug("Parsed Date: {}", parsedDate);

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

}

public static void displayCountries() throws CountryNotFoundException{

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

ApplicationContext context1 = new ClassPathXmlApplicationContext("country.xml");

*@SuppressWarnings*("unchecked")

List<Country> countries = (List<Country>) context1.getBean("countryList");

for (Country country : countries) {

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

}

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

}

}

Output:

