**Week 4 : Spring REST using Spring Boot**

**Hands on 1**

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

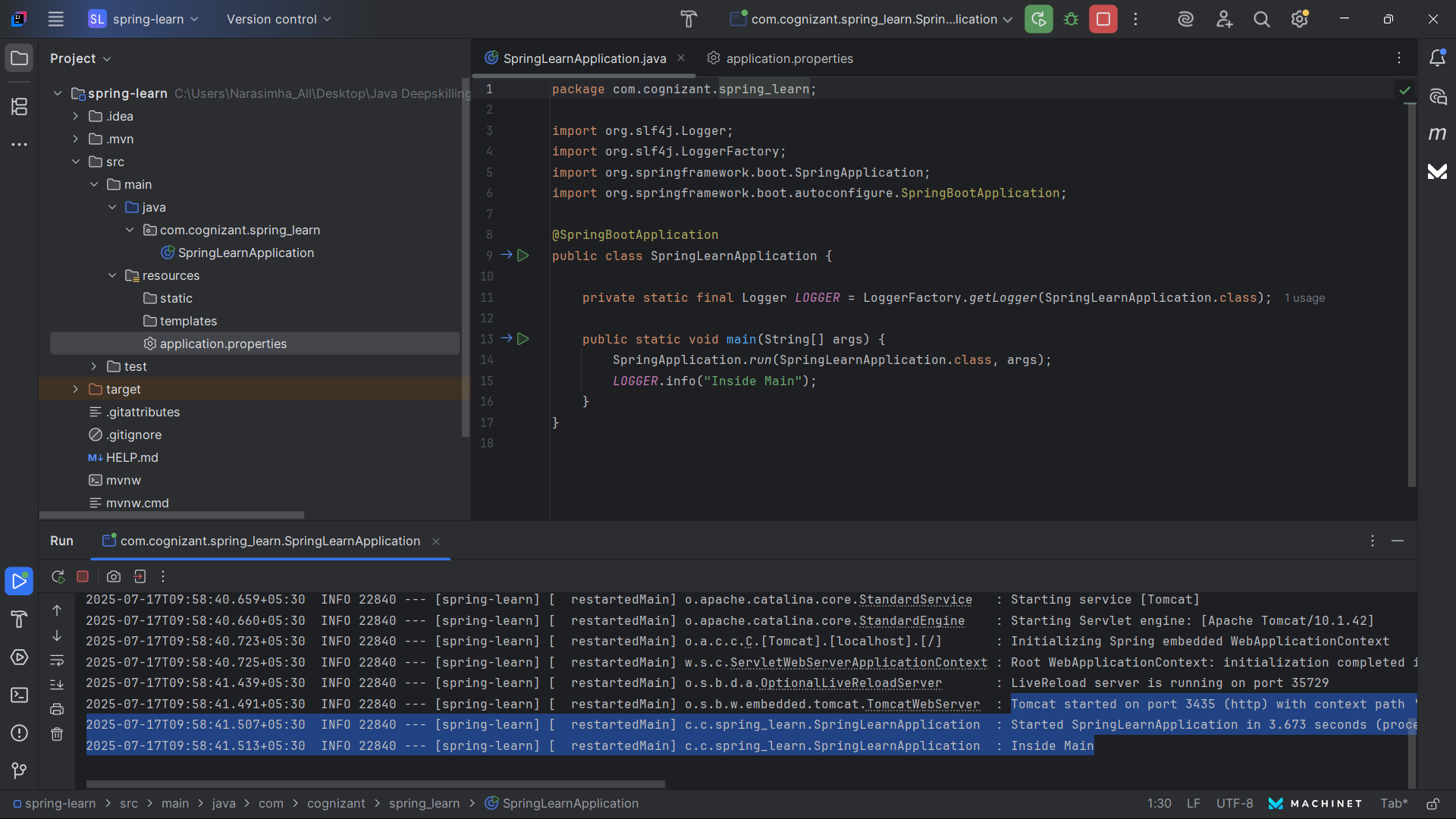
**SpringLearnApplication.java**

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("Inside Main");  
 }  
}

**application.properties**

spring.application.name=spring-learn  
logging.level.root=INFO  
server.port=3435

**Output :**



**Hands on 2**

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

**Main.java**

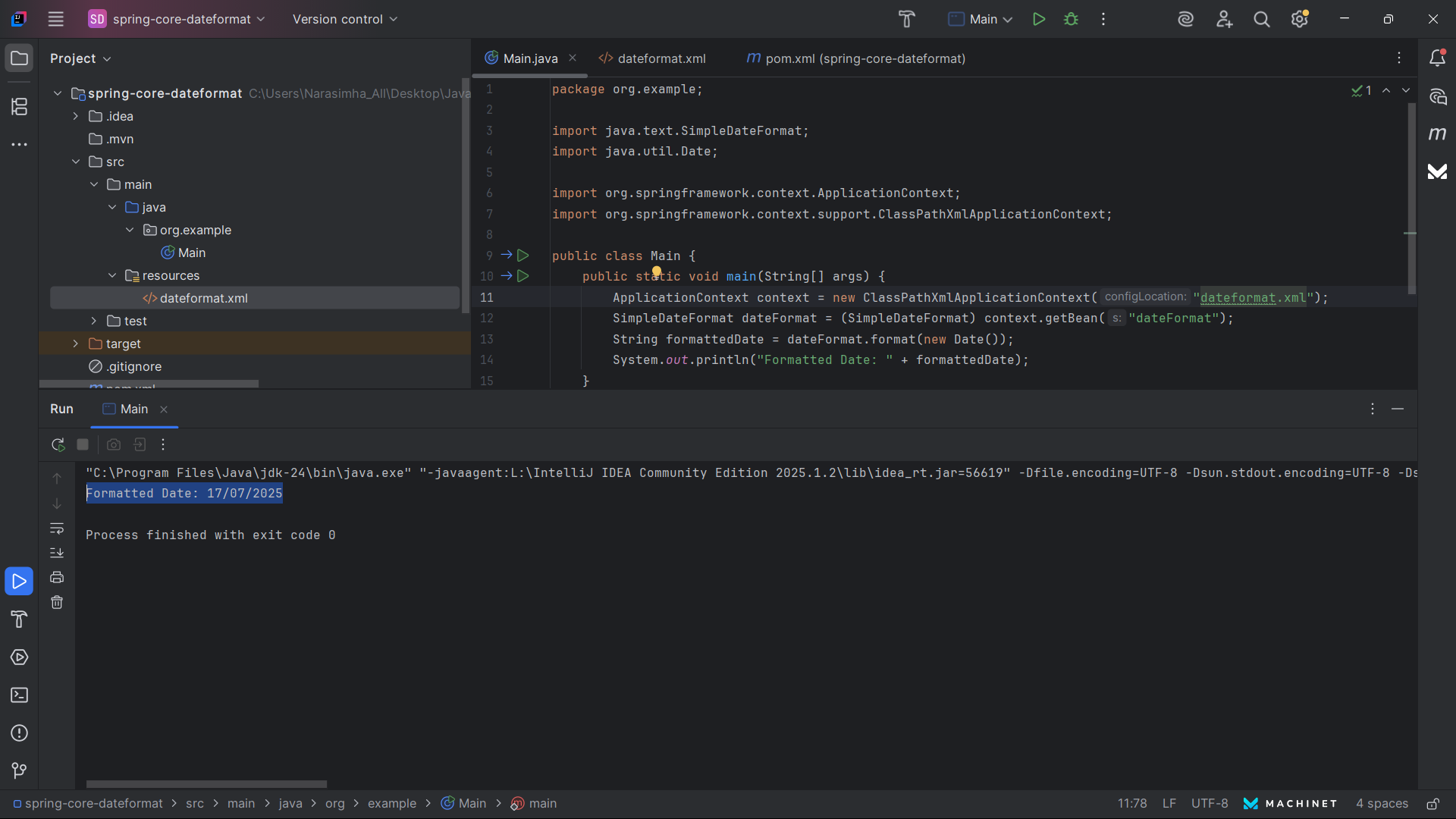
package org.example;  
  
import java.text.SimpleDateFormat;  
import java.util.Date;  
  
import org.springframework.context.ApplicationContext;  
import org.springframework.context.support.ClassPathXmlApplicationContext;  
  
public class Main {  
 public static void main(String[] args) {  
 ApplicationContext context = new ClassPathXmlApplicationContext("dateformat.xml");  
 SimpleDateFormat dateFormat = (SimpleDateFormat) context.getBean("dateFormat");  
 String formattedDate = dateFormat.format(new Date());  
 System.*out*.println("Formatted Date: " + formattedDate);  
 }  
}

v

**dateformat.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"/>  
 </bean>  
  
</beans>

**Output :**



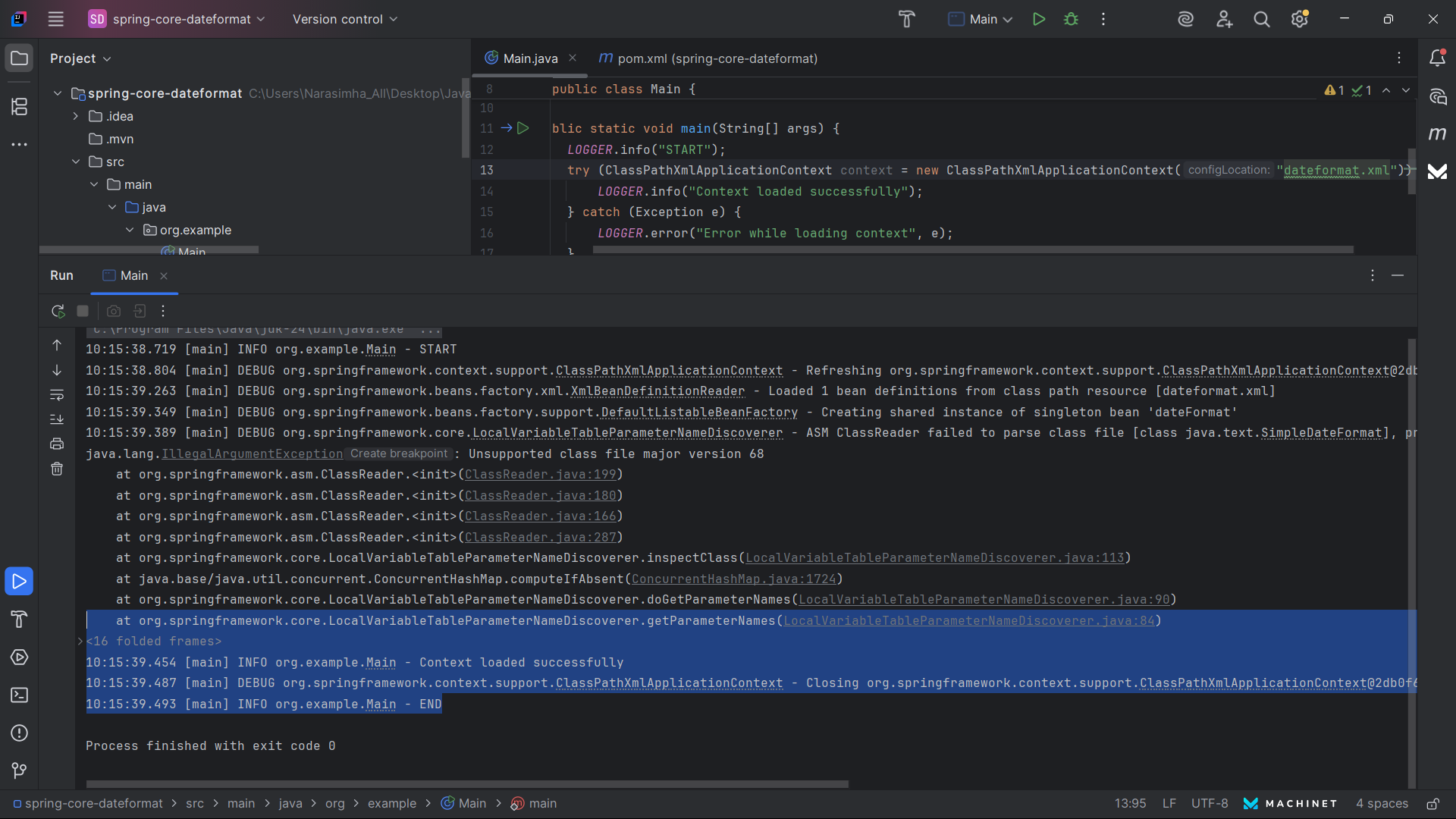
**Hands on 3**

**Spring Core - Incorporate Logging**   
**Code :**

**Main.java**

package org.example;  
  
  
import org.slf4j.Logger;  
import org.slf4j.LoggerFactory;  
import org.springframework.context.support.ClassPathXmlApplicationContext;  
  
public class Main {  
 private static final Logger *LOGGER* = LoggerFactory.*getLogger*(Main.class);  
  
 public static void main(String[] args) {  
 *LOGGER*.info("START");  
 try (ClassPathXmlApplicationContext context = new ClassPathXmlApplicationContext("dateformat.xml")) {  
 *LOGGER*.info("Context loaded successfully");  
 } catch (Exception e) {  
 *LOGGER*.error("Error while loading context", e);  
 }  
 *LOGGER*.info("END");  
 }  
}

**Output :**



**Hands on 4**

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

**Country.java**

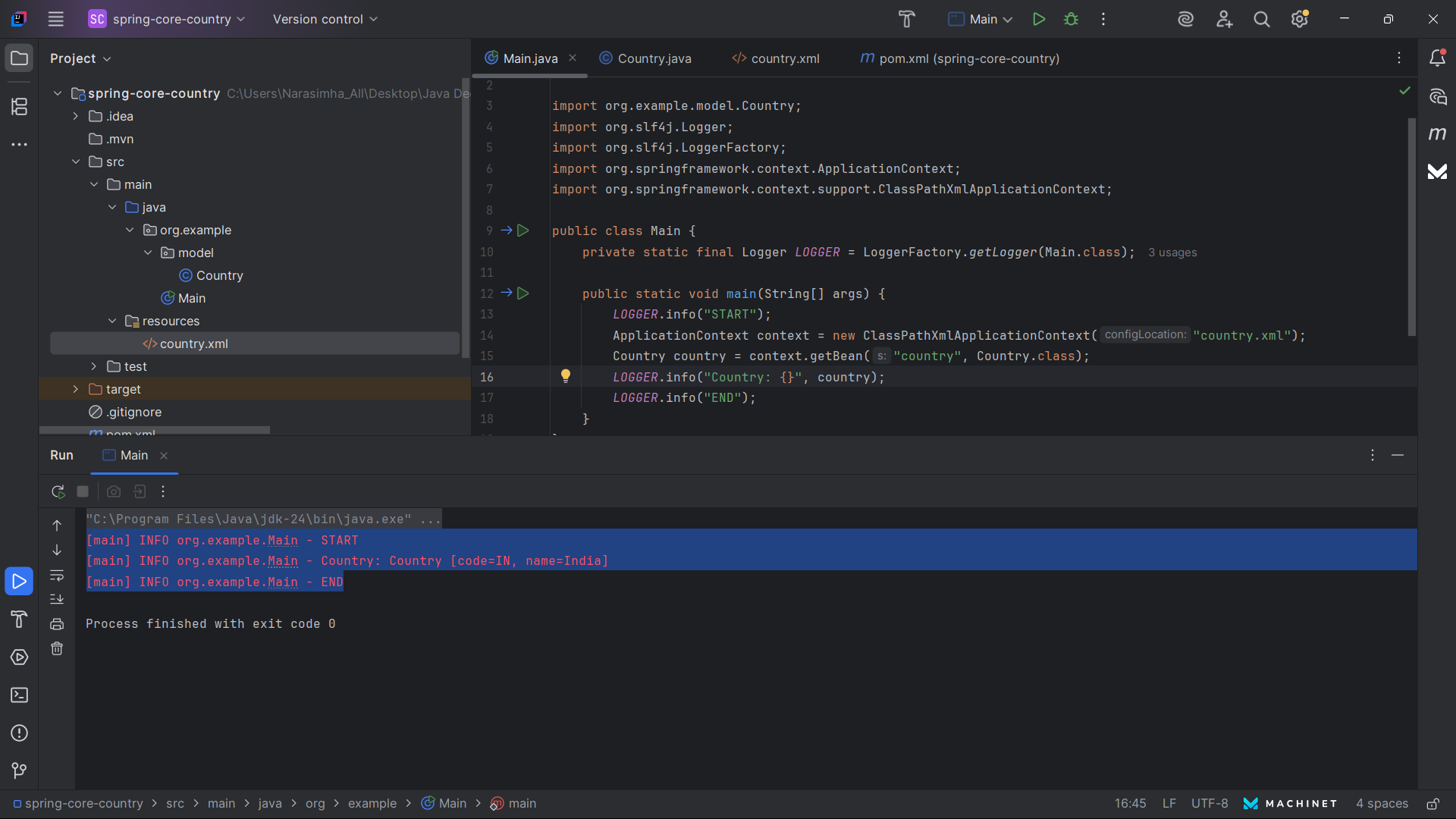
package org.example.model;  
  
public class Country {  
 private String code;  
 private String name;  
  
 public Country() {}  
  
 public String getCode() {  
 return code;  
 }  
  
 public void setCode(String code) {  
 this.code = code;  
 }  
  
 public String getName() {  
 return name;  
 }  
  
 public void setName(String name) {  
 this.name = name;  
 }  
  
 @Override  
 public String toString() {  
 return "Country [code=" + code + ", name=" + name + "]";  
 }  
}

**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="org.example.model.Country">  
 <property name="code" value="IN"/>  
 <property name="name" value="India"/>  
 </bean>  
  
</beans>

**Main.java**

package org.example;  
  
import org.example.model.Country;  
import org.slf4j.Logger;  
import org.slf4j.LoggerFactory;  
import org.springframework.context.ApplicationContext;  
import org.springframework.context.support.ClassPathXmlApplicationContext;  
  
public class Main {  
 private static final Logger *LOGGER* = LoggerFactory.*getLogger*(Main.class);  
  
 public static void main(String[] args) {  
 *LOGGER*.info("START");  
 ApplicationContext context = new ClassPathXmlApplicationContext("country.xml");  
 Country country = context.getBean("country", Country.class);  
 *LOGGER*.info("Country: {}", country);  
 *LOGGER*.info("END");  
 }  
}

**Output :**  


**Hands on 5**

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

**Country.java**

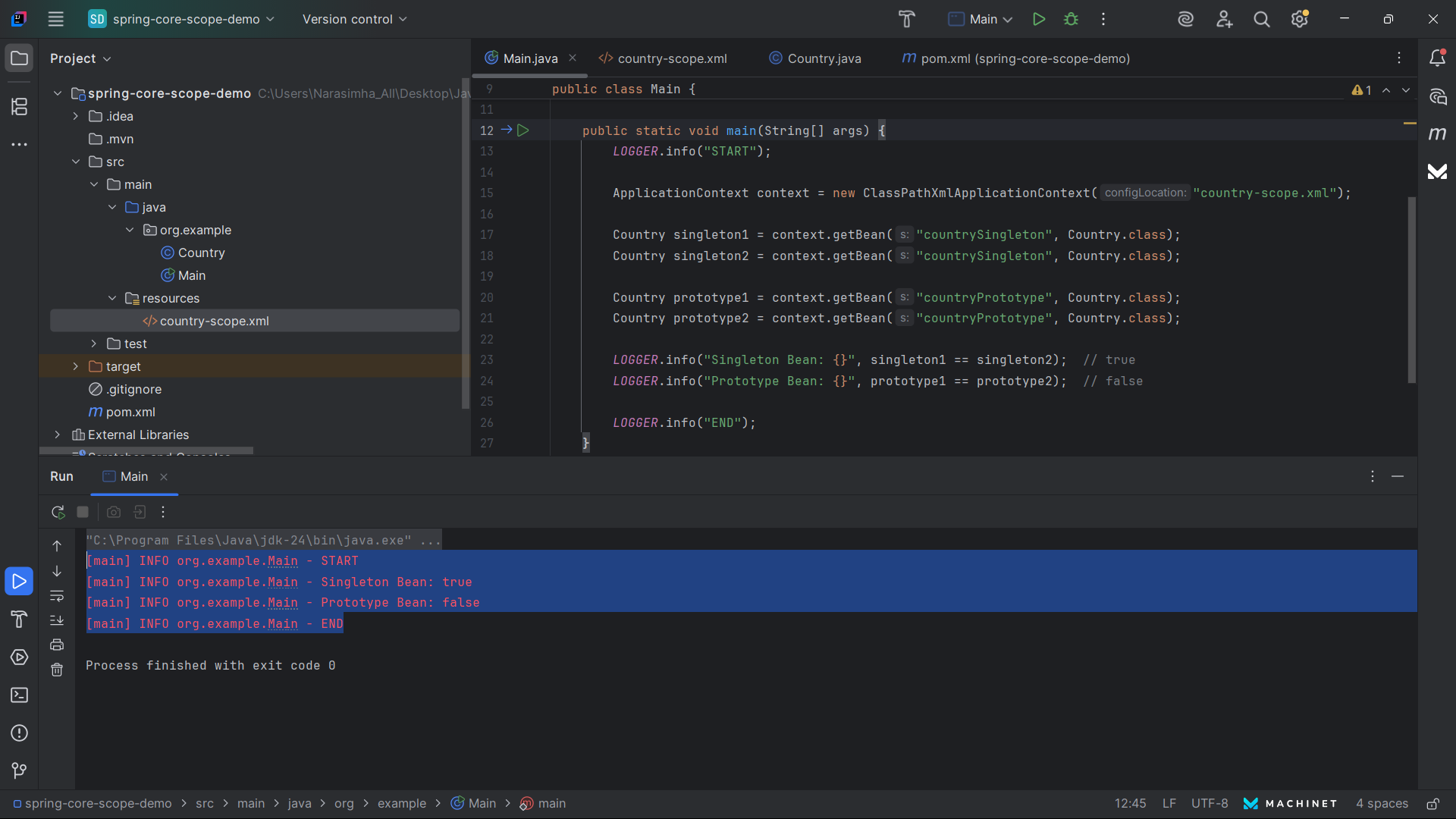
package org.example;  
  
public class Country {  
 private String code;  
 private String name;  
  
 public Country() {}  
  
 public String getCode() {  
 return code;  
 }  
  
 public void setCode(String code) {  
 this.code = code;  
 }  
  
 public String getName() {  
 return name;  
 }  
  
 public void setName(String name) {  
 this.name = name;  
 }  
  
 @Override  
 public String toString() {  
 return "Country [code=" + code + ", name=" + name + "]";  
 }  
}

**Main.java**

package org.example;  
  
import org.example.Country;  
import org.slf4j.Logger;  
import org.slf4j.LoggerFactory;  
import org.springframework.context.ApplicationContext;  
import org.springframework.context.support.ClassPathXmlApplicationContext;  
  
public class Main {  
 private static final Logger *LOGGER* = LoggerFactory.*getLogger*(Main.class);  
  
 public static void main(String[] args) {  
 *LOGGER*.info("START");  
  
 ApplicationContext context = new ClassPathXmlApplicationContext("country-scope.xml");  
  
 Country singleton1 = context.getBean("countrySingleton", Country.class);  
 Country singleton2 = context.getBean("countrySingleton", Country.class);  
  
 Country prototype1 = context.getBean("countryPrototype", Country.class);  
 Country prototype2 = context.getBean("countryPrototype", Country.class);  
  
 *LOGGER*.info("Singleton Bean: {}", singleton1 == singleton2); // true  
 *LOGGER*.info("Prototype Bean: {}", prototype1 == prototype2); // false  
  
 *LOGGER*.info("END");  
 }  
}

**country-scope.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="countrySingleton" class="org.example.Country" scope="singleton">  
 <property name="code" value="SG"/>  
 <property name="name" value="Singapore"/>  
 </bean>  
  
 <bean id="countryPrototype" class="org.example.Country" scope="prototype">  
 <property name="code" value="JP"/>  
 <property name="name" value="Japan"/>  
 </bean>  
  
</beans>

**Output :**  


**Hands on 6**

**Spring Core – Load list of countries from Spring Configuration XML**   
**Code :**

**Main.java**

package org.example;  
  
import org.example.Country;  
import org.slf4j.Logger;  
import org.slf4j.LoggerFactory;  
import org.springframework.context.ApplicationContext;  
import org.springframework.context.support.ClassPathXmlApplicationContext;  
  
import java.util.List;  
  
public class Main {  
 private static final Logger *LOGGER* = LoggerFactory.*getLogger*(Main.class);  
  
 public static void main(String[] args) {  
 *LOGGER*.info("START");  
  
 ApplicationContext context = new ClassPathXmlApplicationContext("country-list.xml");  
  
 List<Country> countries = context.getBean("countryList", List.class);  
 *LOGGER*.info("Country List: {}", countries);  
  
 *LOGGER*.info("END");  
 }  
}

**Country.java**

package org.example;  
  
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; }  
  
 @Override  
 public String toString() {  
 return "Country [code=" + code + ", name=" + name + "]";  
 }  
}

**country-list.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="country1" class="org.example.Country">  
 <property name="code" value="IN"/>  
 <property name="name" value="India"/>  
 </bean>  
  
 <bean id="country2" class="org.example.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="country1"/>  
 <ref bean="country2"/>  
 </list>  
 </constructor-arg>  
 </bean>  
  
</beans>

**Output :**

