# Spring REST using Spring Boot 3

## Create a Spring Web Project using Maven

### CODE:

**SpringLearnApplication.java**

package com.cognizant.spring\_learn;

import org.springframework.boot.SpringApplication;

import org.springframework.boot.autoconfigure.SpringBootApplication;

import org.slf4j.Logger;

import org.slf4j.LoggerFactory;

@SpringBootApplication

public class SpringLearnApplication {

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

public static void main(String[] args) {

LOGGER.info("Starting SpringLearnApplication");

SpringApplication.run(SpringLearnApplication.class, args);

LOGGER.info("SpringLearnApplication started successfully");

}

}

**SpringLearnApplicationTests.java**

package com.cognizant.spring\_learn;

import org.junit.jupiter.api.Test;

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

@SpringBootTest

class SpringLearnApplicationTests {

@Test

void contextLoads() {

}

}

**application.properties**

spring.application.name=spring-learn

**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>

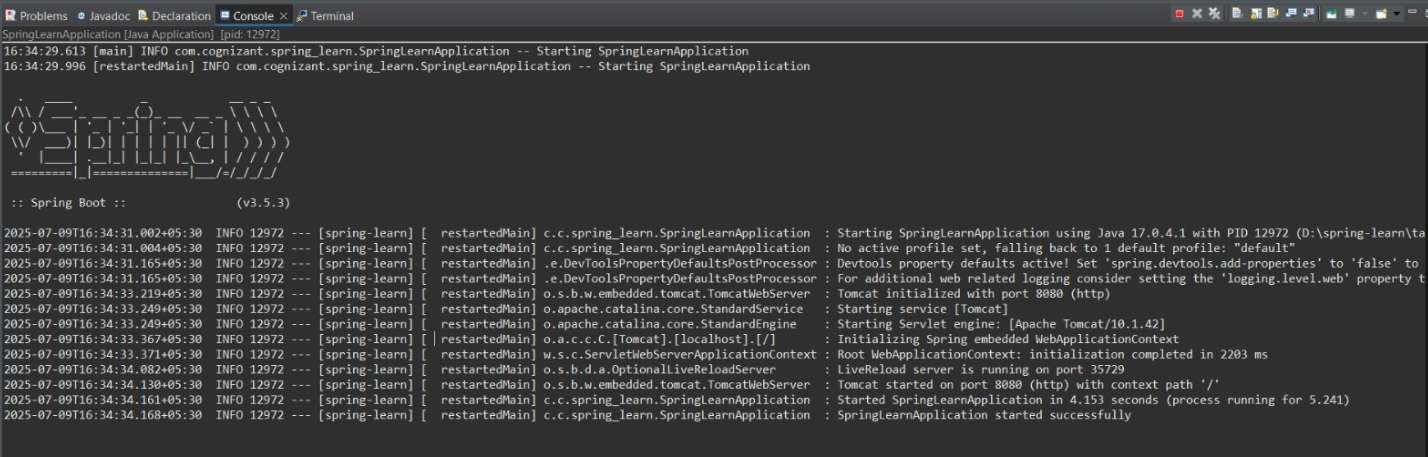
</plugin>

</plugins>

</build>

</project>

#### OUTPUT:



## Spring Core – Load Country from Spring Configuration XML

### CODE:

**Country.java**

package com.cognizant.spring\_core\_country.model;

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() {

return code;

}

public void setCode(String code) {

LOGGER.debug("Inside setCode()");

this.code = code;

}

public String 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 + "]";

}

}

**SpringCoreCountryApplication.java**

package com.cognizant.spring\_core\_country;

import org.springframework.boot.SpringApplication;

import org.springframework.boot.autoconfigure.SpringBootApplication;

import org.springframework.context.ApplicationContext;

import org.springframework.context.support.ClassPathXmlApplicationContext;

import org.slf4j.Logger;

import org.slf4j.LoggerFactory;

import com.cognizant.spring\_core\_country.model.Country;

@SpringBootApplication

public class SpringCoreCountryApplication {

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

public static void main(String[] args) {

SpringApplication.run(SpringCoreCountryApplication.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 us = context.getBean("us", Country.class);

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

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

LOGGER.debug("All Countries:");

LOGGER.debug("1. {}", us.toString());

LOGGER.debug("2. {}", de.toString());

LOGGER.debug("3. {}", jp.toString());

LOGGER.debug("4. {}", country.toString());

}

}

**SpringCoreCountryApplicationTests.java**

package com.cognizant.spring\_core\_country;

import org.springframework.boot.SpringApplication;

import org.springframework.boot.autoconfigure.SpringBootApplication;

import org.springframework.context.ApplicationContext;

import org.springframework.context.support.ClassPathXmlApplicationContext;

import org.slf4j.Logger;

import org.slf4j.LoggerFactory;

import com.cognizant.spring\_core\_country.model.Country;

@SpringBootApplication

public class SpringCoreCountryApplication {

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

public static void main(String[] args) {

SpringApplication.run(SpringCoreCountryApplication.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 us = context.getBean("us", Country.class);

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

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

LOGGER.debug("All Countries:");

LOGGER.debug("1. {}", us.toString());

LOGGER.debug("2. {}", de.toString());

LOGGER.debug("3. {}", jp.toString());

LOGGER.debug("4. {}", country.toString());

}

}

**application.properties**

<?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="com.cognizant.spring\_core\_country.model.Country">

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

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

</bean>

<bean id="us" class="com.cognizant.spring\_core\_country.model.Country">

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

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

</bean>

<bean id="de" class="com.cognizant.spring\_core\_country.model.Country">

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

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

</bean>

<bean id="jp" class="com.cognizant.spring\_core\_country.model.Country">

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

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

</bean>

</beans>

**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="com.cognizant.spring\_core\_country.model.Country">

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

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

</bean>

<bean id="us" class="com.cognizant.spring\_core\_country.model.Country">

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

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

</bean>

<bean id="de" class="com.cognizant.spring\_core\_country.model.Country">

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

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

</bean>

<bean id="jp" class="com.cognizant.spring\_core\_country.model.Country">

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

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

</bean>

</beans>

**logback.xml**

<configuration>

<appender name="STDOUT" class="ch.qos.logback.core.ConsoleAppender">

<encoder>

<pattern>%d{HH:mm:ss.SSS} [%thread] %-5level %logger{36} - %msg%n</pattern>

</encoder>

</appender>

<root level="DEBUG">

<appender-ref ref="STDOUT" />

</root>

</configuration>

**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-core-country</artifactId>

<version>0.0.1-SNAPSHOT</version>

<name>spring-core-country</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>

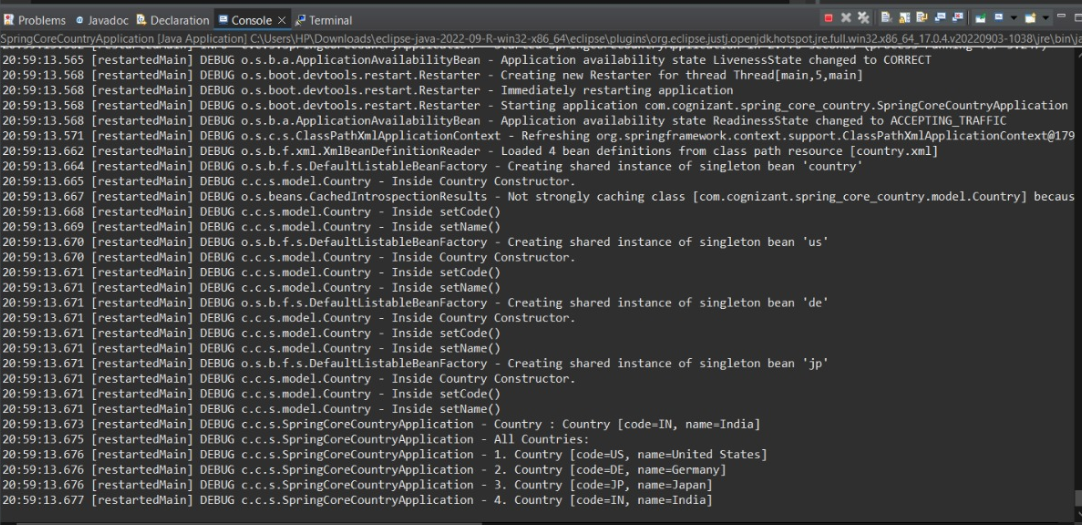
</plugin>

</plugins>

</build>

</project>

#### OUTPUT:



## Hello World RESTful Web Service

### CODE:

**HelloController.java**

package com.cognizant.hello\_world\_service.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() method");

String response = "Hello World from our new service!!";

LOGGER.info("End - sayHello() method");

return response;

}}

**HelloWorldServiceApplication.java**

package com.cognizant.hello\_world\_service;

import org.springframework.boot.SpringApplication;

import org.springframework.boot.autoconfigure.SpringBootApplication;

import org.springframework.core.env.Environment;

@SpringBootApplication

public class HelloWorldServiceApplication {

public static void main(String[] args) {

System.setProperty("server.port", "8083"); // Force port 8083

var app = new SpringApplication(HelloWorldServiceApplication.class);

Environment env = app.run(args).getEnvironment();

String protocol = env.getProperty("server.ssl.key-store") != null ? "https" : "http";

String port = env.getProperty("server.port");

String contextPath = env.getProperty("server.servlet.context-path", "");

System.out.printf("\n----------------------------------------------------------\n" +

"Application '%s' is running! Access URLs:\n" +

"Local: \t\t%s://localhost:%s%s/hello\n" +

"Profile(s): \t%s\n" +

"----------------------------------------------------------\n",

env.getProperty("spring.application.name"),

protocol,

port,

contextPath,

env.getActiveProfiles().length == 0 ? env.getDefaultProfiles() : env.getActiveProfiles());

}

}

**HelloWorldServiceApplicationTests.java**

package com.cognizant.hello\_world\_service;

import org.junit.jupiter.api.Test;

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

@SpringBootTest

class HelloWorldServiceApplicationTests {

@Test

void contextLoads() {

}

}

**application.properties**

# Server configuration

server.port=8083

# Spring application name

spring.application.name=hello-world-service

# Logging configuration

logging.level.root=INFO

logging.level.com.cognizant.hello\_world\_service=DEBUG

**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>3.2.4</version>

<relativePath/>

</parent>

<groupId>com.cognizant</groupId>

<artifactId>hello-world-service</artifactId>

<version>0.0.1-SNAPSHOT</version>

<name>hello-world-service</name>

<description>RESTful Hello World Web Service</description>

<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>

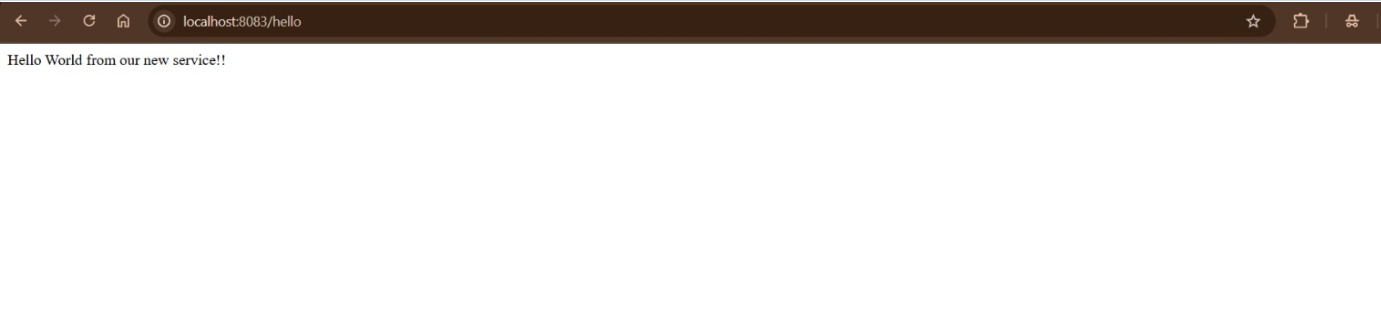
</plugin>

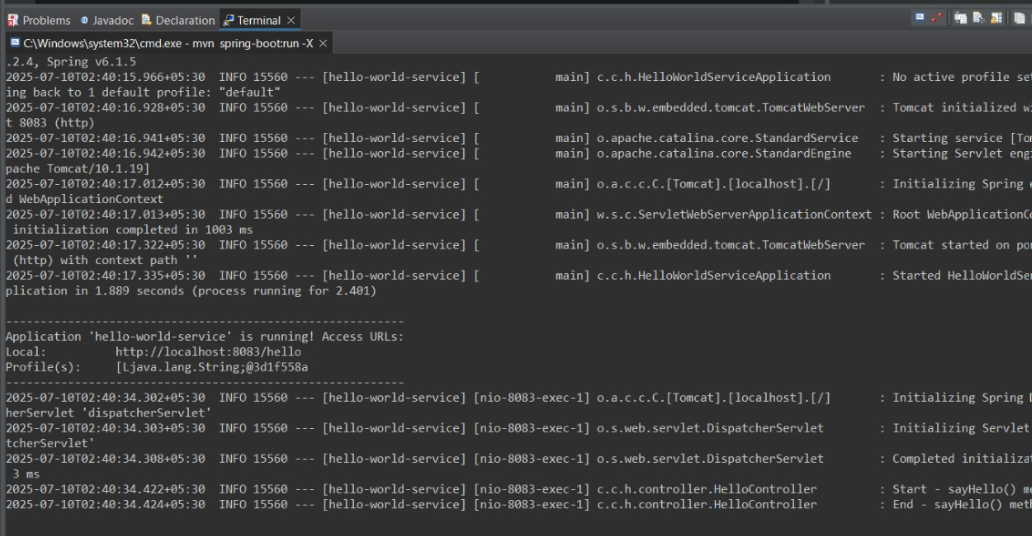
</plugins>

</build>

</project>

#### OUTPUT:





## REST - Country Web Service

### CODE:

**Country.java**

package com.cognizant.countrywebservice.model;

public class Country {

private String code;

private String name;

// Getters and Setters

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 + "]";

}

}

**CountryController.java**

package com.cognizant.countrywebservice.controller;

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

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

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

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

import com.cognizant.countrywebservice.model.Country;

@RestController

@RequestMapping("/country")

public class CountryController {

@Autowired

private Country in; // This will be injected from beans.xml

@GetMapping

public Country getCountryIndia() {

return in;

}

}

**CountrywebserviceApplication.java**

package com.cognizant.countrywebservice;

import org.springframework.boot.SpringApplication;

import org.springframework.boot.autoconfigure.SpringBootApplication;

import org.springframework.context.annotation.ImportResource;

@SpringBootApplication

@ImportResource({"classpath:beans.xml"})

public class CountrywebserviceApplication {

public static void main(String[] args) {

SpringApplication.run(CountrywebserviceApplication.class, args);

}

}

**CountrywebserviceApplicationTests.java**

package com.cognizant.countrywebservice;

import org.junit.jupiter.api.Test;

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

@SpringBootTest

class CountrywebserviceApplicationTests {

@Test

void contextLoads() {

}

}

**application.properties**

# Server port configuration

server.port=8080

# Spring boot banner (optional)

spring.main.banner-mode=off

**beans.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="in" class="com.cognizant.countrywebservice.model.Country">

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

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

</bean>

</beans>

**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>countrywebservice</artifactId>

<version>0.0.1-SNAPSHOT</version>

<name>countrywebservice</name>

<description>Spring REST Country Web Service</description>

<properties>

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

<project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>

</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.projectlombok</groupId>

<artifactId>lombok</artifactId>

<scope>provided</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>

<configuration>

<excludes>

<exclude>

<groupId>org.projectlombok</groupId>

<artifactId>lombok</artifactId>

</exclude>

</excludes>

</configuration>

</plugin>

<plugin>

<groupId>org.apache.maven.plugins</groupId>

<artifactId>maven-compiler-plugin</artifactId>

<version>3.11.0</version>

<configuration>

<source>${java.version}</source>

<target>${java.version}</target>

</configuration>

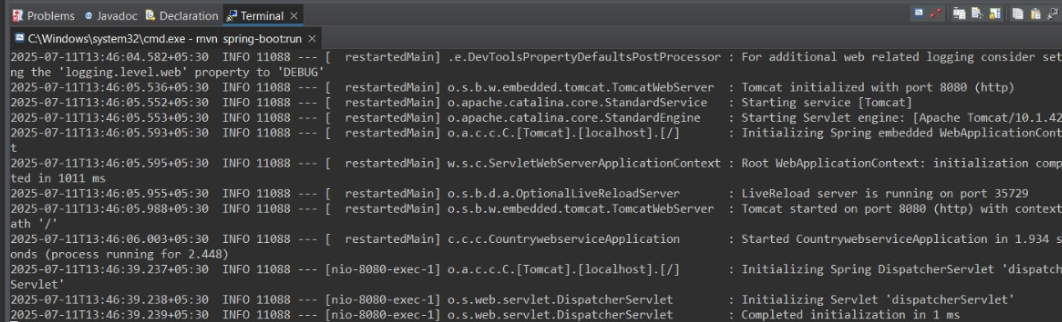
</plugin>

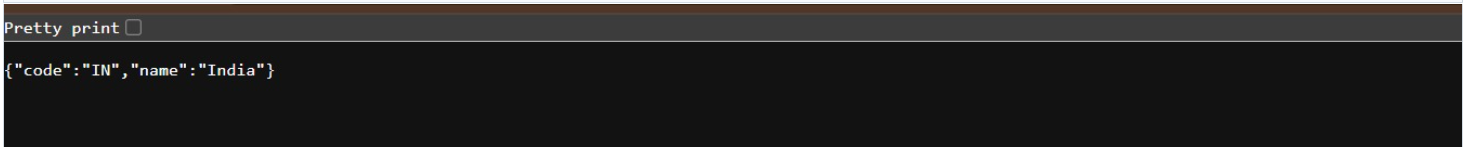
</plugins>

</build>

</project>

#### OUTPUT:





## REST - Get country based on country code

### CODE:

**CodeApplication.java**

package com.cognizant.code;

import org.springframework.boot.SpringApplication;

import org.springframework.boot.autoconfigure.SpringBootApplication;

@SpringBootApplication

public class CodeApplication {

public static void main(String[] args) {

SpringApplication.run(CodeApplication.class, args);

}

}

**CodeApplicationTests.java**

package com.cognizant.code;

import org.junit.jupiter.api.Test;

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

@SpringBootTest

class CodeApplicationTests {

@Test

void contextLoads() {

}

}

**Country.java**

package com.cognizant.code.model;

public class Country {

private String code;

private String name;

// Getters and setters

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 + "'}";

}

}

**CountryController.java**

package com.cognizant.code.controller;

import com.cognizant.code.model.Country;

import com.cognizant.code.service.CountryService;

import org.springframework.http.ResponseEntity;

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;

@RestController

@RequestMapping("/api")

public class CountryController {

private final CountryService countryService;

public CountryController(CountryService countryService) {

this.countryService = countryService;

}

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

public ResponseEntity<Country> getCountryByCode(@PathVariable String code) {

Country country = countryService.getCountry(code);

if (country != null) {

return ResponseEntity.ok(country);

}

return ResponseEntity.notFound().build();

}

}

**CountryService.java**

package com.cognizant.code.service;

import com.cognizant.code.model.Country;

import org.springframework.core.io.ClassPathResource;

import org.springframework.stereotype.Service;

import org.w3c.dom.Document;

import org.w3c.dom.Element;

import org.w3c.dom.NodeList;

import javax.xml.parsers.DocumentBuilder;

import javax.xml.parsers.DocumentBuilderFactory;

import java.io.InputStream;

import java.util.ArrayList;

import java.util.List;

@Service

public class CountryService {

private final List<Country> countries;

public CountryService() throws Exception {

this.countries = loadCountries();

}

private List<Country> loadCountries() throws Exception {

List<Country> countryList = new ArrayList<>();

InputStream is = new ClassPathResource("country.xml").getInputStream();

DocumentBuilderFactory factory = DocumentBuilderFactory.newInstance();

DocumentBuilder builder = factory.newDocumentBuilder();

Document document = builder.parse(is);

NodeList nodeList = document.getElementsByTagName("country");

for (int i = 0; i < nodeList.getLength(); i++) {

Element element = (Element) nodeList.item(i);

Country country = new Country();

country.setCode(getElementValue(element, "code"));

country.setName(getElementValue(element, "name"));

countryList.add(country);

}

return countryList;

}

private String getElementValue(Element parent, String tagName) {

return parent.getElementsByTagName(tagName).item(0).getTextContent();

}

public Country getCountry(String code) {

return countries.stream()

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

.findFirst()

.orElse(null);

}

}

**application.properties**

server.port=8081

spring.application.name=country-service

**country.xml**

<?xml version="1.0" encoding="UTF-8"?>

<countries>

<country>

<code>IN</code>

<name>India</name>

</country>

<country>

<code>US</code>

<name>United States</name>

</country>

<country>

<code>UK</code>

<name>United Kingdom</name>

</country>

<country>

<code>JP</code>

<name>Japan</name>

</country>

</countries>

**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>

</parent>

<groupId>com.cognizant</groupId>

<artifactId>code</artifactId>

<version>0.0.1-SNAPSHOT</version>

<name>country-service</name>

<properties>

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

</properties>

<dependencies>

<dependency>

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

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

</dependency>

</dependencies>

<build>

<plugins>

<plugin>

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

<artifactId>spring-boot-maven-plugin</artifactId>

</plugin>

</plugins>

</build>

</project>

### OUTPUT:

