Assignment: REST – Country Web Service (Spring Learn Application)

# ✅ Objective:

Create a RESTful web service that returns the details of the country India, configured as a Spring bean.

# ✅ Task Details

• URL: http://localhost:8083/country

• Controller: com.cognizant.spring-learn.controller.CountryController

• Method: getCountryIndia()

• Method Annotation: @RequestMapping

• Response Format: JSON

# ✅ 1. Model Class: Country.java

package com.cognizant.springlearn.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;  
 }  
}

# ✅ 2. Spring XML Configuration: 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="in" class="com.cognizant.springlearn.model.Country">  
 <property name="code" value="IN"/>  
 <property name="name" value="India"/>  
 </bean>  
</beans>

# ✅ 3. REST Controller: CountryController.java

package com.cognizant.springlearn.controller;  
  
import com.cognizant.springlearn.model.Country;  
import org.springframework.beans.factory.xml.XmlBeanFactory;  
import org.springframework.core.io.ClassPathResource;  
import org.springframework.web.bind.annotation.RequestMapping;  
import org.springframework.web.bind.annotation.RestController;  
  
@RestController  
public class CountryController {  
  
 @RequestMapping("/country")  
 public Country getCountryIndia() {  
 XmlBeanFactory factory = new XmlBeanFactory(new ClassPathResource("country.xml"));  
 Country country = (Country) factory.getBean("in");  
 return country;  
 }  
}

# ✅ 4. Sample Request

URL: http://localhost:8083/country

Method: GET

# ✅ 5. Sample Response (JSON)

{  
 "code": "IN",  
 "name": "India"  
}

# ✅ 6. SME Explanation

## 📌 a) What happens in the controller method?

• When the endpoint /country is accessed, Spring calls the getCountryIndia() method.  
• The method loads the Spring XML configuration (country.xml) using XmlBeanFactory.  
• It fetches the bean with ID "in" which represents the country India.  
• The bean object is returned directly from the method.

## 📌 b) How the bean is converted into JSON?

• The controller is annotated with @RestController, which implies @ResponseBody.  
• Spring Boot uses Jackson (a built-in library) to convert Java objects into JSON automatically.  
• The Country object returned from the controller method is serialized to JSON.

## 📌 c) Viewing HTTP Headers – Browser (Developer Tools > Network)

Request Headers:  
GET /country HTTP/1.1  
Host: localhost:8083  
Accept: application/json  
  
Response Headers:  
Content-Type: application/json  
Content-Length: 35

## 📌 d) Viewing HTTP Headers – Postman (Headers Tab)

Request Headers:  
Accept: application/json  
  
Response Headers:  
Content-Type: application/json  
Content-Length: 35  
Date: Sun, 13 Jul 2025

# ✅ Conclusion

The REST web service was successfully implemented using Spring Boot. It loads a country bean defined in an XML configuration and returns it as a JSON response via a REST endpoint. The response and headers were verified using browser developer tools and Postman.