**5. REST - Get country based on country code:**

**CODE:**

**SpringLearnApplication.java:**  
package com.example.cognizant.springlearn;  
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);  
 }  
}

**CountryService.java:**

package com.example.cognizant.springlearn.service;  
import com.example.cognizant.springlearn.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> countries = context.getBean("countryList", List.class);  
 // Using lambda expression for case-insensitive search  
 return countries.stream()  
 .filter(c -> c.getCode().equalsIgnoreCase(code))  
 .findFirst()  
 .orElse(null); // or throw exception if preferred  
 }  
}

**CountryController.java:**

package com.example.cognizant.springlearn.controller;  
import com.example.cognizant.springlearn.Country;  
import com.example.cognizant.springlearn.service.CountryService;  
import org.springframework.beans.factory.annotation.Autowired;  
import org.springframework.web.bind.annotation.\*;  
import java.util.List;  
@RestController  
public class CountryController {  
 @Autowired  
 private CountryService countryService;  
  
 @RequestMapping("/country")  
 public Country getCountryIndia() {  
 return countryService.getCountry("IN");  
 }  
 @RequestMapping("/countries")  
 public List<Country> getAllCountries() {  
 return new org.springframework.context.support.ClassPathXmlApplicationContext("country.xml")  
 .getBean("countryList", List.class);  
 }  
 @GetMapping("/country/{code}")  
 public Country getCountry(@PathVariable String code) {  
 return countryService.getCountry(code);  
 }  
}

**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">  
  
 <!-- Individual Country Beans -->  
 <bean id="in" class="com.example.cognizant.springlearn.Country">  
 <property name="code" value="IN"/>  
 <property name="name" value="India"/>  
 </bean>  
  
 <bean id="us" class="com.example.cognizant.springlearn.Country">  
 <property name="code" value="US"/>  
 <property name="name" value="United States"/>  
 </bean>  
  
 <bean id="de" class="com.example.cognizant.springlearn.Country">  
 <property name="code" value="DE"/>  
 <property name="name" value="Germany"/>  
 </bean>  
  
 <bean id="jp" class="com.example.cognizant.springlearn.Country">  
 <property name="code" value="JP"/>  
 <property name="name" value="Japan"/>  
 </bean>  
  
 <bean id="fr" class="com.example.cognizant.springlearn.Country">  
 <property name="code" value="FR"/>  
 <property name="name" value="France"/>  
 </bean>  
  
 <bean id="au" class="com.example.cognizant.springlearn.Country">  
 <property name="code" value="AU"/>  
 <property name="name" value="Australia"/>  
 </bean>  
  
 <bean id="cn" class="com.example.cognizant.springlearn.Country">  
 <property name="code" value="CN"/>  
 <property name="name" value="China"/>  
 </bean>  
  
 <bean id="br" class="com.example.cognizant.springlearn.Country">  
 <property name="code" value="BR"/>  
 <property name="name" value="Brazil"/>  
 </bean>  
  
 <bean id="za" class="com.example.cognizant.springlearn.Country">  
 <property name="code" value="ZA"/>  
 <property name="name" value="South Africa"/>  
 </bean>  
  
 <bean id="sg" class="com.example.cognizant.springlearn.Country">  
 <property name="code" value="SG"/>  
 <property name="name" value="Singapore"/>  
 </bean>  
  
 <!-- List of All Countries -->  
 <bean id="countryList" class="java.util.ArrayList">  
 <constructor-arg>  
 <list>  
 <ref bean="in"/>  
 <ref bean="us"/>  
 <ref bean="de"/>  
 <ref bean="jp"/>  
 <ref bean="fr"/>  
 <ref bean="au"/>  
 <ref bean="cn"/>  
 <ref bean="br"/>  
 <ref bean="za"/>  
 <ref bean="sg"/>  
 </list>  
 </constructor-arg>  
 </bean>  
</beans>

**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  
 http://maven.apache.org/xsd/maven-4.0.0.xsd">  
 <modelVersion>4.0.0</modelVersion>  
 <groupId>com.example.cognizant</groupId>  
 <artifactId>spring-learn</artifactId>  
 <version>0.0.1-SNAPSHOT</version>  
 <packaging>jar</packaging>  
 <name>spring-learn</name>  
 <description>Spring Boot Project for Country REST Services</description>  
  
 <parent>  
 <groupId>org.springframework.boot</groupId>  
 <artifactId>spring-boot-starter-parent</artifactId>  
 <version>3.2.0</version>  
 <relativePath/> <!-- lookup parent from repository -->  
 </parent>  
  
 <properties>  
 <java.version>17</java.version> <!-- Change to 11 if needed -->  
 </properties>  
  
 <dependencies>  
  
 <!-- Spring Boot Web Starter (for REST API) -->  
 <dependency>  
 <groupId>org.springframework.boot</groupId>  
 <artifactId>spring-boot-starter-web</artifactId>  
 </dependency>  
  
 <!-- Spring Context (needed for XML config: ClassPathXmlApplicationContext) -->  
 <dependency>  
 <groupId>org.springframework</groupId>  
 <artifactId>spring-context</artifactId>  
 </dependency>  
  
 <!-- Optional: DevTools (hot reload) -->  
 <dependency>  
 <groupId>org.springframework.boot</groupId>  
 <artifactId>spring-boot-devtools</artifactId>  
 <scope>runtime</scope>  
 <optional>true</optional>  
 </dependency>  
  
 <!-- Spring Boot Test Starter -->  
 <dependency>  
 <groupId>org.springframework.boot</groupId>  
 <artifactId>spring-boot-starter-test</artifactId>  
 <scope>test</scope>  
</dependency>  
  
 </dependencies>  
 <build>  
 <plugins>  
 <!-- Spring Boot Maven Plugin -->  
 <plugin>  
 <groupId>org.springframework.boot</groupId>  
 <artifactId>spring-boot-maven-plugin</artifactId>  
 </plugin>  
 </plugins>  
 </build>  
</project>

**OUTPUT:**

