**Hands on**

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

Solution-

1. Modifying ‘spring-learn’ project created previously

2. Created Controller class: com.cognizant.spring\_learn.controller.CountryController.java

CountryController-

package com.cognizant.spring\_learn.controller;  
  
import com.cognizant.spring\_learn.Country;  
import com.cognizant.spring\_learn.service.CountryService;  
import org.slf4j.Logger;  
import org.slf4j.LoggerFactory;  
import org.springframework.context.ApplicationContext;  
import org.springframework.context.support.ClassPathXmlApplicationContext;  
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 java.util.List;  
  
@RestController  
public class CountryController {  
  
 private static final Logger *LOGGER* = LoggerFactory.*getLogger*(CountryController.class);  
  
 @RequestMapping("/country")  
 public Country getCountryIndia() {  
 *LOGGER*.debug("START getCountryIndia()");  
 ApplicationContext context = new ClassPathXmlApplicationContext("country.xml");  
 Country country = context.getBean("in", Country.class);  
 *LOGGER*.debug("END getCountryIndia()");  
 return country;  
 }  
  
 private final CountryService countryService;  
  
 public CountryController(CountryService countryService) {  
 this.countryService = countryService;  
 }  
  
 @GetMapping("/countries/{code}")  
 public Country getCountry(@PathVariable String code) {  
 *LOGGER*.debug("START getCountry({})", code);  
 Country country = countryService.getCountry(code);  
 *LOGGER*.debug("END getCountry() -> {}", country);  
 return country;  
 }  
  
}

3. Modify service class

CountryService.java

package com.cognizant.spring\_learn.service;  
  
import com.cognizant.spring\_learn.Country;  
import jakarta.annotation.PostConstruct;  
import org.springframework.context.ApplicationContext;  
import org.springframework.context.support.ClassPathXmlApplicationContext;  
import org.springframework.stereotype.Service;  
  
import java.util.List;  
  
@Service  
public class CountryService {  
  
 private List<Country> countryList;  
  
 @PostConstruct  
 @SuppressWarnings("unchecked")  
 public void init() {  
 ApplicationContext context = new ClassPathXmlApplicationContext("country.xml");  
 this.countryList = (List<Country>) context.getBean("countryList");  
 }  
  
 public Country getCountry(String code) {  
 return countryList.stream()  
 .filter(c -> c.getCode().equalsIgnoreCase(code))  
 .findFirst()  
 .orElseThrow(() -> new RuntimeException("Country not found: " + code));  
 }  
}

Country.xml-

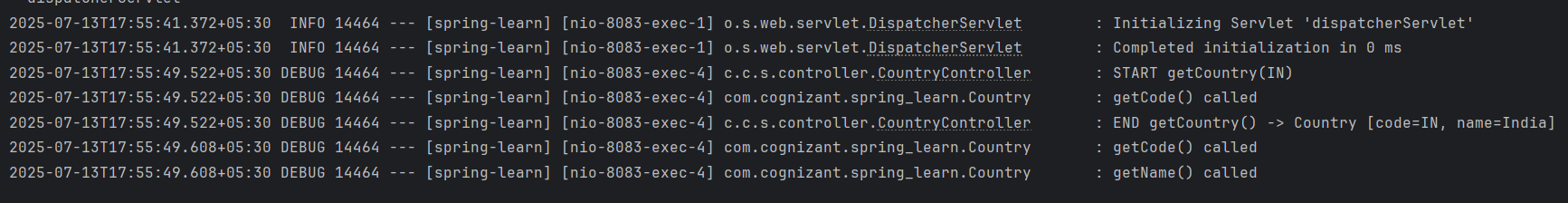
<?xml version="1.0" encoding="UTF-8"?>  
<beans xmlns="http://www.springframework.org/schema/beans"  
 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"  
 xsi:schemaLocation="http://www.springframework.org/schema/beans  
 https://www.springframework.org/schema/beans/spring-beans.xsd">  
  
<bean id="in" class="com.cognizant.spring\_learn.Country">  
 <property name="code" value="IN"/>  
 <property name="name" value="India"/>  
</bean>  
  
<bean id="us" class="com.cognizant.spring\_learn.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="in"/>  
 <ref bean="us"/>  
 </list>  
 </constructor-arg>  
</bean>  
</beans>

3. Run the application

Output-

On port

Logs



Project Structure

