**Handson Exercise 5 - REST - Get country exceptional scenario**

**CODE:**

**Country Class: -**

package com.cognizant.ex4handson2;  
public class Country {  
 private String code;  
 private String name;  
  
 public Country() {}  
  
 public Country(String code, String name) {  
 this.code = code;  
 this.name = 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;  
 }  
}

**Country.xml file: -**

<?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.ex4handson2.Country">  
 <property name="code" value="IN"/>  
 <property name="name" value="India"/>  
 </bean>  
  
 <bean id="us" class="com.cognizant.ex4handson2.Country">  
 <property name="code" value="US"/>  
 <property name="name" value="United States"/>  
 </bean>  
  
 <bean id="jp" class="com.cognizant.ex4handson2.Country">  
 <property name="code" value="JP"/>  
 <property name="name" value="Japan"/>  
 </bean>  
  
 <bean id="de" class="com.cognizant.ex4handson2.Country">  
 <property name="code" value="DE"/>  
 <property name="name" value="Germany"/>  
 </bean>  
  
 <!-- List of all countries -->  
 <bean id="countryList" class="java.util.ArrayList">  
 <constructor-arg>  
 <list>  
 <ref bean="in"/>  
 <ref bean="us"/>  
 <ref bean="jp"/>  
 <ref bean="de"/>  
 </list>  
 </constructor-arg>  
 </bean>  
</beans>

**CountryService Class: -**

package com.cognizant.ex4handson2;  
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) throws CountryNotFoundException {  
 ApplicationContext context = new ClassPathXmlApplicationContext("country.xml");  
 List<Country> countryList = (List<Country>) context.getBean("countryList");  
  
 return countryList.stream()  
 .filter(c -> c.getCode().equalsIgnoreCase(code))  
 .findFirst()  
 .orElseThrow(CountryNotFoundException::new);  
 }  
}

**CountryController Class: -**

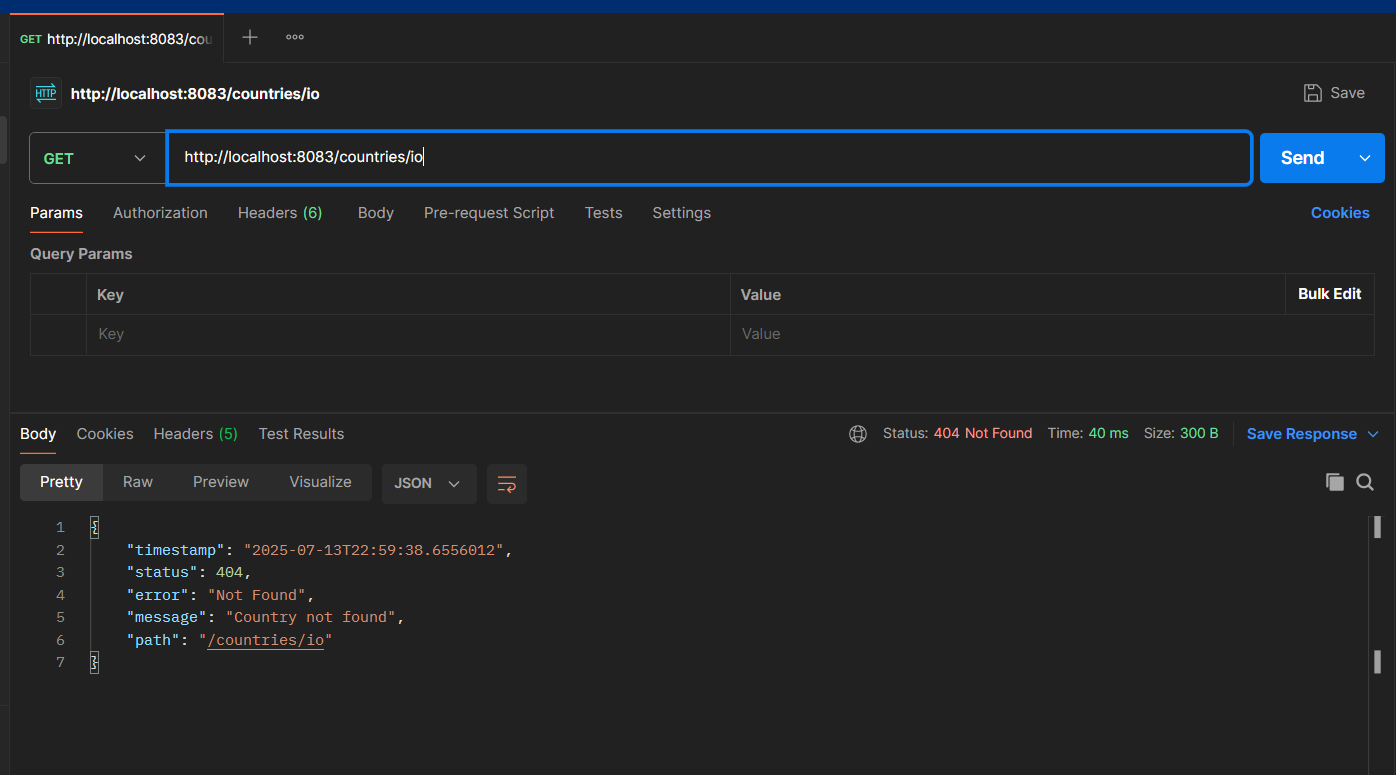
package com.cognizant.ex4handson2;  
import org.slf4j.Logger;  
import org.slf4j.LoggerFactory;  
import org.springframework.beans.factory.annotation.Autowired;  
import org.springframework.web.bind.annotation.\*;  
import java.util.Map;  
import java.util.LinkedHashMap;  
import java.time.LocalDateTime;  
import org.springframework.http.ResponseEntity;  
import org.springframework.http.HttpStatus;  
  
  
@RestController  
public class CountryController {  
  
 private static final Logger *LOGGER* = LoggerFactory.*getLogger*(CountryController.class);  
  
 @Autowired  
 private CountryService countryService;  
 @GetMapping("/countries/{code}")  
 public ResponseEntity<?> getCountry(@PathVariable String code) {  
 try {  
 Country country = countryService.getCountry(code);  
 return ResponseEntity.*ok*(country);  
 } catch (CountryNotFoundException e) {  
 Map<String, Object> body = new LinkedHashMap<>();  
 body.put("timestamp", LocalDateTime.*now*());  
 body.put("status", HttpStatus.*NOT\_FOUND*.value());  
 body.put("error", "Not Found");  
 body.put("message", e.getMessage());  
 body.put("path", "/countries/" + code);  
 return new ResponseEntity<>(body, HttpStatus.*NOT\_FOUND*);  
 }  
 }  
}

**CountryNotFoundException Class: -**

package com.cognizant.ex4handson2;  
import org.springframework.http.HttpStatus;  
import org.springframework.web.bind.annotation.ResponseStatus;  
  
@ResponseStatus(value = HttpStatus.*NOT\_FOUND*, reason = "Country not found")  
public class CountryNotFoundException extends Exception {  
  
 public CountryNotFoundException() {  
 super("Country not found");  
 }  
}

**OUTPUT:**

When incorrect code is given we get the output as: -



When correct code is given we get the output as: -

