**Hello World RESTful Web Service** 

application.properties

spring.application.name=spring-learn

logging.level.org.springframework=info

logging.level.com.cognizant.springlearn=debug

logging.pattern.console=%d{yyMMdd}|%d{HH:mm:ss.SSS}|%-20.20thread|%5p|%-25.25logger{25}|%25M|%m%n

server.port=8083

HelloController.java

package com.cognizant.springlearn.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");

String message = "Hello World!!";

LOGGER.info("END");

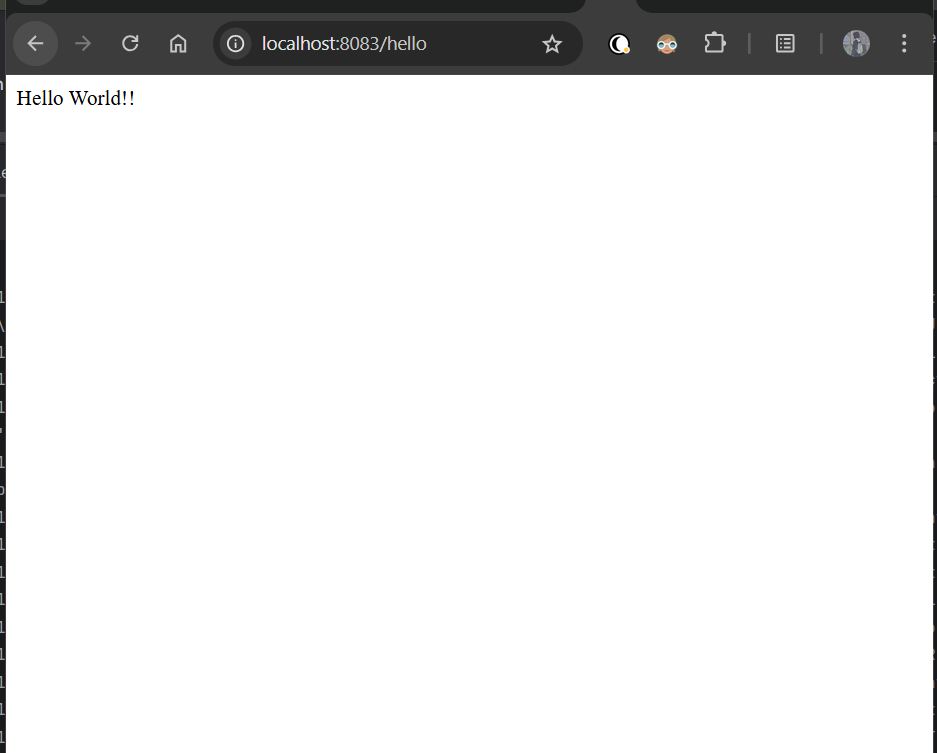
return message;

}

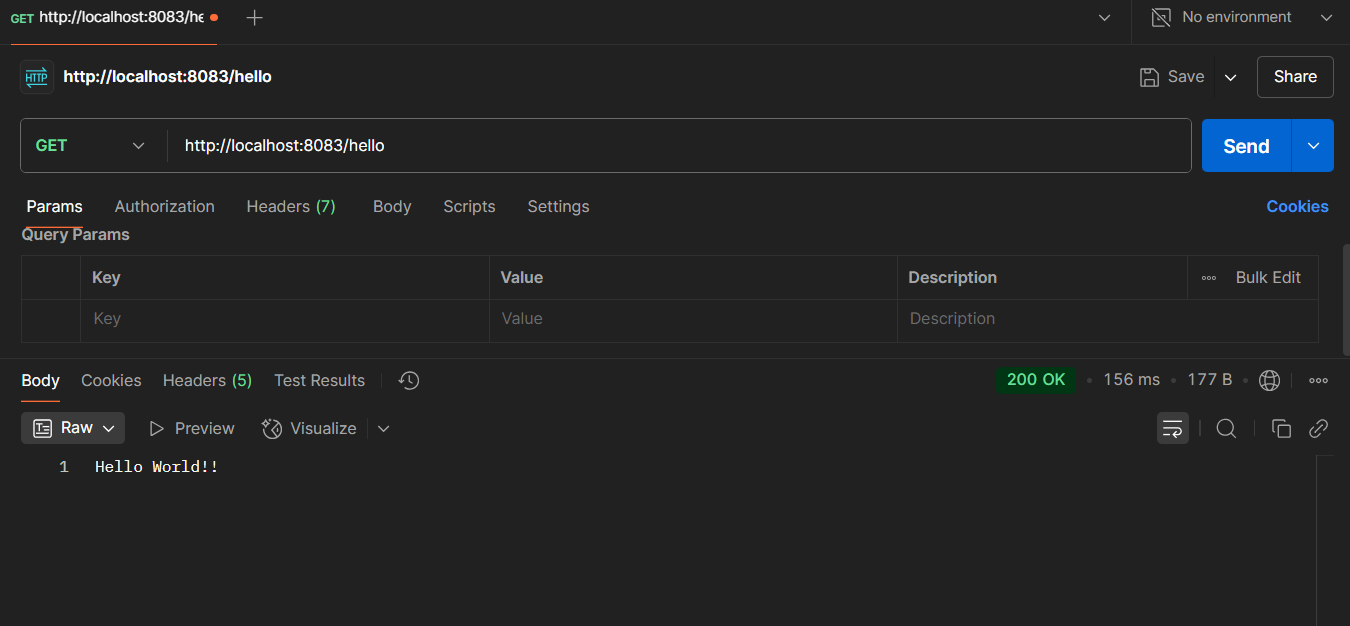
}

Output:

Chrome Browser:



Postman:



**REST - Country Web Service** 

CountryController.java

package com.cognizant.springlearn.controller;

import com.cognizant.springlearn.Country;

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.RequestMapping;

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

@RestController

public class CountryController {

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

@RequestMapping("/country")

public Country getCountryIndia() {

LOGGER.info("START getIndia");

ApplicationContext context = new ClassPathXmlApplicationContext("country.xml");

Country country = context.getBean("in", Country.class);

LOGGER.info("END getIndia");

return country;

}

}

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="country" class="com.cognizant.springlearn.Country" scope="prototype">

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

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

</bean>

<bean id="in" class="com.cognizant.springlearn.Country">

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

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

</bean>

<bean id="us" class="com.cognizant.springlearn.Country">

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

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

</bean>

<bean id="de" class="com.cognizant.springlearn.Country">

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

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

</bean>

<bean id="jp" class="com.cognizant.springlearn.Country">

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

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

</bean>

<bean id="countryList" class="java.util.ArrayList">

<constructor-arg>

<list>

<ref bean="in"></ref>

<ref bean="us"></ref>

<ref bean="de"></ref>

<ref bean="jp"></ref>

</list>

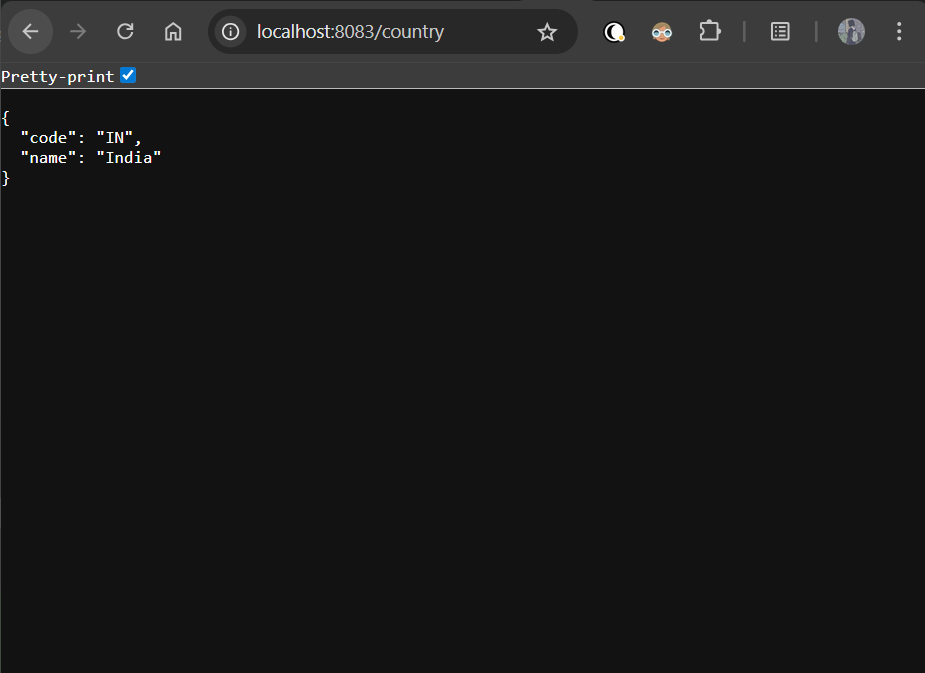
</constructor-arg>

</bean>

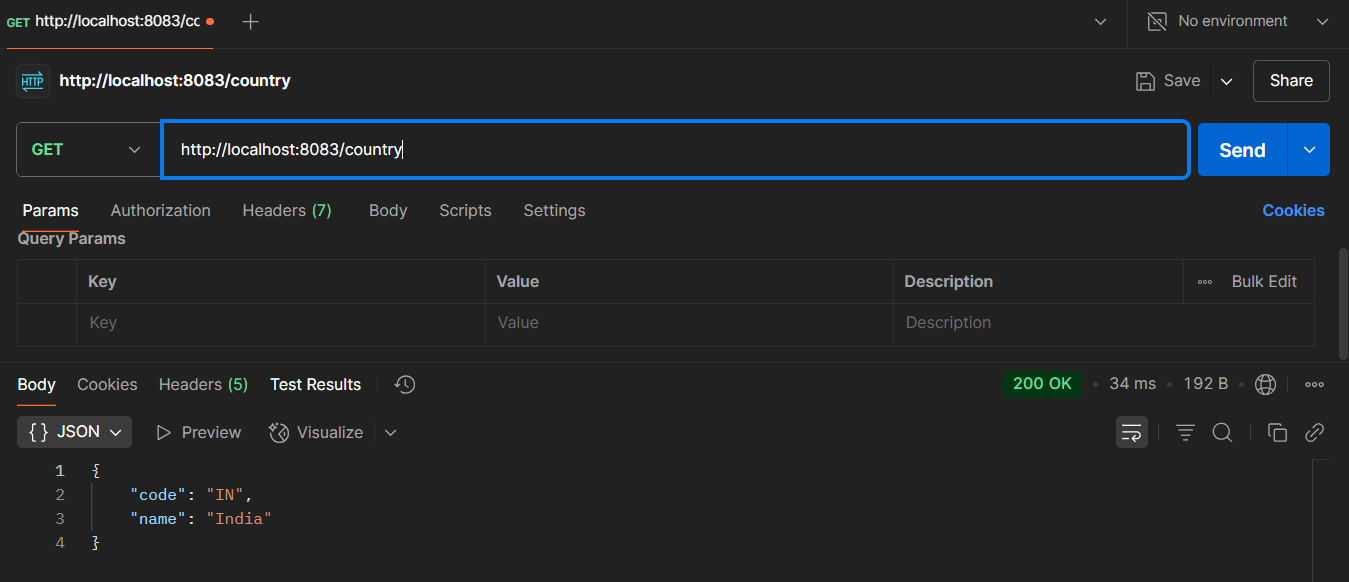
</beans>

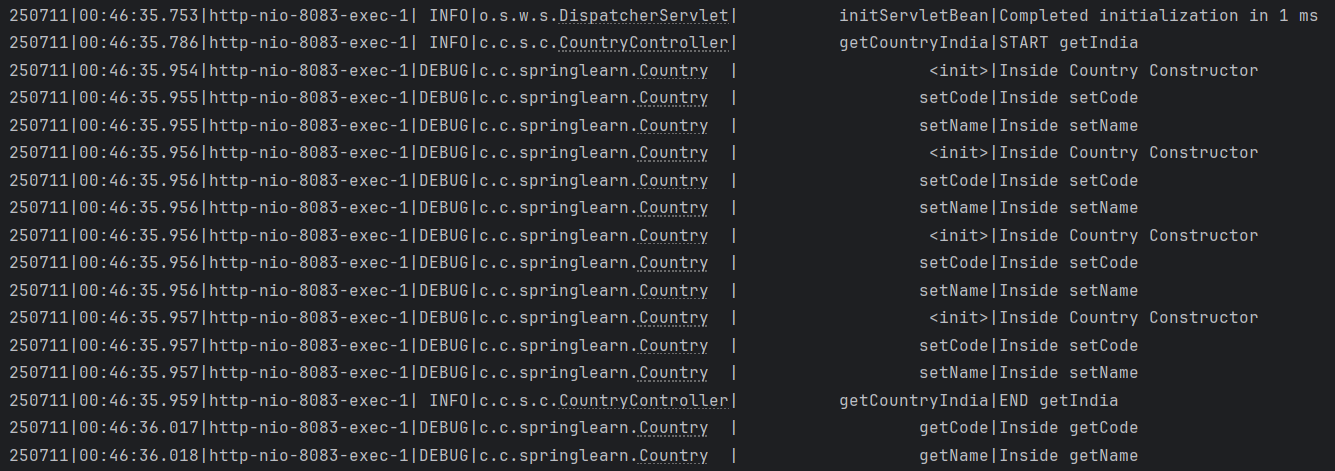
Output:

Chrome Browser



Postman



Terminal:  
  


**REST - Get all countries**

CountryController.java

package com.cognizant.springlearn.controller;

import com.cognizant.springlearn.Country;

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.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.info("START getIndia");

ApplicationContext context = new ClassPathXmlApplicationContext("country.xml");

Country country = context.getBean("in", Country.class);

LOGGER.info("END getIndia");

return country;

}

@GetMapping("/countries")

public List<Country> getAllCountries() {

LOGGER.info("START getAllCountries");

ApplicationContext context = new ClassPathXmlApplicationContext("country.xml");

List<Country> countryList = context.getBean("countryList", List.class);

LOGGER.debug("Countries: {}", countryList);

LOGGER.info("END getAllCountries");

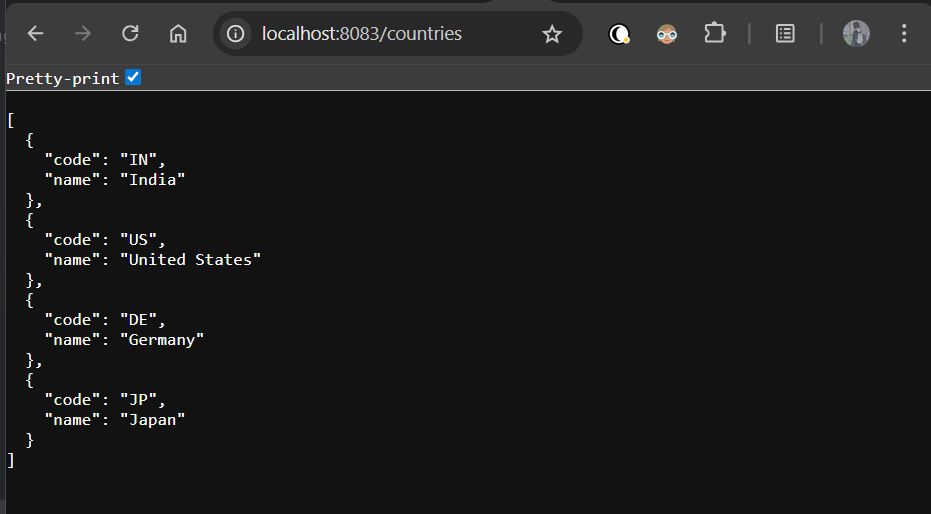
return countryList;

}

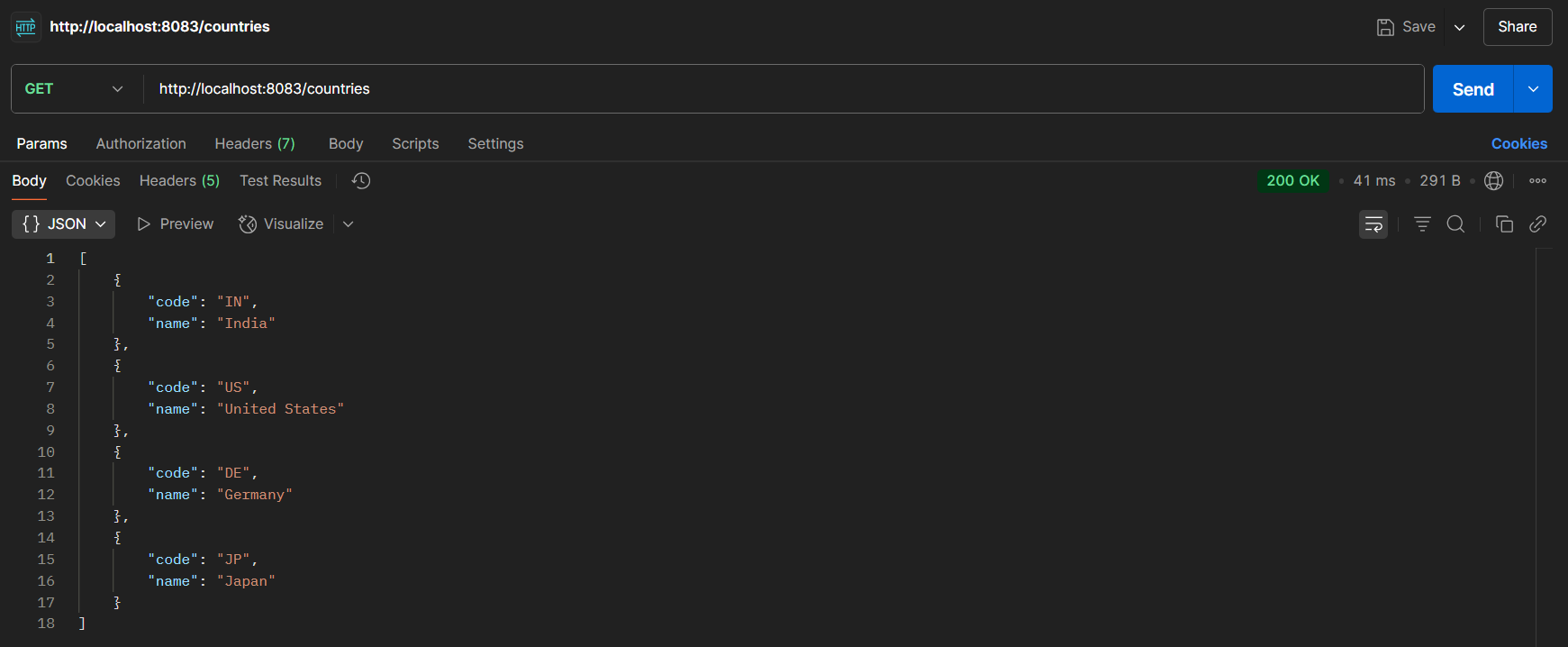
}

Output:

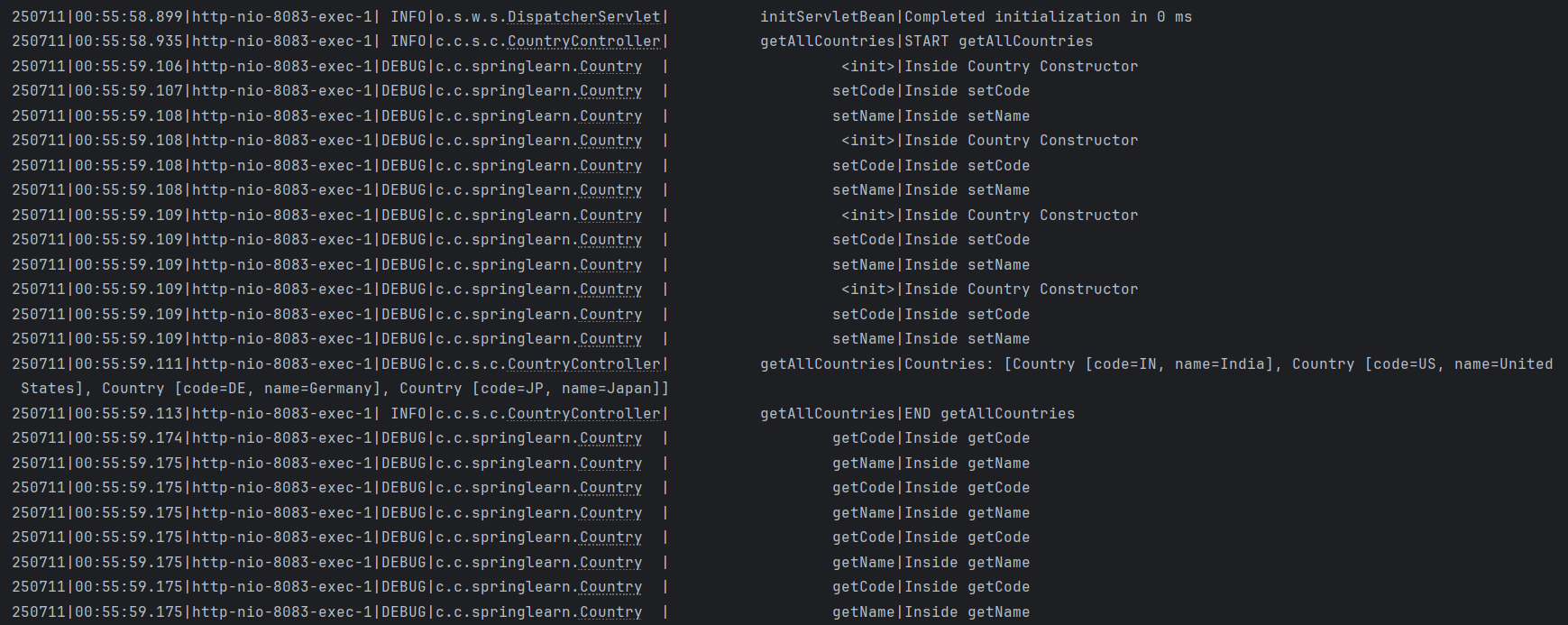
Chrome Browser:



Postman:



Terminal:



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

CountryService.java

package com.cognizant.springlearn.service;

import com.cognizant.springlearn.Country;

import org.slf4j.Logger;

import org.slf4j.LoggerFactory;

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

public Country getCountry(String code) {

LOGGER.info("START");

ApplicationContext context = new ClassPathXmlApplicationContext("country.xml");

List<Country> countries = context.getBean("countryList", List.class);

Country result = null;

for (Country country : countries) {

if (country.getCode().equalsIgnoreCase(code)) {

result = country;

break;

}

}

LOGGER.debug("Matched Country: {}", result);

LOGGER.info("END");

return result;

}

}

CountryController.java

package com.cognizant.springlearn.controller;

import com.cognizant.springlearn.Country;

import com.cognizant.springlearn.service.CountryService;

import org.slf4j.Logger;

import org.slf4j.LoggerFactory;

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

import org.springframework.context.ApplicationContext;

import org.springframework.context.support.ClassPathXmlApplicationContext;

import org.springframework.web.bind.annotation.\*;

import java.util.List;

@RestController

public class CountryController {

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

@Autowired

private CountryService countryService;

@RequestMapping("/country")

public Country getCountryIndia() {

LOGGER.info("START getIndia");

ApplicationContext context = new ClassPathXmlApplicationContext("country.xml");

Country country = context.getBean("in", Country.class);

LOGGER.info("END getIndia");

return country;

}

@GetMapping("/countries")

public List<Country> getAllCountries() {

LOGGER.info("START getAllCountries");

ApplicationContext context = new ClassPathXmlApplicationContext("country.xml");

List<Country> countryList = context.getBean("countryList", List.class);

LOGGER.debug("Countries: {}", countryList);

LOGGER.info("END getAllCountries");

return countryList;

}

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

public Country getCountry(@PathVariable String code) {

LOGGER.info("START");

Country country = countryService.getCountry(code);

LOGGER.debug("Country found: {}", country);

LOGGER.info("END");

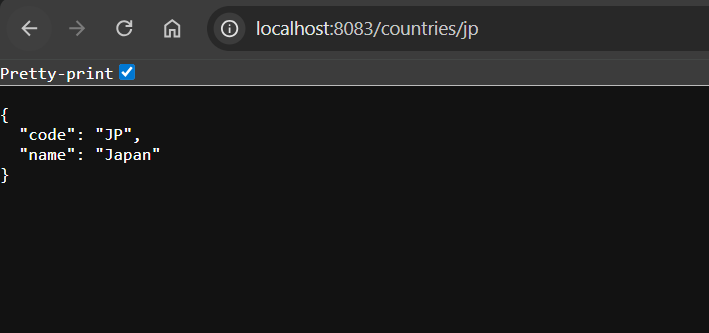
return country;

}

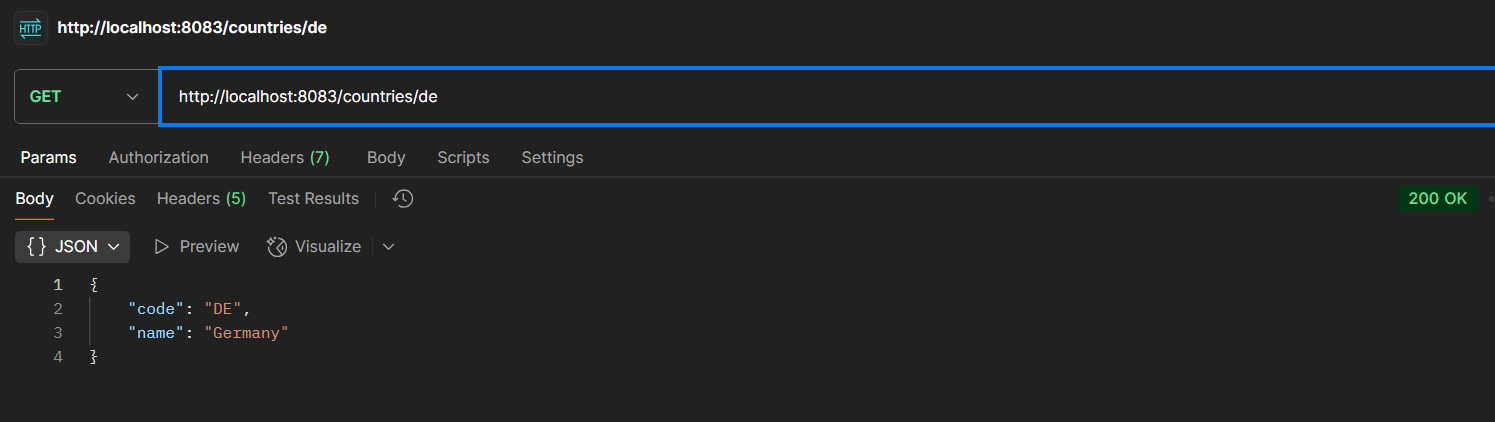
}

Output:

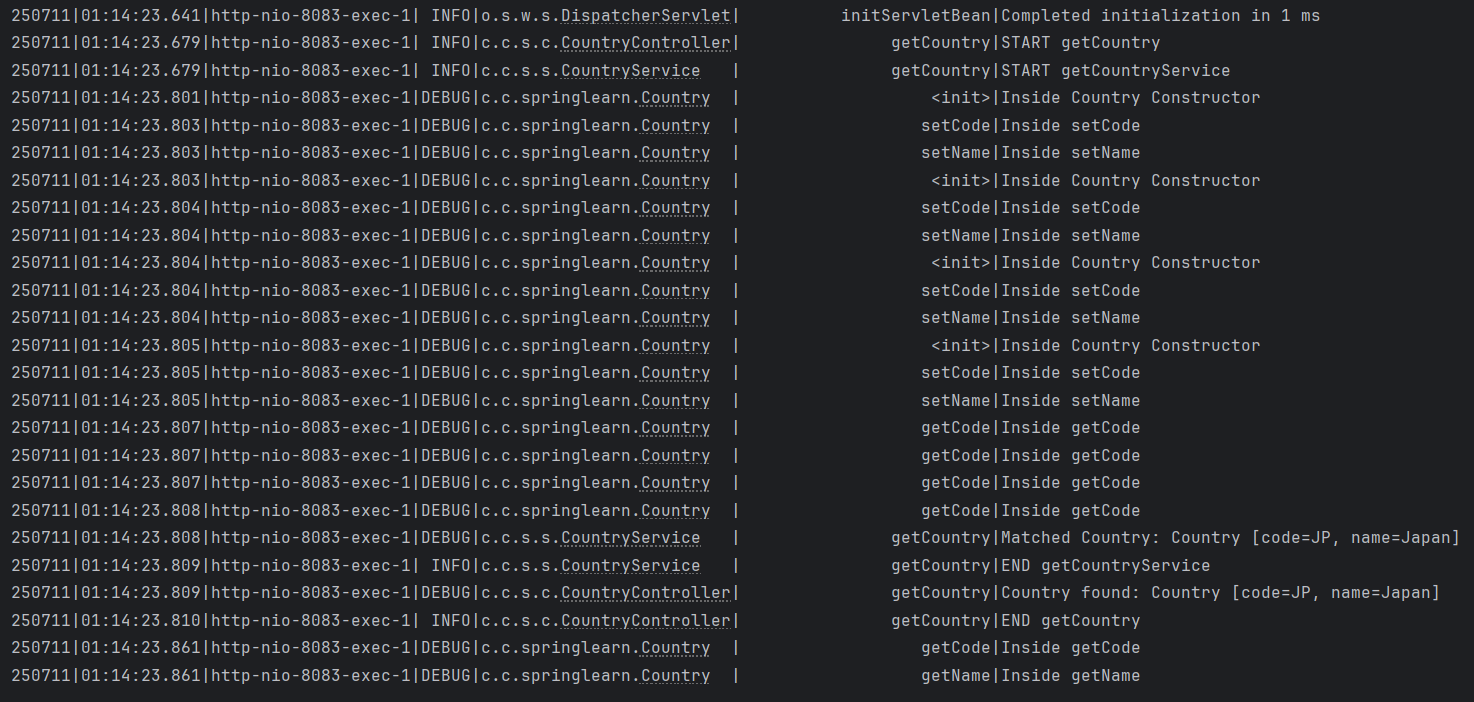
Chrome Browser:



Postman:



Terminal:



**REST - Get country exceptional scenario**

CountryNotFoundException.java

package com.cognizant.springlearn.service.exception;

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();

}

}

CountryService.java

package com.cognizant.springlearn.service;

import com.cognizant.springlearn.Country;

import com.cognizant.springlearn.service.exception.CountryNotFoundException;

import org.slf4j.Logger;

import org.slf4j.LoggerFactory;

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

public Country getCountry(String code) throws CountryNotFoundException {

LOGGER.info("START");

ApplicationContext context = new ClassPathXmlApplicationContext("country.xml");

List<Country> countries = context.getBean("countryList", List.class);

for (Country country : countries) {

if (country.getCode().equalsIgnoreCase(code)) {

LOGGER.debug("Matched Country: {}", country);

LOGGER.info("END");

return country;

}

}

LOGGER.error("Country not found: {}", code);

throw new CountryNotFoundException();

}

}

CountryController.java

package com.cognizant.springlearn.controller;

// import statements …

import java.util.List;

@RestController

public class CountryController {

// Existing code…

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

public Country getCountry(@PathVariable String code) throws CountryNotFoundException {

LOGGER.info("START");

Country country = countryService.getCountry(code);

LOGGER.debug("Country found: {}", country);

LOGGER.info("END");

return country;

}

}

application.properties:

spring.application.name=spring-learn

logging.level.org.springframework=info

logging.level.com.cognizant.springlearn=debug

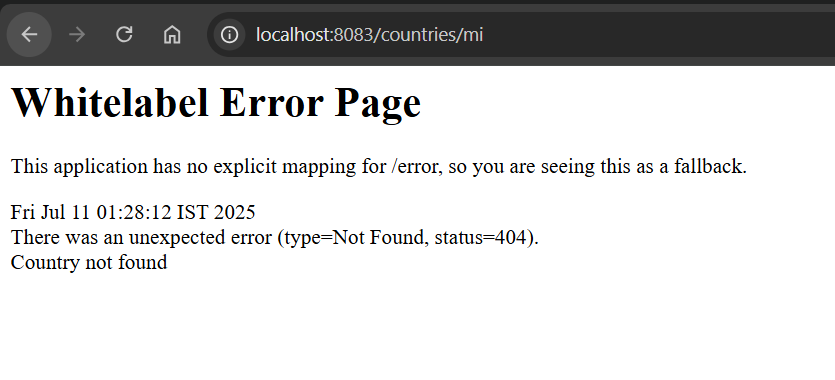
logging.pattern.console=%d{yyMMdd}|%d{HH:mm:ss.SSS}|%-20.20thread|%5p|%-25.25logger{25}|%25M|%m%n

server.port=8083

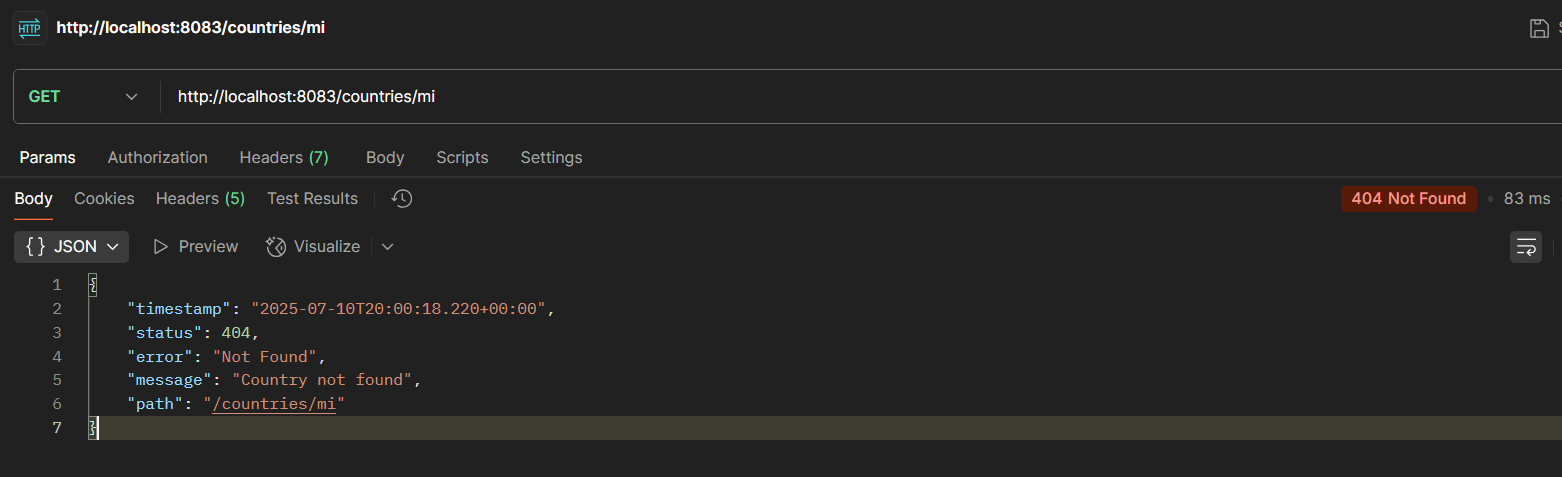
server.error.include-stacktrace=never //to exclude stack trace

Output:

Chrome Browser:



Postman:



Terminal:



**MockMVC - Test get country service**

SpringLearnApplicationTest.java

package com.cognizant.springlearn;

import static org.junit.jupiter.api.Assertions.assertNotNull;

import static org.springframework.test.web.servlet.request.MockMvcRequestBuilders.get;

import static org.springframework.test.web.servlet.result.MockMvcResultMatchers.status;

import static org.springframework.test.web.servlet.result.MockMvcResultMatchers.jsonPath;

import com.cognizant.springlearn.controller.CountryController;

import org.junit.jupiter.api.Test;

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

import org.springframework.boot.test.autoconfigure.web.servlet.AutoConfigureMockMvc;

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

import org.springframework.test.web.servlet.MockMvc;

import org.springframework.test.web.servlet.ResultActions;

@SpringBootTest

@AutoConfigureMockMvc

class SpringLearnApplicationTests {

@Autowired

private CountryController countryController;

@Autowired

private MockMvc mvc;

@Test

public void contextLoads() {

assertNotNull(countryController);

}

@Test

public void testGetCountry() throws Exception {

ResultActions actions = mvc.perform(get("/country"));

actions.andExpect(status().isOk())

.andExpect(jsonPath("$.code").exists())

.andExpect(jsonPath("$.code").value("IN"))

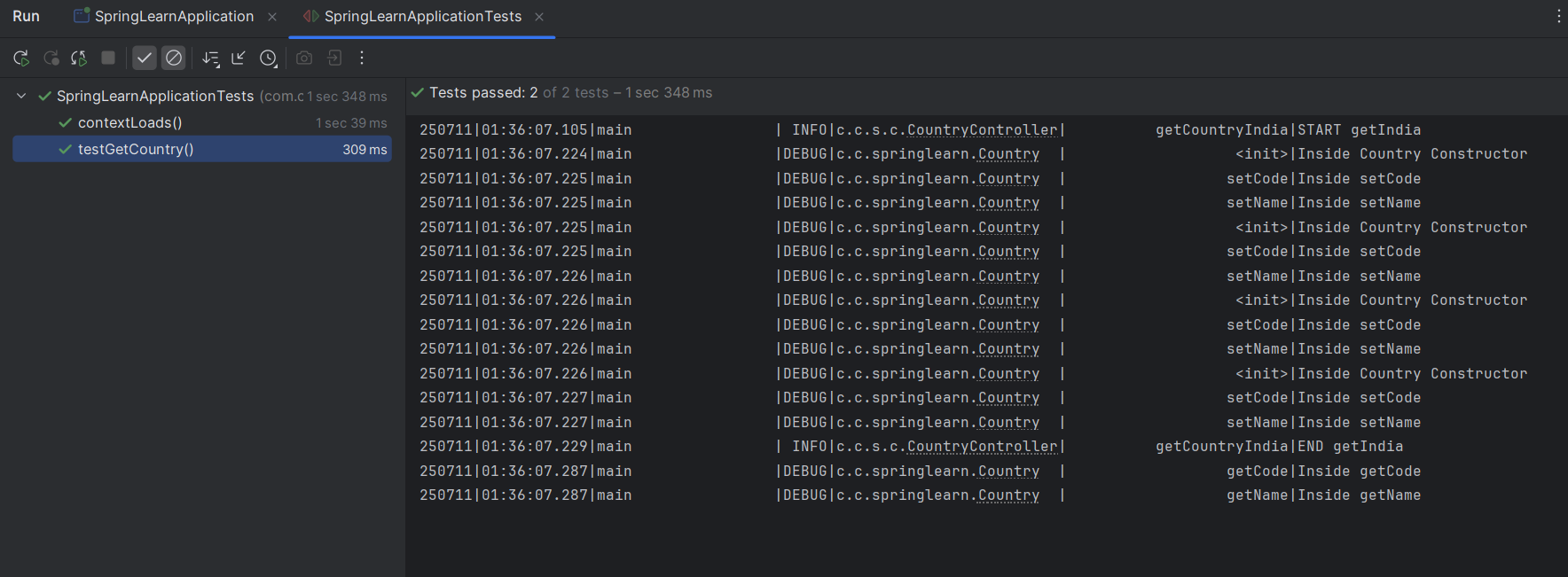
.andExpect(jsonPath("$.name").exists())

.andExpect(jsonPath("$.name").value("India"));

}

}

Output:



**MockMVC - Test get country service for exceptional scenario**

SpringLearnApplicationTest.java

package com.cognizant.springlearn;

import static org.junit.jupiter.api.Assertions.assertNotNull;

import static org.springframework.test.web.servlet.request.MockMvcRequestBuilders.get;

import static org.springframework.test.web.servlet.result.MockMvcResultMatchers.status;

import static org.springframework.test.web.servlet.result.MockMvcResultMatchers.jsonPath;

import com.cognizant.springlearn.controller.CountryController;

import org.junit.jupiter.api.Test;

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

import org.springframework.boot.test.autoconfigure.web.servlet.AutoConfigureMockMvc;

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

import org.springframework.test.web.servlet.MockMvc;

import org.springframework.test.web.servlet.ResultActions;

@SpringBootTest

@AutoConfigureMockMvc

class SpringLearnApplicationTests {

@Autowired

private CountryController countryController;

@Autowired

private MockMvc mvc;

@Test

public void contextLoads() {

assertNotNull(countryController);

}

@Test

public void testGetCountry() throws Exception {

ResultActions actions = mvc.perform(get("/country"));

actions.andExpect(status().isOk())

.andExpect(jsonPath("$.code").exists())

.andExpect(jsonPath("$.code").value("IN"))

.andExpect(jsonPath("$.name").exists())

.andExpect(jsonPath("$.name").value("India"));

}

@Test

public void testGetCountryException() throws Exception {

ResultActions actions = mvc.perform(get("/countries/zz"));

actions.andExpect(status().isNotFound())

.andExpect(status().reason("Country not found"));

}

}

Output:

