**WEEK – 4**

**HANDS ON**

**2. spring-rest-handson**

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

**CODE:**

1. **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/maven-v4\_0\_0.xsd">

<modelVersion>4.0.0</modelVersion>

<groupId>com.cognizant</groupId>

<artifactId>spring-learn</artifactId>

<version>0.0.1-SNAPSHOT</version>

<packaging>jar</packaging>

<parent>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-starter-parent</artifactId>

<version>3.2.5</version>

<relativePath/>

</parent>

<properties>

<java.version>17</java.version>

</properties>

<dependencies>

<dependency>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-starter-web</artifactId>

</dependency>

<dependency>

<groupId>jakarta.xml.bind</groupId>

<artifactId>jakarta.xml.bind-api</artifactId>

<version>3.0.1</version>

</dependency>

<dependency>

<groupId>org.glassfish.jaxb</groupId>

<artifactId>jaxb-runtime</artifactId>

<version>3.0.2</version>

</dependency>

<dependency>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-starter-test</artifactId>

<scope>test</scope>

</dependency>

</dependencies>

<build>

<plugins>

<plugin>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-maven-plugin</artifactId>

</plugin>

</plugins>

</build>

</project>

1. **Country.json**

[

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

{ "code": "US", "name": "USA" },

{ "code": "JP", "name": "Japan" },

{ "code": "DE", "name": "Germany" }

]

1. **Application.properties**

Server.port=8083

1. **CountryController.java**

package com.cognizant.controller;

import com.cognizant.model.Country;

import com.cognizant.service.CountryService;

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

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

@RestController

public class CountryController {

@Autowired

private CountryService countryService;

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

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

return countryService.getCountry(code);

}

}

1. **Country.java**

package com.cognizant.model;

import java.io.Serializable;

public class Country implements Serializable {

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;

}

}

1. **CountryService.java**

package com.cognizant.service;

import com.cognizant.model.Country;

import com.fasterxml.jackson.core.type.TypeReference;

import com.fasterxml.jackson.databind.ObjectMapper;

import org.springframework.core.io.ClassPathResource;

import org.springframework.stereotype.Service;

import java.io.InputStream;

import java.util.List;

@Service

public class CountryService {

public Country getCountry(String code) throws Exception {

List<Country> countries = getAllCountries();

return countries.stream()

.filter(c -> c.getCode().equalsIgnoreCase(code))

.findFirst()

.orElseThrow(() -> new Exception("Country not found with code: " + code));

}

private List<Country> getAllCountries() throws Exception {

ObjectMapper mapper = new ObjectMapper();

InputStream is = new ClassPathResource("country.json").getInputStream();

return mapper.readValue(is, new TypeReference<List<Country>>() {});

}

}

1. **SpringLearnApplication**

package com.cognizant;

import org.springframework.boot.SpringApplication;

import org.springframework.boot.autoconfigure.SpringBootApplication;

@SpringBootApplication

public class SpringLearnApplication {

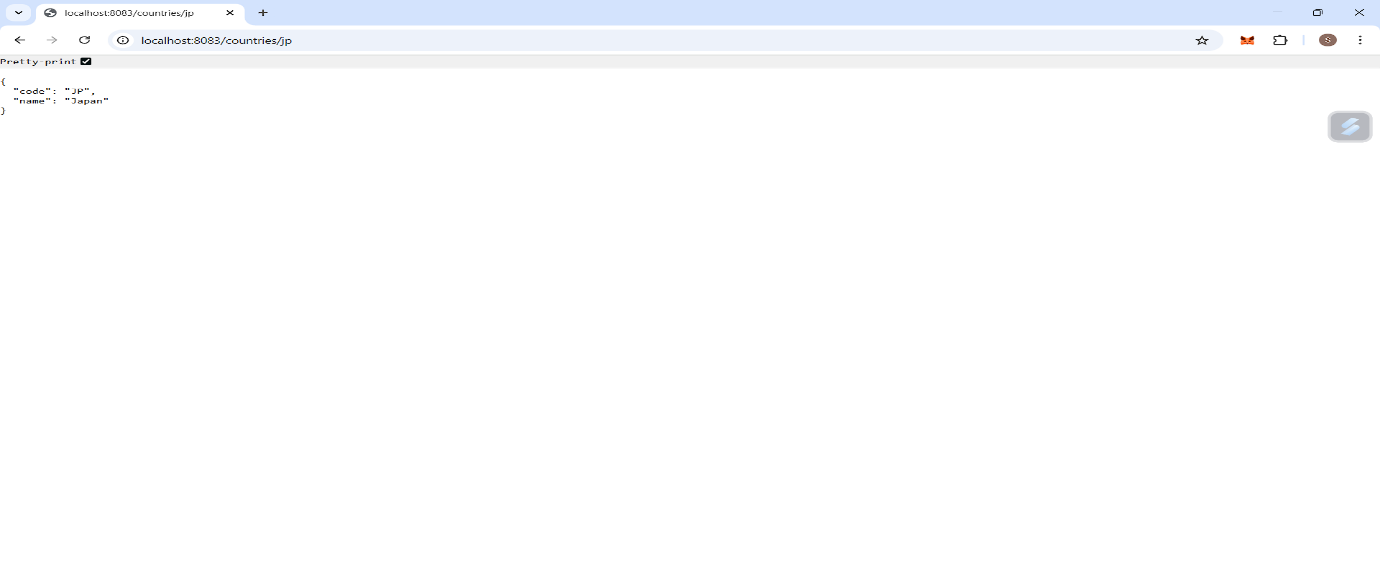
public static void main(String[] args) {

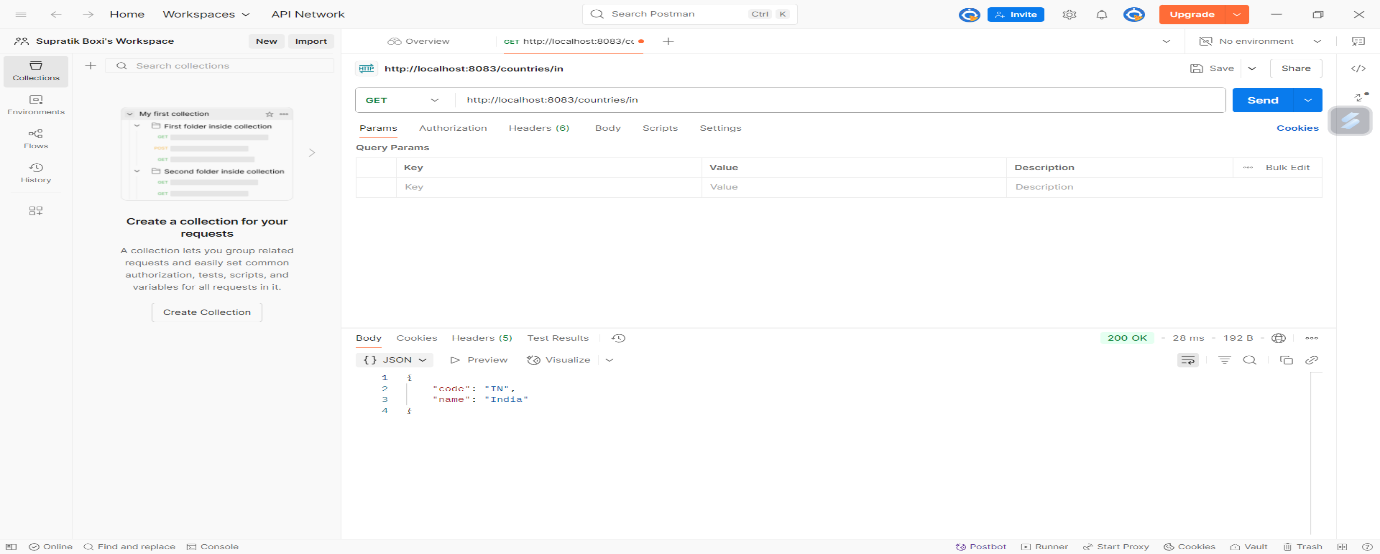
SpringApplication.run(SpringLearnApplication.class, args);

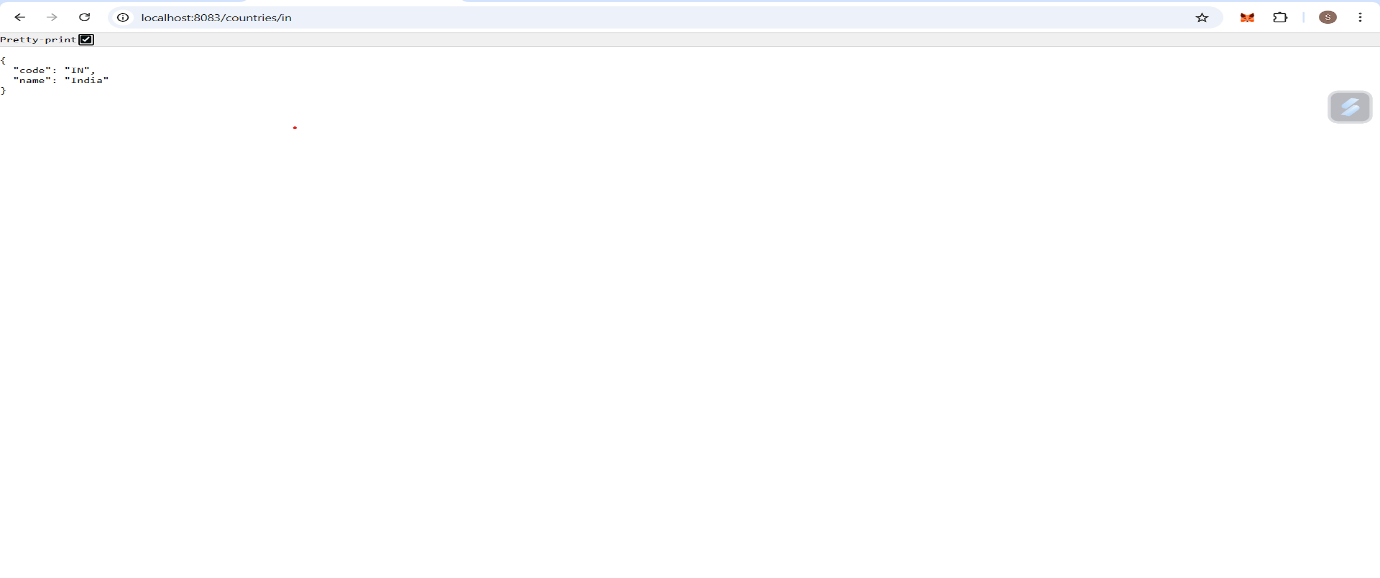
}

}

**OUTPUT**

****

****

****