**REST - Country Web Service**   
  
Write a REST service that returns India country details in the earlier created spring learn application.

**Country.java:**

package com.cognizant.spring\_learn.model;

public class Country {

private String code;

private String 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;

}

}

**CountryController.java:**

package com.cognizant.spring\_learn.controller;

import com.cognizant.spring\_learn.model.Country;

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 {

@RequestMapping("/country")

public Country getCountryIndia() {

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

Country india = (Country) context.getBean("in");

return india;

}

}  
**spring.xml:**

<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">

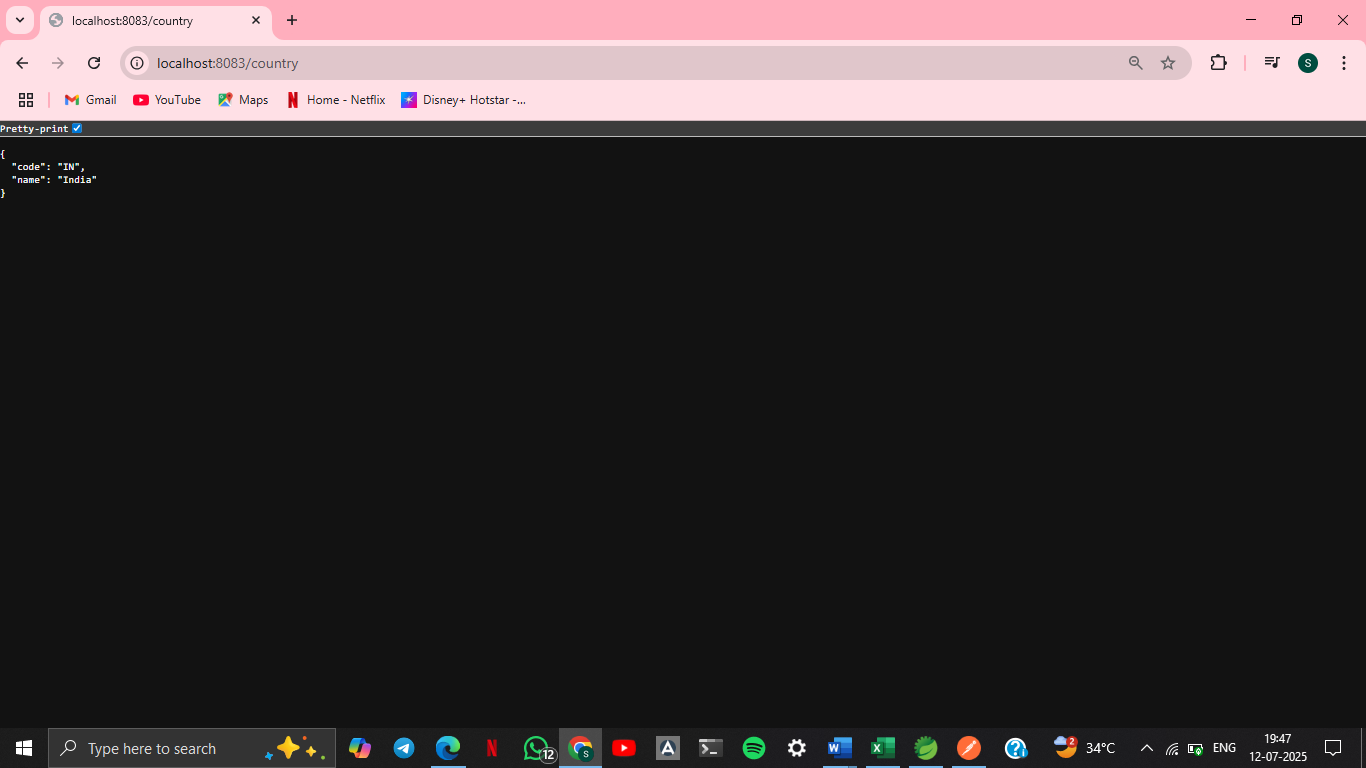
<bean id="in" class="com.cognizant.spring\_learn.model.Country">

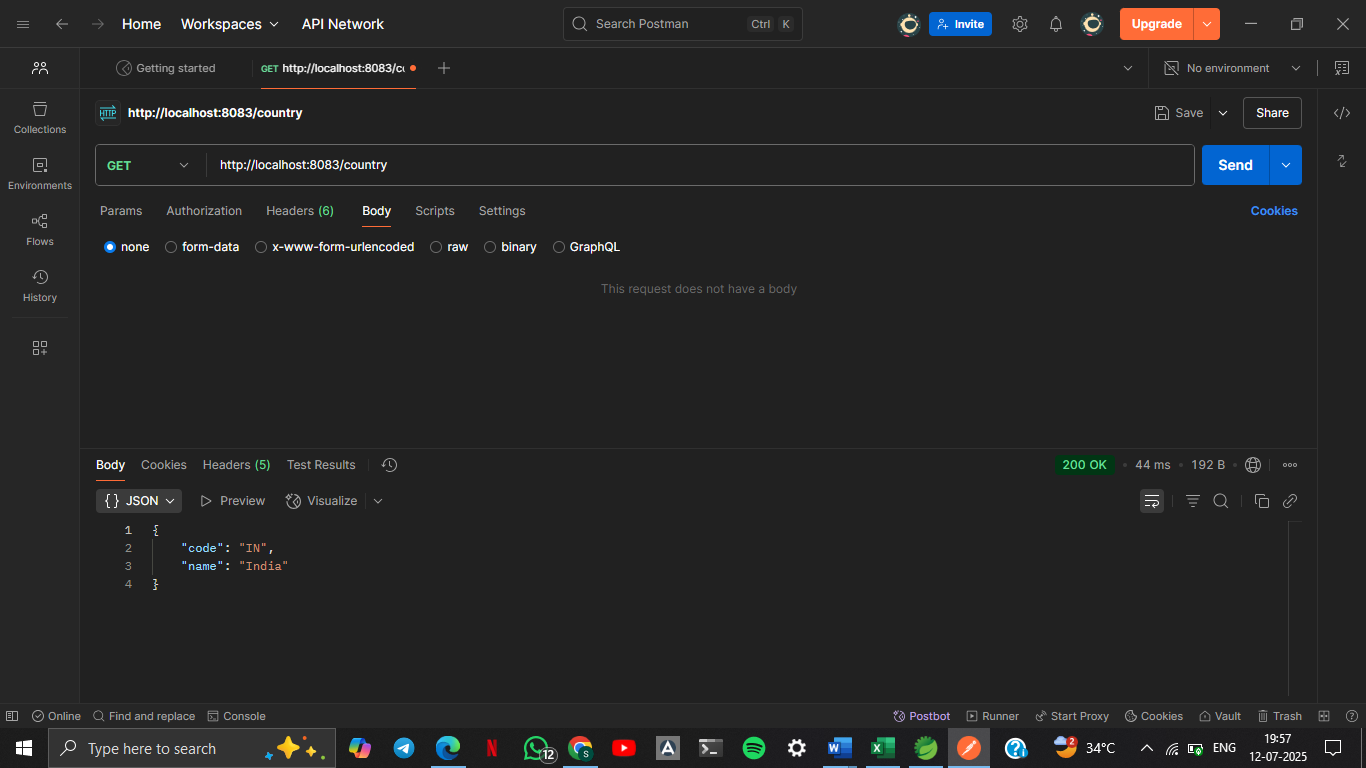
<property name="code" value="IN"/>

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

</bean>

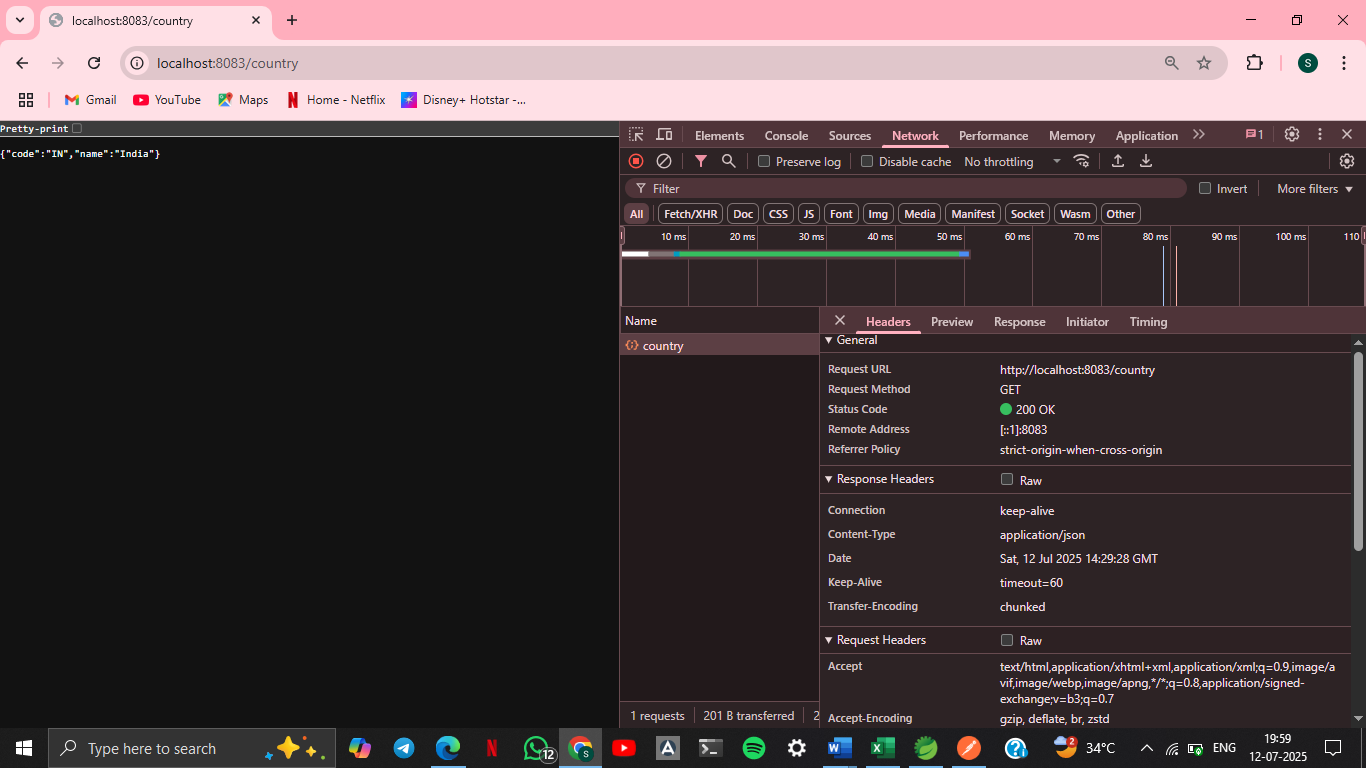
</beans>

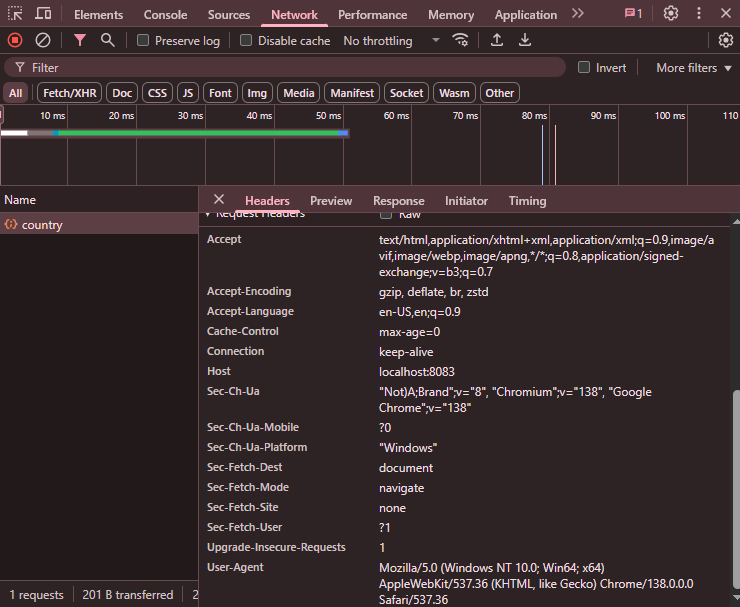




**SME to explain the following aspects:**

* What happens in the controller method?
* @RequestMapping("/country") maps the method to a GET request.
* Spring loads the spring.xml file from src/main/resources.
* Retrieves the bean id="in" of type Country.
* Returns the Country object.
* **Spring Boot + Jackson** auto-converts this Java object to JSON.
* How the bean is converted into JSON reponse?
* Spring Boot has **Jackson** in its dependencies.
* When a controller (annotated with @RestController) returns an object, Spring uses Jackson to serialize it.
* return india;
* becomes:
  + json
  + {
  + "code": "IN",
  + "name": "India"
  + }
* In network tab of developer tools show the HTTP header details received





* In postman click on "Headers" tab to view the HTTP header details received

