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

**Path:** src/main/resources/country.xml  
**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

http://www.springframework.org/schema/beans/spring-beans.xsd">

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

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

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

</bean>

</beans>

**Path :**src/main/java/com/cognizant/spring\_learn/controller/CountryController.java

**File: CountryController.java**

package com.cognizant.spring\_learn.controller;

import com.cognizant.spring\_learn.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() {

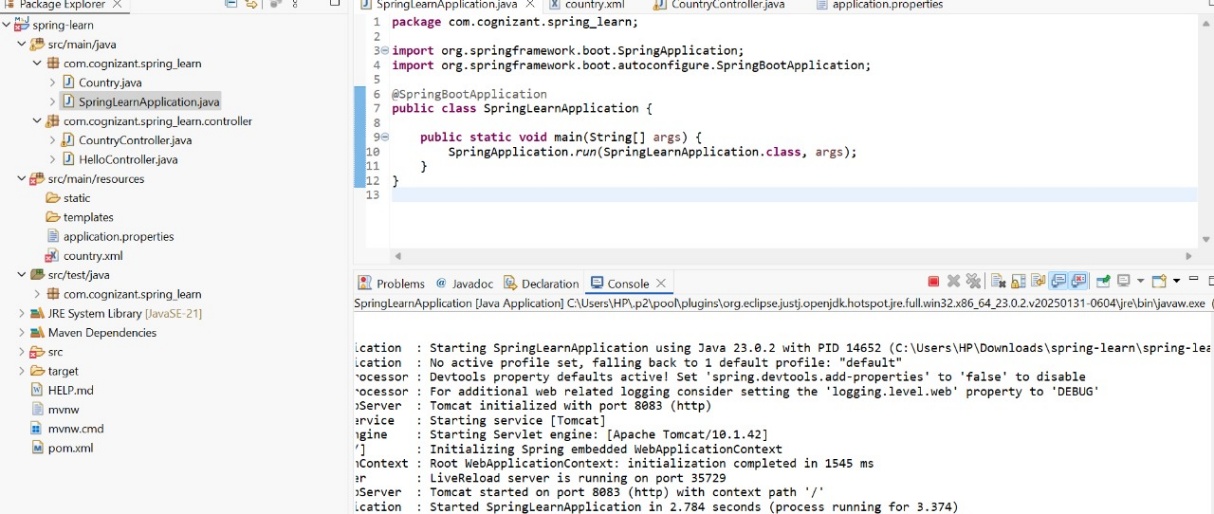
ApplicationContext context = new ClassPathXmlApplicationContext("country.xml");

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

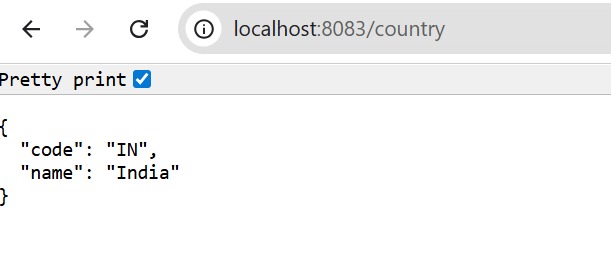
return country;

}

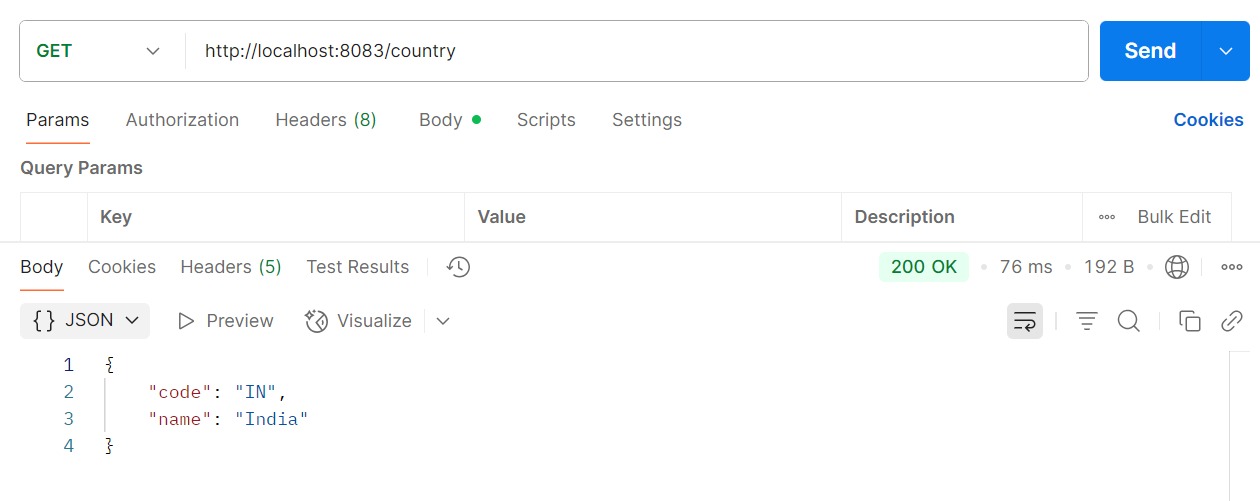
}



**BROWSER :-**



**POSTMAN :-**



**SME to explain the following aspects:**

**What happens in the controller method?**

* When /country is requested, getCountryIndia() is invoked.
* It loads the country bean from country.xml.
* Spring automatically converts the returned Country object to JSON using Jackson (a default dependency in Spring Boot)

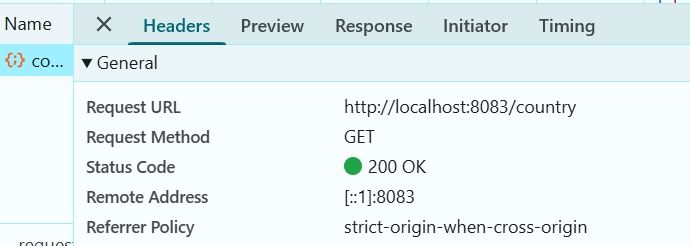
**How the bean is converted into JSON reponse?**

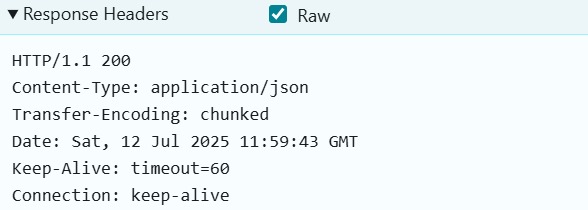
* Spring Boot includes **Jackson** library by default.
* It uses **@RestController + Return Type = Object** to auto-convert object to JSON.

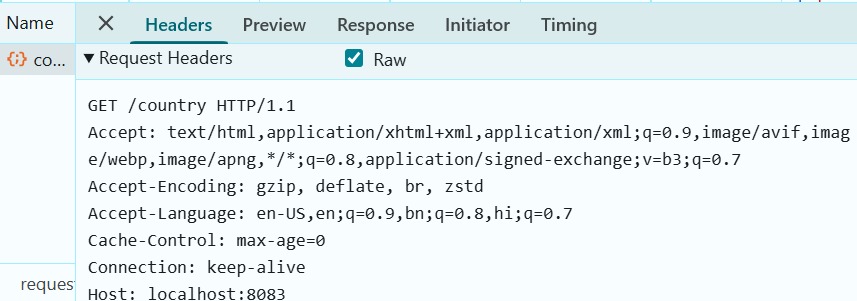
**In network tab of developer tools show the HTTP header details received**

**In Chrome :** Right-click > Inspect > Network tab

* Reload http://localhost:8083/country
* Click the /country request
* Go to the **"Headers" tab**
* View Request Headers & Response Headers







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

**In Postman:**

* Open Postman
* Send GET request to http://localhost:8083/country
* Click the **"Headers" tab**

