Hello World REST API:

Steps:

1. CountryController.java:

```java
package com.cognizant.spring_learn.controller;

import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.web.bind.annotation.GetMapping;
import org.springframework.web.bind.annotation.PathVariable;
import org.springframework.web.bind.annotation.RestController;

import com.cognizant.spring_learn.model.Country;
import com.cognizant.spring_learn.service.CountryService;

@RestController
public class CountryController {

    @Autowired
    private CountryService countryService;

    @GetMapping("/countries/{code}")
    public Country getCountry(@PathVariable String code) {
        return countryService.getCountry(code);
    }
}
```

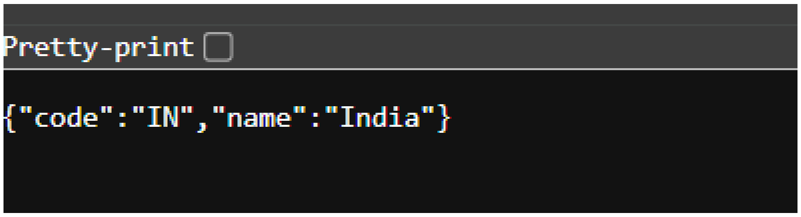
2. Country.java:

```
package com.cognizant.spring_learn.model;


public class Country {
    private String code;
    private String name;


    // Default constructor
    public Country() {}


    // Parameterized constructor
    public Country(String code, String name) {
        this.code = code;
        this.name = name;
    }


    // Getters and Setters
    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;
    }

    @Override
    public String toString() {
        return "Country{" +
                "code='" + code + '\'' +
                ", name='" + name + '\'' +
                '}';
    }
}
```

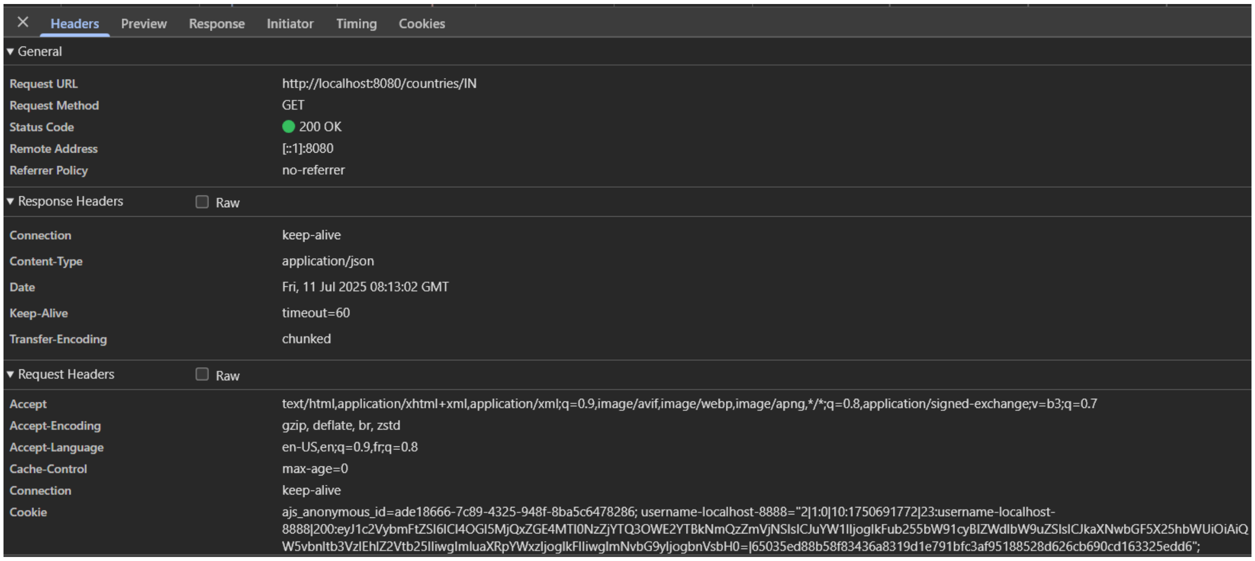
3. CountryService.java:

```
// src/main/java/com/cognizant/springlearn/service/CountryService.java

package com.cognizant.spring_learn.service;


import java.util.ArrayList;

import java.util.List;


import org.springframework.stereotype.Service;


import com.cognizant.spring_learn.model.Country;


@Service

public class CountryService {


    private List<Country> countries = new ArrayList<>();


    public CountryService() {

        // Initialize with some countries

        countries.add(new Country("IN", "India"));

        countries.add(new Country("US", "United States"));

        countries.add(new Country("DE", "Germany"));

        countries.add(new Country("JP", "Japan"));
    }

    public Country getCountry(String code) {
        return countries.stream()
                .filter(country -> country.getCode().equalsIgnoreCase(code))
                .findFirst()
                .orElse(null);
    }
}
```

Output:



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



▾ General

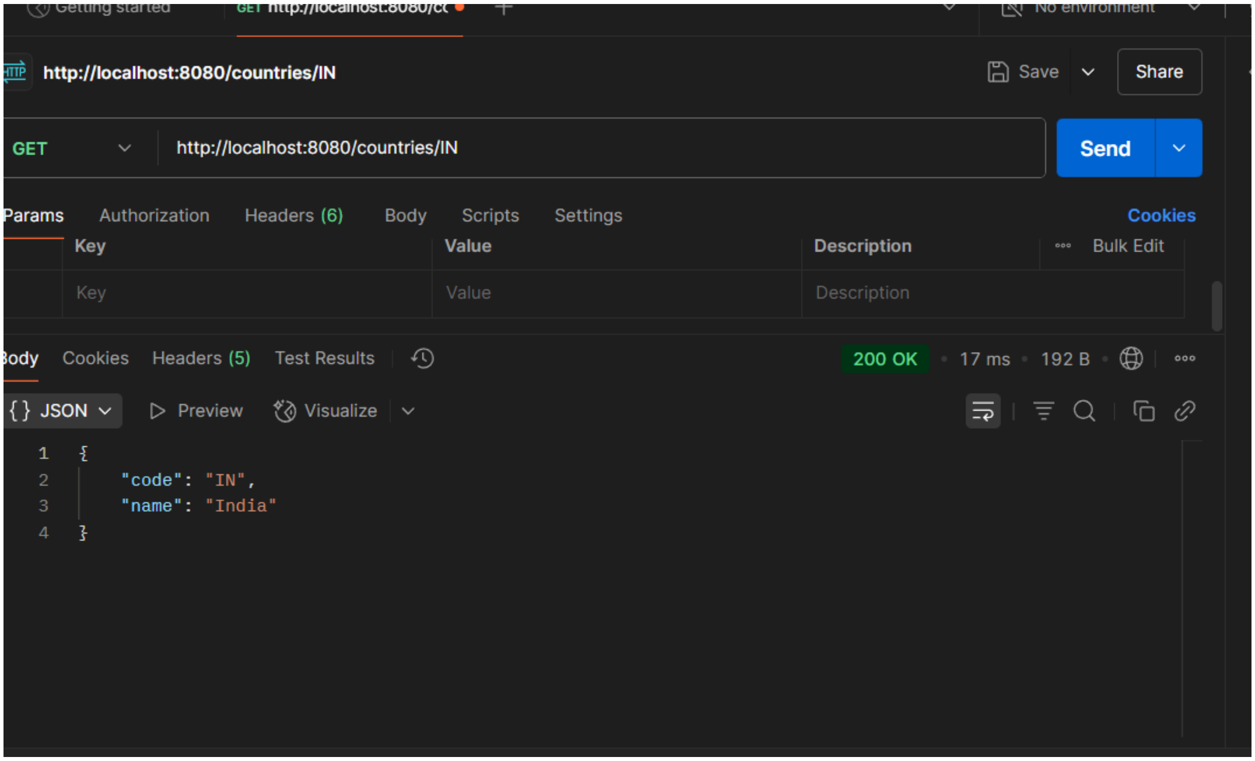
Request URL	http://localhost:8080/countries/IN
Request Method	GET
Status Code	● 200 OK
Remote Address	[::1]:8080
Referrer Policy	no-referrer

▾ Response Headers □ Raw

Connection	keep-alive
Content-Type	application/json
Date	Fri, 11 Jul 2025 08:13:02 GMT
Keep-Alive	timeout=60
Transfer-Encoding	chunked

▾ Request Headers □ Raw

Accept	text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.7
Accept-Encoding	gzip, deflate, br, zstd
Accept-Language	en-US,en;q=0.9,fr;q=0.8
Cache-Control	max-age=0
Connection	keep-alive
Cookie	ajs_anonymous_id=ade18666-7c89-4325-948f-8ba5c6478286; username-localhost-8888="2|1:0|10:1750691772|23:username-localhost-8888|200:eyJ1c2VybmFtZSI6ICI4OGI5MjQxZGE4MTI0NzZjYTQ3OWE2YTBkNmQzZmVjNSIsICJuYW1lIjogIkFub255bW91cyBIZWdlbW9uZSIsICJkaXNwbGF5X25hbWUiOiAiQW5vbnltb3VzIEhlZ2Vtb25lIiwgImluaXRpYWxzIjogIkFIIiwgImNvbG9yIjogbnVsbH0=|65035ed88b58f83436a8319d1e791bfc3af95188528d626cb690cd163325edd6";



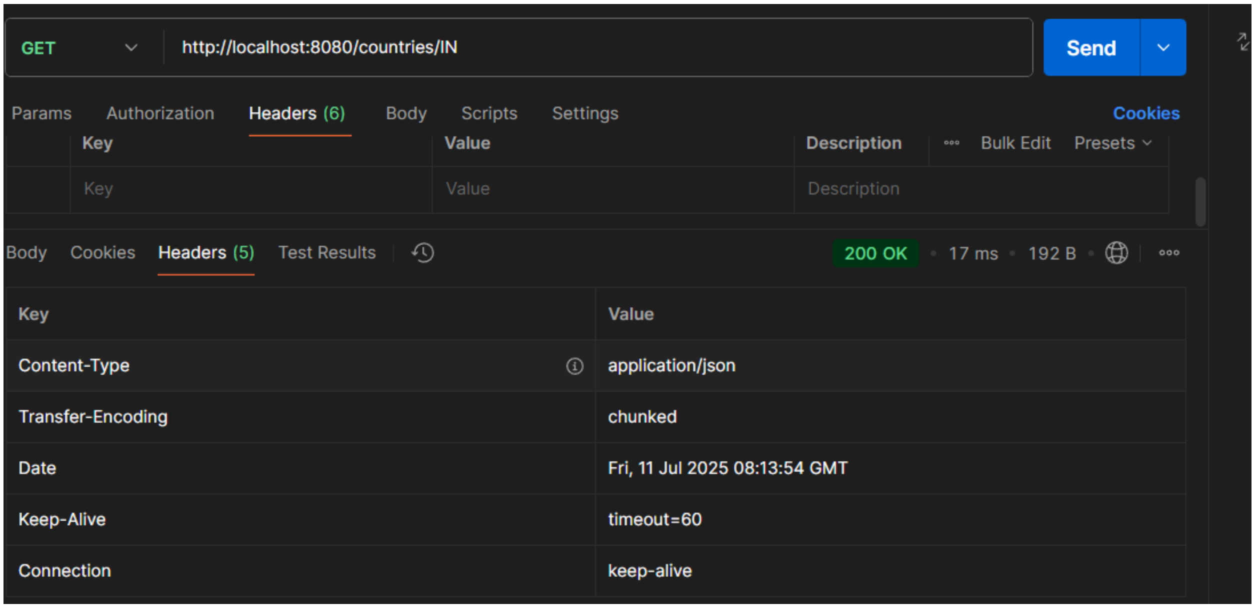
http://localhost:8080/countries/IN

GET http://localhost:8080/countries/IN

Key	Value	Description
Key	Value	Description

200 OK • 17 ms • 192 B

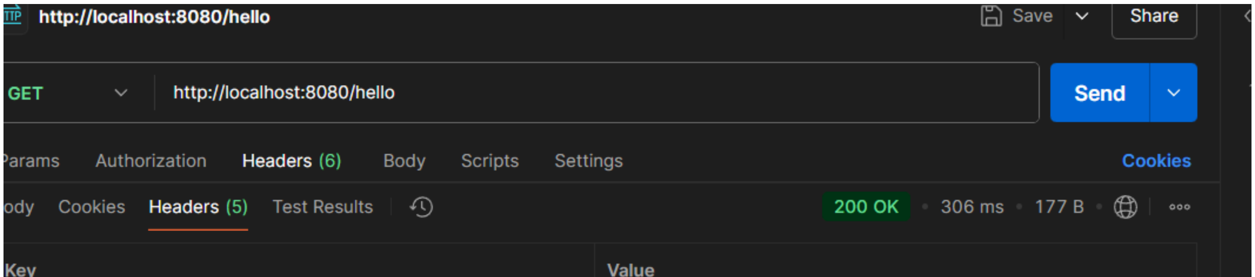
```
{
    "code": "IN",
    "name": "India"
}
```



GET http://localhost:8080/countries/IN

200 OK • 17 ms • 192 B

Key	Value
Content-Type	application/json
Transfer-Encoding	chunked
Date	Fri, 11 Jul 2025 08:13:54 GMT
Keep-Alive	timeout=60
Connection	keep-alive



http://localhost:8080/hello

GET http://localhost:8080/hello

200 OK • 306 ms • 177 B

Key	Value