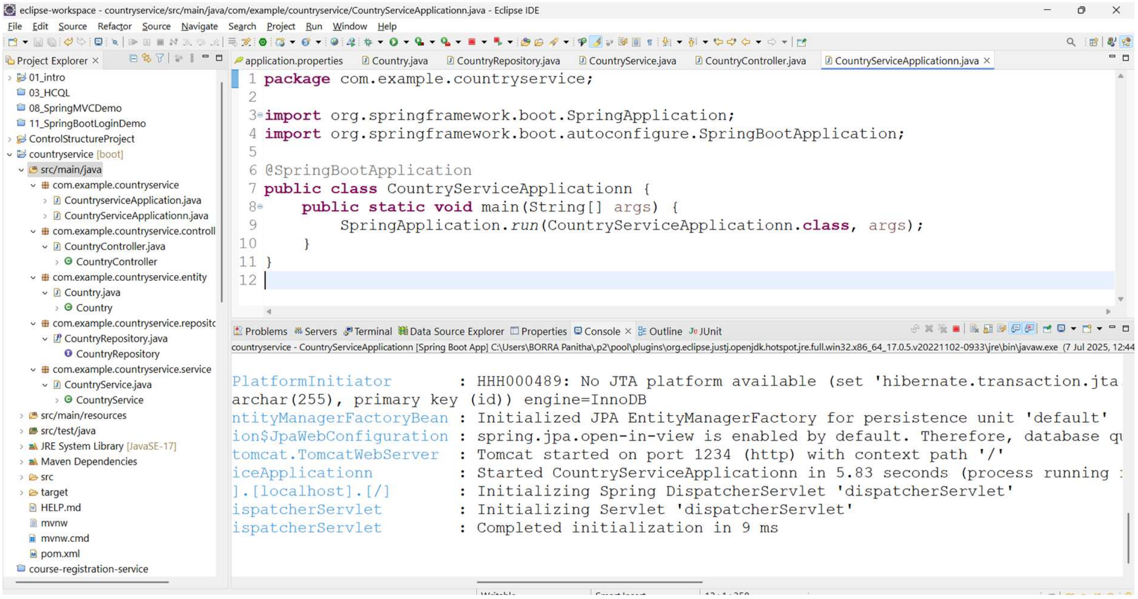
Exercise 9: Creating a Spring Boot Application

Implement services for managing Country

```
package com.example.countryservice.entity;


import jakarta.persistence.*;


@Entity
public class Country {


    @Id
    @GeneratedValue(strategy = GenerationType.IDENTITY)
    private int id;


    private String name;
    private String capital;




    public int getId() { return id; }
    public void setId(int id) { this.id = id; }


    public String getName() { return name; }
    public void setName(String name) { this.name = name; }


    public String getCapital() { return capital; }
    public void setCapital(String capital) { this.capital = capital; }
}
package com.example.countryservice.repository;


import com.example.countryservice.entity.Country;
import org.springframework.data.jpa.repository.JpaRepository;

public interface CountryRepository extends JpaRepository<Country, Integer> {
}
package com.example.countryservice.service;

import com.example.countryservice.entity.Country;
import com.example.countryservice.repository.CountryRepository;
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.stereotype.Service;

import java.util.List;
import java.util.Optional;

@Service
public class CountryService {

    @Autowired
    private CountryRepository countryRepository;

    public Country addCountry(Country country) {
        return countryRepository.save(country);
    }

    public List<Country> getAllCountries() {
        return countryRepository.findAll();
    }
    public Optional<Country> getCountryById(int id) {
        return countryRepository.findById(id);
    }

    public Country updateCountry(Country country) {
        return countryRepository.save(country);
    }

    public void deleteCountry(int id) {
        countryRepository.deleteById(id);
    }
}
package com.example.countryservice.controller;

import com.example.countryservice.entity.Country;
import com.example.countryservice.service.CountryService;
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.web.bind.annotation.*;

import java.util.List;

@RestController
@RequestMapping("/countries")
public class CountryController {

    @Autowired
    private CountryService countryService;
    @PostMapping
    public Country add(@RequestBody Country country) {
        return countryService.addCountry(country);
    }

    @GetMapping
    public List<Country> getAll() {
        return countryService.getAllCountries();
    }

    @GetMapping("/{id}")
    public Country getById(@PathVariable int id) {
        return countryService.getCountryById(id).orElse(null);
    }

    @PutMapping
    public Country update(@RequestBody Country country) {
        return countryService.updateCountry(country);
    }

    @DeleteMapping("/{id}")
    public String delete(@PathVariable int id) {
        countryService.deleteCountry(id);
        return "Country with ID " + id + " deleted";
    }
}
package com.example.countryservice;
```

```
import org.springframework.boot.SpringApplication;

import org.springframework.boot.autoconfigure.SpringBootApplication;


@SpringBootApplication

public class CountryServiceApplicationn {

    public static void main(String[] args) {

        SpringApplication.run(CountryServiceApplicationn.class, args);

    }

}
```

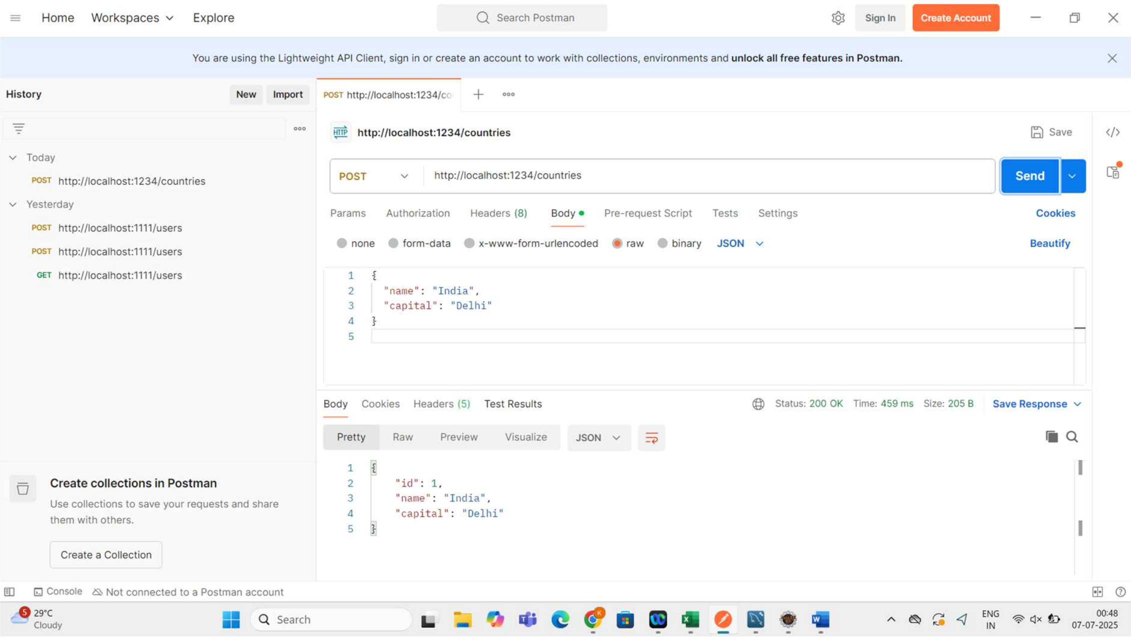
OUTPUT:



```
package com.example.countryservice;

import org.springframework.boot.SpringApplication;
import org.springframework.boot.autoconfigure.SpringBootApplication;

@SpringBootApplication
public class CountryServiceApplicationn {
    public static void main(String[] args) {
        SpringApplication.run(CountryServiceApplicationn.class, args);
    }
}

```

countryservice - CountryServiceApplicationn [Spring Boot App] C:\Users\BORRA Panitha\.p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_17.0.5.v20221102-0933\jre\bin\javaw.exe (7 Jul 2025, 12:44

```
PlatformInitiator       : HHH000489: No JTA platform available (set 'hibernate.transaction.jta
archar(255), primary key (id)) engine=InnoDB
ntityManagerFactoryBean : Initialized JPA EntityManagerFactory for persistence unit 'default'
ion$JpaWebConfiguration : spring.jpa.open-in-view is enabled by default. Therefore, database qu
tomcat.TomcatWebServer  : Tomcat started on port 1234 (http) with context path '/'
iceApplicationn         : Started CountryServiceApplicationn in 5.83 seconds (process running :
].[localhost].[/]       : Initializing Spring DispatcherServlet 'dispatcherServlet'
ispatcherServlet        : Initializing Servlet 'dispatcherServlet'
ispatcherServlet        : Completed initialization in 9 ms
```



You are using the Lightweight API Client, sign in or create an account to work with collections, environments and **unlock all free features in Postman.**

History:

- POST http://localhost:1234/countries
- POST http://localhost:1111/users
- POST http://localhost:1111/users
- GET http://localhost:1111/users

POST http://localhost:1234/countries

Body: ○ none ○ form-data ○ x-www-form-urlencoded ● raw ○ binary JSON

```
{
  "name": "India",
  "capital": "Delhi"
}

```

Status: 200 OK Time: 459 ms Size: 205 B

JSON

```
{
    "id": 1,
    "name": "India",
    "capital": "Delhi"
}
```