**Spring Data JPA - Quick Example**

**Find a country based on country code**

**CODE:**

controller 🡪 CountryController.java

package org.example.springdatajpahandson.controller;  
  
import org.example.springdatajpahandson.model.Country;  
import org.example.springdatajpahandson.service.CountryService;  
import org.springframework.beans.factory.annotation.Autowired;  
import org.springframework.http.ResponseEntity;  
import org.springframework.web.bind.annotation.\*;  
  
import java.util.List;  
  
@RestController  
@RequestMapping("/countries")  
public class CountryController {  
  
 @Autowired  
 private CountryService countryService;  
  
 @PostMapping  
 public Country addCountry(@RequestBody Country country) {  
 return countryService.saveCountry(country);  
 }  
  
 @GetMapping  
 public List<Country> getAllCountries() {  
 return countryService.getAllCountries();  
 }  
  
 @GetMapping("/{id}")  
 public ResponseEntity<Country> getCountryById(@PathVariable Long id) {  
 return countryService.getCountryById(id)  
 .map(ResponseEntity::*ok*)  
 .orElse(ResponseEntity.*notFound*().build());  
 }  
  
 @DeleteMapping("/{id}")  
 public ResponseEntity<Void> deleteCountry(@PathVariable Long id) {  
 countryService.deleteCountry(id);  
 return ResponseEntity.*noContent*().build();  
 }  
  
 @GetMapping("/code/{code}")  
 public ResponseEntity<Country> getCountryByCode(@PathVariable String code) {  
 return countryService.getCountryByCode(code)  
 .map(ResponseEntity::*ok*)  
 .orElse(ResponseEntity.*notFound*().build());  
 }  
}

model🡪Country.java

package org.example.springdatajpahandson.model;  
import jakarta.persistence.\*;  
@Entity  
public class Country {  
 @Id  
 @GeneratedValue(strategy = GenerationType.*IDENTITY*)  
 private Long id;  
 private String name;  
 private String code;  
 public Long getId() {  
 return id;  
 }  
 public void setId(Long id) {  
 this.id = id;  
 }  
 public String getName() {  
 return name;  
 }  
 public void setName(String name) {  
 this.name = name;  
 }  
 public String getCode() {  
 return code;  
 }  
 public void setCode(String code) {  
 this.code = code;  
 }  
}

repository🡪CountryRepository.java

package org.example.springdatajpahandson.repository;  
  
import org.example.springdatajpahandson.model.Country;  
import org.springframework.data.jpa.repository.JpaRepository;  
import java.util.Optional;  
  
public interface CountryRepository extends JpaRepository<Country, Long> {  
 Optional<Country> findByCode(String code);  
}

service🡪CountryService

package org.example.springdatajpahandson.service;  
  
import org.example.springdatajpahandson.model.Country;  
import org.example.springdatajpahandson.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 saveCountry(Country country) {  
 return countryRepository.save(country);  
 }  
  
 public List<Country> getAllCountries() {  
 return countryRepository.findAll();  
 }  
  
 public Optional<Country> getCountryById(Long id) {  
 return countryRepository.findById(id);  
 }  
  
 public void deleteCountry(Long id) {  
 countryRepository.deleteById(id);  
 }  
 public Optional<Country> getCountryByCode(String code) {  
 return countryRepository.findByCode(code);  
 }  
}

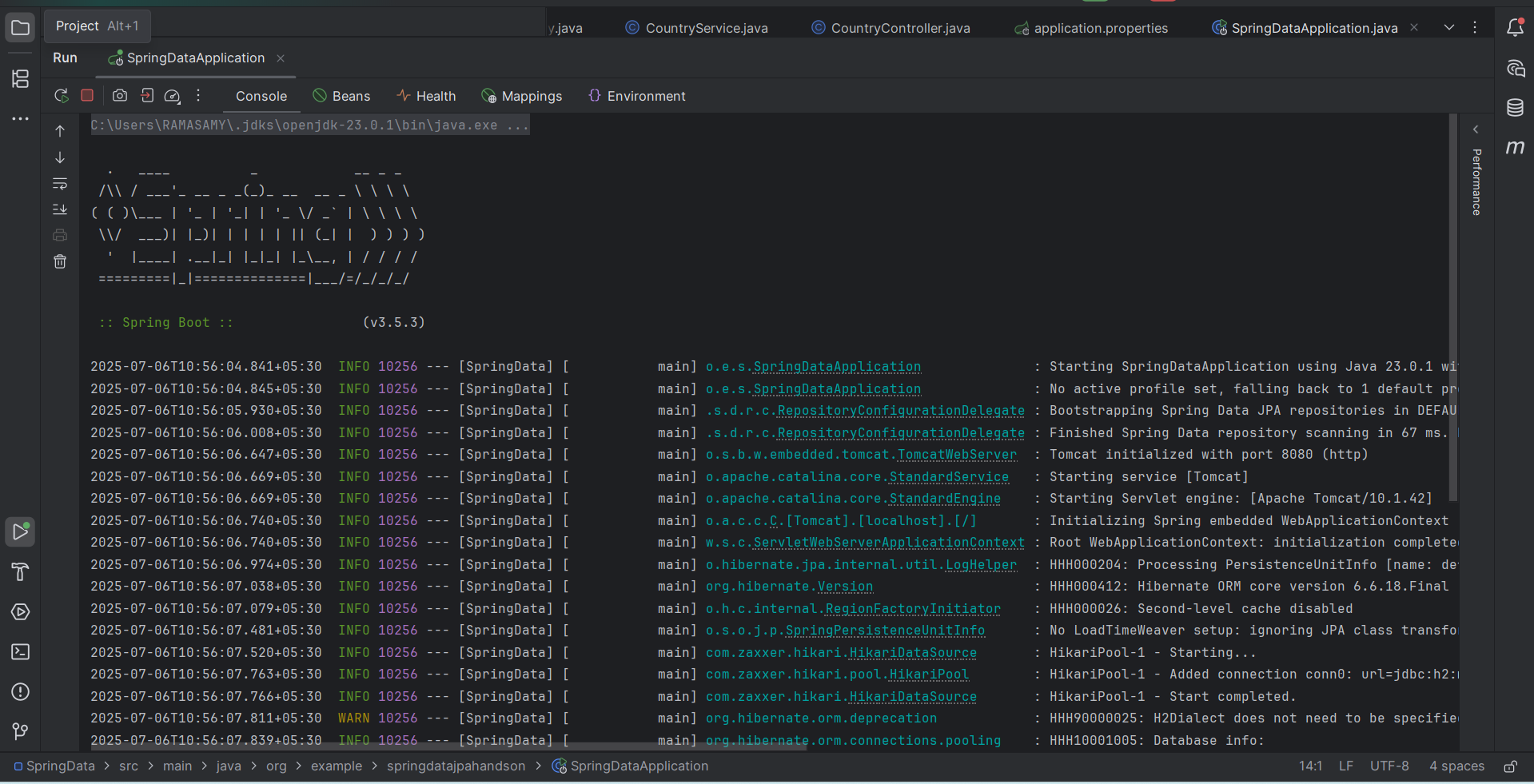
applicaton.properties

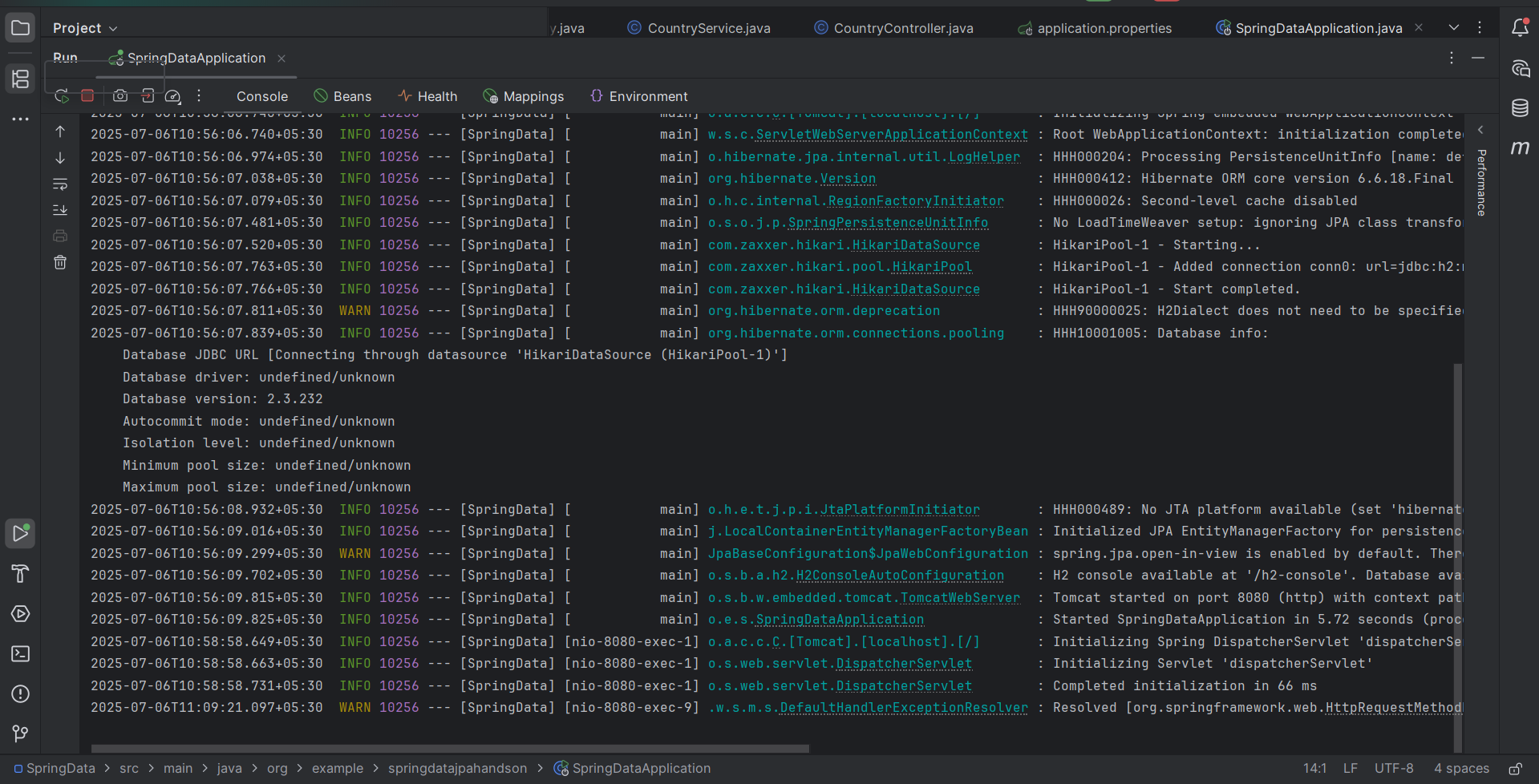
spring.application.name=SpringData  
spring.datasource.url=jdbc:h2:mem:testdb  
spring.datasource.driverClassName=org.h2.Driver  
spring.datasource.username=sa  
spring.datasource.password=  
spring.jpa.hibernate.ddl-auto=update  
spring.jpa.database-platform=org.hibernate.dialect.H2Dialect  
spring.h2.console.enabled=true

SpringDataApplication.java

package org.example.springdatajpahandson;  
  
import org.springframework.boot.SpringApplication;  
import org.springframework.boot.autoconfigure.SpringBootApplication;  
  
@SpringBootApplication  
public class SpringDataApplication {  
  
 public static void main(String[] args) {  
 SpringApplication.*run*(SpringDataApplication.class, args);  
 }  
  
}

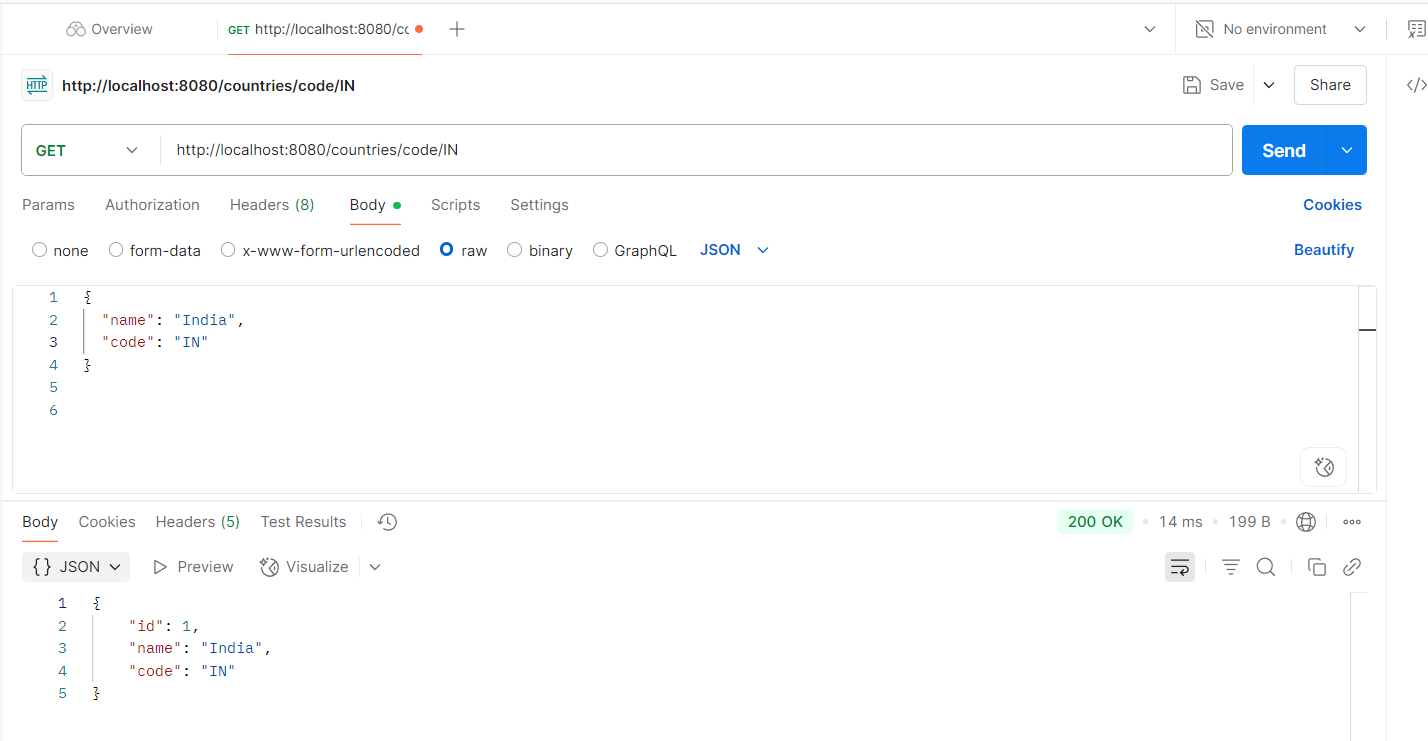
**OUTPUT**

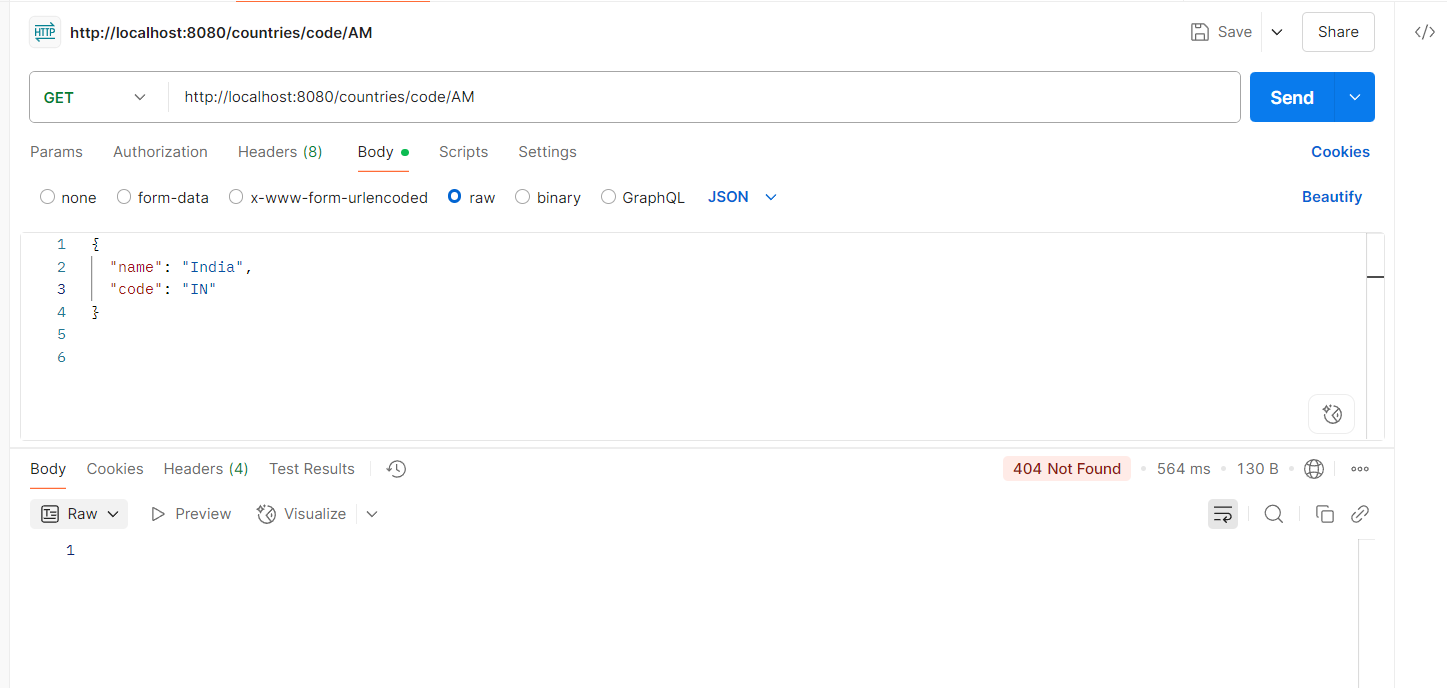
****

****

**POSTMAN:**

**GET**

****

****