**Hands on 1**

**Spring Data JPA - Quick Example  
  
Queries:**

SHOW DATABASES;

CREATE TABLE country (

code VARCHAR(2) PRIMARY KEY,

name VARCHAR(50)

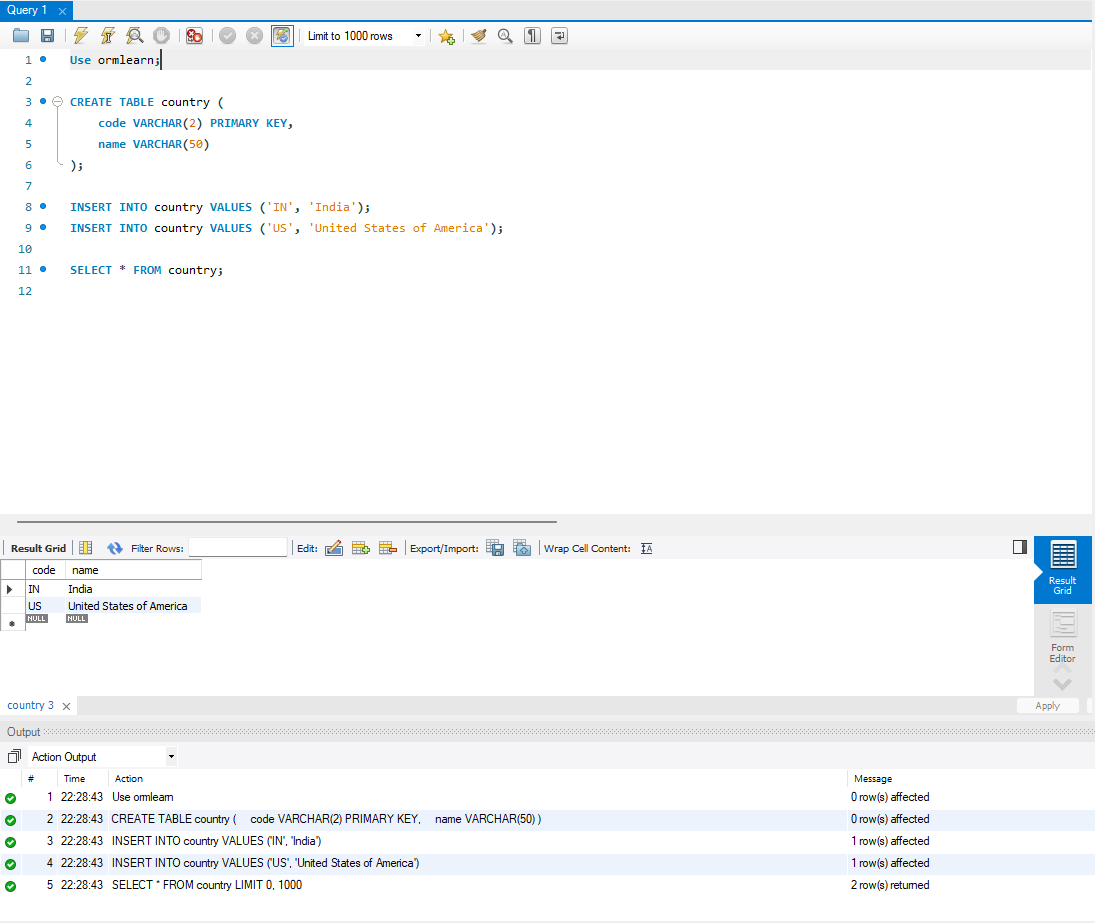
);

INSERT INTO country VALUES ('IN', 'India');

INSERT INTO country VALUES ('US', 'United States of America');

SELECT \* FROM country;

**Output:**



**Code:**

**application.properties**

# Logging configuration  
logging.level.org.springframework=info  
logging.level.com.cognizant=debug  
logging.level.org.hibernate.SQL=trace  
logging.level.org.hibernate.type.descriptor.sql=trace  
  
logging.pattern.console=%d{dd-MM-yy} %d{HH:mm:ss.SSS} %-20.20thread %5p %-25.25logger{25} %25M %4L %m%n  
  
  
# Hibernate  
spring.jpa.hibernate.ddl-auto=validate  
spring.jpa.properties.hibernate.dialect=org.hibernate.dialect.MySQLDialect  
spring.application.name=orm-learn  
spring.datasource.driver-class-name= com.mysql.cj.jdbc.Driver  
spring.datasource.url=jdbc:mysql://localhost:3306/ormlearn  
spring.datasource.username=root  
[spring.datasource.password=SUy@$h2612](mailto:spring.datasource.password=SUy@$h2612)

**Country.java**

package com.cognizant.orm\_learn.model;  
  
import jakarta.persistence.Column;  
import jakarta.persistence.Entity;  
import jakarta.persistence.Id;  
import jakarta.persistence.Table;  
  
@Entity  
@Table(name = "country")  
public class Country {  
  
 @Id  
 @Column(name = "code")  
 private String code;  
  
 @Column(name = "name")  
 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;  
 }  
  
 @Override  
 public String toString() {  
 return "Country [code=" + code + ", name=" + name + "]";  
 }  
}

**CountryRepository.java**

package com.cognizant.orm\_learn.repository;  
  
import org.springframework.data.jpa.repository.JpaRepository;  
import org.springframework.stereotype.Repository;  
import com.cognizant.orm\_learn.model.Country;  
  
@Repository  
public interface CountryRepository extends JpaRepository<Country, String> {  
  
}

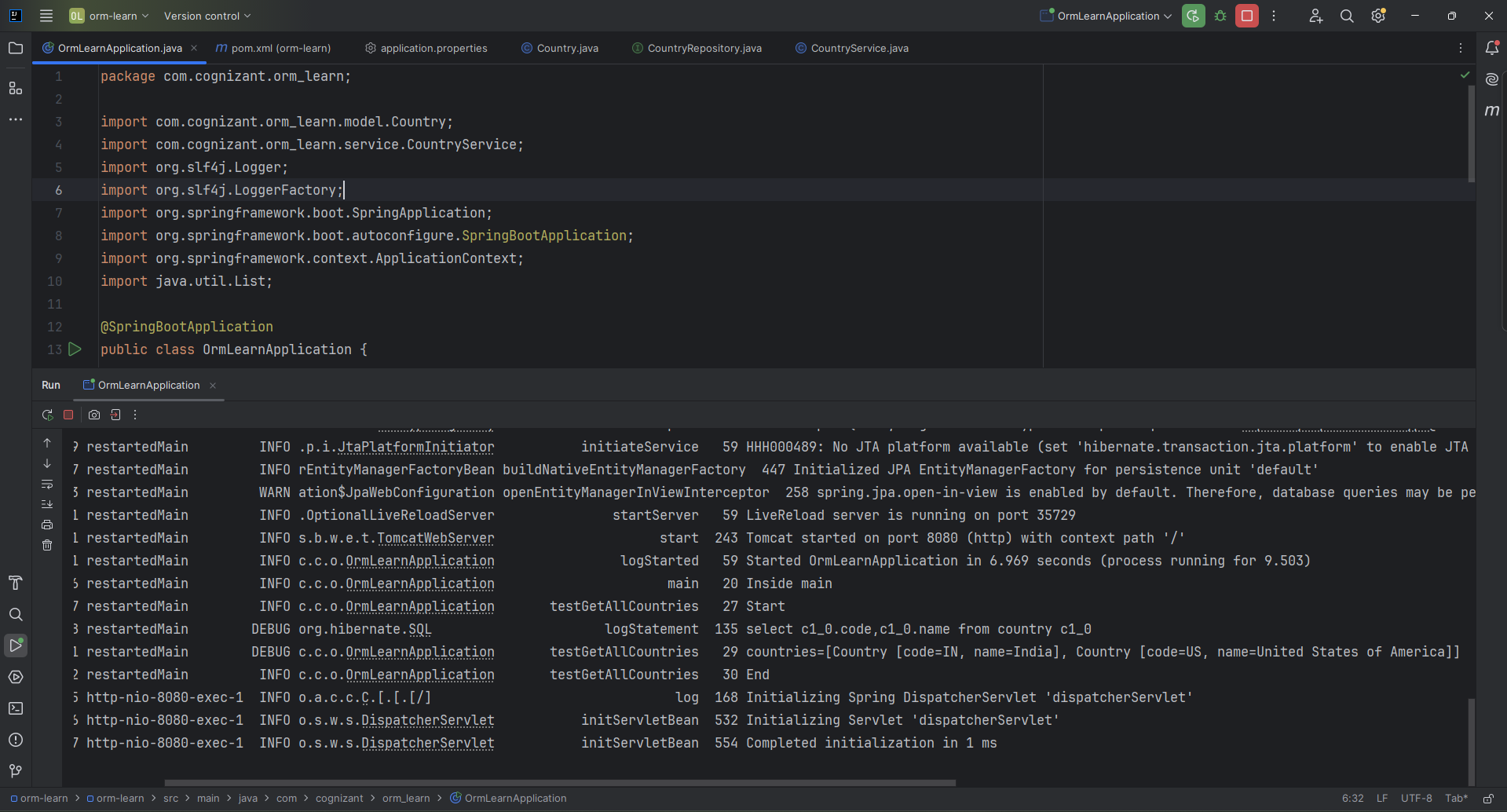
**CountryService.java**

package com.cognizant.orm\_learn.service;  
  
import java.util.List;  
import org.springframework.beans.factory.annotation.Autowired;  
import org.springframework.stereotype.Service;  
import org.springframework.transaction.annotation.Transactional;  
import com.cognizant.orm\_learn.model.Country;  
import com.cognizant.orm\_learn.repository.CountryRepository;  
  
@Service  
public class CountryService {  
  
 @Autowired  
 private CountryRepository countryRepository;  
  
 @Transactional  
 public List<Country> getAllCountries() {  
 return countryRepository.findAll();  
 }  
}

**OrmLearnApplication.java**

package com.cognizant.orm\_learn;  
  
import com.cognizant.orm\_learn.model.Country;  
import com.cognizant.orm\_learn.service.CountryService;  
import org.slf4j.Logger;  
import org.slf4j.LoggerFactory;  
import org.springframework.boot.SpringApplication;  
import org.springframework.boot.autoconfigure.SpringBootApplication;  
import org.springframework.context.ApplicationContext;  
import java.util.List;  
  
@SpringBootApplication  
public class OrmLearnApplication {  
  
 private static final Logger *LOGGER* = LoggerFactory.*getLogger*(OrmLearnApplication.class);  
 private static CountryService *countryService*;  
  
 public static void main(String[] args) {  
 ApplicationContext context = SpringApplication.*run*(OrmLearnApplication.class, args);  
 *LOGGER*.info("Inside main");  
 *countryService* = context.getBean(CountryService.class);  
  
 *testGetAllCountries*();  
 }  
  
 private static void testGetAllCountries() {  
 *LOGGER*.info("Start");  
 List<Country> countries = *countryService*.getAllCountries();  
 *LOGGER*.debug("countries={}", countries);  
 *LOGGER*.info("End");  
 }  
}

**Output:**



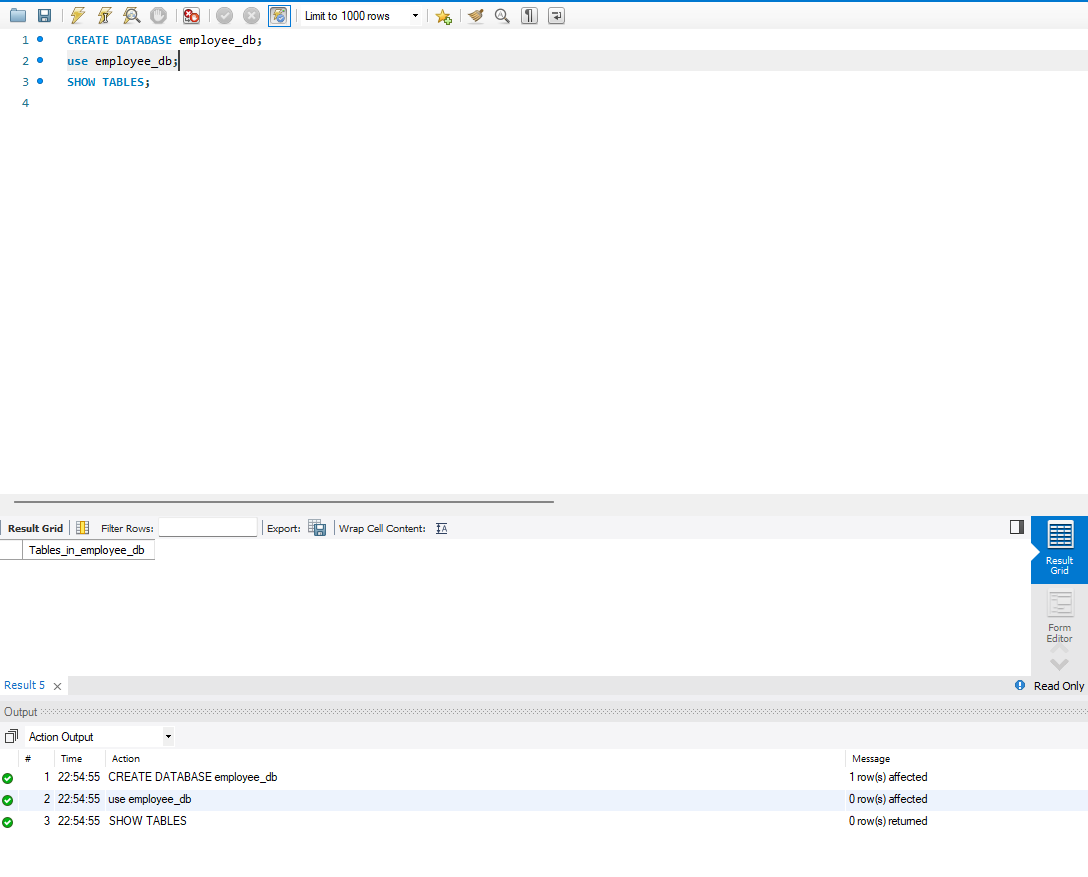
**Hands on 4**

**Difference between JPA, Hibernate and Spring Data JPA  
  
Queries:**

CREATE DATABASE employee\_db;

SHOW TABLES;

**Output:**



**Code:**

**application.properties**

spring.application.name=employeeapp  
  
# Database Configuration  
spring.datasource.url=jdbc:mysql://localhost:3306/employee\_db  
spring.datasource.username=root  
spring.datasource.password=SUy@$h2612  
  
# JPA / Hibernate Settings  
spring.jpa.hibernate.ddl-auto=update  
spring.jpa.show-sql=true  
spring.jpa.properties.hibernate.dialect=org.hibernate.dialect.MySQLDialect  
  
# Optional but clean logging  
logging.level.org.hibernate.SQL=DEBUG  
logging.level.org.hibernate.type.descriptor.sql.BasicBinder=TRACE

**Employee.java**

package com.example.employeeapp.model;  
  
import jakarta.persistence.\*;  
  
  
@Entity  
@Table(name = "employee")  
public class Employee {  
  
 @Id  
 @GeneratedValue(strategy = GenerationType.*IDENTITY*)  
 private Integer id;  
  
 @Column(nullable = false)  
 private String name;  
  
 private String department;  
  
 // Getters and setters  
 public Integer getId() {  
 return id;  
 }  
  
 public void setId(Integer id) {  
 this.id = id;  
 }  
  
 public String getName() {  
 return name;  
 }  
  
 public void setName(String name) {  
 this.name = name;  
 }  
  
 public String getDepartment() {  
 return department;  
 }  
  
 public void setDepartment(String department) {  
 this.department = department;  
 }  
  
 // toString  
 @Override  
 public String toString() {  
 return "Employee [id=" + id + ", name=" + name + ", department=" + department + "]";  
 }  
}

**EmployeeRepository.java**

package com.example.employeeapp.repository;  
  
import com.example.employeeapp.model.Employee;  
import org.springframework.data.jpa.repository.JpaRepository;  
import org.springframework.stereotype.Repository;  
  
@Repository  
public interface EmployeeRepository extends JpaRepository<Employee, Integer> {  
  
}

**EmployeeService.java**

package com.example.employeeapp.service;  
  
import com.example.employeeapp.model.Employee;  
import com.example.employeeapp.repository.EmployeeRepository;  
import org.springframework.beans.factory.annotation.Autowired;  
import org.springframework.stereotype.Service;  
  
import jakarta.transaction.Transactional;  
import java.util.List;  
  
@Service  
public class EmployeeService {  
  
 @Autowired  
 private EmployeeRepository employeeRepository;  
  
 @Transactional  
 public void addEmployee(Employee employee) {  
 employeeRepository.save(employee);  
 }  
  
 public List<Employee> getAllEmployees() {  
 return employeeRepository.findAll();  
 }  
}

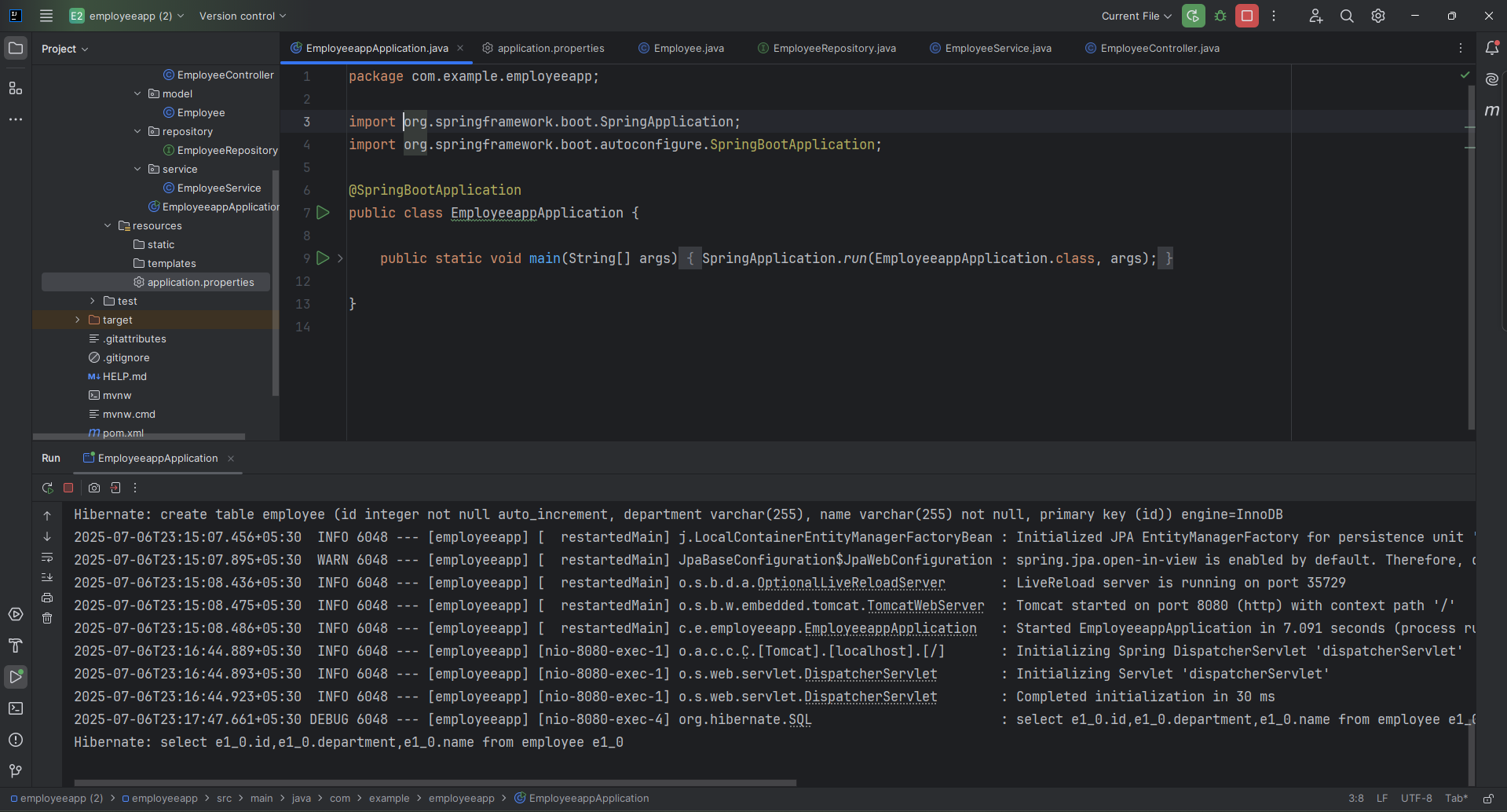
**EmployeeController.java**

package com.example.employeeapp.controller;  
  
import com.example.employeeapp.model.Employee;  
import com.example.employeeapp.service.EmployeeService;  
import org.springframework.beans.factory.annotation.Autowired;  
import org.springframework.web.bind.annotation.\*;  
  
import java.util.List;  
  
@RestController  
@RequestMapping("/employees")  
public class EmployeeController {  
  
 @Autowired  
 private EmployeeService employeeService;  
  
 // Add an employee  
 @PostMapping  
 public String addEmployee(@RequestBody Employee employee) {  
 employeeService.addEmployee(employee);  
 return "Employee added successfully!";  
 }  
  
 // Get all employees  
 @GetMapping  
 public List<Employee> getAllEmployees() {  
 return employeeService.getAllEmployees();  
 }  
}

**EmployeeappApplication.java**

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

**Output:**



**Table Created:**

