**WEEK-03**

**MODULE-06:SPRING DATA JPA WITH**

**SPRING BOOT,HIBERNATE**

**MANDATORY HANDSON**

**EXERCISE-01:Spring Data JPA - Quick Example**

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

# Database configuration

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=Niran\*\*\*\*@03

# Hibernate configuration

spring.jpa.hibernate.ddl-auto=validate

spring.jpa.properties.hibernate.dialect=org.hibernate.dialect.MySQLDialect

OrmLearnApplication.java

**package** com.cognizant;

**import** java.util.List;

**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** com.cognizant.orm\_learn.model.Country;

**import** com.cognizant.orm\_learn.service.CountryService;

@SpringBootApplication

**public** **class** OrmLearnApplication {

**private** **static** CountryService *countryService*;

**private** **static** **final** Logger ***LOGGER*** = LoggerFactory.*getLogger*(OrmLearnApplication.**class**);

**public** **static** **void** main(String[] args) {

ApplicationContext context = SpringApplication.*run*(OrmLearnApplication.**class**, args);

*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");

}

}

MySQL

Create ormlearn;

use ormlearn;

create table country(co\_code varchar(2) primary key, co\_name varchar(50));

insert into country values ('IN', 'India');

insert into country values ('US', 'United States of America');

Country.java

**package** com.cognizant.orm\_learn.model;

**import** jakarta.persistence.\*;

@Entity

@Table(name="country")

**public** **class** Country {

@Id

@Column(name="co\_code")

**private** String code;

@Column(name="co\_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** jakarta.transaction.Transactional;

**import** org.springframework.beans.factory.annotation.Autowired;

**import** org.springframework.stereotype.Service;

**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;

**import** java.util.List;

**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** com.cognizant.orm\_learn.model.Country;

**import** com.cognizant.orm\_learn.service.CountryService;

@SpringBootApplication

**public** **class** OrmLearnApplication {

**private** **static** CountryService *countryService*;

**private** **static** **final** Logger ***LOGGER*** = LoggerFactory.*getLogger*(OrmLearnApplication.**class**);

**public** **static** **void** main(String[] args) {

ApplicationContext context = SpringApplication.*run*(OrmLearnApplication.**class**, args);

*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");

}

}

Pom.xml

</properties>

<dependencies>

<dependency>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-starter-data-jpa</artifactId>

</dependency>

<dependency>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-devtools</artifactId>

<scope>runtime</scope>

<optional>true</optional>

</dependency>

<dependency>

<groupId>mysql</groupId>

<artifactId>mysql-connector-java</artifactId>

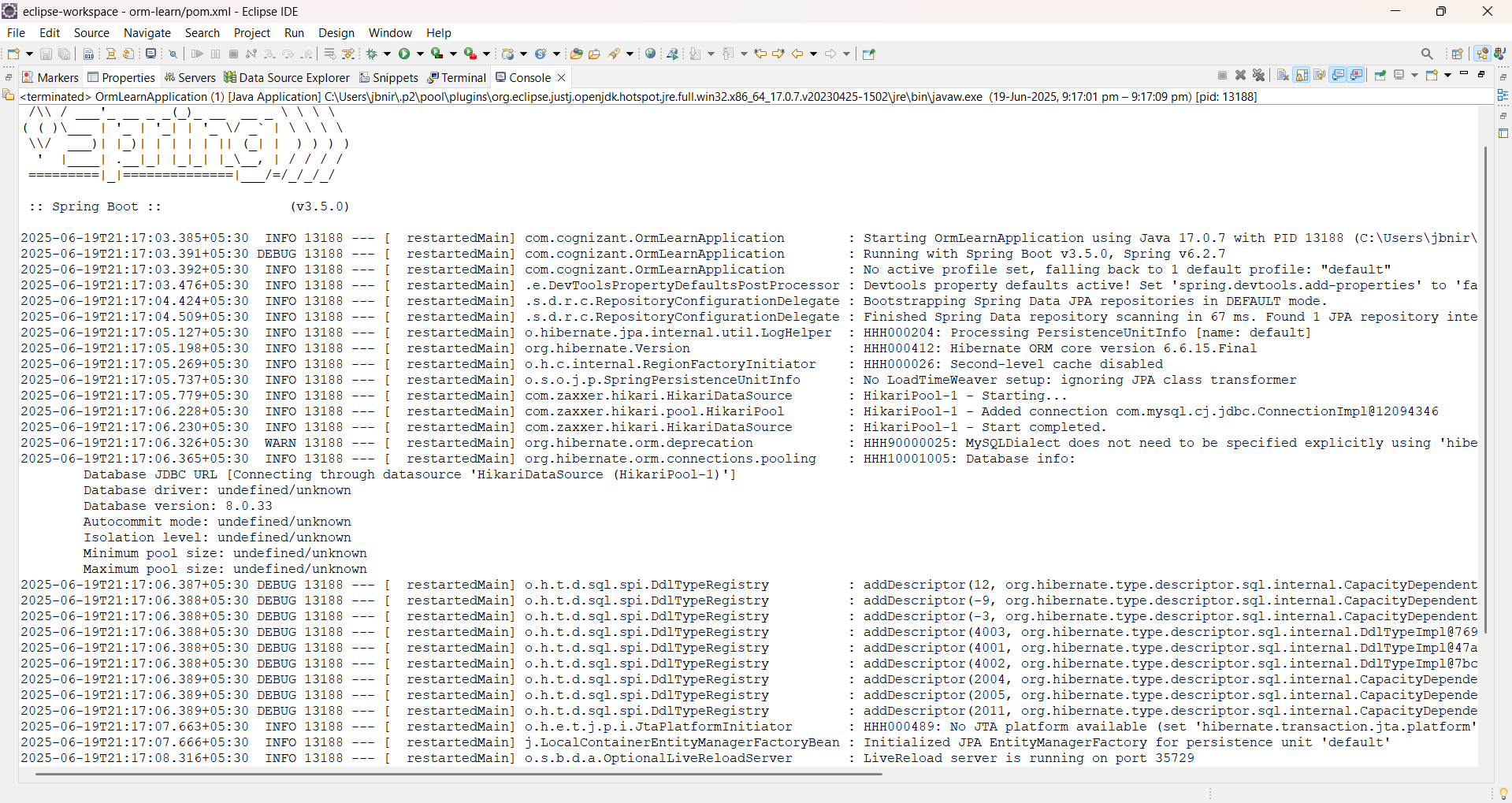
<version>8.0.33</version>

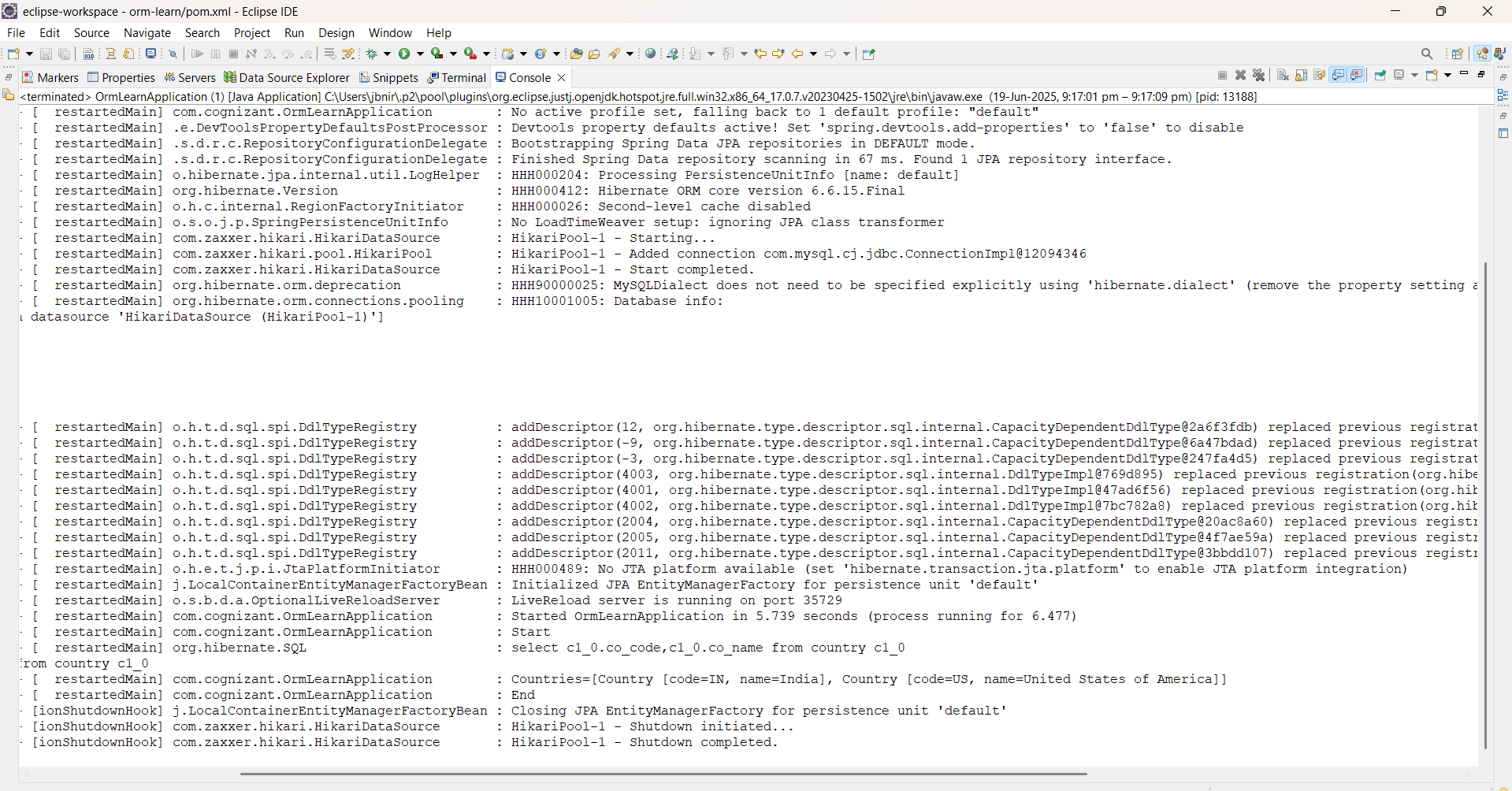
<scope>runtime</scope>

</dependency>

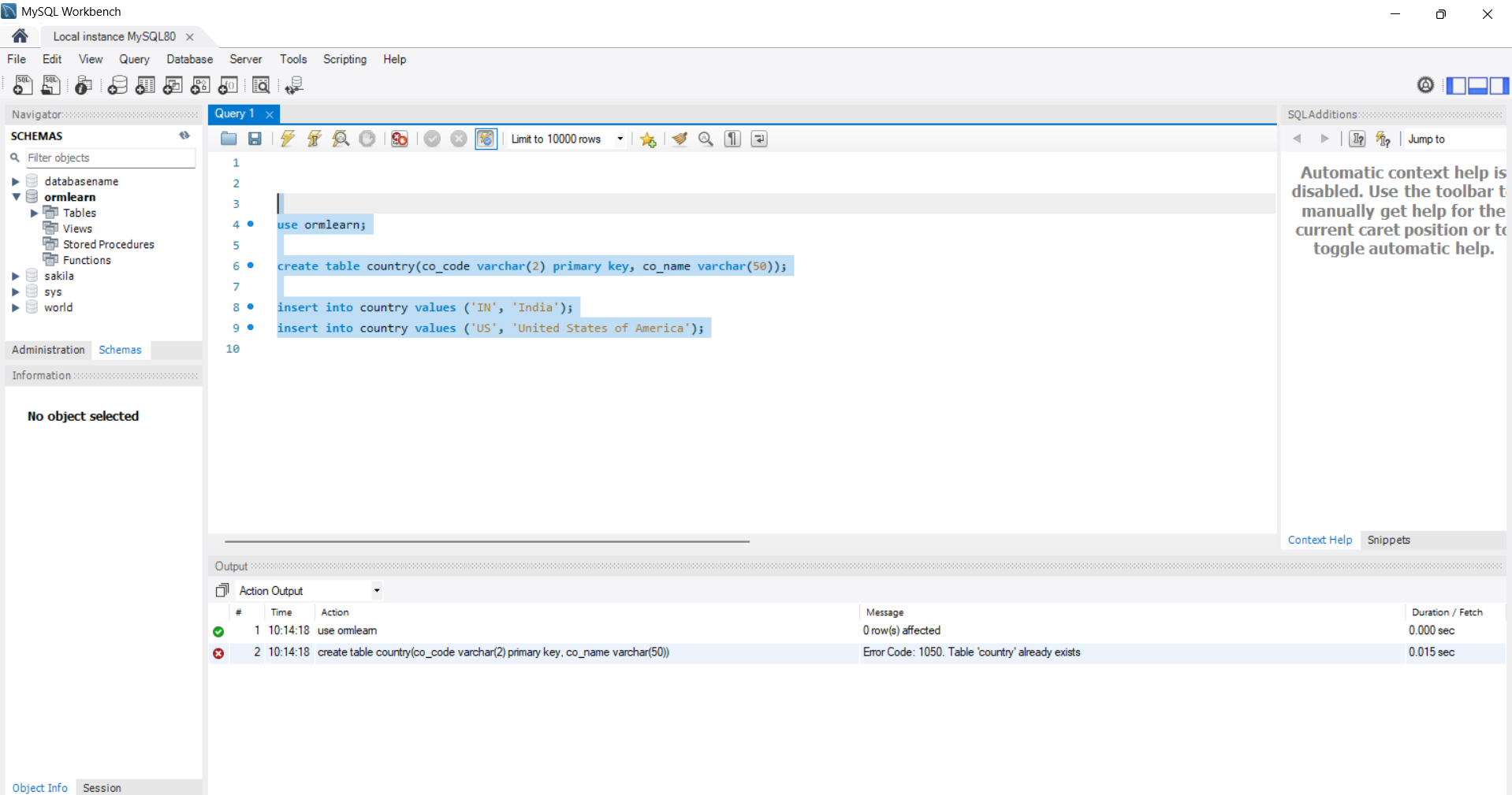
**OUTPUT**

**CONSOLE**

****

****

**MYSQL-CMD**

****

**EXERCISE-04:Difference between JPA, Hibernate and Spring Data JPA**

**Project with JPA and Hibernate**

Pom.xml

<dependencies>

<!-- Hibernate Core -->

<dependency>

<groupId>org.hibernate</groupId>

<artifactId>hibernate-core</artifactId>

<version>5.6.15.Final</version>

</dependency>

<dependency>

<groupId>org.slf4j</groupId>

<artifactId>slf4j-api</artifactId>

<version>1.7.36</version>

</dependency>

<dependency>

<groupId>org.slf4j</groupId>

<artifactId>slf4j-simple</artifactId>

<version>1.7.36</version>

</dependency>

<!-- MySQL Connector -->

<dependency>

<groupId>mysql</groupId>

<artifactId>mysql-connector-java</artifactId>

<version>8.0.32</version>

</dependency>

<!-- JUnit (Optional, for testing) -->

<dependency>

<groupId>junit</groupId>

<artifactId>junit</artifactId>

<version>4.13.2</version>

<scope>test</scope>

</dependency>

<!-- Logging Dependency -->

<dependency>

<groupId>org.slf4j</groupId>

<artifactId>slf4j-api</artifactId>

<version>1.7.36</version>

</dependency>

<dependency>

<groupId>org.slf4j</groupId>

<artifactId>slf4j-simple</artifactId>

<version>1.7.36</version>

</dependency>

</dependencies>

Hibernate.cfg.xml

<?xml version='1.0' encoding='utf-8'?>

<!DOCTYPE hibernate-configuration PUBLIC

"-//Hibernate/Hibernate Configuration DTD 3.0//EN"

"http://www.hibernate.org/dtd/hibernate-configuration-3.0.dtd">

<hibernate-configuration>

<session-factory>

<!-- Database connection settings -->

<property name="hibernate.connection.driver\_class">com.mysql.cj.jdbc.Driver</property>

<property name="hibernate.connection.url">jdbc:mysql://localhost:3306/employee\_db</property>

<property name="hibernate.connection.username">root</property>

<property name="hibernate.connection.password">Niranjana@03</property>

<!-- SQL Dialect -->

<property name="hibernate.dialect">org.hibernate.dialect.MySQLDialect</property>

<!-- Automatically create/update tables -->

<property name="hibernate.hbm2ddl.auto">update</property>

<!-- Show SQL in console -->

<property name="show\_sql">true</property>

<!-- Mapping the entity class -->

<mapping class="com.example.hibernate.entity.Employee"/>

</session-factory>

</hibernate-configuration>

Employee.java

**package** com.example.hibernate.entity;

**import** javax.persistence.\*;

@Entity

**public** **class** Employee {

@Id

@GeneratedValue(strategy=GenerationType.***IDENTITY***)

**private** **int** id;

**private** String name;

**private** String department;

**public** Employee() {}

**public** Employee(String name, String department) {

**this**.name = name;

**this**.department = department;

}

**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 getDepartment() { **return** department; }

**public** **void** setDepartment(String department) { **this**.department = department; }

}

HibernateUtil.java

**package** com.example.hibernate.util;

**import** org.hibernate.SessionFactory;

**import** org.hibernate.cfg.Configuration;

**public** **class** HibernateUtil {

**private** **static** **final** SessionFactory ***sessionFactory***;

**static** {

**try** {

***sessionFactory*** = **new** Configuration().configure().buildSessionFactory();

} **catch** (Throwable ex) {

**throw** **new** ExceptionInInitializerError(ex);

}

}

**public** **static** SessionFactory getSessionFactory() {

**return** ***sessionFactory***;

}

}

EmployeeDao.java

**package** com.example.hibernate.dao;

**import** com.example.hibernate.entity.Employee;

**import** com.example.hibernate.util.HibernateUtil;

**import** org.hibernate.Session;

**import** org.hibernate.Transaction;

**public** **class** EmployeeDao {

**public** Integer addEmployee(Employee employee) {

Session session = HibernateUtil.*getSessionFactory*().openSession();

Transaction tx = **null**;

Integer employeeID = **null**;

**try** {

tx = session.beginTransaction();

employeeID = (Integer) session.save(employee);

tx.commit();

} **catch** (Exception e) {

**if** (tx != **null**) tx.rollback();

e.printStackTrace();

} **finally** {

session.close();

}

**return** employeeID;

}

}

MainApp.java

**package** com.example.hibernate.main;

**import** com.example.hibernate.dao.EmployeeDao;

**import** com.example.hibernate.entity.Employee;

**public** **class** MainApp {

**public** **static** **void** main(String[] args) {

EmployeeDao dao = **new** EmployeeDao();

Employee emp = **new** Employee("John", "IT");

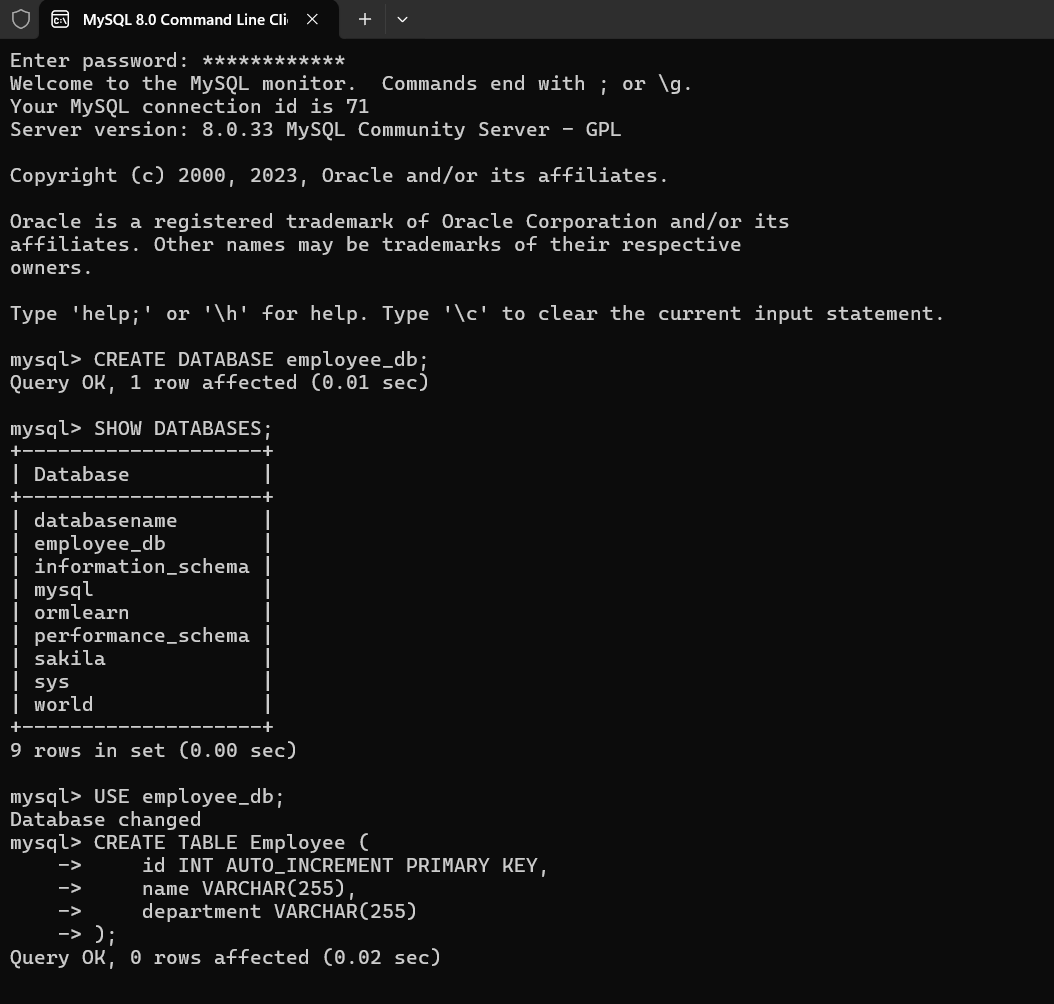
Integer id = dao.addEmployee(emp);

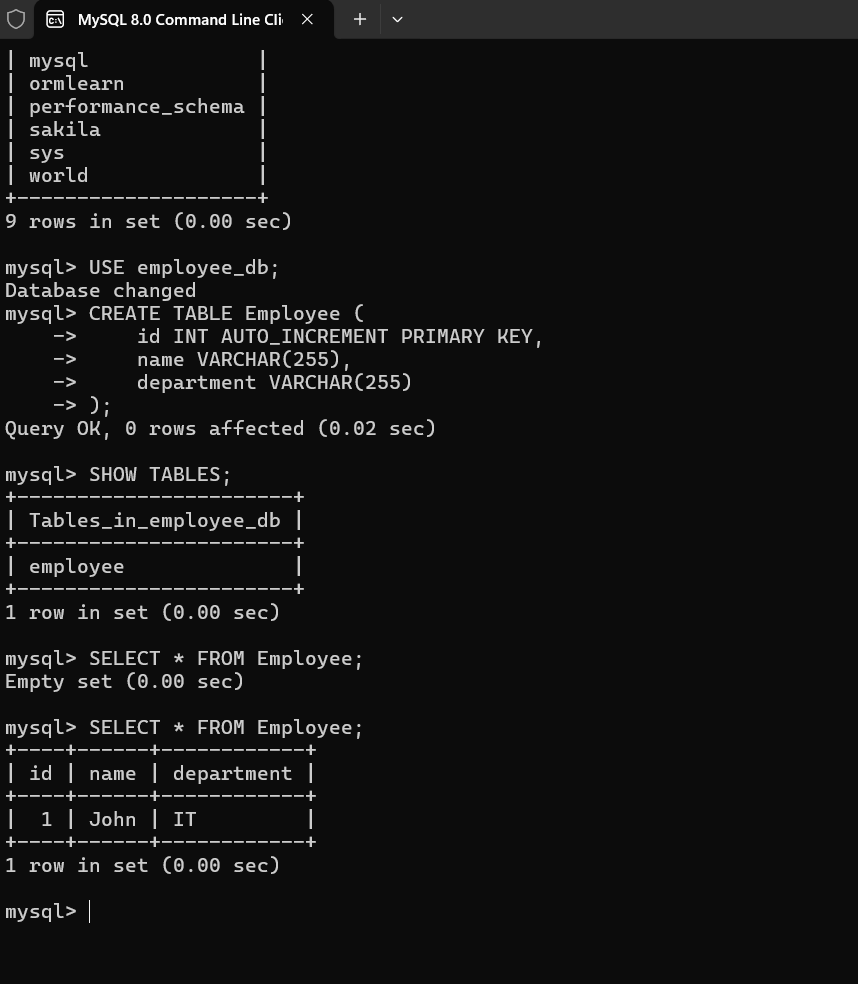
System.***out***.println("Employee saved with ID: " + id);

}

}

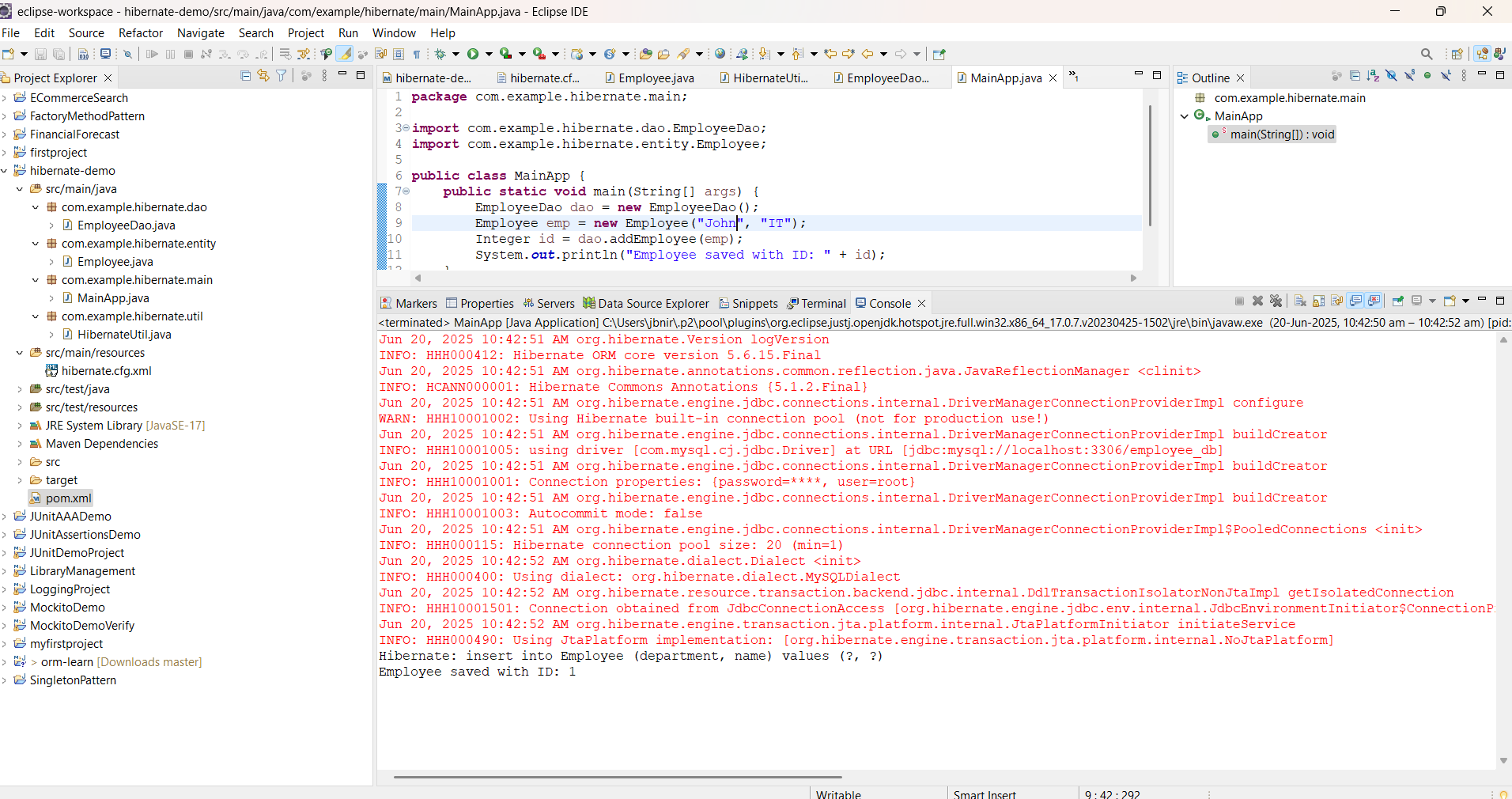
MySQL





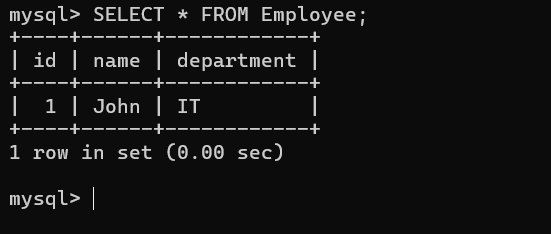
**OUTPUT**

**CONSOLE**

****



**MySQL CMD LINE**

****

**Project with Spring data JPA**

pom.xml

<dependencies>

<dependency>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-starter-data-jpa</artifactId>

</dependency>

<dependency>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-starter-web</artifactId>

</dependency>

<dependency>

<groupId>mysql</groupId>

<artifactId>mysql-connector-java</artifactId>

<version>8.0.33</version>

</dependency>

</dependencies>

SpringjpaApplication.java

package com.example.springjpa;

import org.springframework.boot.SpringApplication;

import org.springframework.boot.autoconfigure.SpringBootApplication;

@SpringBootApplication

public class SpringjpaApplication {

public static void main(String[] args) {

SpringApplication.run(SpringjpaApplication.class, args); } }

Employee.java

package com.example.springjpa.entity;

import javax.persistence.Entity;

import javax.persistence.GeneratedValue;

import javax.persistence.GenerationType;

import javax.persistence.Id;

@Entity

public class Employee {

@Id

@GeneratedValue(strategy = GenerationType.IDENTITY)

private Long id;

private String name;

private String department;

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 getDepartment() {

return department;

}

public void setDepartment(String department) {

this.department = department;

}

}

EmployeeRepository.java

package com.example.springjpa.repository;

import com.example.springjpa.entity.Employee;

import org.springframework.data.jpa.repository.JpaRepository;

public interface EmployeeRepository extends JpaRepository<Employee, Long> {

}

EmployeeController.java

package com.example.springjpa.controller;

import com.example.springjpa.entity.Employee;

import com.example.springjpa.repository.EmployeeRepository;

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 EmployeeRepository employeeRepository;

@PostMapping

public Employee addEmployee(@RequestBody Employee employee) {

return employeeRepository.save(employee);

}

@GetMapping

public List<Employee> getAllEmployees() {

return employeeRepository.findAll();

}

}

application.properties

spring.datasource.url=jdbc:mysql://localhost:3306/employee\_db

spring.datasource.username=root

spring.datasource.password=your\_password

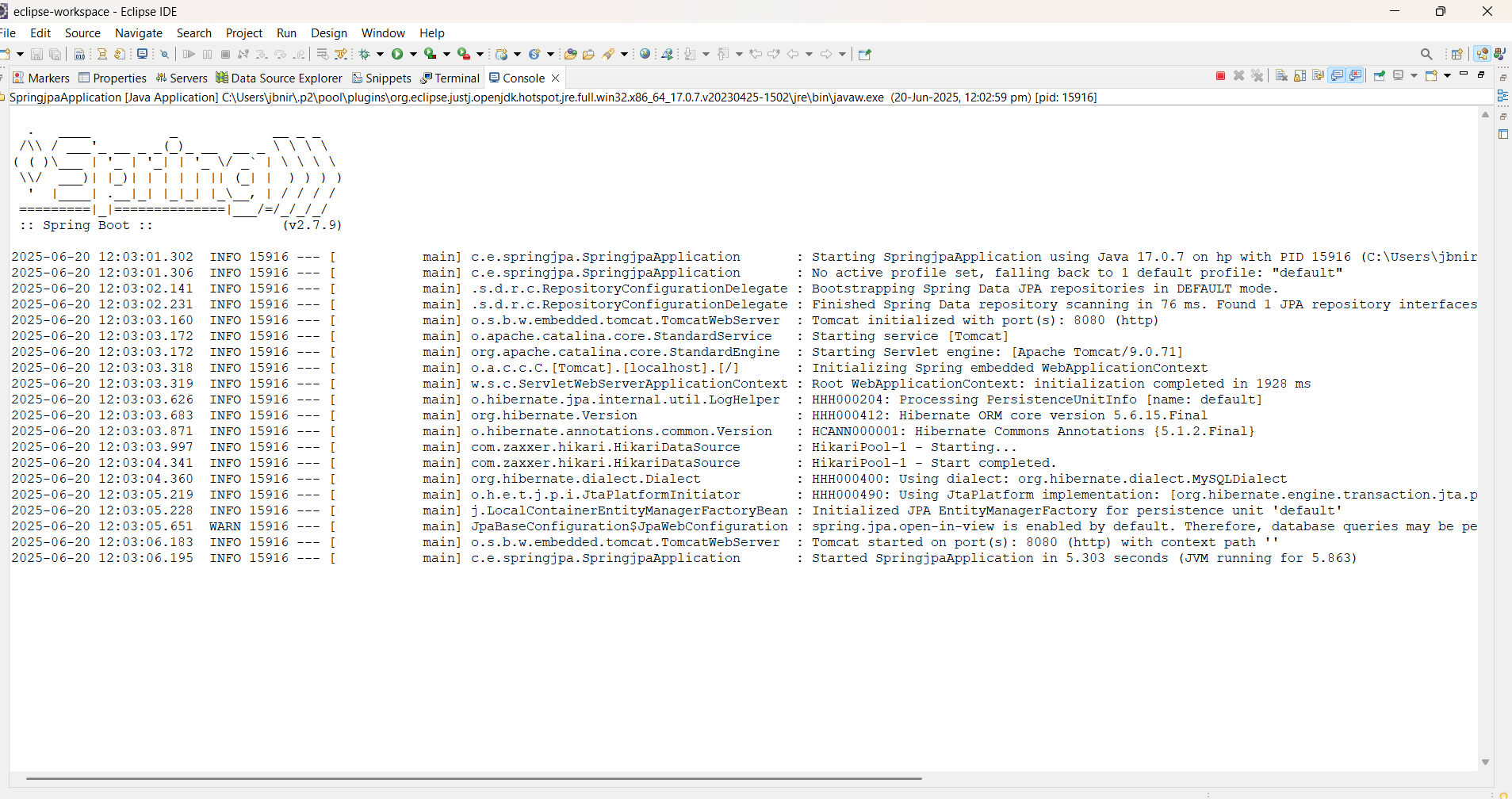
spring.jpa.hibernate.ddl-auto=update

spring.jpa.show-sql=true

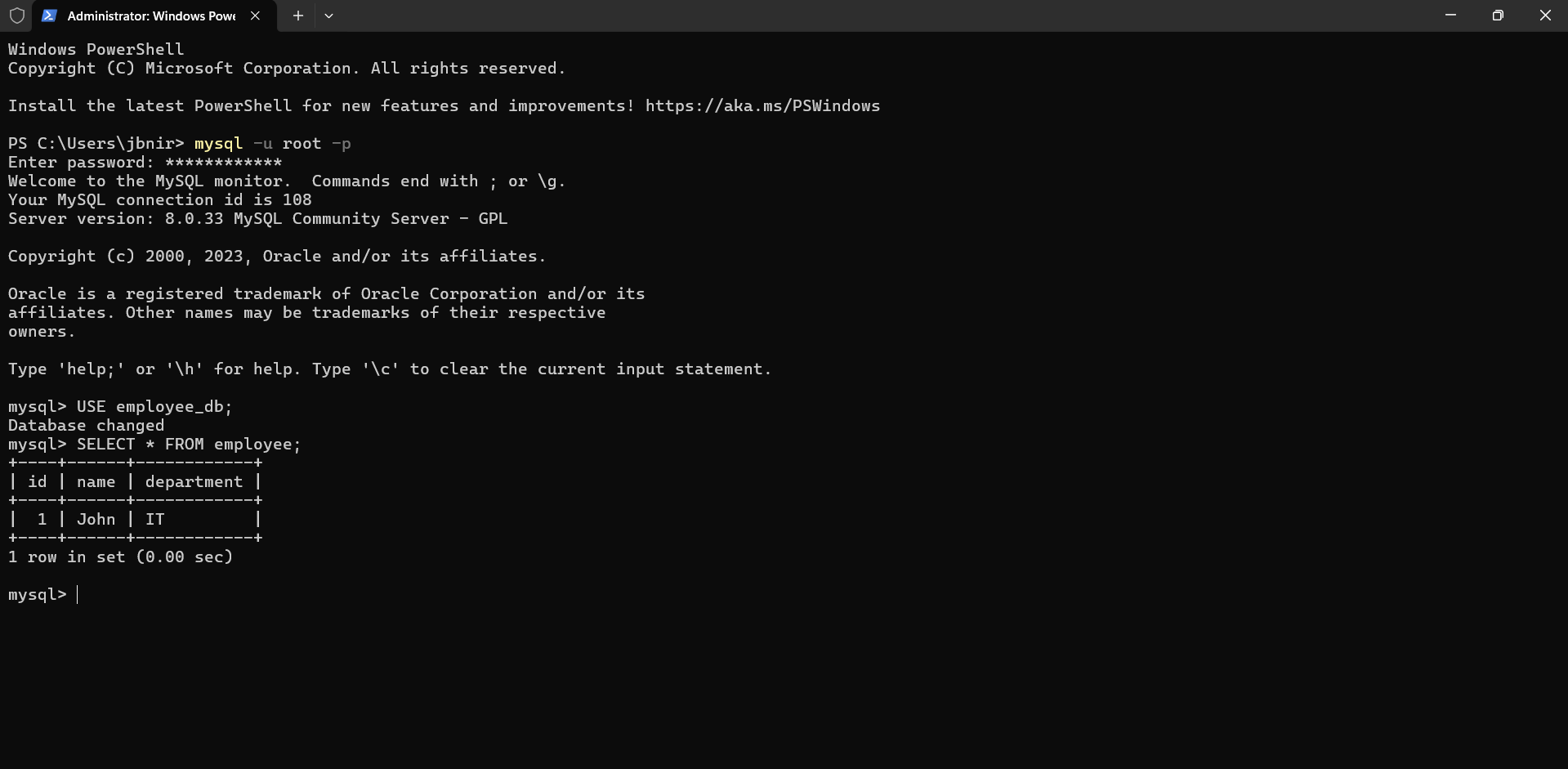
spring.jpa.database-platform=org.hibernate.dialect.MySQLDialect

**OUTPUT**

**CONSOLE**

****

**MySQL CMD LINE:**

****