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

**Employee.java**

package com.cognizant;

import jakarta.persistence.\*;

@Entity

@Table(name = "employees")

public class Employee {

@Id

@GeneratedValue(strategy = GenerationType.IDENTITY)

private Integer id;

private String firstName;

private String lastName;

private String email;

public Employee() {}

public Employee(String first, String last, String email) {

this.firstName = first; this.lastName = last; this.email = email;

}

public Integer getId() {

return id;

}

public void setId(Integer id) {

this.id = id;

}

public String getFirstName() {

return firstName;

}

public void setFirstName(String firstName) {

this.firstName = firstName;

}

public String getLastName() {

return lastName;

}

public void setLastName(String lastName) {

this.lastName = lastName;

}

public String getEmail() {

return email;

}

public void setEmail(String email) {

this.email = email;

}

}

# **HibernateDemoRunner.java**

package com.cognizant;

import org.hibernate.Session;

import org.hibernate.SessionFactory;

import org.hibernate.Transaction;

import org.hibernate.cfg.Configuration;

public class HibernateDemoRunner {

private static final SessionFactory sessionFactory =

new Configuration().configure() // loads hibernate.cfg.xml

.buildSessionFactory();

public static void main(String[] args) {

Integer id = createEmployee("Alice", "Doe", "alice@example.com");

Employee e = getEmployee(id);

updateEmail(id, "alice@updated.com");

deleteEmployee(id);

sessionFactory.close(); // clean shutdown

}

private static Integer createEmployee(String fn, String ln, String email) {

Session s = sessionFactory.openSession();

Transaction tx = null; Integer generatedId = null;

try {

tx = s.beginTransaction();

generatedId = (Integer) s.save(new Employee(fn, ln, email));

tx.commit();

} catch (Exception ex) { if (tx!=null) tx.rollback(); }

finally { s.close(); }

return generatedId;

}

private static Employee getEmployee(Integer id) {

try (Session s = sessionFactory.openSession()) {

return s.get(Employee.class, id);

}

}

private static void updateEmail(Integer id, String newEmail) {

Transaction tx = null;

try (Session s = sessionFactory.openSession()) {

tx = s.beginTransaction();

Employee e = s.get(Employee.class, id);

e.setEmail(newEmail);

s.update(e);

tx.commit();

} catch (Exception ex) { if (tx!=null) tx.rollback(); }

}

private static void deleteEmployee(Integer id) {

Transaction tx = null;

try (Session s = sessionFactory.openSession()) {

tx = s.beginTransaction();

Employee e = s.get(Employee.class, id);

if (e != null) s.delete(e);

tx.commit();

} catch (Exception ex) { if (tx!=null) tx.rollback(); }

}

}

**hibernate.cfg.xml**

<!DOCTYPE hibernate-configuration PUBLIC

"-//Hibernate/Hibernate Configuration DTD 5.4//EN"

"http://hibernate.org/dtd/hibernate-configuration-5.4.dtd">

<hibernate-configuration>

<session-factory>

<property name="hibernate.connection.driver\_class">org.h2.Driver</property>

<property name="hibernate.connection.url">jdbc:h2:mem:testdb;DB\_CLOSE\_DELAY=-1</property>

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

<property name="hibernate.dialect">org.hibernate.dialect.H2Dialect</property>

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

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

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

<mapping class="com.cognizant.Employee"/>

</session-factory>

</hibernate-configuration>

**logback.xml**

<configuration>

<appender name="STDOUT" class="ch.qos.logback.core.ConsoleAppender">

<encoder>

<pattern>%d{HH:mm:ss.SSS} %-5level %logger - %msg%n</pattern>

</encoder>

</appender>

<root level="INFO">

<appender-ref ref="STDOUT"/>

</root>

<logger name="org.hibernate.SQL" level="DEBUG"/>

<logger name="org.hibernate.type.descriptor.sql" level="TRACE"/>

</configuration>

**Pom.xml**

<?xml version="1.0" encoding="UTF-8"?>

<project xmlns="http://maven.apache.org/POM/4.0.0"

xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"

xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd">

<modelVersion>4.0.0</modelVersion>

<groupId>com.cognizant</groupId>

<artifactId>HibernateDemo</artifactId>

<version>1.0-SNAPSHOT</version>

<dependencies>

<dependency>

<groupId>org.hibernate.orm</groupId>

<artifactId>hibernate-core</artifactId>

<version>6.5.2.Final</version>

</dependency>

<dependency>

<groupId>jakarta.persistence</groupId>

<artifactId>jakarta.persistence-api</artifactId>

<version>3.2.0</version>

</dependency>

<dependency>

<groupId>com.h2database</groupId>

<artifactId>h2</artifactId>

<version>2.3.232</version>

</dependency>

<dependency>

<groupId>ch.qos.logback</groupId>

<artifactId>logback-classic</artifactId>

<version>1.5.18</version>

<scope>compile</scope>

</dependency>

</dependencies>

<properties>

<maven.compiler.source>23</maven.compiler.source>

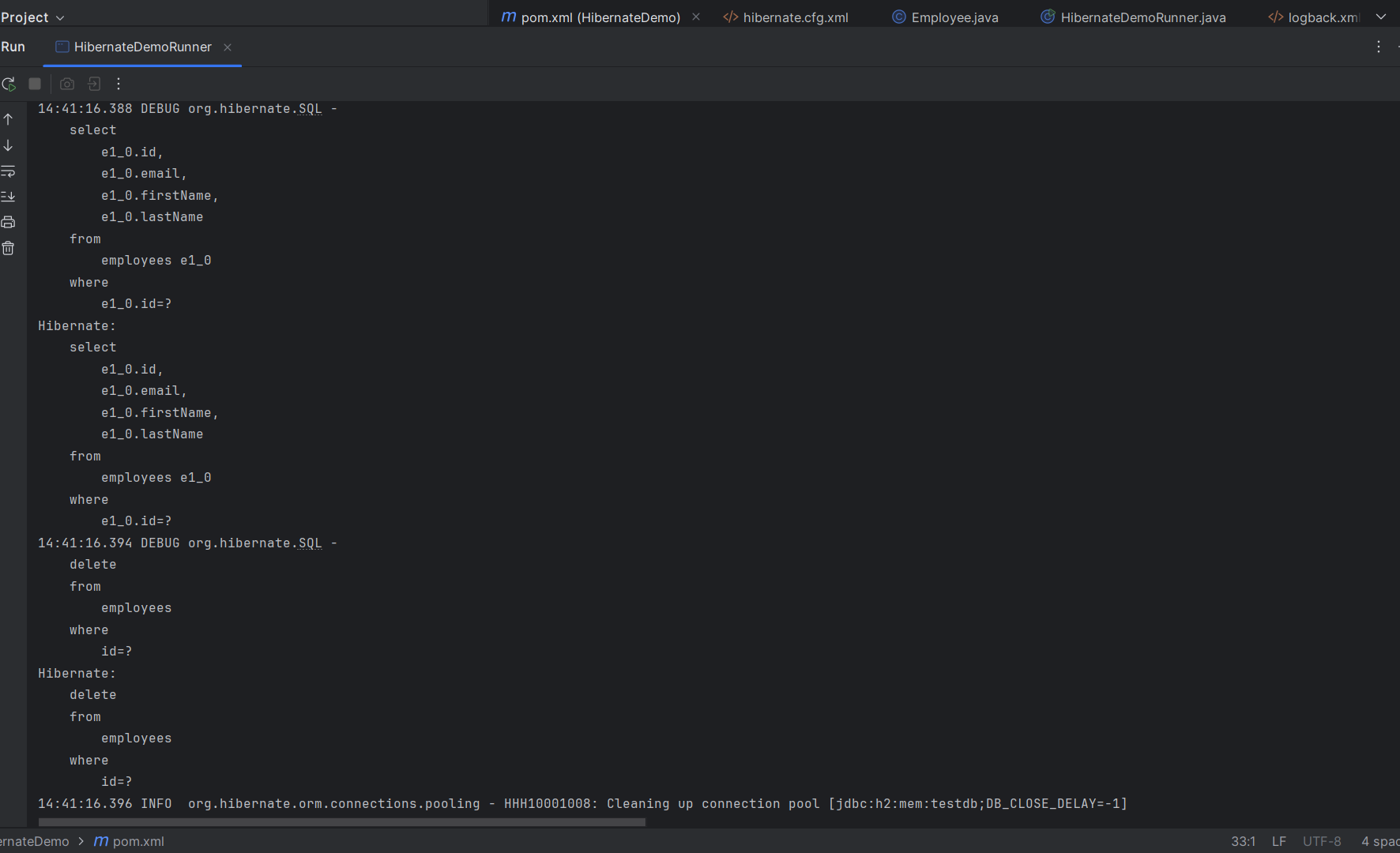
<maven.compiler.target>23</maven.compiler.target>

<project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>

</properties>

</project>

**OUTPUT :**

****