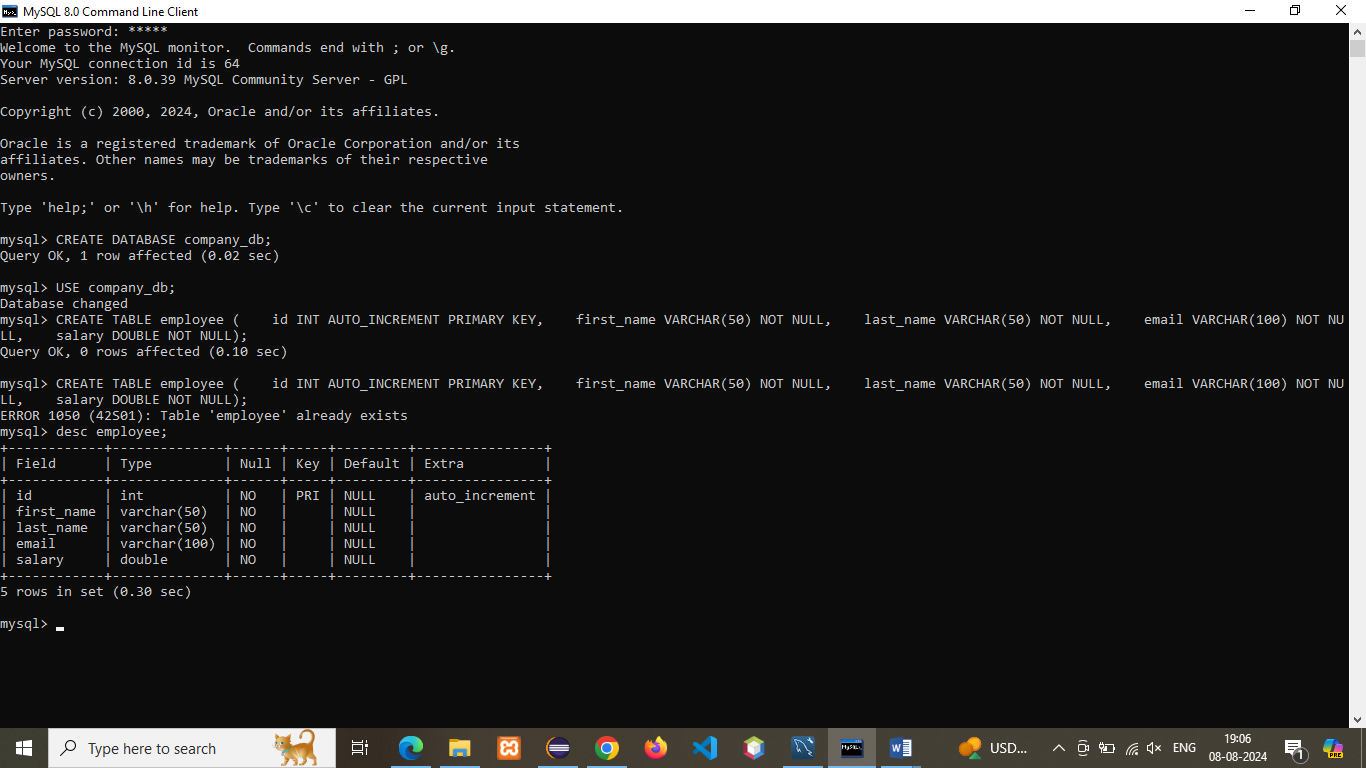
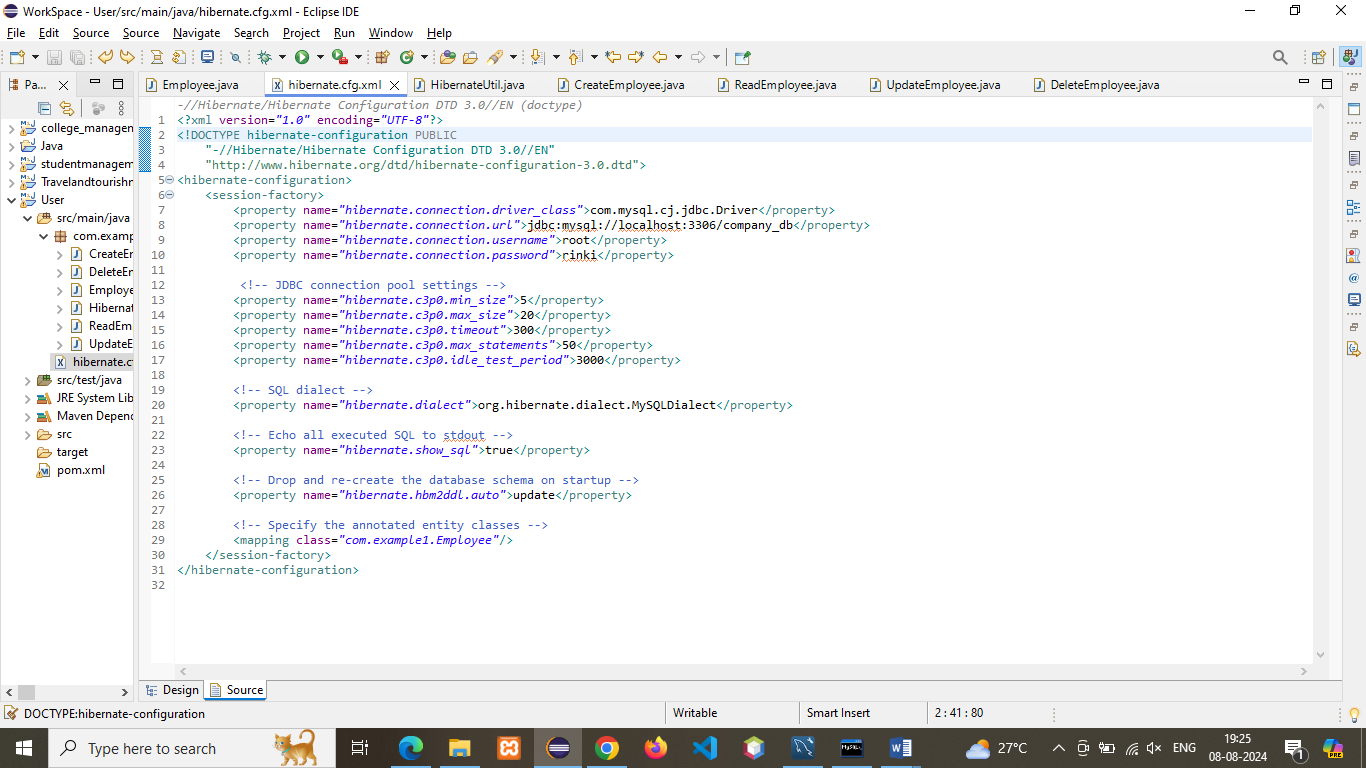
Q1)     CREATE HIBERNATE CRUD OPERATIONS USING entity of your choice. Get the details from respective table using SQL. Define the necessary tables/entities to represent relevant information. Perform update and delete operation.





**Program:**

1. **CreateEmployee**

**package** com.example1;

**import** org.hibernate.Session;

**import** org.hibernate.Transaction;

**public** **class** CreateEmployee {

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

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

Transaction transaction = **null**;

**try** {

transaction = session.beginTransaction();

Employee employee = **new** Employee();

employee.setFirstName("John");

employee.setLastName("Doe");

employee.setEmail("john.doe@example.com");

employee.setSalary(50000);

session.save(employee);

transaction.commit();

System.***out***.println("Employee created successfully! Employee ID: " + employee.getId());

} **catch** (Exception e) {

**if** (transaction != **null**) {

transaction.rollback();

}

e.printStackTrace();

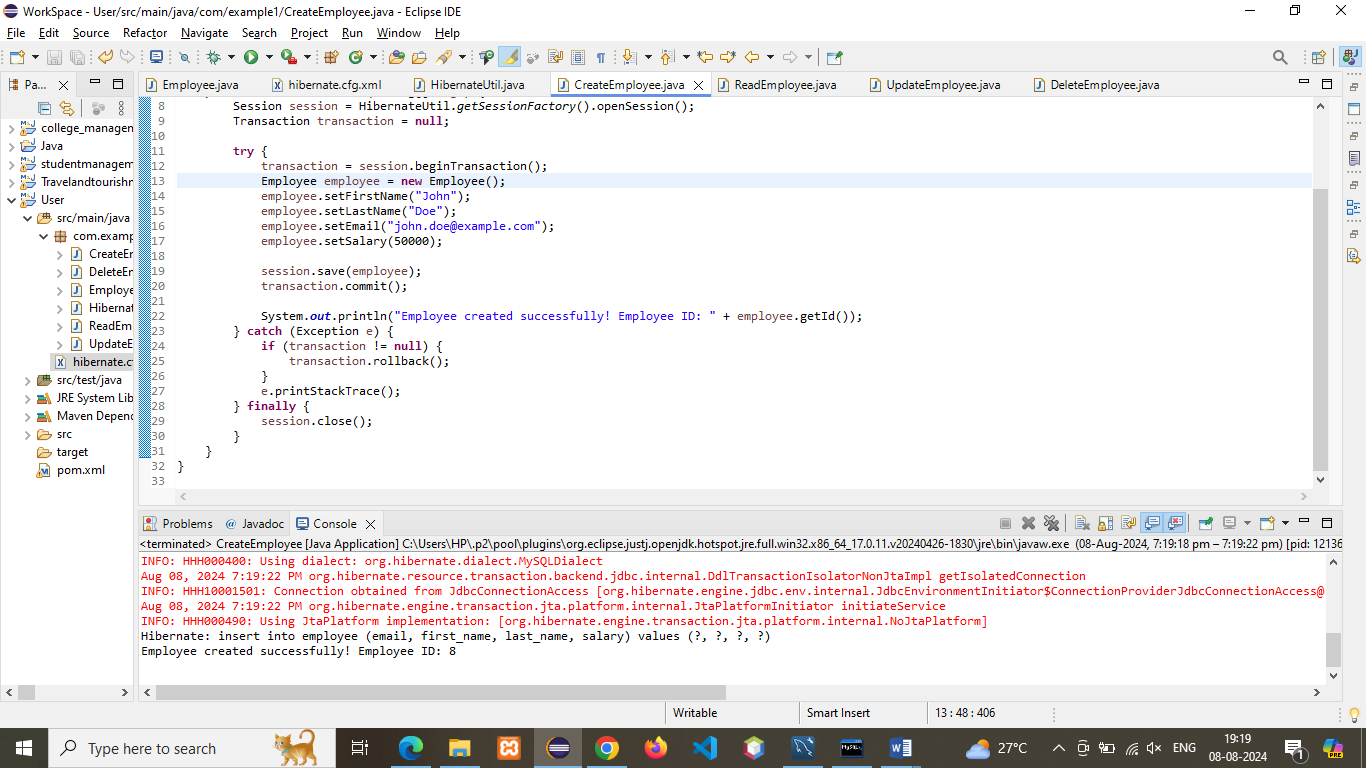
} **finally** {

session.close();

}

}

}



1. **ReadEmployee:**

**package** com.example1;

**import** org.hibernate.Session;

**public** **class** ReadEmployee {

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

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

**try** {

**int** employeeId = 7;

Employee employee = session.get(Employee.**class**, employeeId);

**if** (employee != **null**) {

System.***out***.println("Employee ID: " + employee.getId());

System.***out***.println("First Name: " + employee.getFirstName());

System.***out***.println("Last Name: " + employee.getLastName());

System.***out***.println("Email: " + employee.getEmail());

System.***out***.println("Salary: " + employee.getSalary());

} **else** {

System.***out***.println("Employee not found with ID " + employeeId);

}

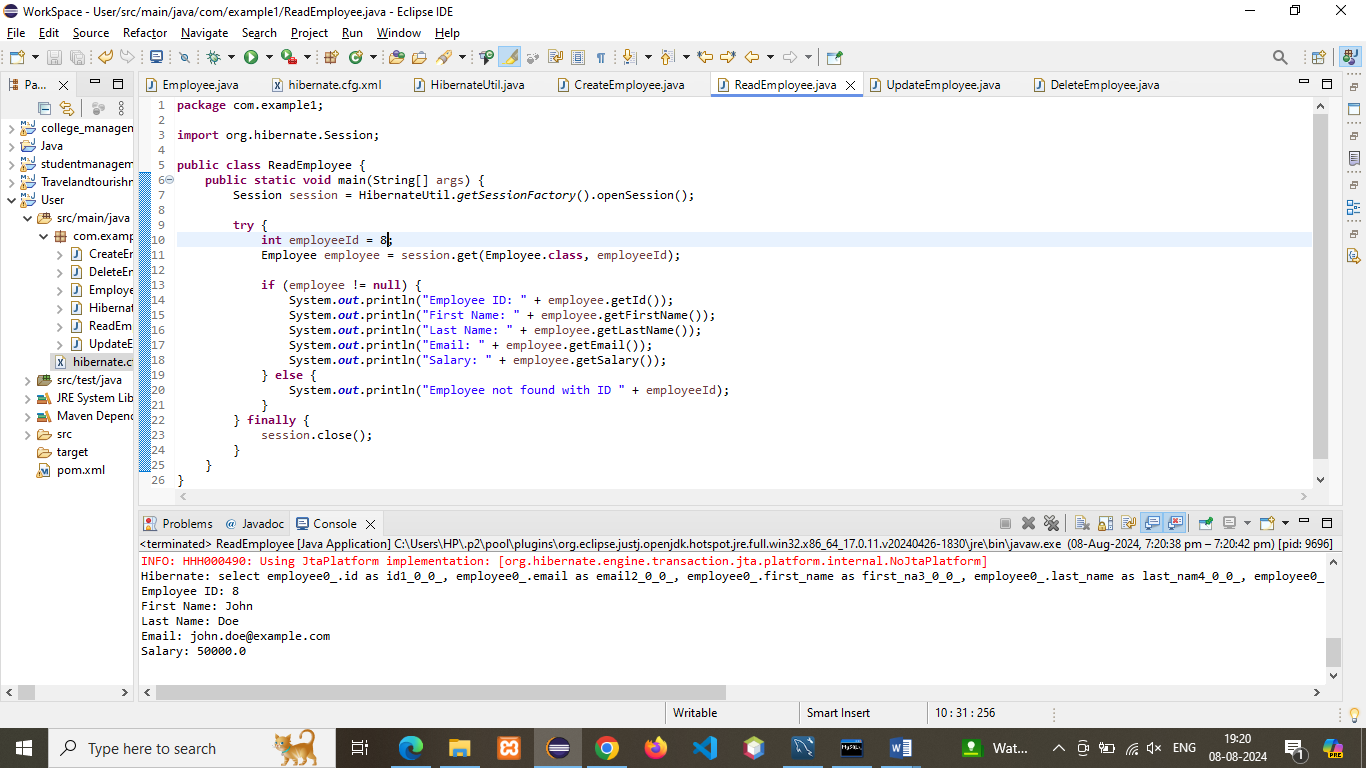
} **finally** {

session.close();

}

}

}



1. **UpdateEmployee:**

**package** com.example1;

**import** org.hibernate.Session;

**import** org.hibernate.Transaction;

**public** **class** UpdateEmployee {

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

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

Transaction transaction = **null**;

**try** {

**int** employeeId = 8;

transaction = session.beginTransaction();

Employee employee = session.get(Employee.**class**, employeeId);

**if** (employee != **null**) {

employee.setSalary(60000);

session.update(employee);

transaction.commit();

System.***out***.println("Employee updated successfully! Updated Salary: " + employee.getSalary());

} **else** {

System.***out***.println("Employee not found with ID " + employeeId);

}

} **catch** (Exception e) {

**if** (transaction != **null**) {

transaction.rollback();

}

e.printStackTrace();

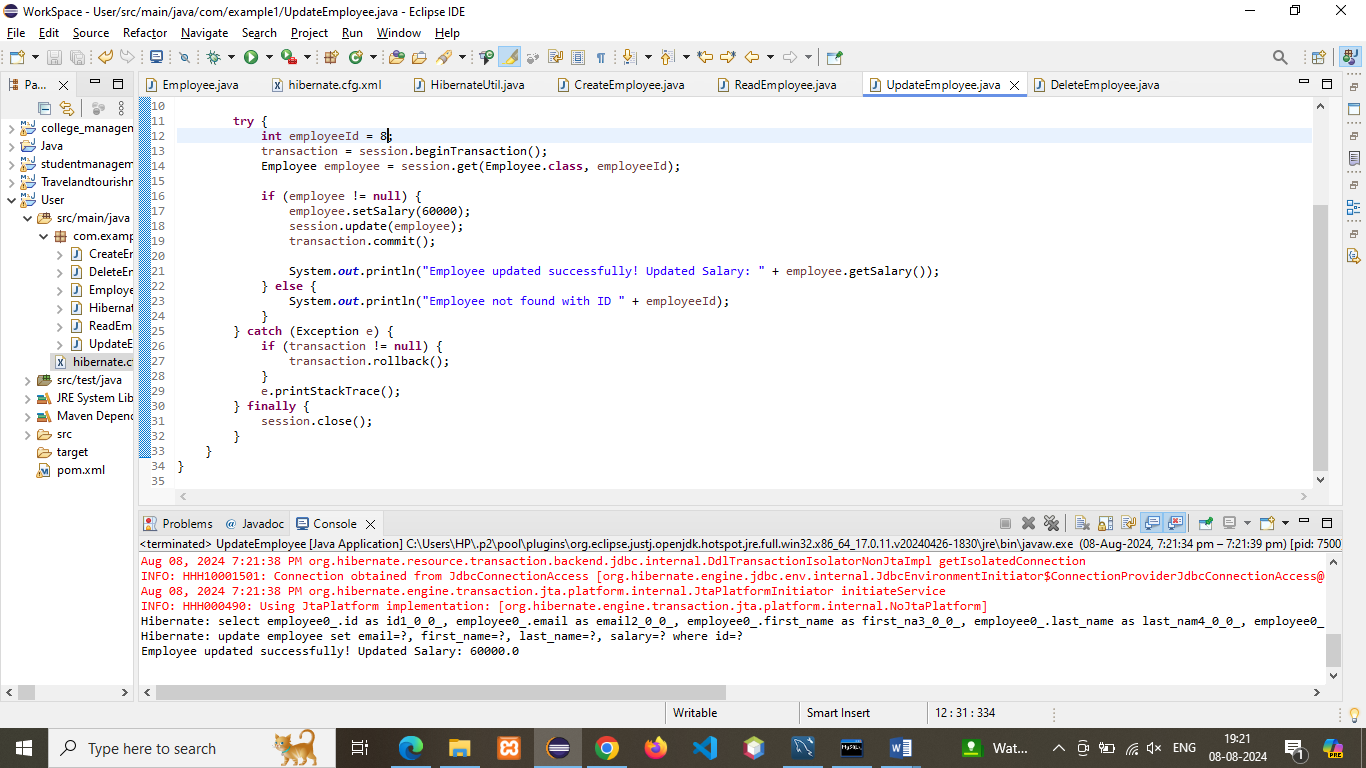
} **finally** {

session.close();

}

}

}



1. **DeleteEmployee:**

**package** com.example1;

**import** org.hibernate.Session;

**import** org.hibernate.Transaction;

**public** **class** DeleteEmployee {

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

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

Transaction transaction = **null**;

**try** {

**int** employeeId = 8;

transaction = session.beginTransaction();

Employee employee = session.get(Employee.**class**, employeeId);

**if** (employee != **null**) {

session.delete(employee);

transaction.commit();

System.***out***.println("Employee deleted successfully!");

} **else** {

System.***out***.println("Employee not found with ID " + employeeId);

}

} **catch** (Exception e) {

**if** (transaction != **null**) {

transaction.rollback();

}

e.printStackTrace();

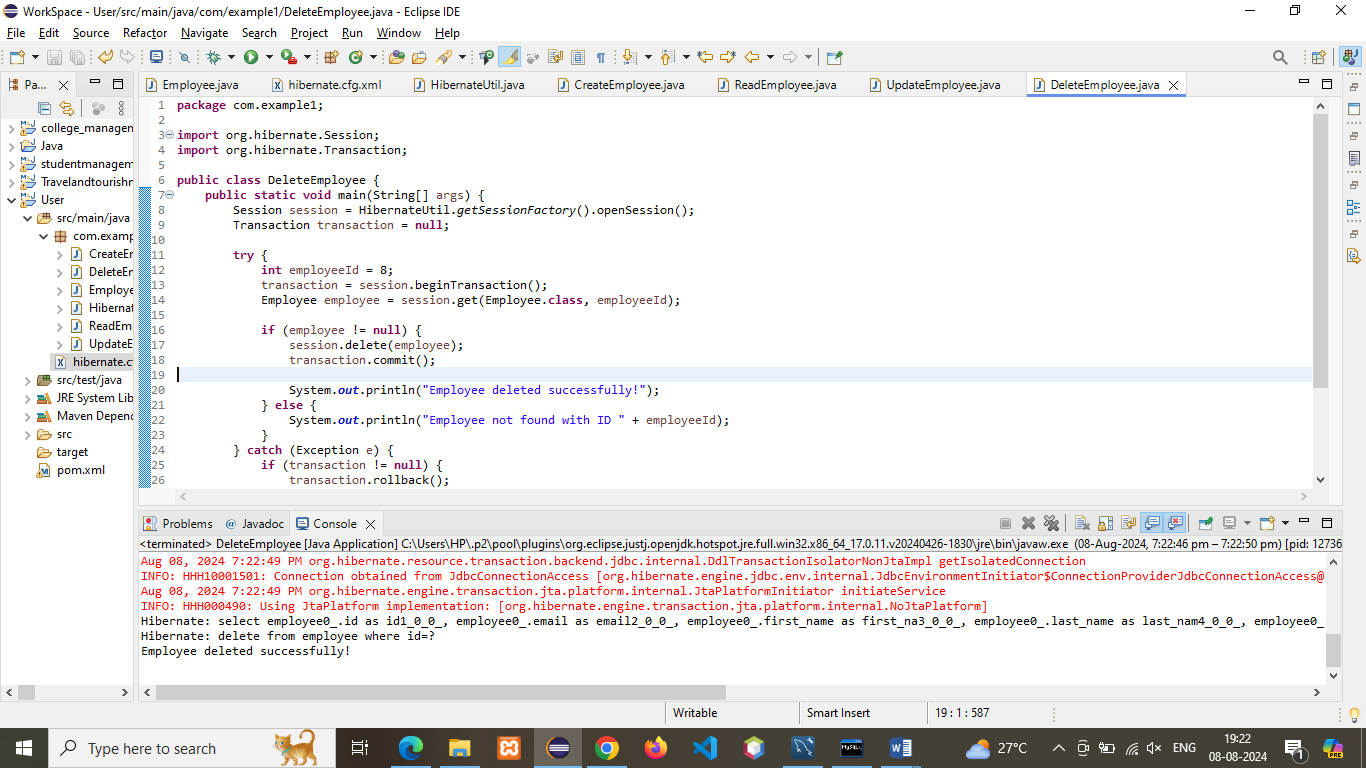
} **finally** {

session.close();

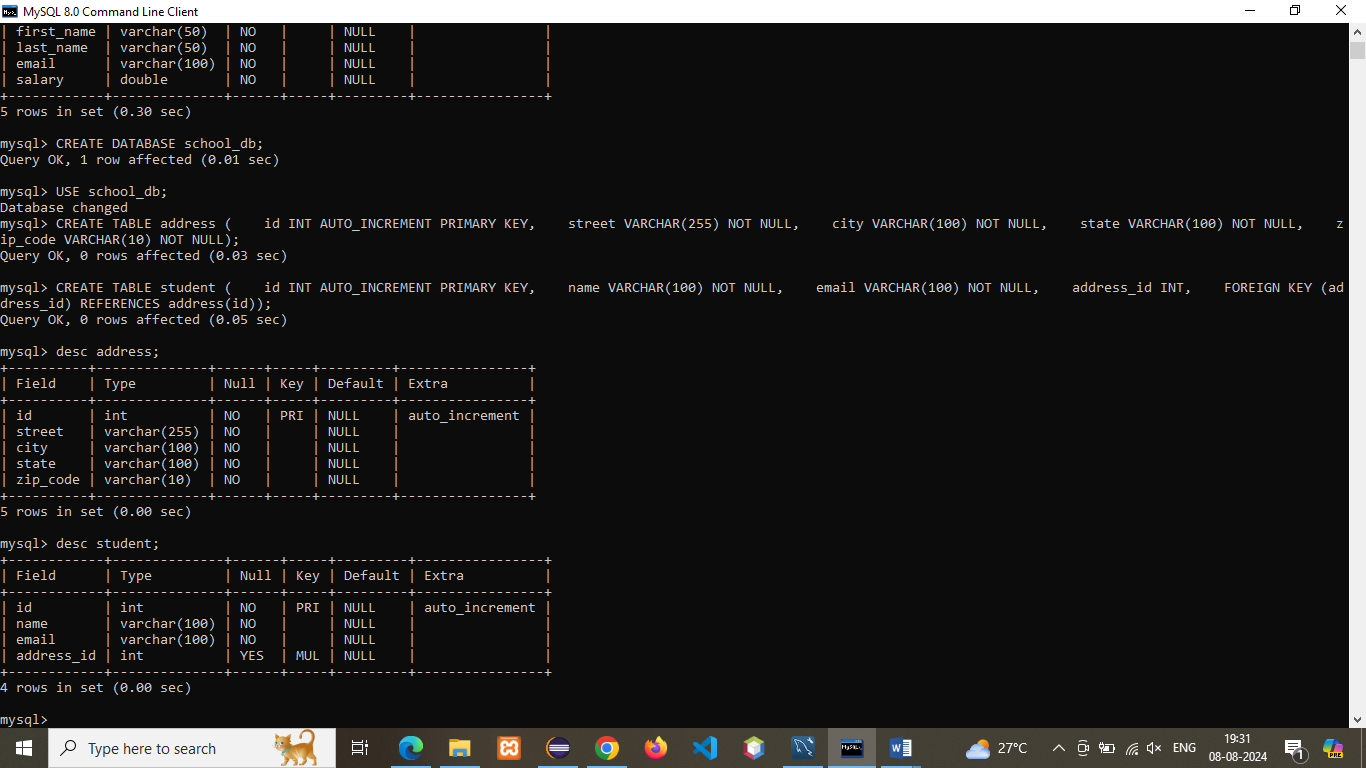
}

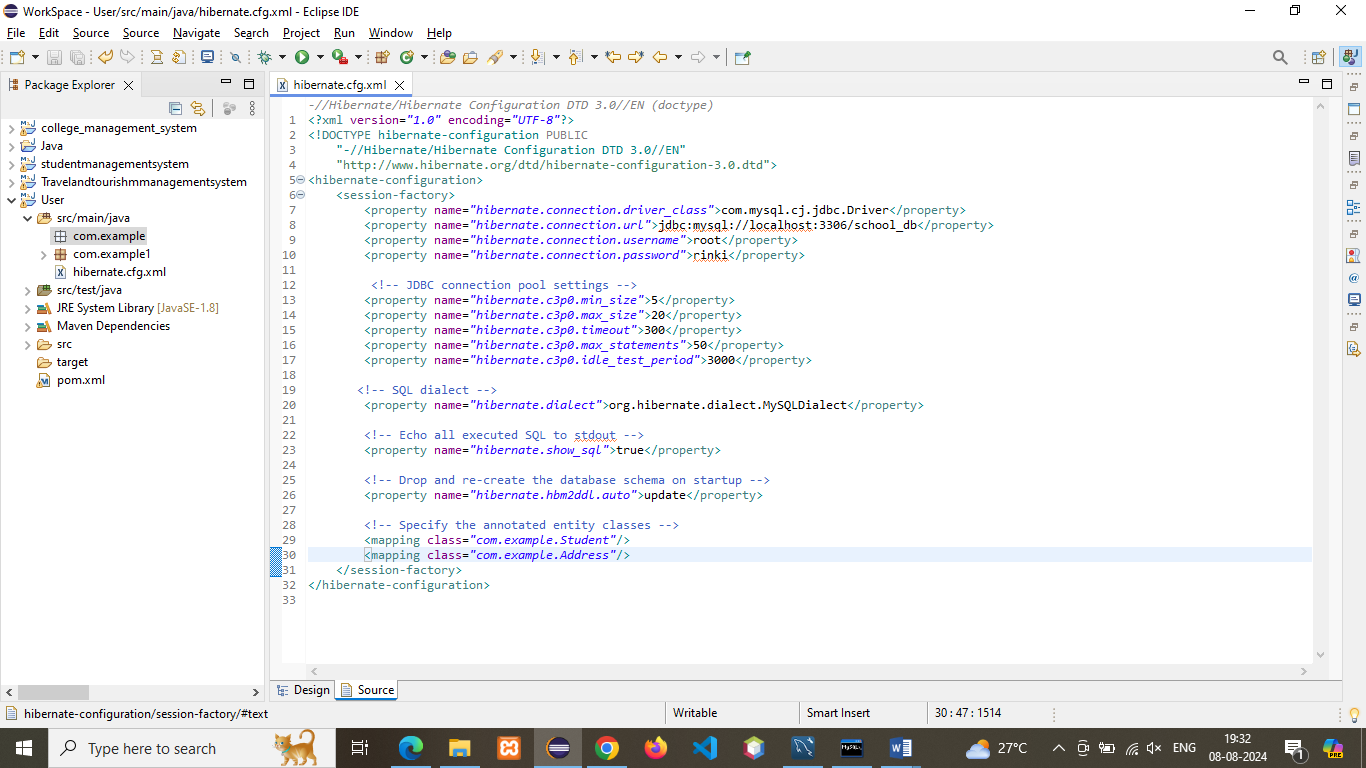
}

}



Q2)  You are working on a Java application to manage information about students and their respective addresses. Implement a one-to-one association between the Student and Address entities using Hibernate.





**Student.Java**:

**package** com.example;

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

@Entity

@Table(name = "student")

**public** **class** Student {

@Id

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

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

@Column(name = "name")

**private** String name;

@Column(name = "email")

**private** String email;

@OneToOne(cascade = CascadeType.***ALL***)

@JoinColumn(name = "address\_id")

**private** Address address;

// Getters and Setters

**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 getEmail() {

**return** email;

}

**public** **void** setEmail(String email) {

**this**.email = email;

}

**public** Address getAddress() {

**return** address;

}

**public** **void** setAddress(Address address) {

**this**.address = address;

}

}

ADDRESS:

**package** com.example;

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

@Entity

@Table(name = "address")

**public** **class** Address {

@Id

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

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

@Column(name = "street")

**private** String street;

@Column(name = "city")

**private** String city;

@Column(name = "state")

**private** String state;

@Column(name = "zip\_code")

**private** String zipCode;

// Getters and Setters

**public** **int** getId() {

**return** id;

}

**public** **void** setId(**int** id) {

**this**.id = id;

}

**public** String getStreet() {

**return** street;

}

**public** **void** setStreet(String street) {

**this**.street = street;

}

**public** String getCity() {

**return** city;

}

**public** **void** setCity(String city) {

**this**.city = city;

}

**public** String getState() {

**return** state;

}

**public** **void** setState(String state) {

**this**.state = state;

}

**public** String getZipCode() {

**return** zipCode;

}

**public** **void** setZipCode(String zipCode) {

**this**.zipCode = zipCode;

}

}

CreateStudent

**package** com.example;

**import** org.hibernate.Session;

**import** org.hibernate.Transaction;

**public** **class** CreateStudent {

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

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

Transaction transaction = **null**;

**try** {

transaction = session.beginTransaction();

Address address = **new** Address();

address.setStreet("123 Main St");

address.setCity("Springfield");

address.setState("IL");

address.setZipCode("62704");

Student student = **new** Student();

student.setName("John Doe");

student.setEmail("john.doe@example.com");

student.setAddress(address);

session.save(student);

transaction.commit();

System.***out***.println("Student created successfully with ID: " + student.getId());

} **catch** (Exception e) {

**if** (transaction != **null**) {

transaction.rollback();

}

e.printStackTrace();

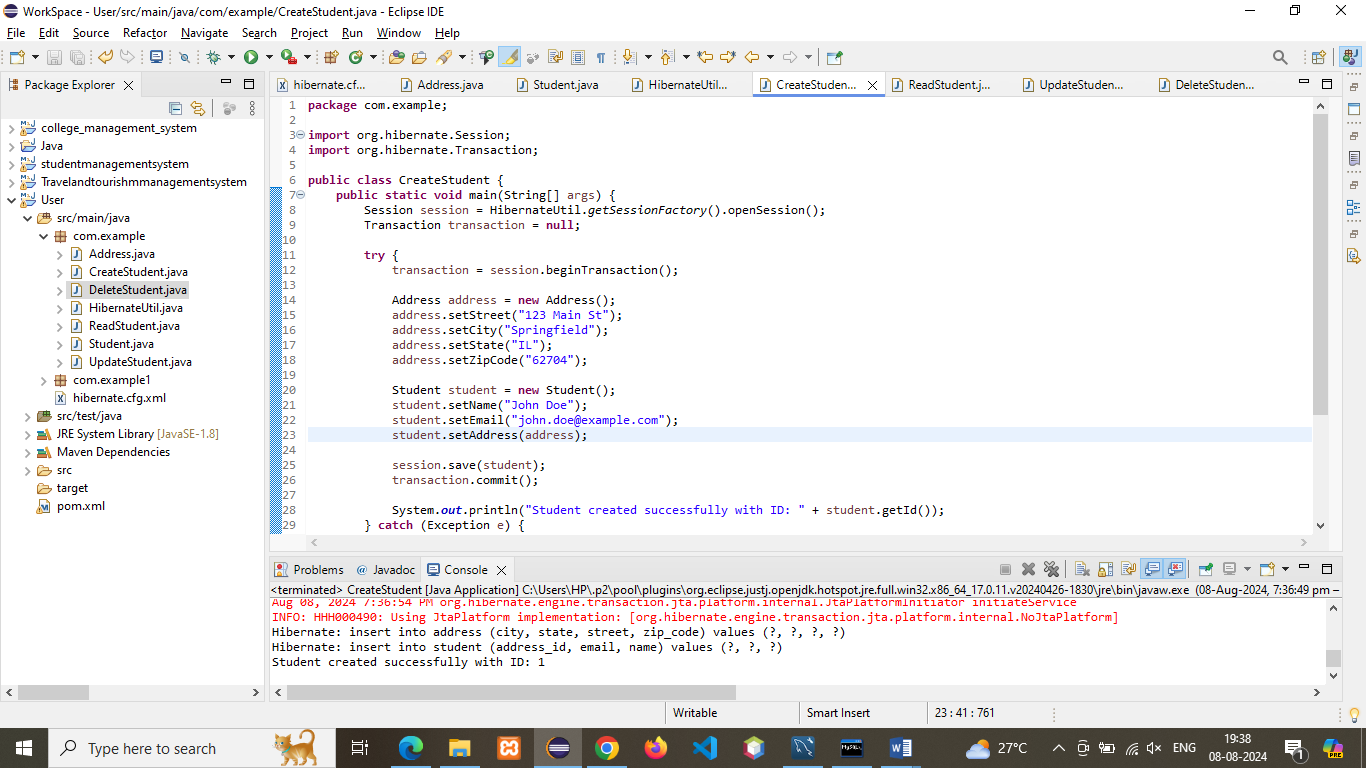
} **finally** {

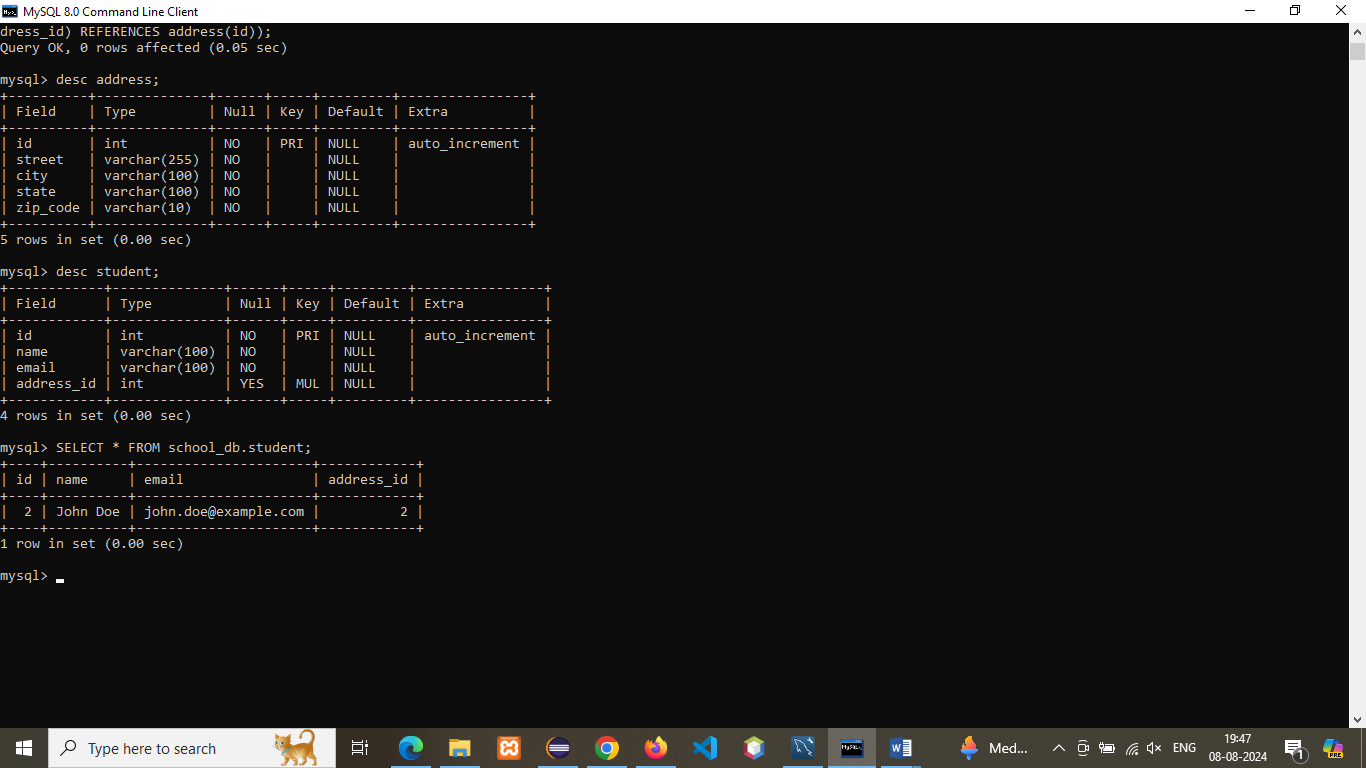
session.close();

}

}

}





**ReadStudent.java:**

**package** com.example;

**import** org.hibernate.Session;

**public** **class** ReadStudent {

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

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

**try** {

**int** studentId = 1;

Student student = session.get(Student.**class**, studentId);

**if** (student != **null**) {

System.***out***.println("Student ID: " + student.getId());

System.***out***.println("Name: " + student.getName());

System.***out***.println("Email: " + student.getEmail());

System.***out***.println("Address: " + student.getAddress().getStreet() + ", " +

student.getAddress().getCity() + ", " +

student.getAddress().getState() + " " +

student.getAddress().getZipCode());

} **else** {

System.***out***.println("Student not found with ID " + studentId);

}

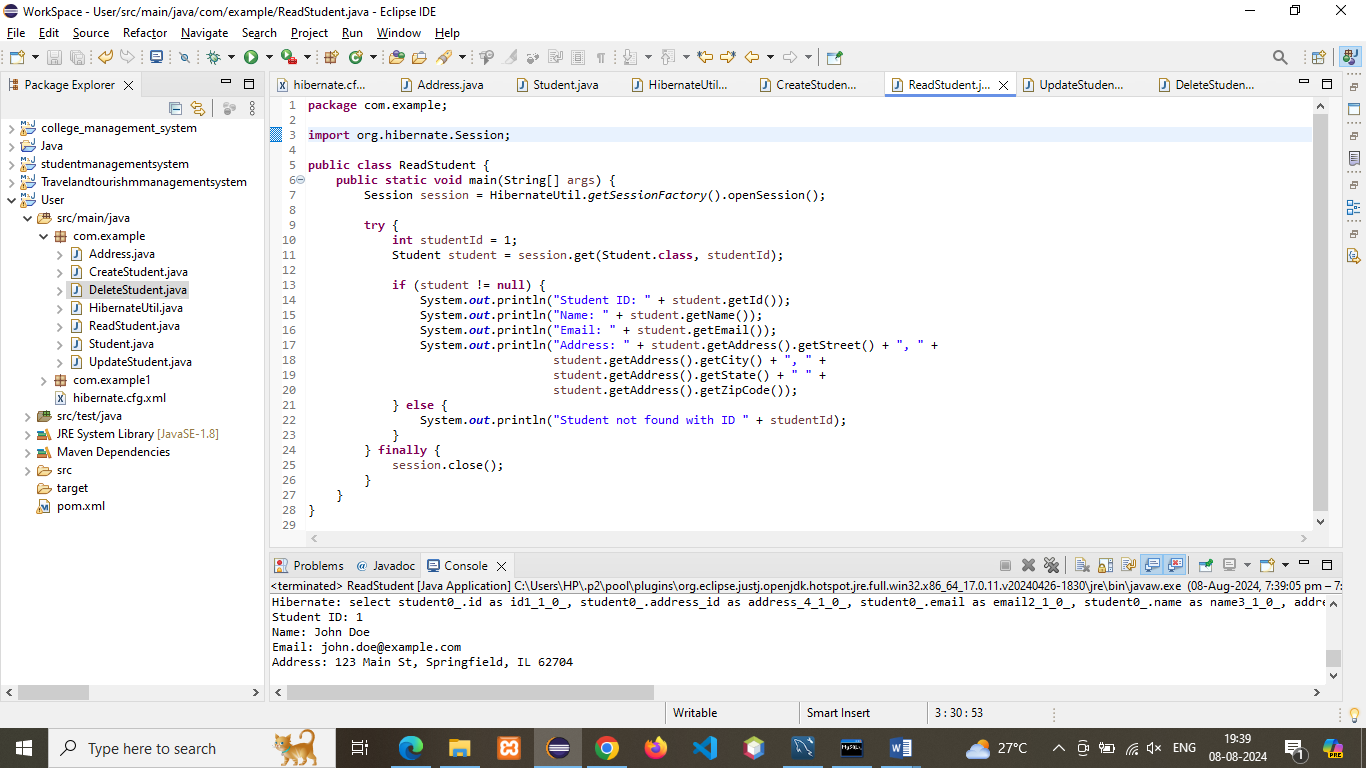
} **finally** {

session.close();

}

}

}



**UpdateStudent.java:**

**package** com.example;

**import** org.hibernate.Session;

**import** org.hibernate.Transaction;

**public** **class** UpdateStudent {

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

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

Transaction transaction = **null**;

**try** {

**int** studentId = 1;

transaction = session.beginTransaction();

Student student = session.get(Student.**class**, studentId);

**if** (student != **null**) {

student.getAddress().setCity("New Springfield");

session.update(student);

transaction.commit();

System.***out***.println("Student updated successfully!");

} **else** {

System.***out***.println("Student not found with ID " + studentId);

}

} **catch** (Exception e) {

**if** (transaction != **null**) {

transaction.rollback();

}

e.printStackTrace();

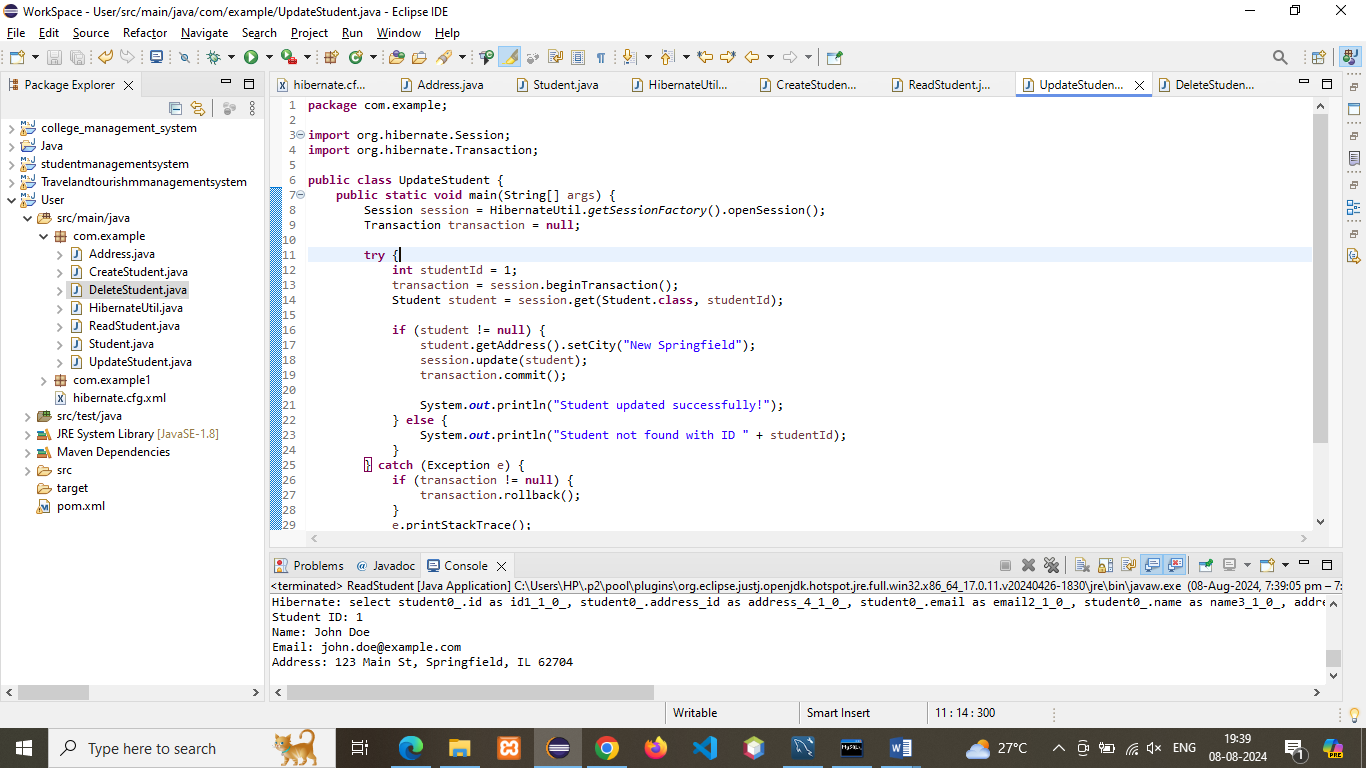
} **finally** {

session.close();

}

}

}



**DeleteStudent.java:**

**package** com.example;

**import** org.hibernate.Session;

**import** org.hibernate.Transaction;

**public** **class** DeleteStudent {

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

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

Transaction transaction = **null**;

**try** {

**int** studentId = 1;

transaction = session.beginTransaction();

Student student = session.get(Student.**class**, studentId);

**if** (student != **null**) {

session.delete(student);

transaction.commit();

System.***out***.println("Student deleted successfully!");

} **else** {

System.***out***.println("Student not found with ID " + studentId);

}

} **catch** (Exception e) {

**if** (transaction != **null**) {

transaction.rollback();

}

e.printStackTrace();

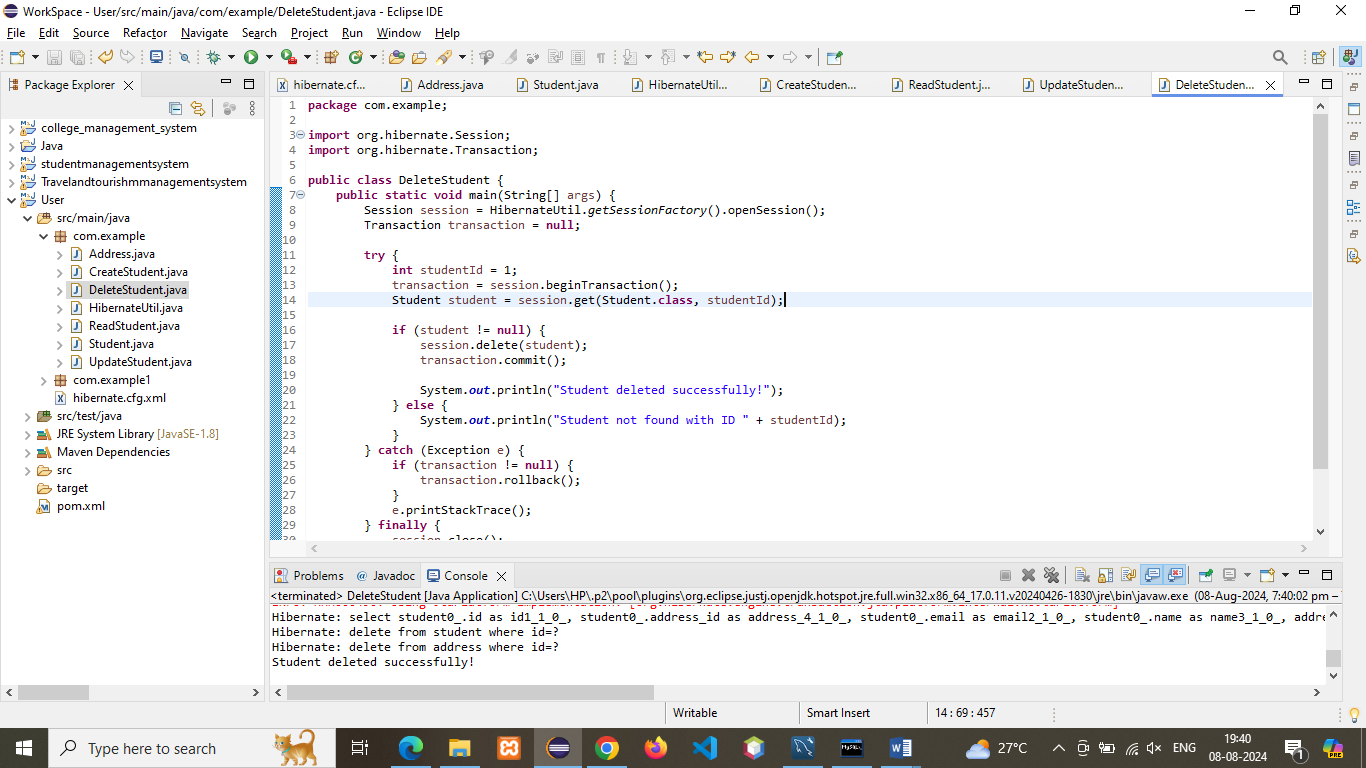
} **finally** {

session.close();

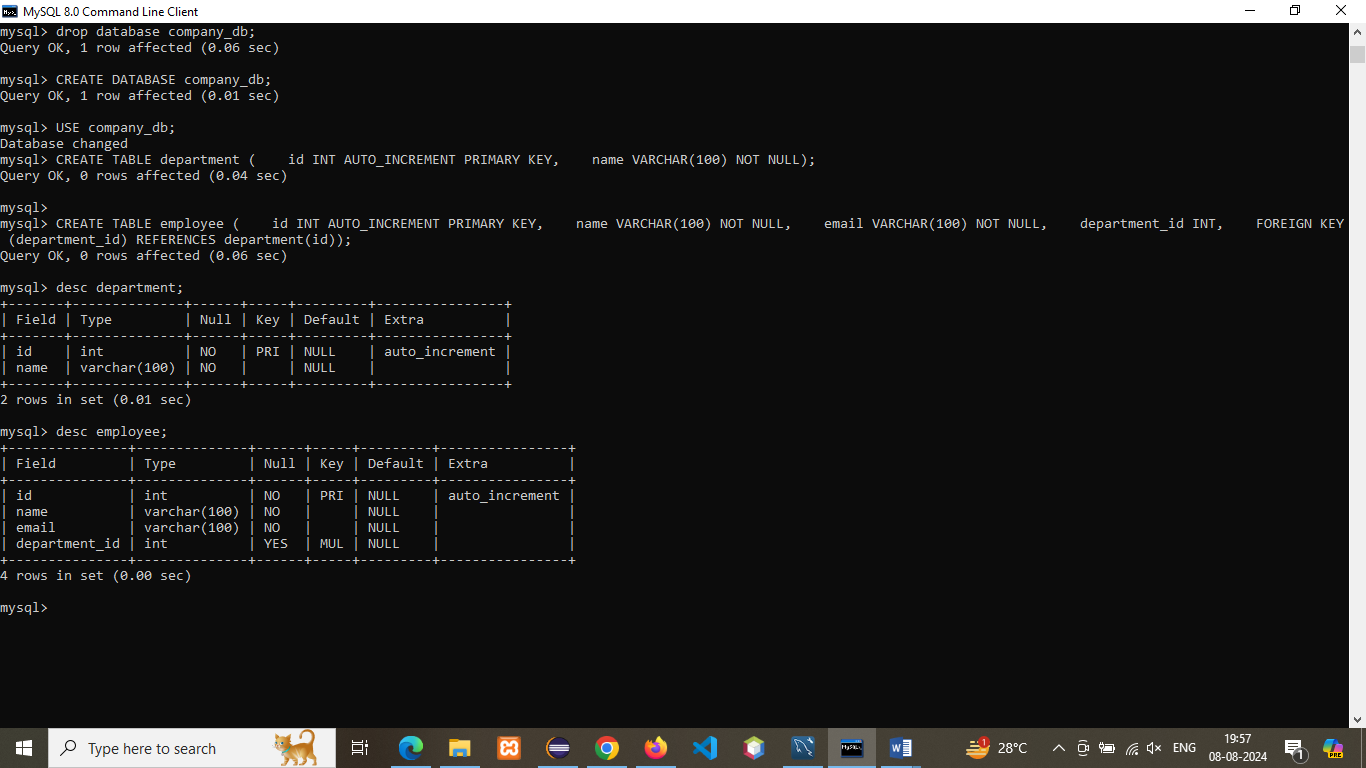
}

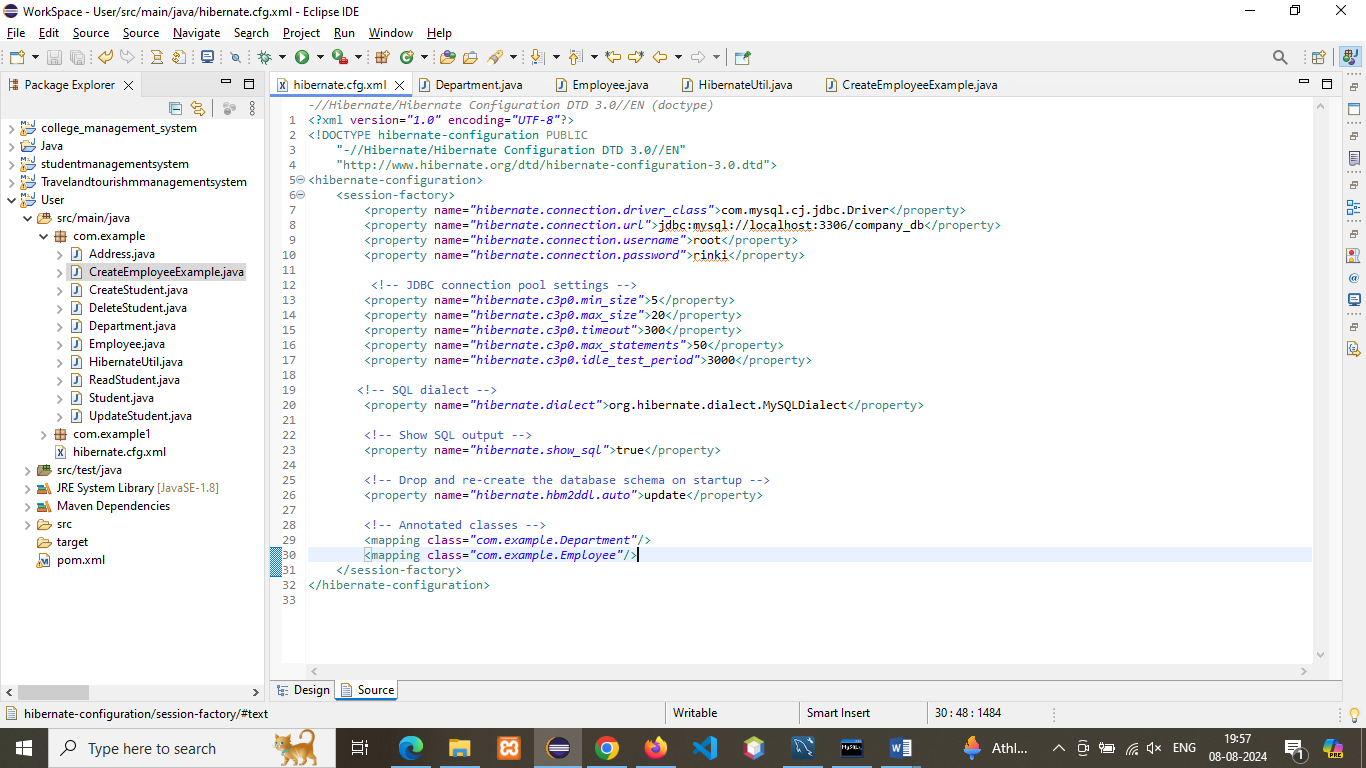
}

}



 Q3)   You are working on a Java application to manage information about employees and their respective departments. Implement a one-to-many association between the Employee and Department entities using Hibernate.





**Department.java:**

**package** com.example;

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

**import** java.util.ArrayList;

**import** java.util.List;

@Entity

@Table(name = "department")

**public** **class** Department {

@Id

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

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

@Column(name = "name")

**private** String name;

@OneToMany(mappedBy = "department", cascade = CascadeType.***ALL***, fetch = FetchType.***LAZY***)

**private** List<Employee> employees = **new** ArrayList<>();

// Constructors, getters, and setters

**public** Department() {

}

**public** Department(String name) {

**this**.name = name;

}

**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** List<Employee> getEmployees() {

**return** employees;

}

**public** **void** setEmployees(List<Employee> employees) {

**this**.employees = employees;

}

}

**Employee.java:**

**package** com.example;

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

@Entity

@Table(name = "employee")

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

@Id

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

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

@Column(name = "name")

**private** String name;

@Column(name = "email")

**private** String email;

@ManyToOne(fetch = FetchType.***LAZY***)

@JoinColumn(name = "department\_id")

**private** Department department;

// Constructors, getters, and setters

**public** Employee() {

}

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

**this**.name = name;

**this**.email = email;

}

**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 getEmail() {

**return** email;

}

**public** **void** setEmail(String email) {

**this**.email = email;

}

**public** Department getDepartment() {

**return** department;

}

**public** **void** setDepartment(Department department) {

**this**.department = department;

}

}

HibernateUtil.Java

**package** com.example;

**import** org.hibernate.SessionFactory;

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

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

**private** **static** **final** SessionFactory ***sessionFactory*** = *buildSessionFactory*();

**private** **static** SessionFactory buildSessionFactory() {

**try** {

**return** **new** Configuration().configure().buildSessionFactory();

} **catch** (Throwable ex) {

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

}

}

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

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

}

**public** **static** **void** shutdown() {

*getSessionFactory*().close();

}

}

**CreateEmployee.java:**

**package** com.example;

**import** org.hibernate.Session;

**import** org.hibernate.Transaction;

**public** **class** CreateEmployeeExample {

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

// Create Department object

Department itDepartment = **new** Department("IT");

// Create Employee objects

Employee emp1 = **new** Employee("Alice Johnson", "alice.johnson@company.com");

Employee emp2 = **new** Employee("Bob Smith", "bob.smith@company.com");

// Associate employees with department

emp1.setDepartment(itDepartment);

emp2.setDepartment(itDepartment);

itDepartment.getEmployees().add(emp1);

itDepartment.getEmployees().add(emp2);

// Start Hibernate session

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

Transaction transaction = **null**;

**try** {

// Begin transaction

transaction = session.beginTransaction();

// Save department (cascades to employees)

session.save(itDepartment);

// Commit transaction

transaction.commit();

// Output

System.***out***.println("Department and Employees saved successfully.");

} **catch** (Exception e) {

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

e.printStackTrace();

} **finally** {

session.close();

}

}

}

