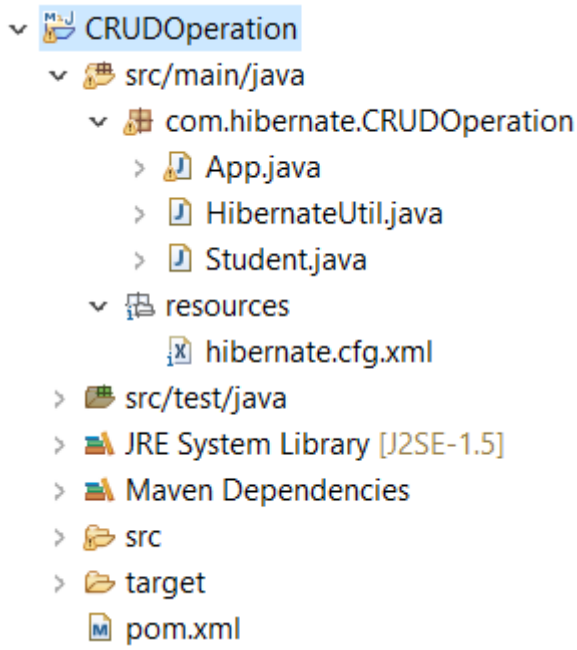
Project Structure



Step 1: Create Maven Project

1. Open your IDE (e.g., IntelliJ IDEA, Eclipse).
2. Create a new Maven project.
    - Group ID: com.hibernate
    - Artifact ID: CRUDusingHQL

Step 2: Add the below Dependencies to pom.xml

1. mysql-connector-j
2. hibernate-core (hibernate-core)
3. jakarta.persistence-api (jakarta.persistence)

Step-3 Create the below Classes

1. Create the Student class in the com.hibernate.CRUDOperation package.

Class : Student

```
package com.hibernate.CRUDOperation;

import jakarta.persistence.Column;
import jakarta.persistence.Entity;
import jakarta.persistence.GeneratedValue;
import jakarta.persistence.GenerationType;
import jakarta.persistence.Id;
import jakarta.persistence.Table;

@Entity
@Table( name = "StudentMaster")
public class Student {
        @Id
```

```
    @GeneratedValue(strategy = GenerationType.IDENTITY)
    private int id;

    @Column(name = "studentname", nullable = false)
    private String name;
    @Column(name = "studentmobile", nullable = false)
    private String mobile;

    public Student() {}
    public Student(int id, String name, String mobile) {
        this.id = id;
        this.name = name;
        this.mobile = mobile;
    }
    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 getMobile() { return mobile; }
    public void setMobile(String mobile) { this.mobile = mobile; }
}
```

Step 4: Create a resources package inside the src/man/java folder.

Create a XML file named hibernate.cfg.xml with the following content inside the resources folder.

```
<?xml version="1.0" encoding="UTF-8"?>
<hibernate-configuration>
    <session-factory>
        <property name="hibernate.connection.driver_class"> com.mysql.cj.jdbc.Driver </property>
        <property name="hibernate.connection.url">
            jdbc:mysql://localhost:3306/hibernateDB
        </property>
        <property name="hibernate.connection.username">root</property>
        <property name="hibernate.connection.password">password</property>
        <property name="hibernate.dialect">org.hibernate.dialect.MySQLDialect</property>
        <property name="show_sql">true</property>
        <property name="format_sql">true</property>
        <property name="hbm2ddl.auto">create </property>

        <mapping class="com.hibernate.CRUDOperation.Student"></mapping>
    </session-factory>
```

```
</hibernate-configuration>
```

Note: Ensure the MySQL database hibernateDB exists, and replace the username and password with your actual MySQL credentials.

Step 5: Create the HibernateUtil Class

Create the HibernateUtil class in the com.hibernate.one_to_one_example package:

```
package com.hibernate.one_to_one_example;
import org.hibernate.SessionFactory;
import org.hibernate.cfg.Configuration;

public class HibernateUtil {
	private static final SessionFactory sessionFactory = buildSessionFactory();
	private static SessionFactory buildSessionFactory() {
		SessionFactory sessionFactory = null;
		try {
			Configuration configuration = new Configuration();
			configuration.configure("resources/hibernate.cfg.xml");
			sessionFactory = configuration.buildSessionFactory();
		}
		catch (Exception e) { e.printStackTrace(); }
		return sessionFactory;
	}

	public static SessionFactory getSessionFactory()  {
		return sessionFactory;
	}
}
```

Step-6 Now add the below lines of code into **App.java** class file.

```
package com.hibernate.CRUDOperation;

import org.hibernate.Transaction;
import org.hibernate.Session;

import java.util.*;

class StudentHandler{

	private int m_srollno;
	private String m_sname,m_smobile;
```

```
private Transaction tran = null;
private Session session;

private Scanner scn = new Scanner(System.in);

public void addstudent() {

	try {
		System.out.println("-------------------------");
		System.out.println("Add record");
		System.out.println("-------------------------");

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

		tran = session.beginTransaction();

	System.out.print("Enter the rollno  :");
	m_srollno = scn.nextInt();
	System.out.print("Enter the name    :");
	m_sname = scn.next();
	System.out.print("Enter the mobile  :");
	m_smobile = scn.next();

	Student student = new Student(m_srollno, m_sname, m_smobile);
	session.save(student);
	tran.commit();
	System.out.println("Record added successfully!");
}
catch (Exception e) {
	System.out.println("Error :" + e.getMessage());
}
session.close();
}

@SuppressWarnings("unchecked")
public void getallstudents() {

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

	try {
		System.out.println("-------------------------");
```

```
                System.out.println("Show all records");
                System.out.println("-------------------------");

                List<Student> lststds =
                        session.createQuery("from Student").list();
                System.out.println("-----------------------------------");
                System.out.println("Rollno\tStudent Name\tMobile");
                System.out.println("-----------------------------------");
                for(Student std : lststds) {
                        System.out.println(
                                std.getId() + "\t" + std.getName() +
                                "\t\t" + std.getMobile());
                }
}
catch (Exception e) {
        System.out.println(e.getMessage());
}
session.close();
}

public void getByStudentID() {

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

        try {
                System.out.println("-------------------------");
                System.out.println("Search record by rollno");
                System.out.println("-------------------------");

        System.out.print("Enter the rollno to be search:");
        m_srollno = scn.nextInt();
        Student tempstd = (Student) session.get(Student.class, m_srollno);
        if(tempstd != null) {
                System.out.println("Rollno      :" + tempstd.getId());
                System.out.println("Student Name:" + tempstd.getName());
                System.out.println("Mobile      :" + tempstd.getMobile());
        }
        else
                System.out.println("Record does not exists!");
        }
        catch (Exception e) {
```

```
                System.out.println(e.getMessage());
        }
        session.close();
}

public void updatestudent() {
        session = HibernateUtil.getSessionFactory().openSession();

        try {
                System.out.println("-------------------------");
                System.out.println("Update record");
                System.out.println("-------------------------");

        tran = session.beginTransaction();
        System.out.print("Enter the rollno to be update:");
        m_srollno = scn.nextInt();
        Student tempstd = (Student) session.get(Student.class, m_srollno);
        if(tempstd != null) {
                System.out.println("Old Data");
                System.out.println("Rollno :" + tempstd.getId());
                System.out.println("Name   :" + tempstd.getName());
                System.out.println("Mobile :" + tempstd.getMobile());

                System.out.println("New Data");
        System.out.print("Enter the name    :");
        m_sname = scn.next();
        System.out.print("Enter the mobile  :");
        m_smobile = scn.next();

        tempstd.setId(m_srollno);
        tempstd.setName(m_sname);
        tempstd.setMobile(m_smobile);

        session.persist(tempstd);
        tran.commit();
        System.out.println("Record updated successfully!");
        }
        else
                System.out.println("Record does not exists!");
        }
        catch (Exception e) {
```

```
                System.out.println(e.getMessage());
            }
            session.close();
        }

        public void deletestudent() {
            session = HibernateUtil.getSessionFactory().openSession();

            try {
                System.out.println("-------------------------");
                System.out.println("Delete record");
                System.out.println("-------------------------");

                tran = session.beginTransaction();
                System.out.print("Enter the rollno to be delete:");
                m_srollno = scn.nextInt();
                Student tempstd = (Student) session.get(Student.class, m_srollno);
                if(tempstd != null) {
                    session.delete(tempstd);
                    tran.commit();
                    System.out.println("Record deleted successfully!");
                }
                else
                    System.out.println("Record does not exists!");
                }
                catch (Exception e) {
                    System.out.println(e.getMessage());
                }
                session.close();
        }
}

public class App
{
   public static void main( String[] args )
   {
        StudentHandler sh = new StudentHandler();
        Scanner scn = new Scanner(System.in);
        int choice = 0;
```

```
        while(choice != 6) {
                System.out.println("1...Insert Record");
                System.out.println("2...Edit Record");
                System.out.println("3...Delete Record");
                System.out.println("4...Show All");
                System.out.println("5...Search by Rollno");
                System.out.println("6...Exit");

                System.out.print("Enter the valid choice :");
                choice = scn.nextInt();

                switch(choice) {
                        case 1:
                                sh.addstudent();
                                break;
                        case 2:
                                sh.updatestudent();
                                break;
                        case 3:
                                sh.deletestudent();
                                break;
                        case 4:
                                sh.getallstudents();
                                break;
                        case 5:
                                sh.getByStudentID();
                                break;
                        case 6:
                                break;
                        default:
                                System.out.println("Invalid choice !");
                                break;
                }
        }
    }
}
```

Step-7 Right click on App.java file and Run the application