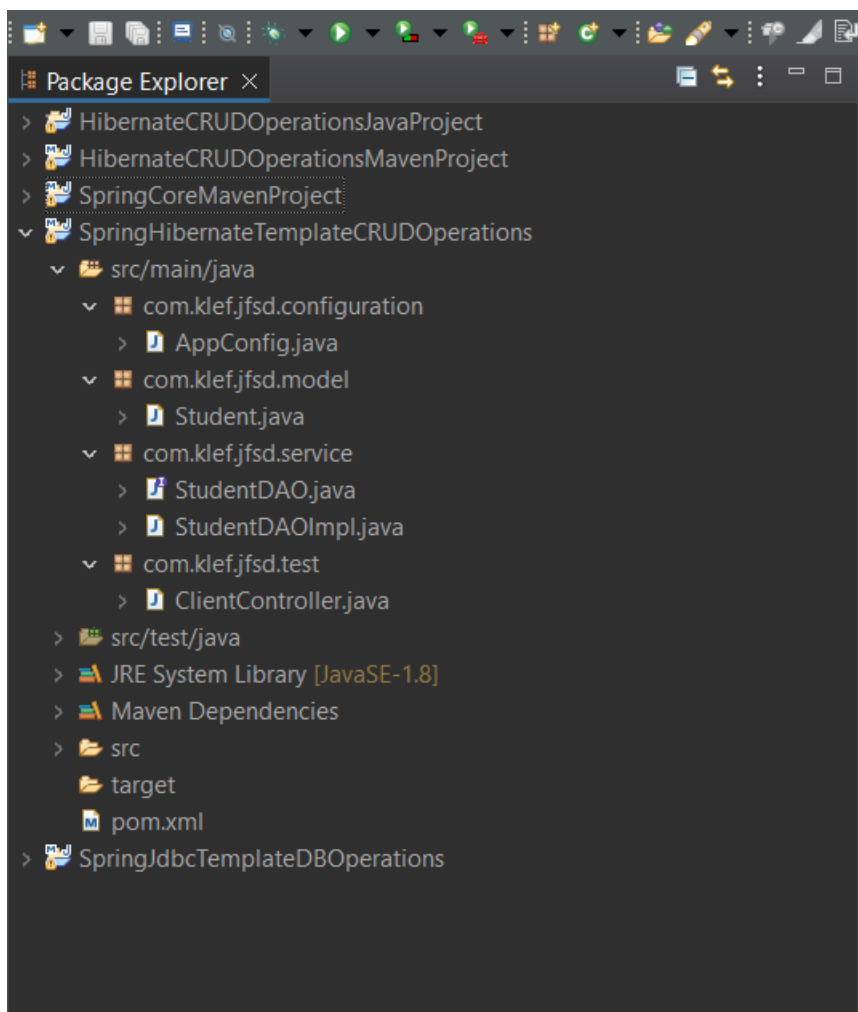ID:2000030185

Name: Suprith Ch

Question:

1.The Navodaya Johar school is wanting to direct a get-together of their 2021-2022 batch students. To design this occasion and oversee, principal appointed a student from a similar batch. To do this the student needed to make hibernate application with spring application. He initially made a table in database to store the subtleties of his companions who are going to the gathering. He made the table with the name Reunion with properties ID No, Name, Contact Number, Amount paid, and Status. ID.NO being the primary key has the most extreme size of 4 digits and Status speaks to how much sum has been paid i.e., completely paid, or partially paid, or not paid. Note that the sum should be paid to go to the gathering is 2000 rupees. Now he needs to make a java project with springs and hibernate integration. Help him in Creating three unique classes for inserting subtleties in database. First make a class for inserting the details of the students who are going to the get-together gathering. Presently make another class for retrieving the amount paid by the students.

2. Now to the extension to the last question create a class file in the same java project for updating the database If the student wants to pay the amount, then update the amount in data base and print the total amount he paid till then and if he had paid the total amount the change the status to fully paid. If not, then show the amount that need to be paid by him. Also create a class file for deleting the details of students who cancelled their plan of going go get-together because of some issues.
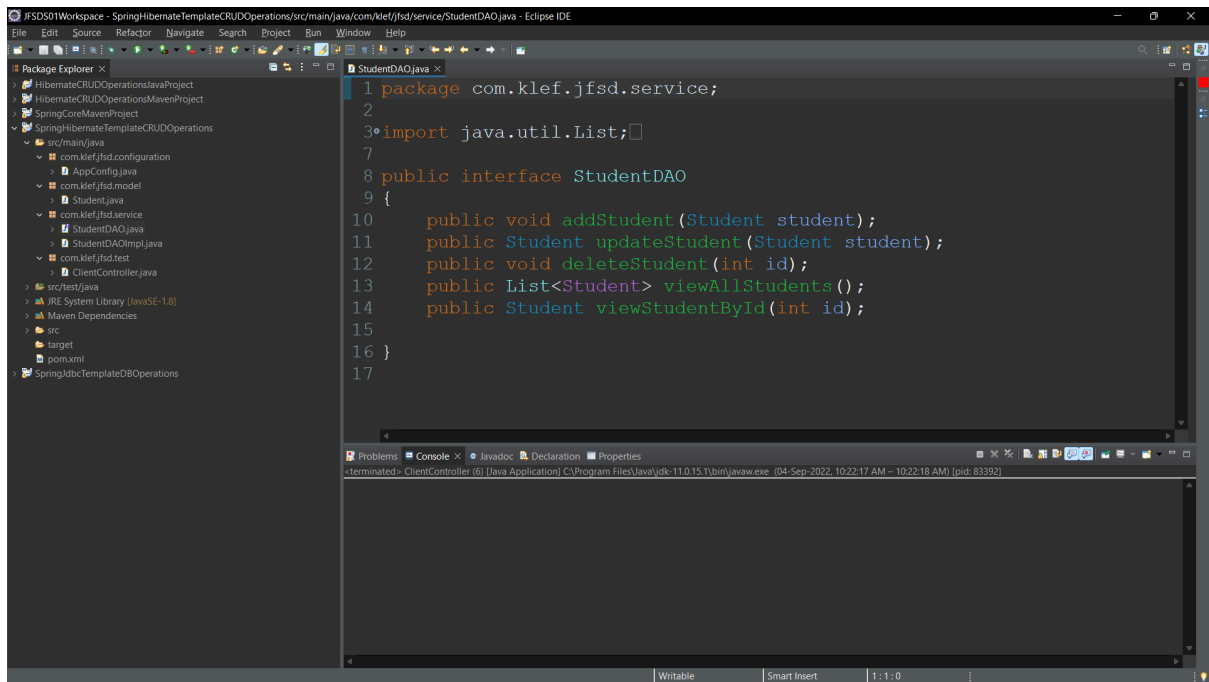
Architecture of Project:

Student.java

```java
package com.klef.jfsd.model;

import javax.persistence.*;

@Entity
@Table(name="student_table")
public class Student
{
    @Id
    @GeneratedValue //auto generator class
    @Column(name="student_id")
    private int id;
    @Column(name="student_name",nullable = false,length = 200)
    private String name;
    @Column(name="student_gender",nullable = false)
    private String gender;
    @Column(name="student_dob",nullable = false)
    private String dob;//date of birth
    @Column(name="student_department",nullable = false)
    private String department;//CSE or ECE or CSIT
    @Column(name="student_program",nullable = false)
    private String program;//B.Tech or M.Tech
    @Column(name="student_cgpa",nullable = false)
    private double cgpa;
    @Column(name="student_backlogs",nullable = false)
    private int backlogs;
```

```java
    private String emailid;
    @Column(name="student_contactno",nullable = false,length = 100,unique = true)
    private String contactno;
    @Column(name="student_location",nullable = false,length = 200)
    private String location;
    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 getGender() {
        return gender;
    }
    public void setGender(String gender) {
        this.gender = gender;
    }
    public String getDob() {
        return dob;
    }
```

File   Edit   Source   Refactor   Navigate   Search   Project   Run   Window   Help

AppConfig.java   Employee.java   EmployeeService.java   ClientController.java   AppConfig.java   Student.java

```java
71        }
72•     public void setCgpa(double cgpa) {
73            this.cgpa = cgpa;
74        }
75•     public int getBacklogs() {
76            return backlogs;
77        }
78•     public void setBacklogs(int backlogs) {
79            this.backlogs = backlogs;
80        }
81•     public String getEmailid() {
82            return emailid;
83        }
84•     public void setEmailid(String emailid) {
85            this.emailid = emailid;
86        }
87•     public String getContactno() {
88            return contactno;
89        }
90•     public void setContactno(String contactno) {
91            this.contactno = contactno;
92        }
93•     public String getLocation() {
94            return location;
95        }
96•     public void setLocation(String location) {
```

Writable   Smart Insert   1 : 1 : 0

File   Edit   Source   Refactor   Navigate   Search   Project   Run   Window   Help

AppConfig.java   Employee.java   EmployeeService.java   ClientController.java   AppConfig.java   Student.java

```java
83        }
84•     public void setEmailid(String emailid) {
85            this.emailid = emailid;
86        }
87•     public String getContactno() {
88            return contactno;
89        }
90•     public void setContactno(String contactno) {
91            this.contactno = contactno;
92        }
93•     public String getLocation() {
94            return location;
95        }
96•     public void setLocation(String location) {
97            this.location = location;
98        }
99•     @Override
100     public String toString() {
101         return "Student [id=" + id + ", name=" + name + ", gender=" + gender + ", dob=" + dob + ",
102                 + department + ", program=" + program + ", cgpa=" + cgpa + ", backlogs=" + backlog
103                 + emailid + ", contactno=" + contactno + ", location=" + location + "]";
104     }
105
106
107 }
108
```
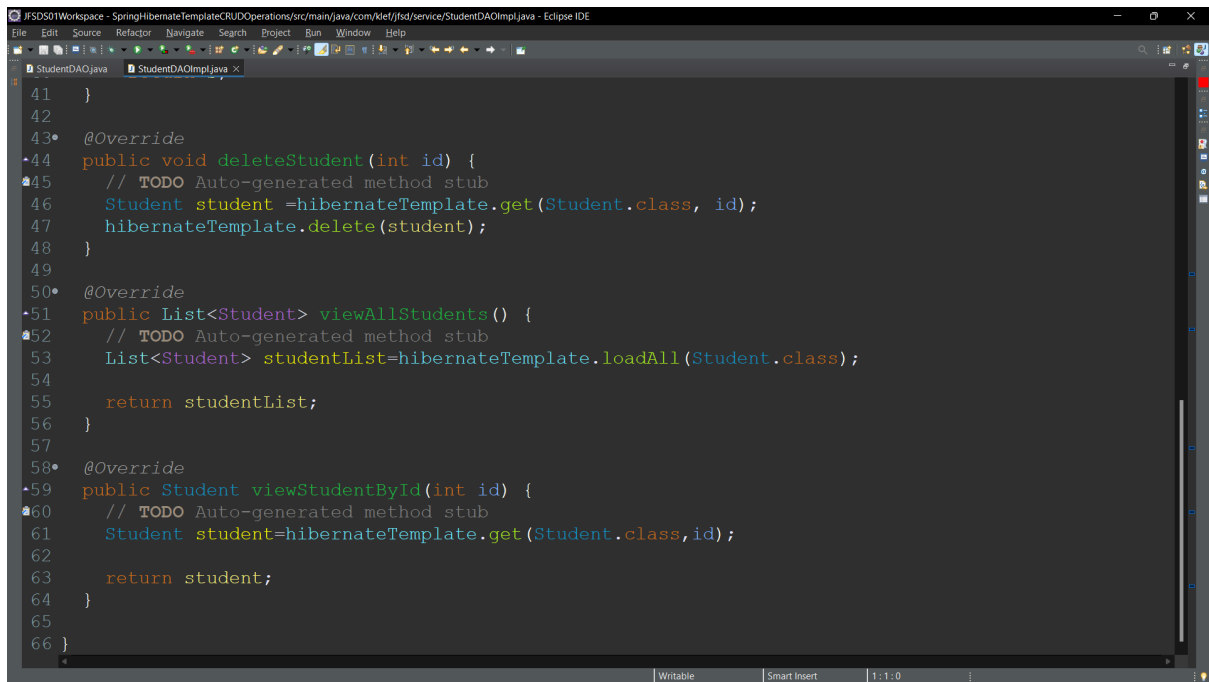
Writable   Smart Insert   1 : 1 : 0

StudentDAO.java interface

StudentDAOImpl.java class

```java
package com.klef.jfsd.service;

import java.util.List;

@Transactional
public class StudentDAOImpl implements StudentDAO {

    public HibernateTemplate hibernateTem    [com.klef.jfsd.service.StudentDAO]
                                                          Press 'F2' for focus
    public HibernateTemplate getHibernateTemplate() {
        return hibernateTemplate;
    }

    public void setHibernateTemplate(HibernateTemplate hibernateTemplate) {
        this.hibernateTemplate = hibernateTemplate;
    }

    @Override
    public void addStudent(Student student) {
        // TODO Auto-generated method stub
        hibernateTemplate.save(student);
    }

    @Override
    public Student updateStudent(Student student) {
        // TODO Auto-generated method stub
```

```java
    }

    @Override
    public Student updateStudent(Student student) {
        // TODO Auto-generated method stub
        Student s=  hibernateTemplate.get(Student.class,student.getId());

        s.setContactno(student.getContactno());
        s.setCgpa(student.getCgpa());
        s.setBacklogs(student.getBacklogs());
        s.setLocation(student.getLocation());

        return s;
    }

    @Override
    public void deleteStudent(int id) {
        // TODO Auto-generated method stub
        Student student =hibernateTemplate.get(Student.class, id);
        hibernateTemplate.delete(student);
    }

    @Override
    public List<Student> viewAllStudents() {
        // TODO Auto-generated method stub
        List<Student> studentList=hibernateTemplate.loadAll(Student.class);
```

File   Edit   Source   Refactor   Navigate   Search   Project   Run   Window   Help

StudentDAO.java   StudentDAOImpl.java ×

```java
 41    }
 42
 43•   @Override
-44    public void deleteStudent(int id) {
 45        // TODO Auto-generated method stub
 46        Student student =hibernateTemplate.get(Student.class, id);
 47        hibernateTemplate.delete(student);
 48    }
 49
 50•   @Override
-51    public List<Student> viewAllStudents() {
 52        // TODO Auto-generated method stub
 53        List<Student> studentList=hibernateTemplate.loadAll(Student.class);
 54
 55        return studentList;
 56    }
 57
 58•   @Override
-59    public Student viewStudentById(int id) {
 60        // TODO Auto-generated method stub
 61        Student student=hibernateTemplate.get(Student.class,id);
 62
 63        return student;
 64    }
 65
 66 }
```

Writable   Smart Insert   1 : 1 : 0

ClientController.java

```java
package com.klef.jfsd.test;

import java.util.List;

public class ClientController {

    AnnotationConfigApplicationContext applicationContext=new AnnotationConfigApplicationContext(AppC
    StudentDAO dao=applicationContext.getBean(StudentDAO.class);

    public static void main(String[] args) {
        ClientController clientController=new ClientController();
        //clientController.AddStudentRecord();
        //clientController.UpdateStudentRecord();
        //clientController.DeleteStudentRecord();
//        clientController.ViewStudentRecordByID();
        clientController.ViewAllStudents();
    }
    public void AddStudentRecord() {

        Student student=new Student();
        student.setName("Srinivas");
        student.setGender("MALE");
        student.setDob("09/10/2002");
        student.setDepartment("ECE");
        student.setProgram("M.Tech");
        student.setCgpa(7.6);
```

```java
        student.setEmailid("srinivas@gmail.com");
        student.setContactno("7756788402");
        student.setLocation("Kadapa");

        dao.addStudent(student);
        System.out.println("Student Record added successfully");
    }

    public void UpdateStudentRecord() {

        Student student=new Student();
        student.setId(5);
        student.setContactno("8977295499");
        student.setCgpa(9.45);
        student.setBacklogs(1);
        student.setLocation("Guntur");

        dao.updateStudent(student);
        System.out.println("Student Record updated successfuly");
    }

    public void DeleteStudentRecord() {

        dao.deleteStudent(6);
        System.out.println("Student Record deleted successfully");
    }
```