

# **JAVA SWINGS BASED- SIH IT PORTAL - SQL CONNECTIVITY USING JDBC**

*A*

*Report*

*Submitted in partial fulfillment of the  
Requirements for the award of the Degree  
of*

**BACHELOR OF ENGINEERING**

**IN**

**INFORMATION TECHNOLOGY**

**By**

Singari Tejaswi <1602-20-737-055 > Under  
the guidance of Ms B. Leelavathy



Department of Information Technology

Vasavi College of Engineering

(Autonomous) (Affiliated to Osmania

University) Ibrahimbagh, Hyderabad-31

## BONAFIDE CERTIFICATE

This is to certify that this project report titled '**SIH IT PORTAL**' is a project work of Singari Tejaswi bearing roll no. 1602-20-737-055 who carried out the project under my supervision in the IV semester for the academic year 2021- 2022.

Signature

Internal Examiner  
Examiner

Signature

External

**ABSTRACT:**

In this project, we have to create a Hackathon portal where students can input their innovative ideas on the portal that is under the control of the department of IT. This portal is useful for a department where they can announce any projects in which students can register and submit their problem statements. To implement this portal and to display all the data on the screen we need to build the database with details of students, project information, faculty and previous information as a back end database. To build this portal, we are using SQL for the back end and java for the front end part.

## **REQUIREMENT ANALYSIS:**

### **List of Tables:**

- student\_details
- project\_details
- faculty
- Hackathon\_event

### **List of attributes with their domain types:**

#### Student\_details

- rno(varchar2)
- sname(varchar2)
- year(number)
- sem(number)
- phone\_no(number)

#### Project\_details

- pname(varchar2)
- duration(sysdate)
- year(number)
- explanation(varchar2)

#### Faculty

- fid(varchar2)
- fname(varchar2)
- pname(varchar2)

- year(number)
- phone\_no(number)

Hackathon\_event

- year(number)
- problem\_statement(varchar2)
- pname(varchar2)
- submitted\_time(smалldatetime)

## AIM AND PRIORITY OF THE PROJECT

To create a Java GUI based for sih it portal which takes the values like details of all students, faculty, projects and the registrations of students for projects. These values are to be updated in the database using JDBC connectivity.

## ARCHITECTURE AND TECHNOLOGY

**Software used:** Java Notepad++, Oracle 11g Database, Java SE version 13, SQL\*Plus.

### Java SWINGS:

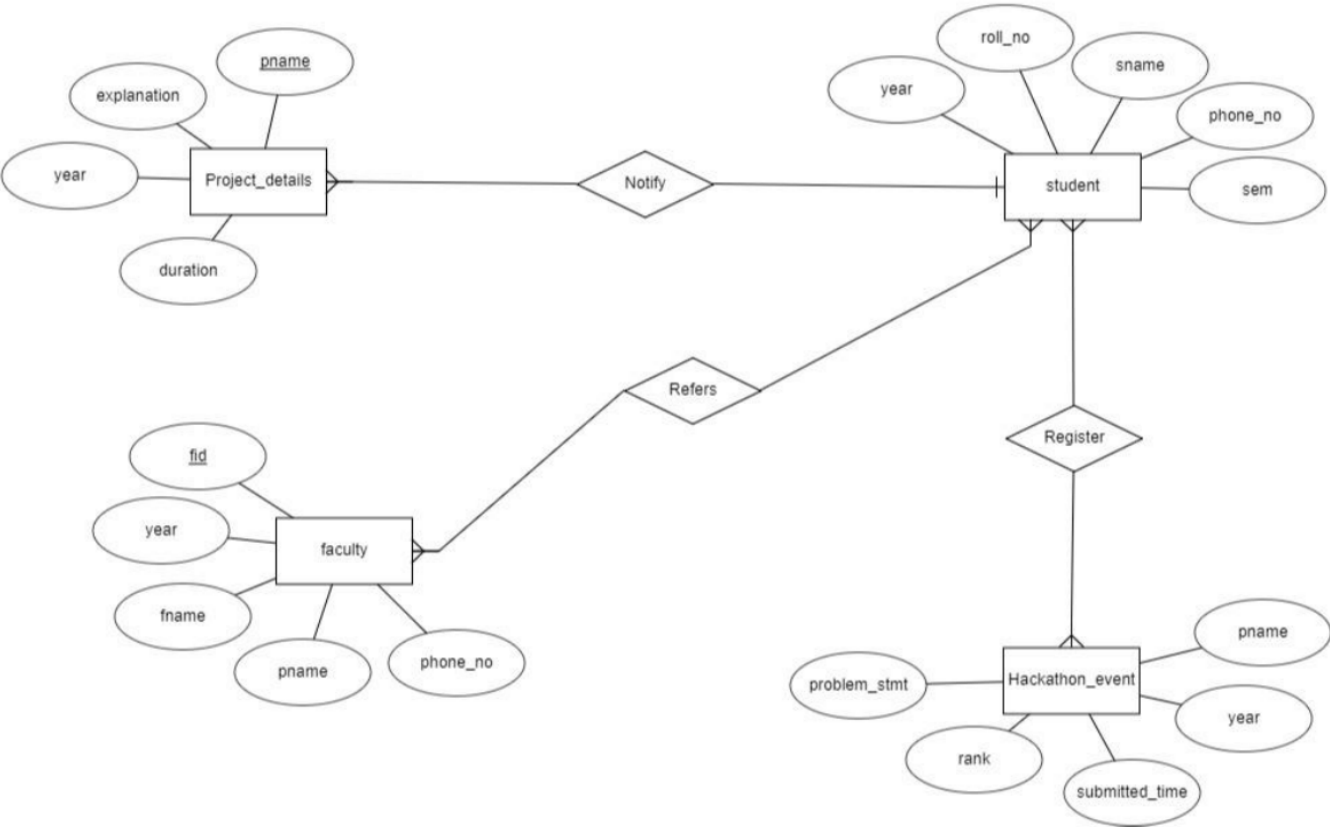
**Java SWINGS** is an API to develop GUI or window-based applications in java. Java SWING components are platform-independent. It is lightweight. The javax.swing package provides classes for SWING API such as JTextField, JLabel, JTextArea, JButton, JChoice, JList etc.

### SQL:

Structure Query Language(SQL) is a database query language used for storing and managing data in Relational DBMS. SQL was the first commercial language introduced for E.F Codd's **Relational** model of database. Today almost all RDBMS (MySQL, Oracle, Infomix, Sybase, MS Access) use **SQL** as the standard database query language. SQL is used to perform all types of data operations in RDBMS.

**DESIGN:**

**ER DIAGRAM:**



**DDL COMMANDS:**

## Run SQL Command Line

```
SQL> spool \Tejaswi\Asn-1.txt;
SQL> create table student_details(
  2  rno varchar2(20),
  3  sname varchar2(30),
  4  year number,
  5  sem number,
  6  phone_no number);
```

Table created.

```
SQL> desc student_details;
```

Name	Null?	Type
RNO		VARCHAR2(20)
SNAME		VARCHAR2(30)
YEAR		NUMBER
SEM		NUMBER
PHONE_NO		NUMBER

```
SQL> alter table student_details add primary key(rno);
```

Table altered.

```
SQL> desc student_details;
```

Name	Null?	Type
RNO	NOT NULL	VARCHAR2(20)
SNAME		VARCHAR2(30)
YEAR		NUMBER
SEM		NUMBER
PHONE_NO		NUMBER

```
SQL>
```

```
SQL> create table project_details(  
  2  pname varchar2(30) primary key,  
  3  duration date,  
  4  year number,  
  5  explanation varchar2(40));
```

Table created.

```
SQL> desc project_details;
```

Name	Null?	Type
PNAME	NOT NULL	VARCHAR2(30)
DURATION		DATE
YEAR		NUMBER
EXPLANATION		VARCHAR2(40)

```
SQL>
```

```
SQL> create table faculty(  
  2  fid varchar2(20) primary key,  
  3  fname varchar2(30),  
  4  year number,  
  5  pname varchar2(30),  
  6  phone_no number,  
  7  constraint fkey_project_name foreign key(pname)  
  8  references project_details(pname)on delete cascade);
```

Table created.

```
SQL> desc faculty;
```

Name	Null?	Type
FID	NOT NULL	VARCHAR2(20)
FNAME		VARCHAR2(30)
YEAR		NUMBER
PNAME		VARCHAR2(30)
PHONE_NO		NUMBER

```
SQL> create table hackathon_event(
  2  pname varchar2(30),
  3  year number,
  4  problem_stmt varchar2(50),
  5  rank number,
  6  submitted_time date,
  7  constraint fkey_proj_name foreign key(pname)
  8  references project_details(pname) on delete cascade);
```

Table created.

```
SQL> desc hackathon_event;
```

Name	Null?	Type
PNAME		VARCHAR2(30)
YEAR		NUMBER
PROBLEM_STMT		VARCHAR2(50)
RANK		NUMBER
SUBMITTED_TIME		DATE

```
SQL>
```

1. create table student\_details(

```
  rno varchar2(20),
  sname varchar2(30),
  year number,
  sem number,
  phone_no number);
```

Setting rno as primary key:

```
alter table student_details add primary key(rno);
```

2. create table project\_details(

```
  pname varchar2(30) primary key,
  duration date,
  year number,
  explanation varchar2(40));
```

3. create table faculty(

```
  fid varchar2(20) primary key,
  fname varchar2(30),
  year number,
  pname varchar2(30),
  phone_no number,
  constraint fkey_project_name foreign key(pname)
  references project_details(pname) on delete cascade);
```

4. create table hackathon\_event(

```
  pname varchar2(30),
  year number,
  problem_stmt varchar2(50),
  rank number,
  submitted_time date,
  constraint fkey_proj_name foreign key(pname)
  references project_details(pname) on delete cascade);
```



## DML COMMANDS:

```
SQL> spool \Tejaswi\As-3.txt;
SQL> insert into student_details values('&rno', '&sname', &year, &sem, &phone_no);
Enter value for rno: S20001
Enter value for sname: Anvitha
Enter value for year: 2021
Enter value for sem: 4
Enter value for phone_no: 7032519223
old 1: insert into student_details values('&rno', '&sname', &year, &sem, &phone_no)
new 1: insert into student_details values('S20001', 'Anvitha', 2021, 4, 7032519223)

1 row created.

SQL> /
Enter value for rno: S21002
Enter value for sname: Sritan
Enter value for year: 2021
Enter value for sem: 2
Enter value for phone_no: 9121567890
old 1: insert into student_details values('&rno', '&sname', &year, &sem, &phone_no)
new 1: insert into student_details values('S21002', 'Sritan', 2021, 2, 9121567890)

1 row created.

SQL> /
Enter value for rno: S20003
Enter value for sname: Poojit
Enter value for year: 2021
Enter value for sem: 4
Enter value for phone_no: 8977623471
old 1: insert into student_details values('&rno', '&sname', &year, &sem, &phone_no)
new 1: insert into student_details values('S20003', 'Poojit', 2021, 4, 8977623471)

1 row created.

SQL> /
Enter value for rno: S21004
Enter value for sname: Sahithi
Enter value for year: 2021
Enter value for sem: 2
Enter value for phone_no: 8019324560
old 1: insert into student_details values('&rno', '&sname', &year, &sem, &phone_no)
new 1: insert into student_details values('S21004', 'Sahithi', 2021, 2, 8019324560)

1 row created.
```

```
SQL> select * from student_details;
```

RNO	SNAME	YEAR	SEM
-----			
PHONE_NO			
-----			
S20001 7032519223	Anvitha	2021	4
S21002 9121567890	Sritan	2021	2
S20003 8977623471	Poojit	2021	4
RNO	SNAME	YEAR	SEM
-----			
PHONE_NO			
-----			
S21004 8019324560	Sahithi	2021	2

```
SQL> insert into project_details values('&pname', '&duration', &year, '&explanation');
```

```
Enter value for pname: Hackathon-1
```

```
Enter value for duration: 31-oct-2020
```

```
Enter value for year: 2020
```

```
Enter value for explanation: topic of developing on mobile apps
```

```
old 1: insert into project_details values('&pname', '&duration', &year, '&explanation')
```

```
new 1: insert into project_details values('Hackathon-1', '31-oct-2020', 2020, 'topic of developing on mobile apps')
```

```
1 row created.
```

```
SQL> /
```

```
Enter value for pname: Hackathon-2
```

```
Enter value for duration: 30-apr-2021
```

```
Enter value for year: 2021
```

```
Enter value for explanation: topic of developing on smart automation
```

```
old 1: insert into project_details values('&pname', '&duration', &year, '&explanation')
```

```
new 1: insert into project_details values('Hackathon-2', '30-apr-2021', 2021, 'topic of developing on smart automation')
```

```
1 row created.
```

```
SQL> █
```

```
SQL> select * from project_details;
```

PNAME	DURATION	YEAR
Hackathon-1	31-OCT-20	2020
developing on mobile apps		
Hackathon-2	30-APR-21	2021
developing on smart automation		

```
SQL> /
```

```
Enter value for fid: F1005
```

```
Enter value for fname: Kavya
```

```
Enter value for year: 2021
```

```
Enter value for pname: Hackathon-1
```

```
Enter value for phone_no: 9212987634
```

```
old 1: insert into faculty values('&fid', '&fname', &year, '&pname', &phone_no)
```

```
new 1: insert into faculty values('F1005', 'Kavya', 2021, 'Hackathon-1', 9212987634)
```

```
1 row created.
```

```
SQL> commit;
```

```
Commit complete.
```

```
SQL> select * from faculty;
```

FID	FNAME	YEAR
PNAME	PHONE_NO	
F1001	Rajendra	2021
Hackathon-1	9010589033	
F1002	Prabhakar	2021
Hackathon-2	9948109871	
F1003	Srinivasulu	2021
Hackathon-1	9618856790	

FID	FNAME	YEAR
PNAME	PHONE_NO	
F1004	Prasanna	2021
Hackathon-2	8349223049	
F1005	Kavya	2021
Hackathon-1	9212987634	

```

SQL> /
Enter value for pname: Hackathon-1
Enter value for year: 2020
Enter value for problem_stmt: mobile app for college website
Enter value for rank: 1
Enter value for submitted_time: 31-oct-2020
old 1: insert into hackathon_event values('&pname', &year, '&problem_stmt', &rank, '&submitted_time')
new 1: insert into hackathon_event values('Hackathon-1', 2020, 'mobile app for college website', 1, '31-oct-2020')

1 row created.

SQL> /
Enter value for pname: Hackathon-2
Enter value for year: 2021
Enter value for problem_stmt: AI based chatbot
Enter value for rank: 1
Enter value for submitted_time: 25-apr-2021
old 1: insert into hackathon_event values('&pname', &year, '&problem_stmt', &rank, '&submitted_time')
new 1: insert into hackathon_event values('Hackathon-2', 2021, 'AI based chatbot', 1, '25-apr-2021')

1 row created.

```

```

SQL>
SQL> /
Enter value for pname: Hackathon-2
Enter value for year: 2021
Enter value for problem_stmt: smart document management system
Enter value for rank: 2
Enter value for submitted_time: 30-apr-2021
old 1: insert into hackathon_event values('&pname', &year, '&problem_stmt', &rank, '&submitted_time')
new 1: insert into hackathon_event values('Hackathon-2', 2021, 'smart document management system', 2, '30-apr-2021')

1 row created.

SQL> commit;

Commit complete.

SQL> select * from hackathon_event;

PNAME                                YEAR
-----
PROBLEM_STMT                                RANK SUBMITTED
-----
Hackathon-1                                2020
aict mobile app                                2 29-OCT-20
Hackathon-1                                2020
mobile app for college website                                1 31-OCT-20
Hackathon-2                                2021
AI based chatbot                                1 25-APR-21

PNAME                                YEAR
-----
PROBLEM_STMT                                RANK SUBMITTED
-----
Hackathon-2                                2021
smart document management system                                2 30-APR-21

```

## IMPLEMENTATION:

### Front end programs and its connectivity

**Java Database Connectivity (JDBC)** is an application programming interface (API) for the programming language Java, which defines how a client may access a database. It is a Java-based data access technology used for Java database connectivity. It is part of the Java Standard Edition platform, from Oracle Corporation. It provides methods to query and update data in a database and is oriented towards relational databases.

The connection to the database can be performed using Java programming (JDBC API) as:

```
public void connectToDB(){
    try {
        Connection con = DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","amruth","vasavi");

    }
    catch (SQLException connectException){
        System.out.println(connectException.getMessage());
        System.out.println(connectException.getSQLState());
        System.out.println(connectException.getErrorCode());    System.exit(1);
    }
}
```

Thus, the connection from Java to Oracle database is performed and therefore, can be used for updating tables in the database directly.

## MAIN METHOD:

```
public class Main {
    public static void main(String[] args) {
        new PortalUI();
    }
}
```

## PortalUI:

```
import java.awt.*;
import javax.swing.*;
import java.awt.Color;
import java.awt.FlowLayout;
import java.awt.Font;
import java.awt.GridLayout;
import java.awt.Label;
import java.awt.event.WindowAdapter;
import java.awt.event.WindowEvent;

import javax.swing.JFrame;
import javax.swing.JLabel;
```

```

import javax.swing.JMenu;
import javax.swing.JMenuBar;
import javax.swing.JMenuItem;
import javax.swing.JOptionPane;
import javax.swing.JPanel;

public class PortalUI extends JFrame{

    private static final long serialVersionUID = 1L;

    private JMenuBar mnu;

    private JMenu mnuProject;
    private JMenu mnuStudents;
    private JMenu mnuFaculty;
    private JMenu mnuEvent;

    private JMenuItem insert1, update1, delete1, view1;
    private JMenuItem insert2, update2, delete2, view2;
    private JMenuItem insert3, update3, delete3, view3;
    private JMenuItem insert4, update4, delete4, view4;

    private JLabel labelName;

    private static JPanel p0, p1;

    public void initialize() {
        mnu = new JMenuBar();
        mnuProject = new JMenu("Project details");
        mnuStudents = new JMenu("Students");
        mnuFaculty = new JMenu("Faculty");
        mnuEvent = new JMenu("Hackathon Event");
        Color lightBlue= new Color(51, 204, 255);

        labelName=new JLabel("SIH IT PORTAL");
        labelName.setFont(new Font("Serif", Font.PLAIN, 40));
        labelName.setForeground(Color.RED);
        p1=new JPanel();
        p0=new JPanel();
        insert1 = new JMenuItem("Insert");
        update1 = new JMenuItem("Update");
        delete1 = new JMenuItem("Delete");
        view1 = new JMenuItem("View");

        insert2 = new JMenuItem("Insert");
        update2 = new JMenuItem("Update");
        delete2 = new JMenuItem("Delete");
        view2 = new JMenuItem("View");

        insert3 = new JMenuItem("Insert");
        update3 = new JMenuItem("Update");
        delete3 = new JMenuItem("Delete");
        view3 = new JMenuItem("View");
    }

```

```

        insert4 = new JMenuItem("Insert");
        update4 = new JMenuItem("Update");
        delete4 = new JMenuItem("Delete");
        view4 = new JMenuItem("View");
    }

    void addComponentsToFrame() {
        mnuProject.add(insert1);
        mnuProject.add(delete1);
        mnuProject.add(update1);
        mnuProject.add(view1);

        mnuStudents.add(insert2);
        mnuStudents.add(delete2);
        mnuStudents.add(update2);
        mnuStudents.add(view2);

        mnuFaculty.add(insert3);
        mnuFaculty.add(delete3);
        mnuFaculty.add(update3);
        mnuFaculty.add(view3);

        mnuEvent.add(insert4);
        mnuEvent.add(delete4);
        mnuEvent.add(update4);
        mnuEvent.add(view4);

        mnu.add(mnuProject);
        mnu.add(mnuStudents);
        mnu.add(mnuFaculty);
        mnu.add(mnuEvent);

        setJMenuBar(mnu);
        p1.add(labelName);
        p1.setAlignmentY(CENTER_ALIGNMENT);
        p1.setBounds(500, 500, 800, 100);
        p0.add(p1);
        p0.setBackground(Color.lightGray);
        add(p0);
    }

    void closeWindow(){
        try {
            int a = JOptionPane.showConfirmDialog(this,"Are you sure want to Quit SIH IT Portal?");
            if(a == JOptionPane.YES_OPTION){
                JOptionPane.showMessageDialog(this, "Thank you!\nExiting SIH IT Portal", "Quit",
                JOptionPane.WARNING_MESSAGE);
                System.exit(0);
            }
            else if (a == JOptionPane.NO_OPTION) {
                setDefaultCloseOperation(JFrame.DO_NOTHING_ON_CLOSE);
            }
            else if (a == JOptionPane.CANCEL_OPTION) {
                setDefaultCloseOperation(JFrame.DO_NOTHING_ON_CLOSE);
            }
        }
        catch(Exception e) {
            System.out.println(e);
        }
    }

```

```

    }
}

void register() {
    Project_details pjd=new Project_details(p0, PortalUI.this, insert1, update1, view1, delete1);
    pjd.buildGUI();
    Students st = new Students(p0, PortalUI.this, insert2, update2, view2, delete2);
    st.buildGUI();
    Faculty fc = new Faculty(p0, PortalUI.this, insert3, update3, view3, delete3);
    fc.buildGUI();
    Hackathon_event hce=new Hackathon_event(p0, PortalUI.this, insert4, update4, view4, delete4);
    hce.buildGUI();

    addWindowListener(new WindowAdapter(){
        public void windowClosing(WindowEvent we)
        {
            closeWindow();
        }
    });
}

public PortalUI() {
    initialize();
    addComponentsToFrame();
    register();
    pack();
    setTitle("SIH IT Portal");
    setSize(800, 800);
    setVisible(true);
}
}

```

## Project\_details:

```

import java.awt.*;
import java.awt.event.*;
import javax.swing.*;
import javax.swing.table.DefaultTableModel;
import java.sql.*;
import java.util.Properties;

public class Project_details{

    private JPanel pn, pn1, pn2, pn3;
    private JFrame jf;
    private JButton JB_insert, JB_modify, JB_view, JB_delete;
    private JLabel JL_pname, JL_expla, JL_duration, JL_year;
    private JTextField JTF_pname, JTF_expla, JTF_duration, JTF_year;

    private JMenuitem insert2, update2, view2, delete2;
    private List ProjectList;
    private Choice Projectid;

    public Project_details(JPanel pn, JFrame jf, JMenuitem insert2, JMenuitem update2, JMenuitem view2,
JMenuitem delete2){

```



```

        try {
            Class.forName("oracle.jdbc.driver.OracleDriver");
        }
        catch (Exception e) {
            System.err.println("Unable to find and load driver");
            System.exit(1);
        }
        this.jf = jf;
        this.insert2 = insert2;
        this.update2 = update2;
        this.view2 = view2;
        this.delete2 = delete2;

        JL_pname = new JLabel("Project Name:");
        JTF_pname = new JTextField(10);
        JL_expla = new JLabel("Explanation:");
        JTF_expla = new JTextField(10);
        JL_duration = new JLabel("Duration:");
        JTF_duration = new JTextField(10);
        JL_year = new JLabel("Year:");
        JTF_year = new JTextField(10);

        this.pn = pn;
    }

    private void displaySQLErrors(SQLException e) {
        JOptionPane.showMessageDialog(pn1, "\nSQLException: " + e.getMessage() + "\n" + "SQLState: " +
e.getSQLState() + "\n" + "VendorError: " + e.getErrorCode() + "\n");
    }
    public void loaddetails(){
        try{
            Projectid = new Choice();
            Projectid.removeAll();
            Connection con1 = DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe",
"it20737055", "teja2002");
            Statement st1 = con1.createStatement();
            ResultSet rs1 = st1.executeQuery("select * from project_details");
            while(rs1.next()) {
                Projectid.add(rs1.getString("pname"));
            }
        }
        catch(SQLException e) {
            displaySQLErrors(e);
        }
    }
    public void loadproject(){
        try{
            ProjectList = new List();
            ProjectList.removeAll();
            Connection con2 = DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe",
"it20737055", "teja2002");
            Statement st2 = con2.createStatement();

```

```

        ResultSet rs2 = st2.executeQuery("select * from project_details");
        while(rs2.next()) {
            ProjectList.add(rs2.getString("pname"));
        }
    }
    catch(SQLException e) {
        displaySQLErrors(e);
    }
}

public void buildGUI(){
    insert2.addActionListener(new ActionListener(){
        @Override
        public void actionPerformed(ActionEvent aevt){

            JB_insert = new JButton("Submit");
            JTF_pname.setText(null);
            JTF_expla.setText(null);
            JTF_duration.setText(null);
            JTF_year.setText(null);

            loadproject();

            pn.removeAll();
            jf.invalidate();
            jf.validate();
            jf.repaint();

            pn1 = new JPanel();
            pn1.setLayout(new GridLayout(10,10));
            pn1.add(JL_pname);
            pn1.add(JTF_pname);
            pn1.add(JL_expla);
            pn1.add(JTF_expla);
            pn1.add(JL_duration);
            pn1.add(JTF_duration);
            pn1.add(JL_year);
            pn1.add(JTF_year);

            pn3 = new JPanel(new FlowLayout());
            pn3.add(JB_insert);
            pn1.setBounds(115, 80, 300, 250);
            pn3.setBounds(200, 350, 75, 35);

            pn2 = new JPanel(new FlowLayout());
            ProjectList = new List(10);
            loadproject();
            pn2.add(ProjectList);
            pn2.setBounds(200, 350, 300, 180);

            pn.add(pn1);
            pn.add(pn3);

```

```

        pn.add(pn2);

        pn.setLayout(new BorderLayout());
        jf.add(pn);
        jf.setSize(800, 800);
        jf.validate();

        JB_insert.addActionListener(new ActionListener() {
            @Override
            public void actionPerformed(ActionEvent aevt){
                try{
                    Connection con =
DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","it20737055","teja2002");
                    String query = "INSERT INTO project_details (pname,
explanation, duration, year) values(?, ?, ?, ?)";
                    PreparedStatement stp1 = con.prepareStatement(query);
                    stp1.setString(1, JTF_pname.getText());
                    stp1.setString(2, JTF_expla.getText());
                    stp1.setString(3, JTF_duration.getText());
                    stp1.setString(4, JTF_year.getText());
                    int i = stp1.executeUpdate();
                    con.close();
                    if(i > 0){
                        JOptionPane.showMessageDialog(pn,"Inserted
successfully");
                    }
                    loadproject();
                }
                catch(SQLException e) {
                    displaySQLErrors(e);
                }
            }
        });

        update2.addActionListener(new ActionListener() {
            @Override
            public void actionPerformed(ActionEvent aevt){
                JB_modify=new JButton("Modify");

                JTF_pname.setText(null);
                JTF_expla.setText(null);
                JTF_duration.setText(null);
                JTF_year.setText(null);

                pn.removeAll();
                jf.invalidate();
                jf.validate();
                jf.repaint();

                pn1=new JPanel();

```

```

pn1.setLayout(new GridLayout(10, 10));

pn1.add(JL_pname);
pn1.add(JTF_pname);
pn1.add(JL_expla);
pn1.add(JTF_expla);
pn1.add(JL_duration);
pn1.add(JTF_duration);
pn1.add(JL_year);
pn1.add(JTF_year);

pn3 = new JPanel(new FlowLayout());
pn3.add(JB_modify);
pn1.setBounds(115, 80, 300, 250);
pn3.setBounds(200, 350, 75, 35);

pn2 =new JPanel(new FlowLayout());
ProjectList = new List(10);
loadproject();
pn2.add(ProjectList);
pn2.setBounds(200, 350, 300, 180);

pn.add(pn1);
pn.add(pn3);
pn.add(pn2);

pn.setLayout(new BorderLayout());
jf.add(pn);
jf.setSize(800, 800);
jf.validate();

ProjectList.addItemListener(new ItemListener() {
    public void itemStateChanged(ItemEvent ievt){
        try {
            Connection con3 =
DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe", "it20737055", "teja2002");
            Statement st3 = con3.createStatement();
            ResultSet rs3 = st3.executeQuery("select * from project_details");
            while (rs3.next()) {
                if
(rs3.getString("pname").equals(ProjectList.getSelectedItem()))
                    break;
            }
            if (!rs3.isAfterLast()) {
                JTF_pname.setText(rs3.getString("pname"));
                JTF_expla.setText(rs3.getString("explanation"));
                JTF_duration.setText(rs3.getString("duration"));
                JTF_year.setText(rs3.getString("year"));
            }
        }
        catch (SQLException selectException) {
            displaySQLErrors(selectException);

```

```

        }
    }
});
JB_modify.addActionListener(new ActionListener() {
    @Override
    public void actionPerformed(ActionEvent aevt){
        try{
            int a = JOptionPane.showConfirmDialog(pn, "Are you sure want
to update:");
            if(a == JOptionPane.YES_OPTION){
                String query = "update project_details set explanation = ?,
year = ?, duration = ? where pname = ?";
                Connection con4 =
DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe", "it20737055", "teja2002");
                PreparedStatement stp2 = con4.prepareStatement(query);
                stp2.setString(1, JTF_expla.getText());
                stp2.setString(2, JTF_year.getText());
                stp2.setString(3, JTF_duration.getText());

                stp2.setString(4, JTF_pname.getText());
                int i = stp2.executeUpdate();
                if(i>0){
                    JOptionPane.showMessageDialog(pn, "\nUpdated
rows succesfully");
                }
                loadproject();
            }
        } catch(SQLException e){
            displaySQLErrors(e);
        }
    }
});
}
});
delete2.addActionListener(new ActionListener() {
    @Override
    public void actionPerformed(ActionEvent aevt){
        JB_delete = new JButton("Delete");

        JTF_pname.setText(null);
        JTF_expla.setText(null);
        JTF_duration.setText(null);
        JTF_year.setText(null);

        pn.removeAll();
        jf.invalidate();
        jf.validate();
        jf.repaint();

        pn1 = new JPanel();
        pn1.setLayout(new GridLayout(10, 10));

```

```

pn1.add(JL_pname);
pn1.add(JTF_pname);
pn1.add(JL_expla);
pn1.add(JTF_expla);
pn1.add(JL_duration);
pn1.add(JTF_duration);
pn1.add(JL_year);
pn1.add(JTF_year);

```

```

pn3 = new JPanel(new FlowLayout());
pn3.add(JB_delete);
pn1.setBounds(115, 80, 300, 250);
pn3.setBounds(200, 350, 75, 35);

```

```

pn2 = new JPanel(new FlowLayout());
ProjectList = new List(10);
loadproject();
pn2.add(ProjectList);
pn2.setBounds(200, 350, 300, 200);

```

```

pn.add(pn1);
pn.add(pn3);
pn.add(pn2);

```

```

pn.setLayout(new BorderLayout());
jf.add(pn);
jf.setSize(800, 800);
jf.validate();

```

```

ProjectList.addItemListener(new ItemListener() {
    public void itemStateChanged(ItemEvent ievt){
        try {

```

```

            Connection con5 =
DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe", "it20737055", "teja2002");
            Statement st4 = con5.createStatement();
            ResultSet rs4 = st4.executeQuery("select * from project_details");
            while (rs4.next()) {
                if
(rs4.getString("pname").equals(ProjectList.getSelectedItem()))
                    break;
            }
            if (!rs4.isAfterLast()) {
                JTF_pname.setText(rs4.getString("pname"));
                JTF_expla.setText(rs4.getString("explanation"));
                JTF_duration.setText(rs4.getString("duration"));
                JTF_year.setText(rs4.getString("year"));
            }
        }
    } catch (SQLException selectException) {
        displaySQLErrors(selectException);
    }
}

```

```

        }
    }
});
JB_delete.addActionListener(new ActionListener() {
    @Override
    public void actionPerformed(ActionEvent aevt){
        try {
            int a = JOptionPane.showConfirmDialog(pn,"Are you sure want to
Delete:");
            if(a == JOptionPane.YES_OPTION){
                String query = "DELETE FROM project_details WHERE
pname = ?";
                Connection con6 =
DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe", "it20737055", "teja2002");
                PreparedStatement stp3 = con6.prepareStatement(query);
                stp3.setString(1, JTF_pname.getText());

                int i = stp3.executeUpdate();
                if(i > 0){
                    JOptionPane.showMessageDialog(pn,"nDeleted
rows succesfully");
                }
                loadproject();
            }
        }
        catch(SQLException e){
            displaySQLErrors(e);
        }
    }
});
}

});
view2.addActionListener(new ActionListener() {
    @Override
    public void actionPerformed(ActionEvent aevt){
        pn.removeAll();
        jf.invalidate();
        jf.validate();
        jf.repaint();

        JLabel view = new JLabel("Project details View");
        view.setForeground(Color.RED);
        JB_view = new JButton("View");
        Font myFont = new Font("Serif",Font.BOLD,25);
        view.setFont((myFont));

        pn1 = new JPanel();
        pn2 = new JPanel();
        pn1.add(view);
        pn2.add(JB_view);
        pn.add(pn1);
        pn.add(pn2);
    }
});

```

```

        pn.setBounds(500, 800, 300, 300);
        pn.setLayout(new FlowLayout());

        jf.add(pn);
        jf.setSize(800, 800);
        jf.validate();

        JB_view.addActionListener(new ActionListener() {
            @Override
            public void actionPerformed(ActionEvent aevt){
                JFrame jfr = new JFrame("Project Details");
                JTable jt;
                DefaultTableModel model = new DefaultTableModel();
                jt = new JTable(model);
                model.addColumn("pname");
                model.addColumn("explanation");
                model.addColumn("duration");
                model.addColumn("year");
                try {
                    Connection con7 =
DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe", "it20737055", "teja2002");
                    Statement st4 = con7.createStatement();
                    ResultSet rs5 = st4.executeQuery("select * from project_details");
                    while(rs5.next()){
                        model.addRow(new Object[]{rs5.getString("pname"),
rs5.getString("explanation"), rs5.getString("duration"), rs5.getString("year")});
                    }
                }
                catch(SQLException e){
                    displaySQLErrors(e);
                }
                jt.setEnabled(false);
                jt.setBounds(30, 40, 300, 300);
                JScrollPane jsp = new JScrollPane(jt);
                jfr.add(jsp);
                jfr.setSize(800, 400);
                jfr.setVisible(true);
            }
        });
    }
}

```

## Students:

```

import java.awt.*;
import java.awt.event.*;
import javax.swing.*;
import javax.swing.table.DefaultTableModel;

```



```

import java.sql.*;
import java.util.Properties;

public class Students{

    private JPanel pn, pn1, pn2, pn3;
    private JFrame jf;
    private JButton JB_insert, JB_modify, JB_view, JB_delete;
    private JLabel JL_rno, JL_sname, JL_sem, JL_year, JL_phone;
    private JTextField JTF_rno, JTF_sname, JTF_sem, JTF_year, JTF_phone;

    private JMenuItem insert2, update2, view2, delete2;
    private List ProjectList;
    private Choice Projectid;

    public Students(JPanel pn, JFrame jf, JMenuItem insert2, JMenuItem update2, JMenuItem view2, JMenuItem delete2){
        try {
            Class.forName("oracle.jdbc.driver.OracleDriver");
        }
        catch (Exception e) {
            System.err.println("Unable to find and load driver");
            System.exit(1);
        }
        this.jf = jf;
        this.insert2 = insert2;
        this.update2 = update2;
        this.view2 = view2;
        this.delete2 = delete2;

        JL_rno = new JLabel("Roll number: ");
        JTF_rno = new JTextField(10);
        JL_sname = new JLabel("Student name: ");
        JTF_sname = new JTextField(10);
        JL_sem = new JLabel("Semester: ");
        JTF_sem = new JTextField(10);
        JL_year = new JLabel("Year: ");
        JTF_year = new JTextField(10);
        JL_phone = new JLabel("Phone number: ");
        JTF_phone = new JTextField(10);

        this.pn=pn;
    }

    private void displaySQLExceptions(SQLException e) {
        JOptionPane.showMessageDialog(pn1,"nSQLException: " + e.getMessage() + "n"+"SQLState: " + e.getSQLState() + "n"+"VendorError: " + e.getErrorCode() + "n");
    }

    public void loaddetails(){
        try{
            Projectid = new Choice();

```

```

        Projectid.removeAll();
        Connection con1 = DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe",
"it20737055", "teja2002");
        Statement st1 = con1.createStatement();
        ResultSet rs1 = st1.executeQuery("select * from student_details");
        while(rs1.next()) {
            Projectid.add(rs1.getString("rno"));
        }
    }
    catch(SQLException e) {
        displaySQLErrors(e);
    }
}

public void loadproject(){
    try{
        ProjectList = new List();
        ProjectList.removeAll();
        Connection con2 = DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe",
"it20737055", "teja2002");
        Statement st2 = con2.createStatement();
        ResultSet rs2 = st2.executeQuery("select * from student_details");
        while(rs2.next()) {
            ProjectList.add(rs2.getString("rno"));
        }
    }
    catch(SQLException e) {
        displaySQLErrors(e);
    }
}

public void buildGUI(){
    insert2.addActionListener(new ActionListener(){
        @Override
        public void actionPerformed(ActionEvent aevt){

            JB_insert = new JButton("Submit");
            JTF_rno.setText(null);
            JTF_sname.setText(null);
            JTF_sem.setText(null);
            JTF_year.setText(null);
            JTF_phone.setText(null);

            loadproject();

            pn.removeAll();
            jf.invalidate();
            jf.validate();
            jf.repaint();

            pn1 = new JPanel();
            pn1.setLayout(new GridLayout(10,10));
            pn1.add(JL_rno);

```

```

pn1.add(JTF_rno);
pn1.add(JL_sname);
pn1.add(JTF_sname);
pn1.add(JL_sem);
pn1.add(JTF_sem);
pn1.add(JL_year);
pn1.add(JTF_year);
pn1.add(JL_phone);
pn1.add(JTF_phone);

```

```

pn3 = new JPanel(new FlowLayout());
pn3.add(JB_insert);
pn1.setBounds(115, 80, 300, 250);
pn3.setBounds(200, 350, 75, 35);

```

```

pn2 = new JPanel(new FlowLayout());
ProjectList = new List(10);
loadproject();
pn2.add(ProjectList);
pn2.setBounds(200, 350, 300, 180);

```

```

pn.add(pn1);
pn.add(pn3);
pn.add(pn2);

```

```

pn.setLayout(new BorderLayout());
jf.add(pn);
jf.setSize(800, 800);
jf.validate();

```

```

JB_insert.addActionListener(new ActionListener() {
    @Override
    public void actionPerformed(ActionEvent aevt){
        try{

```

```

            Connection con =
DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","it20737055","teja2002");
            String query = "INSERT INTO Student_details (rno, sname, sem,
year, phone_no) values(?, ?, ?, ?, ?)";

```

```

            PreparedStatement stp = con.prepareStatement(query);
stp.setString(1, JTF_rno.getText());
stp.setString(2, JTF_sname.getText());
stp.setString(3, JTF_sem.getText());
stp.setString(4, JTF_year.getText());
stp.setString(5, JTF_phone.getText());
int i = stp.executeUpdate();
con.close();
if(i > 0){
                JOptionPane.showMessageDialog(pn,"Inserted

```

```

successfully");

```

```

        }

```

```

        }
        catch(SQLException e) {
            displaySQLErrors(e);
        }
    }
});
}
});
update2.addActionListener(new ActionListener() {
    @Override
    public void actionPerformed(ActionEvent aevt){
        JB_modify = new JButton("Modify");

        JTF_rno.setText(null);
        JTF_sname.setText(null);
        JTF_sem.setText(null);
        JTF_year.setText(null);
        JTF_phone.setText(null);

        pn.removeAll();
        jf.invalidate();
        jf.validate();
        jf.repaint();

        pn1 = new JPanel();
        pn1.setLayout(new GridLayout(10, 10));

        pn1.add(JL_rno);
        pn1.add(JTF_rno);
        pn1.add(JL_sname);
        pn1.add(JTF_sname);
        pn1.add(JL_sem);
        pn1.add(JTF_sem);
        pn1.add(JL_year);
        pn1.add(JTF_year);
        pn1.add(JL_phone);
        pn1.add(JTF_phone);

        pn3 = new JPanel(new FlowLayout());
        pn3.add(JB_modify);
        pn1.setBounds(115, 80, 300, 250);
        pn3.setBounds(200, 350, 75, 35);

        pn2 =new JPanel(new FlowLayout());
        ProjectList = new List(10);
        loadproject();
        pn2.add(ProjectList);
        pn2.setBounds(200, 350, 300, 180);

        pn.add(pn1);

```

```

        pn.add(pn3);
        pn.add(pn2);

        pn.setLayout(new BorderLayout());
        jf.add(pn);
        jf.setSize(800, 800);
        jf.validate();

        ProjectList.addItemListener(new ItemListener() {
            public void itemStateChanged(ItemEvent ievt){
                try {
                    Connection con3 =
DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe", "it20737055", "teja2002");
                    Statement st3 = con3.createStatement();
                    ResultSet rs3 = st3.executeQuery("select * from
Student_details");

                    while (rs3.next()) {
                        if
(rs3.getString("rno").equals(ProjectList.getSelectedItem()))
                            break;
                    }
                    if (!rs3.isAfterLast()) {
                        JTF_rno.setText(rs3.getString("rno"));
                        JTF_sname.setText(rs3.getString("sname"));
                        JTF_sem.setText(rs3.getString("sem"));
                        JTF_year.setText(rs3.getString("year"));
                        JTF_phone.setText(rs3.getString("phone_no"));
                    }
                }
                catch (SQLException selectException) {
                    displaySQLErrors(selectException);
                }
            }
        });
        JB_modify.addActionListener(new ActionListener() {
            @Override
            public void actionPerformed(ActionEvent aevt){
                try{
                    int a = JOptionPane.showConfirmDialog(pn, "Are you sure want
to update:");

                    if(a == JOptionPane.YES_OPTION){
                        String query = "update student_details set sname = ?,
year = ?, sem = ?, phone_no = ? where rno = ?";

                        Connection con4 =
DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe", "it20737055", "teja2002");
                        PreparedStatement stp2 = con4.prepareStatement(query);
                        stp2.setString(1, JTF_sname.getText());
                        stp2.setString(2, JTF_year.getText());
                        stp2.setString(3, JTF_sem.getText());
                        stp2.setString(4, JTF_phone.getText());
                        stp2.setString(5, JTF_rno.getText());
                        int i = stp2.executeUpdate();
                    }
                }
            }
        });
    }
}

```

```

rows succesfully");

        if(i>0){
            JOptionPane.showMessageDialog(pn,"\nUpdated
        }
        loadproject();
    }
    }
    catch(SQLException e){
        displaySQLErrors(e);
    }
}

});

}

});
delete2.addActionListener(new ActionListener() {
    @Override
    public void actionPerformed(ActionEvent aevt){
        JB_delete = new JButton("Delete");

        JTF_rno.setText(null);
        JTF_sname.setText(null);
        JTF_sem.setText(null);
        JTF_year.setText(null);
        JTF_phone.setText(null);

        pn.removeAll();
        jf.invalidate();
        jf.validate();
        jf.repaint();

        pn1 = new JPanel();
        pn1.setLayout(new GridLayout(10, 10));
        pn1.add(JL_rno);
        pn1.add(JTF_rno);
        pn1.add(JL_sname);
        pn1.add(JTF_sname);
        pn1.add(JL_sem);
        pn1.add(JTF_sem);
        pn1.add(JL_year);
        pn1.add(JTF_year);
        pn1.add(JL_phone);
        pn1.add(JTF_phone);

        pn3 = new JPanel(new FlowLayout());
        pn3.add(JB_delete);
        pn1.setBounds(115, 80, 300, 250);
        pn3.setBounds(200, 350, 75, 35);

        pn2 = new JPanel(new FlowLayout());
        ProjectList = new List(10);
        loadproject();

```

```

pn2.add(ProjectList);
pn2.setBounds(200, 350, 300, 200);

pn.add(pn1);
pn.add(pn3);
pn.add(pn2);

pn.setLayout(new BorderLayout());
jf.add(pn);
jf.setSize(800, 800);
jf.validate();

ProjectList.addItemListener(new ItemListener() {
    public void itemStateChanged(ItemEvent ievt){
        try {
            Connection con5 =
DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe", "it20737055", "teja2002");
            Statement st4 = con5.createStatement();
            ResultSet rs4 = st4.executeQuery("select * from student_details");
            while (rs4.next()) {
                if
(rs4.getString("rno").equals(ProjectList.getSelectedItem()))
                    break;
            }
            if (!rs4.isAfterLast()) {
                JTF_rno.setText(rs4.getString("rno"));
                JTF_sname.setText(rs4.getString("sname"));
                JTF_sem.setText(rs4.getString("sem"));
                JTF_year.setText(rs4.getString("year"));
                JTF_phone.setText(rs4.getString("phone_no"));
            }
        }
        catch (SQLException selectException) {
            displaySQLErrors(selectException);
        }
    }
});
JB_delete.addActionListener(new ActionListener() {
    @Override
    public void actionPerformed(ActionEvent aevt){
        try {
            int a = JOptionPane.showConfirmDialog(pn,"Are you sure want to
Delete:");
            if(a == JOptionPane.YES_OPTION){
                String query = "DELETE FROM Student_details WHERE
rno = ?";
                Connection con6 =
DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe", "it20737055", "teja2002");
                PreparedStatement stp3 = con6.prepareStatement(query);
                stp3.setString(1, JTF_rno.getText());
                int i = stp3.executeUpdate();

```

```

rows succesfully");
                                if(i>0){
                                    JOptionPane.showMessageDialog(pn,"\nDeleted
                                }
                                loadproject();
                            }
                        }
                    catch(SQLException e){
                        displaySQLErrors(e);
                    }
                }
            });
        }
    });
    view2.addActionListener(new ActionListener() {
        @Override
        public void actionPerformed(ActionEvent aevt){
            pn.removeAll();
            jf.invalidate();
            jf.validate();
            jf.repaint();

            JLabel view = new JLabel("Student details View");
            view.setForeground(Color.RED);
            JB_view = new JButton("View");
            Font myFont = new Font("Serif",Font.BOLD,25);
            view.setFont((myFont));

            pn1 = new JPanel();
            pn2 = new JPanel();
            pn1.add(view);
            pn2.add(JB_view);
            pn.add(pn1);
            pn.add(pn2);
            pn.setBounds(500, 800, 300, 300);
            pn.setLayout(new FlowLayout());

            jf.add(pn);
            jf.setSize(800, 800);
            jf.validate();

            JB_view.addActionListener(new ActionListener() {
                @Override
                public void actionPerformed(ActionEvent aevt){
                    JFrame jfr = new JFrame("Student Details");
                    JTable jt;
                    DefaultTableModel model = new DefaultTableModel();
                    jt = new JTable(model);
                    model.addColumn("Roll Number");
                    model.addColumn("Student Name");
                    model.addColumn("Semester");
                    model.addColumn("Year");
                }
            });
        }
    });
}

```



```

        model.addColumn("Phone number");
    try {
        Connection con7 =
DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe", "it20737055", "teja2002");
        Statement st4 = con7.createStatement();
        ResultSet rs5 = st4.executeQuery("select * from student_details");

        while(rs5.next()){
            model.addRow(new Object[]{rs5.getString("rno"),
rs5.getString("sname"), rs5.getString("sem"), rs5.getString("year"), rs5.getString("phone_no")});
        }
    }
    catch(SQLException e){
        displaySQLErrors(e);
    }
    jt.setEnabled(false);
    jt.setBounds(30, 40, 300, 300);
    JScrollPane jsp = new JScrollPane(jt);
    jfr.add(jsp);
    jfr.setSize(800, 400);
    jfr.setVisible(true);
    }
    });
    }
}

```

## Faculty:

```

import java.awt.*;
import java.awt.event.*;
import javax.swing.*;
import javax.swing.table.DefaultTableModel;
import java.sql.*;
import java.util.Properties;

public class Faculty{

    private JPanel pn, pn1, pn2, pn3;
    private JFrame jf;
    private JButton JB_insert, JB_modify, JB_view, JB_delete;
    private JLabel JL_fid, JL_fname, JL_pname, JL_year, JL_phone;
    private JTextField JTF_fid, JTF_fname, JTF_pname, JTF_year, JTF_phone;

    private JMenuItem insert2, update2, view2, delete2;
    private List ProjectList;
    private Choice Projectid;

    public Faculty(JPanel pn, JFrame jf, JMenuItem insert2, JMenuItem update2, JMenuItem view2, JMenuItem
delete2){
        try {

```

```

        Class.forName("oracle.jdbc.driver.OracleDriver");
    }
    catch (Exception e) {
        System.err.println("Unable to find and load driver");
        System.exit(1);
    }
    this.jf = jf;
    this.insert2 = insert2;
    this.update2 = update2;
    this.view2 = view2;
    this.delete2 = delete2;

    JL_fid = new JLabel("Faculty ID: ");
    JTF_fid = new JTextField(10);
    JL_fname = new JLabel("Faculty name: ");
    JTF_fname = new JTextField(10);
    JL_pname = new JLabel("Project name: ");
    JTF_pname = new JTextField(10);
    JL_year = new JLabel("Year: ");
    JTF_year = new JTextField(10);
    JL_phone = new JLabel("Phone number: ");
    JTF_phone = new JTextField(10);

    this.pn=pn;
}
private void displaySQLErrors(SQLException e) {
    JOptionPane.showMessageDialog(pn1, "\nSQLException: " + e.getMessage() + "\n" + "SQLState: " +
e.getSQLState() + "\n" + "VendorError: " + e.getErrorCode() + "\n");
}
public void loaddetails(){
    try{
        Projectid = new Choice();
        Projectid.removeAll();
        Connection con1 = DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe",
"it20737055", "teja2002");
        Statement st1 = con1.createStatement();
        ResultSet rs1 = st1.executeQuery("select * from faculty");
        while(rs1.next()) {
            Projectid.add(rs1.getString("fid"));
        }
    }
    catch(SQLException e) {
        displaySQLErrors(e);
    }
}

public void loadproject(){
    try{
        ProjectList = new List();
        ProjectList.removeAll();
        Connection con2 = DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe",
"it20737055", "teja2002");
    }
}

```

```

        Statement st2 = con2.createStatement();
        ResultSet rs2 = st2.executeQuery("select * from faculty");
        while(rs2.next()) {
            ProjectList.add(rs2.getString("fid"));
        }
    }
    catch(SQLException e) {
        displaySQLErrors(e);
    }
}

public void buildGUI(){
    insert2.addActionListener(new ActionListener(){
        @Override
        public void actionPerformed(ActionEvent aevt){

            JB_insert = new JButton("Submit");
            JTF_fid.setText(null);
            JTF_fname.setText(null);
            JTF_pname.setText(null);
            JTF_year.setText(null);
            JTF_phone.setText(null);

            loadproject();

            pn.removeAll();
            jf.invalidate();
            jf.validate();
            jf.repaint();

            pn1 = new JPanel();
            pn1.setLayout(new GridLayout(10, 10));
            pn1.add(JL_fid);
            pn1.add(JTF_fid);
            pn1.add(JL_fname);
            pn1.add(JTF_fname);
            pn1.add(JL_pname);
            pn1.add(JTF_pname);
            pn1.add(JL_year);
            pn1.add(JTF_year);
            pn1.add(JL_phone);
            pn1.add(JTF_phone);

            pn3 = new JPanel(new FlowLayout());
            pn3.add(JB_insert);
            pn1.setBounds(115, 80, 300, 250);
            pn3.setBounds(200, 350, 75, 35);

            pn2 = new JPanel(new FlowLayout());
            ProjectList = new List(10);

```

```

loadproject();
pn2.add(ProjectList);
pn2.setBounds(200, 350, 300, 180);

pn.add(pn1);
pn.add(pn3);
pn.add(pn2);

pn.setLayout(new BorderLayout());
jf.add(pn);
jf.setSize(800, 800);
jf.validate();

JB_insert.addActionListener(new ActionListener() {
    @Override
    public void actionPerformed(ActionEvent aevt){

        try{
            Connection con =
DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","it20737055","teja2002");
            String query = "INSERT INTO faculty (fid, fname, pname, year,
phone_no) values(?, ?, ?, ?, ?)";

            PreparedStatement stp = con.prepareStatement(query);
            stp.setString(1, JTF_fid.getText());
            stp.setString(2, JTF_fname.getText());
            stp.setString(3, JTF_pname.getText());
            stp.setString(4, JTF_year.getText());
            stp.setString(5, JTF_phone.getText());
            int i = stp.executeUpdate();
            con.close();
            if(i > 0){
                JOptionPane.showMessageDialog(pn,"Inserted
successfully");
            }
        }
        catch(SQLException e) {
            displaySQLErrors(e);
        }
    }
});

}

});
update2.addActionListener(new ActionListener() {
    @Override
    public void actionPerformed(ActionEvent aevt){
        JB_modify = new JButton("Modify");

        JTF_fid.setText(null);
        JTF_fname.setText(null);
        JTF_pname.setText(null);
        JTF_year.setText(null);
        JTF_phone.setText(null);

```

```
pn.removeAll();
jf.invalidate();
jf.validate();
jf.repaint();
```

```
pn1 = new JPanel();
pn1.setLayout(new GridLayout(10, 10));
```

```
pn1.add(JL_fid);
pn1.add(JTF_fid);
pn1.add(JL_fname);
pn1.add(JTF_fname);
pn1.add(JL_pname);
pn1.add(JTF_pname);
pn1.add(JL_year);
pn1.add(JTF_year);
pn1.add(JL_phone);
pn1.add(JTF_phone);
```

```
pn3 = new JPanel(new FlowLayout());
pn3.add(JB_modify);
pn1.setBounds(115, 80, 300, 250);
pn3.setBounds(200, 350, 75, 35);
```

```
pn2 = new JPanel(new FlowLayout());
ProjectList = new List(10);
loadproject();
pn2.add(ProjectList);
pn2.setBounds(200, 350, 300, 180);
```

```
pn.add(pn1);
pn.add(pn3);
pn.add(pn2);
```

```
pn.setLayout(new BorderLayout());
jf.add(pn);
jf.setSize(800, 800);
jf.validate();
```

```
ProjectList.addItemListener(new ItemListener() {
    public void itemStateChanged(ItemEvent ievt){
        try {
```

```
            Connection con3 =
DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe", "it20737055", "teja2002");
            Statement st3 = con3.createStatement();
            ResultSet rs3 = st3.executeQuery("select * from faculty");
            while (rs3.next()) {
                if
(rs3.getString("fid").equals(ProjectList.getSelectedItem()))
```

```

        break;
    }
    if (!rs3.isAfterLast()) {
        JTF_fid.setText(rs3.getString("fid"));
        JTF_fname.setText(rs3.getString("fname"));
        JTF_pname.setText(rs3.getString("pname"));
        JTF_year.setText(rs3.getString("year"));
        JTF_phone.setText(rs3.getString("phone_no"));
    }
}
catch (SQLException selectException) {
    displaySQLErrors(selectException);
}
}
});
JB_modify.addActionListener(new ActionListener() {
    @Override
    public void actionPerformed(ActionEvent aevt){
        try{
            int a = JOptionPane.showConfirmDialog(pn, "Are you sure want
to update:");

            if(a == JOptionPane.YES_OPTION){
                String query = "update faculty set fname = ?, pname = ?,
year = ?, phone_no = ? where fid = ?";

                Connection con4 =
DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe", "it20737055", "teja2002");
                PreparedStatement stp2 = con4.prepareStatement(query);
                stp2.setString(1, JTF_fname.getText());
                stp2.setString(2, JTF_pname.getText());
                stp2.setString(3, JTF_year.getText());
                stp2.setString(4, JTF_phone.getText());
                stp2.setString(5, JTF_fid.getText());
                int i = stp2.executeUpdate();{
                    JOptionPane.showMessageDialog(pn, "\nUpdated
rows succesfully");
                }
            }
            loadproject();
        }
    }
}
catch(SQLException e){
    displaySQLErrors(e);
}
}
});
}
});
delete2.addActionListener(new ActionListener() {
    @Override
    public void actionPerformed(ActionEvent aevt){
        JB_delete = new JButton("Delete");

        JTF_fid.setText(null);

```

```
JTF_fname.setText(null);
JTF_pname.setText(null);
JTF_year.setText(null);
JTF_phone.setText(null);
```

```
pn.removeAll();
jf.invalidate();
jf.validate();
jf.repaint();
```

```
pn1 = new JPanel();
pn1.setLayout(new GridLayout(10, 10));
pn1.add(JL_fid);
pn1.add(JTF_fid);
pn1.add(JL_fname);
pn1.add(JTF_fname);
pn1.add(JL_pname);
pn1.add(JTF_pname);
pn1.add(JL_year);
pn1.add(JTF_year);
pn1.add(JL_phone);
pn1.add(JTF_phone);
```

```
pn3 = new JPanel(new FlowLayout());
pn3.add(JB_delete);
pn1.setBounds(115, 80, 300, 250);
pn3.setBounds(200, 350, 75, 35);
```

```
pn2 = new JPanel(new FlowLayout());
ProjectList = new List(10);
loadproject();
pn2.add(ProjectList);
pn2.setBounds(200, 350, 300, 200);
```

```
pn.add(pn1);
pn.add(pn3);
pn.add(pn2);
```

```
pn.setLayout(new BorderLayout());
jf.add(pn);
jf.setSize(800, 800);
jf.validate();
```

```
ProjectList.addItemListener(new ItemListener() {
    public void itemStateChanged(ItemEvent ievt){
        try {
```

```
            Connection con5 =
DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe", "it20737055", "teja2002");
            Statement st4 = con5.createStatement();
            ResultSet rs4 = st4.executeQuery("select * from faculty");
            while (rs4.next()) {
```

```

        if
(rs4.getString("fid").equals(ProjectList.getSelectedItem()))
        break;
    }
    if (!rs4.isAfterLast()) {
        JTF_fid.setText(rs4.getString("fid"));
        JTF_fname.setText(rs4.getString("fname"));
        JTF_pname.setText(rs4.getString("pname"));
        JTF_year.setText(rs4.getString("year"));
        JTF_phone.setText(rs4.getString("phone_no"));
    }
}
catch (SQLException selectException) {
    displaySQLErrors(selectException);
}
}
});
JB_delete.addActionListener(new ActionListener() {
    @Override
    public void actionPerformed(ActionEvent aevt){
        try {
            int a = JOptionPane.showConfirmDialog(pn, "Are you sure want
to Delete:");

            if(a == JOptionPane.YES_OPTION){
                String query = "DELETE FROM faculty WHERE fid = ? ";
                Connection con6 =
DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe", "it20737055", "teja2002");
                PreparedStatement stp3 = con6.prepareStatement(query);
                stp3.setString(1, JTF_fid.getText());
                int i = stp3.executeUpdate(query);
                if(i>0){
                    JOptionPane.showMessageDialog(pn,"nDeleted
rows succesfully");
                }
                loadproject();
            }
        }
        catch(SQLException e){
            displaySQLErrors(e);
        }
    }
});
}
});
view2.addActionListener(new ActionListener() {
    @Override
    public void actionPerformed(ActionEvent aevt){
        pn.removeAll();
        jf.invalidate();
        jf.validate();
        jf.repaint();
    }
}

```



```

JLabel view = new JLabel("Faculty details View");
view.setForeground(Color.RED);
JB_view = new JButton("View");
Font myFont = new Font("Serif", Font.BOLD, 25);
view.setFont(myFont);

pn1 = new JPanel();
pn2 = new JPanel();
pn1.add(view);
pn2.add(JB_view);
pn.add(pn1);
pn.add(pn2);
pn.setBounds(500, 800, 300, 300);
pn.setLayout(new FlowLayout());

jf.add(pn);
jf.setSize(800, 800);
jf.validate();

JB_view.addActionListener(new ActionListener() {
    @Override
    public void actionPerformed(ActionEvent ae) {
        JFrame jfr = new JFrame("Student Details");
        JTable jt;
        DefaultTableModel model = new DefaultTableModel();
        jt = new JTable(model);
        model.addColumn("Faculty ID");
        model.addColumn("Faculty Name");
        model.addColumn("Project name");
        model.addColumn("Year");
        model.addColumn("Phone number");
        try {
            Connection con7 =
DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe", "it20737055", "teja2002");
            Statement st4 = con7.createStatement();
            ResultSet rs5 = st4.executeQuery("select * from faculty");

            while(rs5.next()){
                model.addRow(new Object[]{rs5.getString("fid"),
rs5.getString("fname"), rs5.getString("pname"), rs5.getString("year"), rs5.getString("phone_no")});
            }
        } catch (SQLException e) {
            displaySQLErrors(e);
        }
        jt.setEnabled(false);
        jt.setBounds(30, 40, 300, 300);
        JScrollPane jsp = new JScrollPane(jt);
        jfr.add(jsp);
        jfr.setSize(800, 400);
        jfr.setVisible(true);
    }
});

```

```

    }
    });
}
});
}}

```

## Hackathon\_event:

```

import java.awt.*;
import java.awt.event.*;
import javax.swing.*;
import javax.swing.table.DefaultTableModel;
import java.sql.*;
import java.util.Properties;

```

```

public class Hackathon_event{

```

```

    private JPanel pn, pn1, pn2, pn3;
    private JFrame jf;
    private JButton JB_insert, JB_modify, JB_view, JB_delete;
    private JLabel JL_pname, JL_sub_time, JL_rank, JL_year, JL_stmt;
    private JTextField JTF_pname, JTF_sub_time, JTF_rank, JTF_year, JTF_stmt;

```

```

    private JMenuItem insert2, update2, view2, delete2;
    private List ProjectList;
    private Choice Projectid;

```

```

    public Hackathon_event(JPanel pn, JFrame jf, JMenuItem insert2, JMenuItem update2, JMenuItem view2,
JMenuItem delete2){
        try {
            Class.forName("oracle.jdbc.driver.OracleDriver");
        }
        catch (Exception e) {
            System.err.println("Unable to find and load driver");
            System.exit(1);
        }
        this.jf = jf;
        this.insert2 = insert2;
        this.update2 = update2;
        this.view2 = view2;
        this.delete2 = delete2;

        JL_pname = new JLabel("Project name: ");
        JTF_pname = new JTextField(10);
        JL_sub_time = new JLabel("Submitted time: ");
        JTF_sub_time = new JTextField(10);
        JL_rank = new JLabel("Rank: ");
        JTF_rank = new JTextField(10);
        JL_year = new JLabel("Year: ");
        JTF_year = new JTextField(10);
        JL_stmt = new JLabel("Problem statement: ");
        JTF_stmt = new JTextField(10);

```

```

        this.pn=pn;
    }
    private void displaySQLErrors(SQLException e) {
        JOptionPane.showMessageDialog(pn1,"SQLException: " + e.getMessage() + "\n"+"SQLState: " +
e.getSQLState() + "\n"+"VendorError: " + e.getErrorCode() + "\n");
    }
    public void loadaddetails(){
        try{
            Projectid = new Choice();
            Projectid.removeAll();
            Connection con1 = DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe",
"it20737055", "teja2002");
            Statement st1 = con1.createStatement();
            ResultSet rs1 = st1.executeQuery("select * from hackathon_event");
            while(rs1.next()) {
                Projectid.add(rs1.getString("pname"));
            }
        }
        catch(SQLException e) {
            displaySQLErrors(e);
        }
    }

    public void loadproject(){
        try{
            ProjectList = new List();
            ProjectList.removeAll();
            Connection con2 = DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe",
"it20737055", "teja2002");
            Statement st2 = con2.createStatement();
            ResultSet rs2 = st2.executeQuery("select * from hackathon_event");
            while(rs2.next()) {
                ProjectList.add(rs2.getString("pname"));
            }
        }
        catch(SQLException e) {
            displaySQLErrors(e);
        }
    }

    public void buildGUI(){
        insert2.addActionListener(new ActionListener(){
            @Override
            public void actionPerformed(ActionEvent aevt){

                JB_insert = new JButton("Submit");
                JTF_pname.setText(null);
                JTF_sub_time.setText(null);
                JTF_rank.setText(null);
                JTF_year.setText(null);
                JTF_stmt.setText(null);
            }
        });
    }

```

```
loadproject();
```

```
pn.removeAll();  
jf.invalidate();  
jf.validate();  
jf.repaint();
```

```
pn1 = new JPanel();  
pn1.setLayout(new GridLayout(10, 10));  
pn1.add(JL_pname);  
pn1.add(JTF_pname);  
pn1.add(JL_sub_time);  
pn1.add(JTF_sub_time);  
pn1.add(JL_rank);  
pn1.add(JTF_rank);  
pn1.add(JL_year);  
pn1.add(JTF_year);  
pn1.add(JL_stmt);  
pn1.add(JTF_stmt);
```

```
pn3 = new JPanel(new FlowLayout());  
pn3.add(JB_insert);  
pn1.setBounds(115, 80, 300, 250);  
pn3.setBounds(200, 350, 75, 35);
```

```
pn2 = new JPanel(new FlowLayout());  
ProjectList = new List(10);  
loadproject();  
pn2.add(ProjectList);  
pn2.setBounds(200, 350, 300, 180);
```

```
pn.add(pn1);  
pn.add(pn3);  
pn.add(pn2);
```

```
pn.setLayout(new BorderLayout());  
jf.add(pn);  
jf.setSize(800, 800);  
jf.validate();
```

```
JB_insert.addActionListener(new ActionListener() {  
    @Override  
    public void actionPerformed(ActionEvent aevt){
```

```
        try{
```

```
            Connection con =
```

```
DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","it20737055","teja2002");
```

```
            String query = "INSERT INTO hackathon_event (pname,  
problem_stmt, submitted_time, year, rank) values(?, ?, ?, ?, ?)";
```

```

        PreparedStatement stp = con.prepareStatement(query);
        stp.setString(1, JTF_pname.getText());
        stp.setString(2, JTF_stmt.getText());
        stp.setString(3, JTF_sub_time.getText());
        stp.setString(4, JTF_year.getText());
        stp.setString(5, JTF_rank.getText());
        int i = stp.executeUpdate();
        con.close();
        if(i > 0){
            JOptionPane.showMessageDialog(pn,"Inserted successfully");
        }
    }
    catch(SQLException e) {
        displaySQLErrors(e);
    }
}

});

}

});

update2.addActionListener(new ActionListener() {
    @Override
    public void actionPerformed(ActionEvent aevt){
        JB_modify = new JButton("Modify");

        JTF_pname.setText(null);
        JTF_sub_time.setText(null);
        JTF_rank.setText(null);
        JTF_year.setText(null);
        JTF_stmt.setText(null);

        pn.removeAll();
        jf.invalidate();
        jf.validate();
        jf.repaint();

        pn1=new JPanel();
        pn1.setLayout(new GridLayout(10, 10));

        pn1.add(JL_pname);
        pn1.add(JTF_pname);
        pn1.add(JL_sub_time);
        pn1.add(JTF_sub_time);
        pn1.add(JL_rank);
        pn1.add(JTF_rank);
        pn1.add(JL_year);
        pn1.add(JTF_year);
        pn1.add(JL_stmt);
        pn1.add(JTF_stmt);
    }
});

```

```

pn3 = new JPanel(new FlowLayout());
pn3.add(JB_modify);
pn1.setBounds(115, 80, 300, 250);
pn3.setBounds(200, 350, 75, 35);

pn2 =new JPanel(new FlowLayout());
ProjectList = new List(10);
loadproject();
pn2.add(ProjectList);
pn2.setBounds(200, 350, 300, 180);

pn.add(pn1);
pn.add(pn3);
pn.add(pn2);

pn.setLayout(new BorderLayout());
jf.add(pn);
jf.setSize(800, 800);
jf.validate();

ProjectList.addItemListener(new ItemListener() {
    public void itemStateChanged(ItemEvent ievt){
        try {
            Connection con3 =
DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe", "it20737055", "teja2002");
            Statement st3 = con3.createStatement();
            ResultSet rs3 = st3.executeQuery("select * from
hackathon_event");

            while (rs3.next()) {
                if
(rs3.getString("pname").equals(ProjectList.getSelectedItem()))
                    break;
            }
            if (!rs3.isAfterLast()) {
                JTF_pname.setText(rs3.getString("pname"));
                JTF_sub_time.setText(rs3.getString("submitted_time"));
                JTF_rank.setText(rs3.getString("rank"));
                JTF_year.setText(rs3.getString("year"));
                JTF_stmt.setText(rs3.getString("problem_stmt"));
            }
        }
        catch (SQLException selectException) {
            displaySQLErrors(selectException);
        }
    }
});
JB_modify.addActionListener(new ActionListener() {
    @Override
    public void actionPerformed(ActionEvent aevt){
        try{
            int a = JOptionPane.showConfirmDialog(pn, "Are you sure want
to update:");

```

```

        if(a == JOptionPane.YES_OPTION){
            String query = "update hackathon_event set problem_stmt
= ?, submitted_time = ?, rank = ?, year = ? where pname = ?";
            Connection con4 =
DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe", "it20737055", "teja2002");
            PreparedStatement stp2 = con4.prepareStatement(query);
            stp2.setString(1, JTF_stmt.getText());
            stp2.setString(2, JTF_sub_time.getText());
            stp2.setString(3, JTF_rank.getText());
            stp2.setString(4, JTF_year.getText());
            stp2.setString(5, JTF_pname.getText());
            int i = stp2.executeUpdate();
            if(i>0){
                JOptionPane.showMessageDialog(pn,"Updated
rows succesfully");
            }
            loadproject();
        }
    }
    catch(SQLException e){
        displaySQLErrors(e);
    }
}
});
}
});
delete2.addActionListener(new ActionListener() {
    @Override
    public void actionPerformed(ActionEvent aevt){
        JB_delete = new JButton("Delete");

        JTF_pname.setText(null);
        JTF_sub_time.setText(null);
        JTF_rank.setText(null);
        JTF_year.setText(null);
        JTF_stmt.setText(null);

        pn.removeAll();
        jf.invalidate();
        jf.validate();
        jf.repaint();

        pn1 = new JPanel();
        pn1.setLayout(new GridLayout(10, 10));
        pn1.add(JL_pname);
        pn1.add(JTF_pname);
        pn1.add(JL_sub_time);
        pn1.add(JTF_sub_time);
        pn1.add(JL_rank);
        pn1.add(JTF_rank);
        pn1.add(JL_year);
        pn1.add(JTF_year);

```

```

pn1.add(JL_stmt);
pn1.add(JTF_stmt);

pn3 = new JPanel(new FlowLayout());
pn3.add(JB_delete);
pn1.setBounds(115, 80, 300, 250);
pn3.setBounds(200, 350, 75, 35);

pn2 = new JPanel(new FlowLayout());
ProjectList = new List(10);
loadproject();
pn2.add(ProjectList);
pn2.setBounds(200, 350, 300, 200);

pn.add(pn1);
pn.add(pn3);
pn.add(pn2);

pn.setLayout(new BorderLayout());
jf.add(pn);
jf.setSize(800, 800);
jf.validate();

ProjectList.addItemListener(new ItemListener() {
    public void itemStateChanged(ItemEvent ievt){
        try {
            Connection con5 =
DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe", "it20737055", "teja2002");
            Statement st4 = con5.createStatement();
            ResultSet rs4 = st4.executeQuery("select * from
hackathon_event");

            while (rs4.next()) {
                if
(rs4.getString("pname").equals(ProjectList.getSelectedItem()))
                    break;
            }
            if (!rs4.isAfterLast()) {
                JTF_pname.setText(rs4.getString("pname"));
                JTF_sub_time.setText(rs4.getString("submitted_time"));
                JTF_rank.setText(rs4.getString("rank"));
                JTF_year.setText(rs4.getString("year"));
                JTF_stmt.setText(rs4.getString("problem_stmt"));
            }
        }
        catch (SQLException selectException) {
            displaySQLErrors(selectException);
        }
    }
});
JB_delete.addActionListener(new ActionListener() {
    @Override

```



```

        public void actionPerformed(ActionEvent aevt){
            try {
                int a = JOptionPane.showConfirmDialog(pn,"Are you sure want to
Delete:");
                if(a == JOptionPane.YES_OPTION){
                    String query = "DELETE FROM hackathon_event WHERE
pname = ?";
                    Connection con6 =
DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe", "it20737055", "teja2002");
                    PreparedStatement stp3 = con6.prepareStatement(query);
                    stp3.setString(1, JTF_pname.getText());
                    int i = stp3.executeUpdate();
                    if(i>0){
                        JOptionPane.showMessageDialog(pn,"nDeleted
rows succesfully");
                    }
                    loadproject();
                }
            }
            catch(SQLException e){
                displaySQLErrors(e);
            }
        }
    });
}
});
view2.addActionListener(new ActionListener() {
    @Override
    public void actionPerformed(ActionEvent aevt){
        pn.removeAll();
        jf.invalidate();
        jf.validate();
        jf.repaint();

        JLabel view = new JLabel("Hackathon event details View");
        view.setForeground(Color.RED);
        JB_view = new JButton("View");
        Font myFont = new Font("Serif", Font.BOLD,25);
        view.setFont((myFont));

        pn1 = new JPanel();
        pn2 = new JPanel();
        pn1.add(view);
        pn2.add(JB_view);
        pn.add(pn1);
        pn.add(pn2);
        pn.setBounds(500, 800, 300, 300);
        pn.setLayout(new FlowLayout());

        jf.add(pn);
        jf.setSize(800, 800);
        jf.validate();

```

```

JB_view.addActionListener(new ActionListener() {
    @Override
    public void actionPerformed(ActionEvent aevt){
        JFrame jfr = new JFrame("Hackathon event Details");
        JTable jt;
        DefaultTableModel model = new DefaultTableModel();
        jt = new JTable(model);
        model.addColumn("Project name");
        model.addColumn("Problem statement");
        model.addColumn("Submitted time");
        model.addColumn("Year");
        model.addColumn("Rank");
        try {
            Connection con7 =
DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe", "it20737055", "teja2002");
            Statement st4 = con7.createStatement();
            ResultSet rs5 = st4.executeQuery("select * from
hackathon_event");

            while(rs5.next()){
                model.addRow(new Object[]{rs5.getString("pname"),
rs5.getString("problem_stmt"), rs5.getString("submitted_time"), rs5.getString("year"), rs5.getString("rank")});
            }
        } catch(SQLException e){
            displaySQLErrors(e);
        }
        jt.setEnabled(false);
        jt.setBounds(30, 40, 300, 300);
        JScrollPane jsp = new JScrollPane(jt);
        jfr.add(jsp);
        jfr.setSize(800, 400);
        jfr.setVisible(true);
    }
});
}
}
}

```

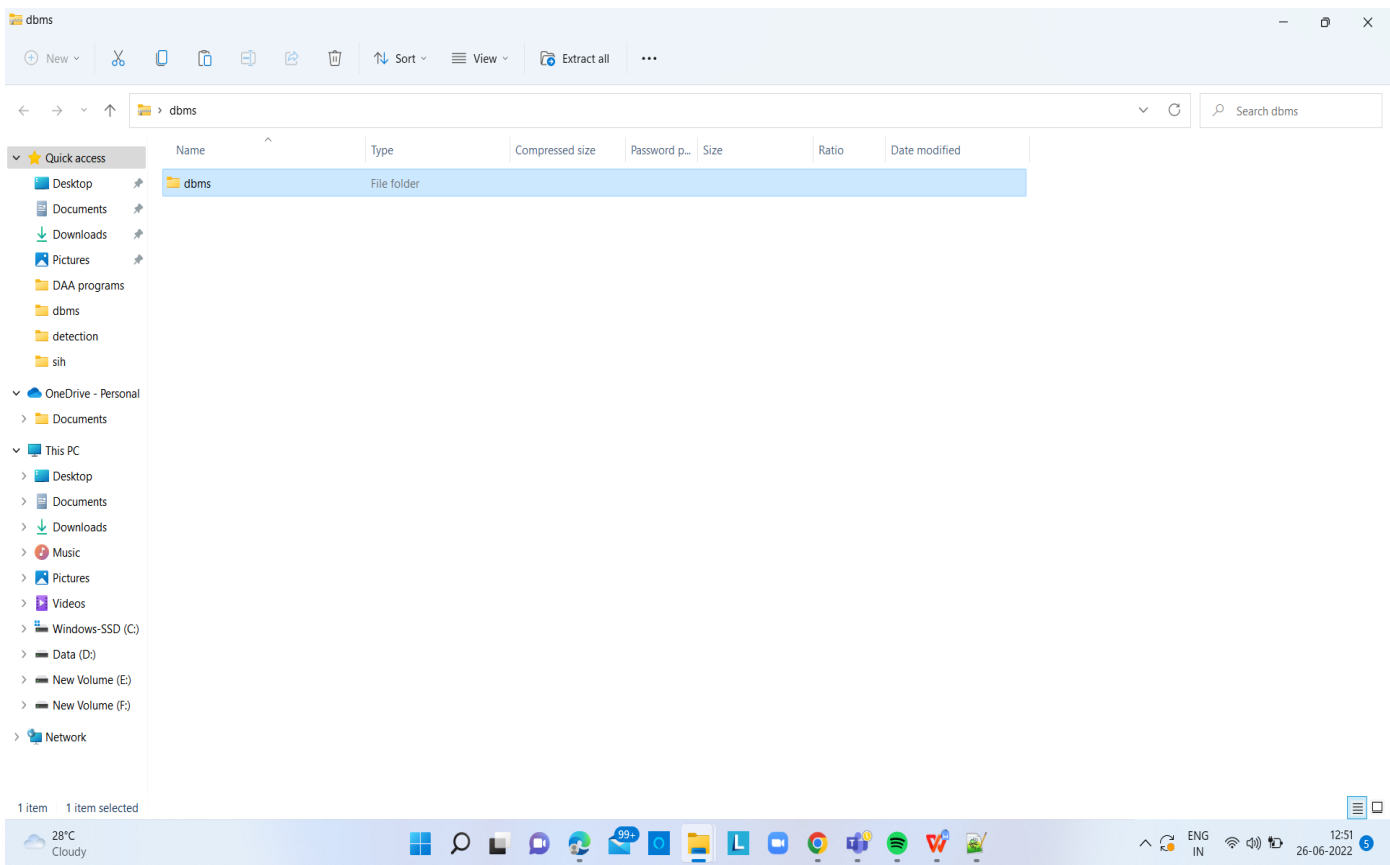
## Github links and folder structure:

Link:

<https://github.com/Tejaswi141/Dbm>

s-assignment-SIH-portal

# Folder structure:



dbms

New

✂

Sort

View

Extract all

dbms > dbms

Search dbms

Quick access

Desktop

Documents

Downloads

Pictures

DAA programs

dbms

Screenshots

sih

OneDrive - Personal

Documents

This PC

Desktop

Documents

Downloads

Music

Pictures

Videos

Windows-SSD (C:)

Data (D:)

New Volume (E:)

New Volume (F:)

Network

Name	Type	Compressed size	Password p...	Size	Ratio	Date modified
class_files	File folder					25-06-2022 17:59
java_files	File folder					25-06-2022 17:36

2 items

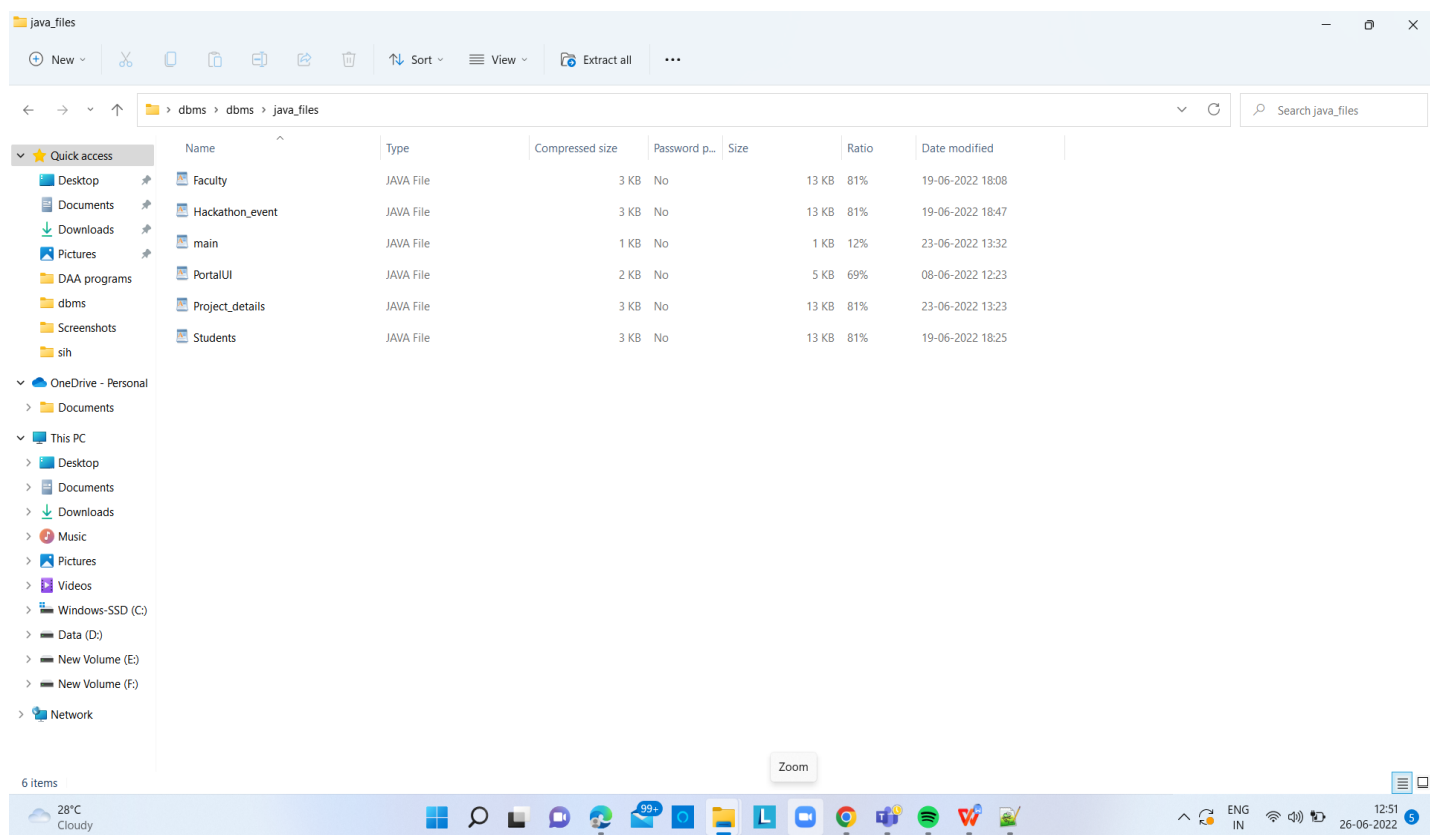
28°C Cloudy

Windows Taskbar Icons

ENG IN

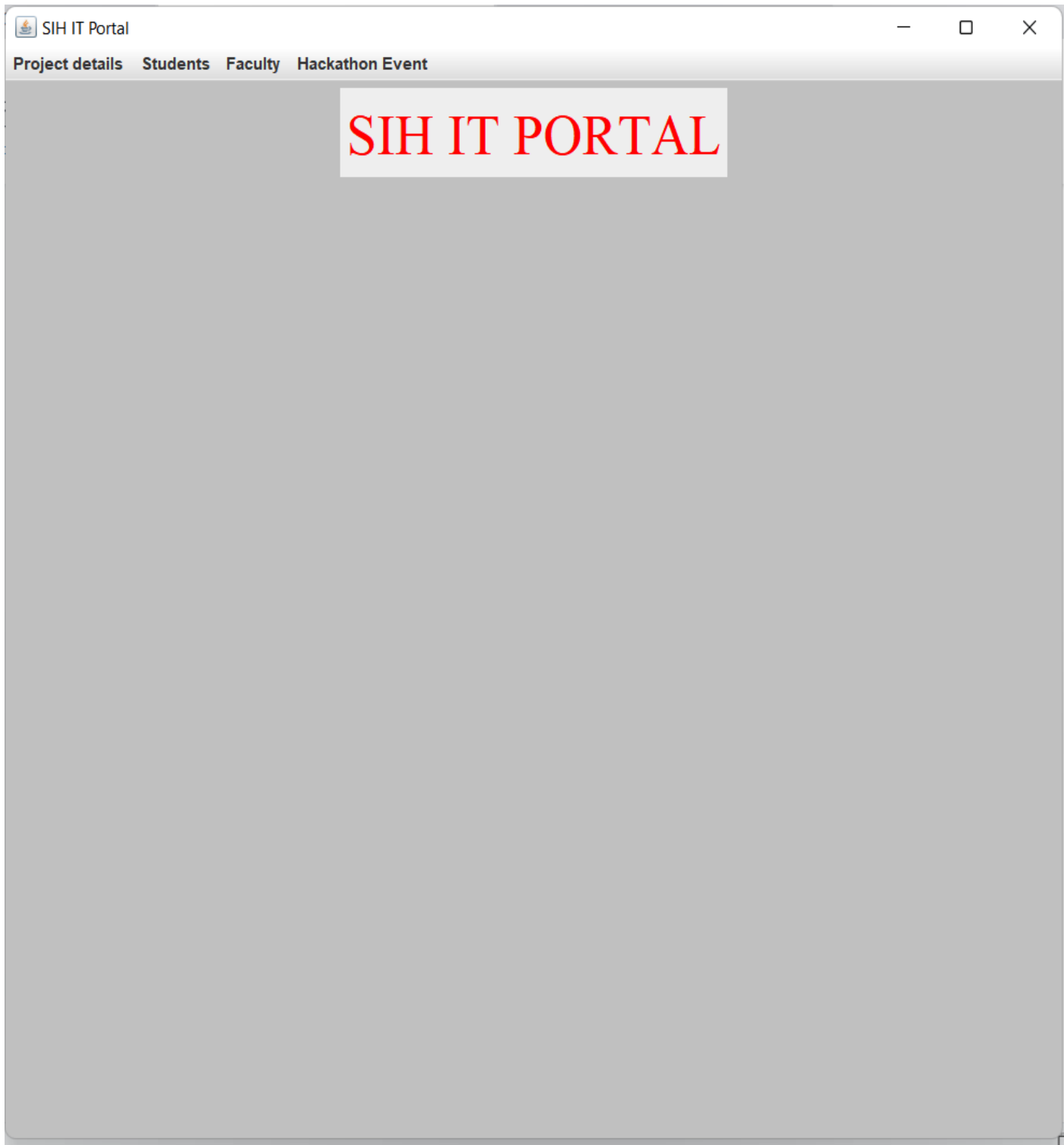
12:51

26-06-2022



# TESTING:

## OUTPUT SCREENSHOTS:



**Project Name:**

Hackathon-7

**Explanation:**

based on apps

**Duration:**

30-July-2022

**Year:**

2022

**Submit**

Hackathon-1  
Hackathon-2  
Hackathon-4  
Hackathon-3

Message ×



Inserted successfully

OK

**Project Name:**

Hackathon-4

**Explanation:**

web technologies

**Duration:**

2021-09-30 00:00:00.0

**Year:**

2021

**Delete**

Hackathon-1  
Hackathon-2  
Hackathon-7  
Hackathon-4

Message

**Deleted rows succesfully****OK**



**Project Name:**

Hackathon-7

**Explanation:**

based on apps of new technologies

**Duration:**

30-July-2022

**Year:**

2022

**Modify**

Hackathon-2

Hackathon-7

Hackathon-3

Hackathon-6

Message



Updated rows succesfully

OK

SIH IT Portal

Project details

Students

Faculty

Hackathon Event

Project details View

View

Project Details

pname	explanation	duration	year
Hackathon-1	developing on mobile apps	2020-10-31 00:00:00.0	2020
Hackathon-2	developing on smart automation	2021-04-30 00:00:00.0	2021
Hackathon-7	based on apps of new technologies	2022-07-30 00:00:00.0	2022
Hackathon-3	based on android apps	2021-07-12 00:00:00.0	2021
Hackathon-6	to develop apps	2022-07-12 00:00:00.0	2022

**Roll number:**

S20009

**Student name:**

Tejaswi

**Semester:**

4

**Year:**

2022

**Phone number:**

7320513476

**Submit**S20005  
S20006  
S20007  
S20008

Message

×

**Inserted successfully****OK**

**Roll number:**

S20008

**Student name:**

Ravi

**Semester:**

4

**Year:**

2021

**Phone number:**

7840589213

**Delete**S20006  
S20007  
S20009  
S20008

Select an Option

×



Are you sure want to Delete:

**Yes****No****Cancel**

Roll number:

S20003

Student name:

poojit

Semester:

4

Year:

2021

Phone number:

8977623471

Modify

S20001  
S21002  
S20003  
S21004

Select an Option ×



Are you sure want to update:

Yes

No

Cancel

SIH IT Portal

Project details

Students

Faculty

Hackathon Event

Student details View

View

Student Details

Roll Number	Student Name	Semester	Year	Phone number
S20001	Anvitha	4	2021	7032519223
S21002	Sritan	2	2021	9121567890
S20003	poojit	4	2021	8977623471
S21004	Sahithi	2	2021	8247680757
S20005	Varshitha	4	2021	8019324560
S20006	Jival	3	2022	6578934566
S20007	Bhanu	4	2022	6789450321
S20009	Tejaswi	4	2022	7320513476
S20008	Ravi	4	2021	7840589213

Faculty ID:

F1006

Faculty name:

Sathya

Project name:

Hackathon-6

Year:

2022

Phone number:

8754012376

Submit

F1003  
F1004  
F1005  
F1007

Message

×



Inserted successfully

OK

Faculty ID:

F1007

Faculty name:

Madhavi

Project name:

Hackathon-6

Year:

2021

Phone number:

7649302811

Delete

F1004  
F1005  
F1007  
F1006

Select an Option

×



Are you sure want to Delete:

Yes

No

Cancel



Faculty ID:

F1004

Faculty name:

Prasanna

Project name:

Hackathon-2

Year:

2021

Phone number:

8349223049

Modify

F1001  
F1002  
F1003  
F1004

Select an Option



Are you sure want to update:

Yes

No

Cancel

SIH IT Portal

Project details

Students

Faculty

Hackathon Event

Faculty details View

View

Student Details

Faculty ID	Faculty Name	Project name	Year	Phone number
F1001	Rajendra	Hackathon-1	2021	9010589033
F1002	Prabhakar	Hackathon-2	2021	9948109871
F1003	Srinivasulu	Hackathon-1	2021	9618856790
F1004	Prasanna	Hackathon-2	2021	8349223049
F1005	Kavya	Hackathon-1	2021	9212987634
F1007	Madhavi	Hackathon-6	2021	7649302811
F1006	Sathya	Hackathon-6	2022	8754012376

**Project name:**

Hackathon-3

**Submitted time:**

10-July-22

**Rank:**

4

**Year:**

2022

**Problem statement:**

building the app for booking tickets

**Submit**Hackathon-2  
Hackathon-2  
Hackathon-6

Message ×

**Inserted successfully****OK**

**Project name:**

Hackathon-2

**Submitted time:**

2021-04-25 00:00:00.0

**Rank:**

1

**Year:**

2021

**Problem statement:**

AI based chatbot

**Delete**

Hackathon-3  
Hackathon-2  
Hackathon-2  
Hackathon-6

Select an Option ×

**Are you sure want to Delete:****Yes****No****Cancel**

**Project name:**

Hackathon-6

**Submitted time:**

2022-07-10 00:00:00.0

**Rank:**

1

**Year:**

2022

**Problem statement:**

building app for travelling trips

**Modify**

Hackathon-3  
Hackathon-2  
Hackathon-2  
Hackathon-6

Select an Option

✕



Are you sure want to update:

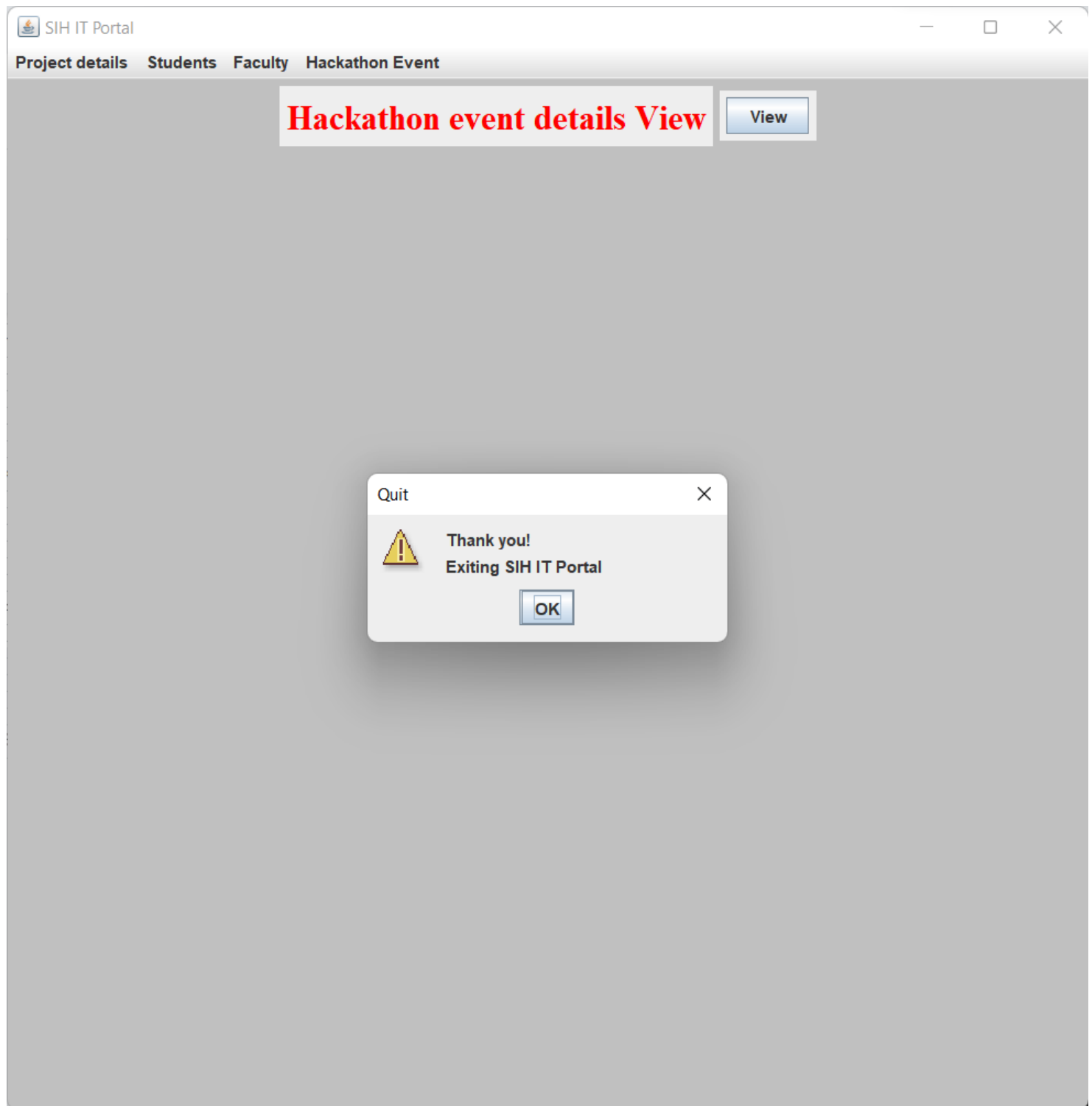
**Yes****No****Cancel**

## Hackathon event details View


[View](#)

### Hackathon event Details

Project name	Problem statement	Submitted time	Year	Rank
Hackathon-3	building the app for booking ti..	2022-07-10 00:00:00.0	2022	4
Hackathon-2	AI based chatbot	2021-04-25 00:00:00.0	2021	1
Hackathon-2	smart document manageme..	2021-04-30 00:00:00.0	2021	2
Hackathon-6	building app for travelling trips	2022-07-10 00:00:00.0	2022	1



## DATABASE OUTPUTS:

 Run SQL Command Line

SQL\*Plus: Release 11.2.0.2.0 Production on Sun Jun 26 13:24:45 2022

Copyright (c) 1982, 2010, Oracle. All rights reserved.

SQL> conn it20737055/teja2002

Connected.

SQL> select \* from project\_details;

PNAME	DURATION	YEAR
-------	----------	------

-----  
EXPLANATION  
-----

Hackathon-1 developing on mobile apps	31-OCT-20	2020
--	-----------	------

Hackathon-2 developing on smart automation	30-APR-21	2021
---	-----------	------

Hackathon-4 web technologies	30-SEP-21	2021
---------------------------------	-----------	------

PNAME	DURATION	YEAR
-------	----------	------

-----  
EXPLANATION  
-----

Hackathon-3 based on android apps	12-JUL-21	2021
--------------------------------------	-----------	------

Hackathon-6 to develop apps	12-JUL-22	2022
--------------------------------	-----------	------

SQL>



# Run SQL Command Line

```
SQL> select * from student_details;
```

RNO	SNAME	YEAR	SEM
-----	-------	------	-----

PHONE_NO
----------

S20001 7032519223	Anvitha	2021	4
----------------------	---------	------	---

S21002 9121567890	Sritan	2021	2
----------------------	--------	------	---

S20003 8977623471	poojit	2021	4
----------------------	--------	------	---

RNO	SNAME	YEAR	SEM
-----	-------	------	-----

PHONE_NO
----------

S21004 8247680757	Sahithi	2021	2
----------------------	---------	------	---

S20005 8019324560	Varshitha	2021	4
----------------------	-----------	------	---

S20006 6578934566	Jival	2022	3
----------------------	-------	------	---

RNO	SNAME	YEAR	SEM
-----	-------	------	-----

PHONE_NO
----------

S20007 6789450321	Bhanu	2022	4
----------------------	-------	------	---

S20009 7320513476	Tejaswi	2022	4
----------------------	---------	------	---

S20008 7840589213	Ravi	2021	4
----------------------	------	------	---

9 rows selected.

```
SQL> █
```

```
SQL> select * from faculty;
```

FID	FNAME	YEAR
-----		
PNAME	PHONE_NO	
-----		
F1001	Rajendra	2021
Hackathon-1	9010589033	
F1002	Prabhakar	2021
Hackathon-2	9948109871	
F1003	Srinivasulu	2021
Hackathon-1	9618856790	

FID	FNAME	YEAR
-----		
PNAME	PHONE_NO	
-----		
F1004	Prasanna	2021
Hackathon-2	8349223049	
F1005	Kavya	2021
Hackathon-1	9212987634	
F1007	Madhavi	2021
Hackathon-6	7649302811	

FID	FNAME	YEAR
-----		
PNAME	PHONE_NO	
-----		
F1006	Sathya	2022
Hackathon-6	8754012376	

```
7 rows selected.
```

```
SQL>
```

## DISCUSSION AND FUTURE WORK

So far this project has helped us to register for any project that is conducted online by faculty and students will be able to register for those projects and easily submit their ideas. In the future, we try to include an option which allows students to review their projects which are checked by the faculty and can clarify their mistakes.

## REFERENCES:

1. Abraham Silberschatz, Henry F Korth, S. Sudarshan, Database System Concepts, 6th Edition, McGraw-Hill International Edition, 2010.
2. [https://docs.oracle.com/cd/E11882\\_01/server.112/e40540/intro.htm#CNCPT001](https://docs.oracle.com/cd/E11882_01/server.112/e40540/intro.htm#CNCPT001)
3. <https://docs.oracle.com/javase/8/docs/>