RAMAIAH UNIVERSITY OF APPLIED SCIENCES

ASSIGNMENT

Course Code.	CSC311A
Course Name	DBMS
Programme	B.Tech
Department	Computer Science Engineering
Faculty	F.E.T

Name of the Student	Sudhanshu shekhar
Reg. No	**17ETCS002213**
Semester/Year	6thSem / 3nd Year
Course Leader/s	AMI MAM

:

Declaration Sheet				
Student Name	Sudhanshu shekhar			
Reg. No	17ETCS002213			
Programme	B.Tech		Semester/Year	6 SEM
Course Code	CSC311A			
Course Title	DATA BASE MANAGEMENT SYSTEM			
Course Date		to		
Course Leader				

Declaration

The assignment submitted herewith is a result of my own investigations and that I have conformed to the guidelines against plagiarism as laid out in the Student Handbook. All sections of the text and results, which have been obtained from other sources, are fully referenced. I understand that cheating and plagiarism constitute a breach of University regulations and will be dealt with accordingly.

Signature of the Student		Date	
Submission date stamp (by Examination & Assessment Section)			

Signature of the Course Leader and date	Signature of the Reviewer and date

PART A

A.1 List of functional and data requirements

List of the functional requirement is as follows:

FR 1: The system should allow the separate login for staff and group members.

FR 2: The system should able to maintain all the details of student and teacher.

FR 3: The system should allow to display the group id and room details for the student.

FR 4: The system should allow to register new project team.

FR 5: The system should allow the group member to delete the registration made

FR 6: The system should allow the project leader to control or take project.

FR 7: The system should allow the user to logout from the system.

Data requirement is as follows:

Data requirement for student:

Name	Type	Key
Pname	character	
Dept	character	
Pdes	character	
Gid	character	Primary key
Gpass	character	
Groom	Character	
Gtable	inteager	
mentor	Character	Foreign key
Leader	Character	
Name2	Character	
Name3	Character	
Name4	Character	

Data requirement for faculty:

Name	Type	key
Fname	character	Parimary Key
FUserId	character	
FPass	character	

A.2 Implementation of relational database schema with appropriate attributes, and constraints using SQL commands

The implementation of the relational Dtabase scheme with appropriate attributes and constraints using sql command:

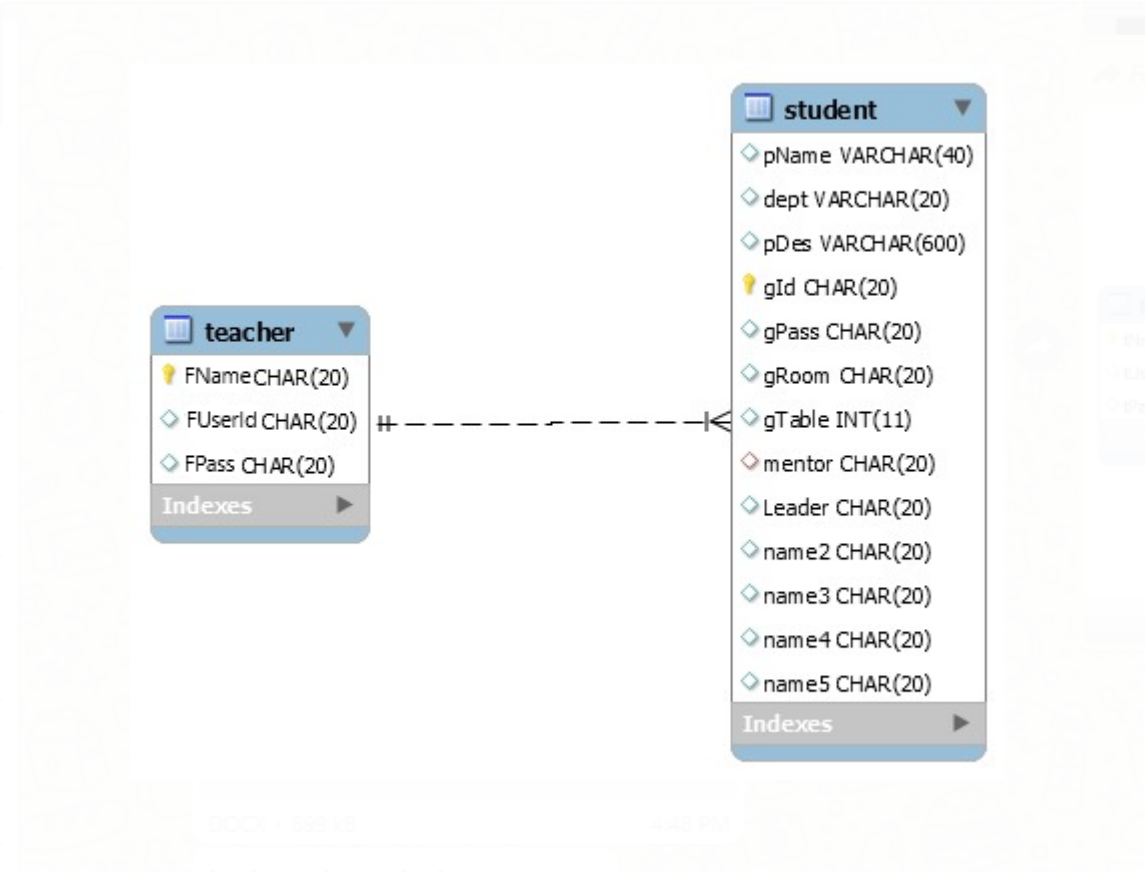
FACALTY:

FName	FUserId	FPass

STUDENT:

pname	dept	pdes	gID	gpass	groom	gtable	leader	Name 1	Name2	Name3	Name4

Relation scheme for above is as follows:



Here we have mentioned the Fname as a primary key in Facalty , we have defined the Gid as the primary key in the student, We have Tname as the Foreign key in the student and as the mentor.

Code for executing the value in table:

The code for executing the values in table is as follows:

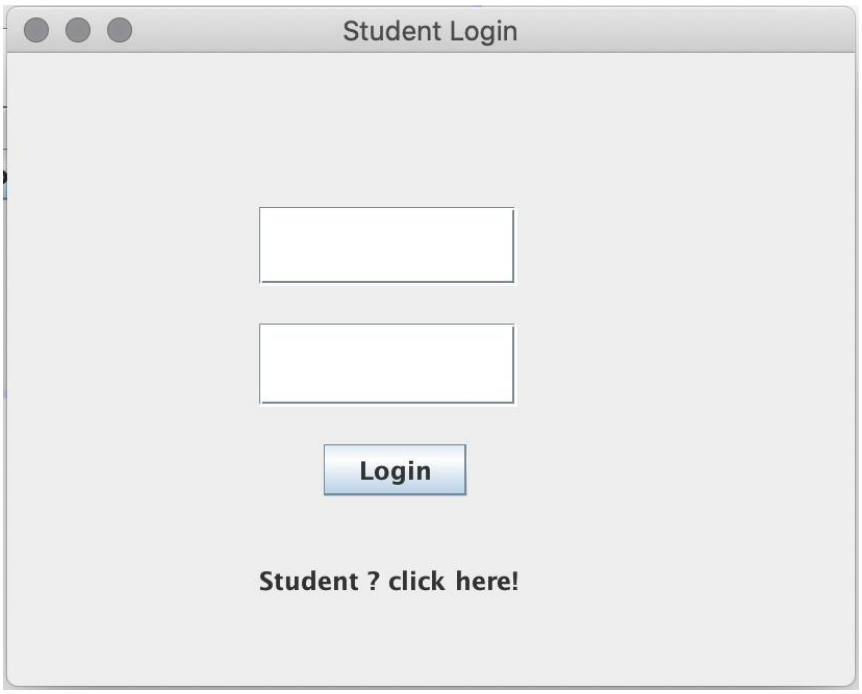
```
create table teacher(tName char(20) primary key,tUserId char(20),tPass char(20));
create table student(pName varchar(40),dept varchar(20),pDes varchar(60),gId char(20) ,gPass char(20),
gRoom char(20),gTable int(11) ,mentor char(20) , Leader char(20),
name2 char(20),name3 char(20),name4 char(20),
name5 char(20),primary key(gId),foreign key(mentor)references teacher (tName));

show tables;
desc teacher;
desc student;

insert into teacher values("nitin","1","ruas");
insert into student(gId ,gPass) values("51","msruas");

show tables;
desc team;
select * from teacher;
select * from student;
```

A.3 Implementation of GUI with options such as login, registration, updating, and cancellation

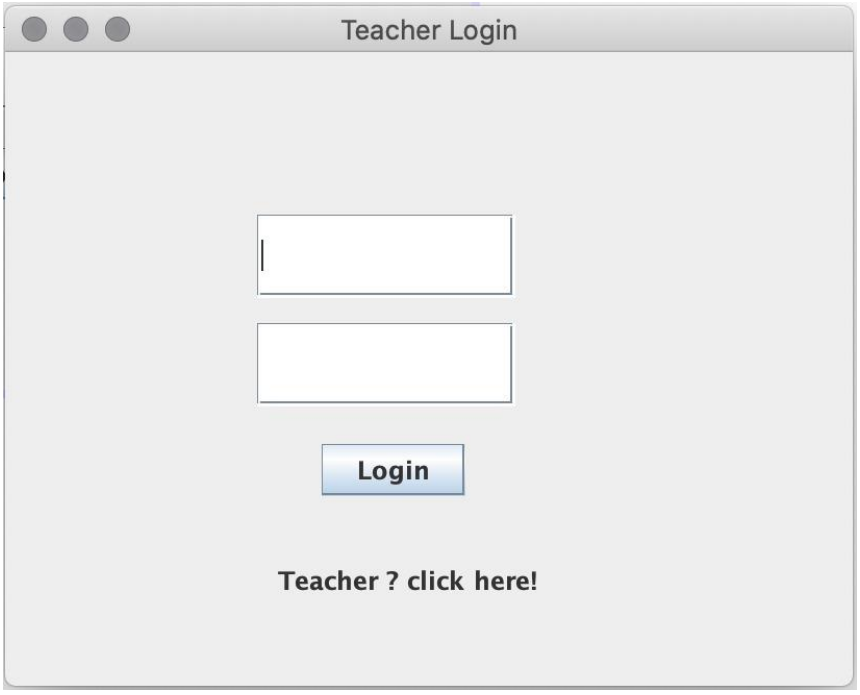
Student login:

The following is the graphical user interface for the login:

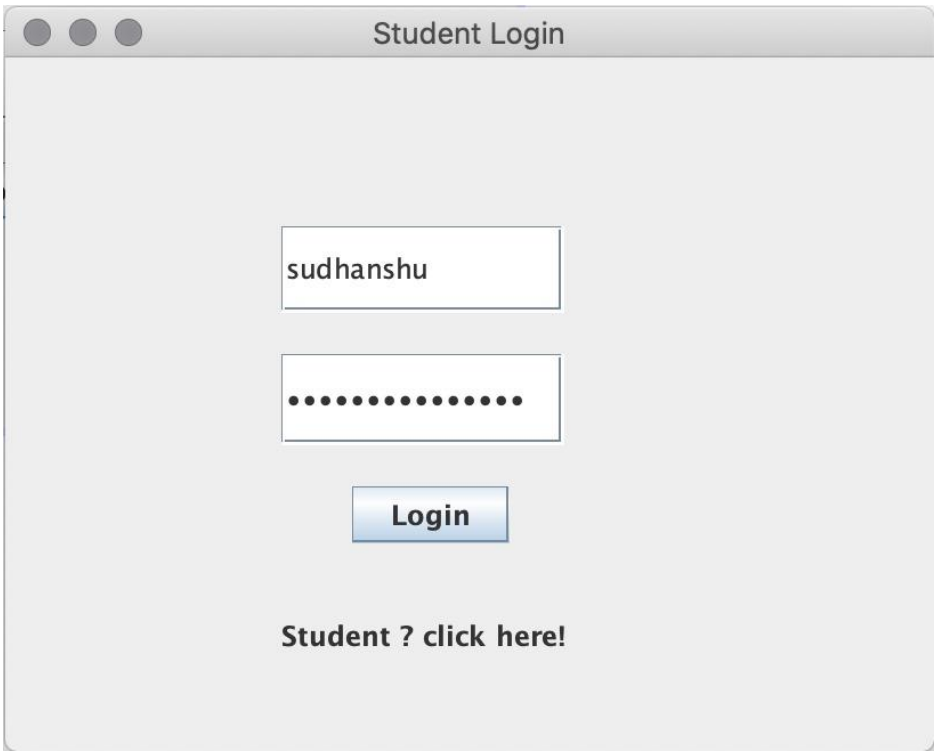
Empty login for student :



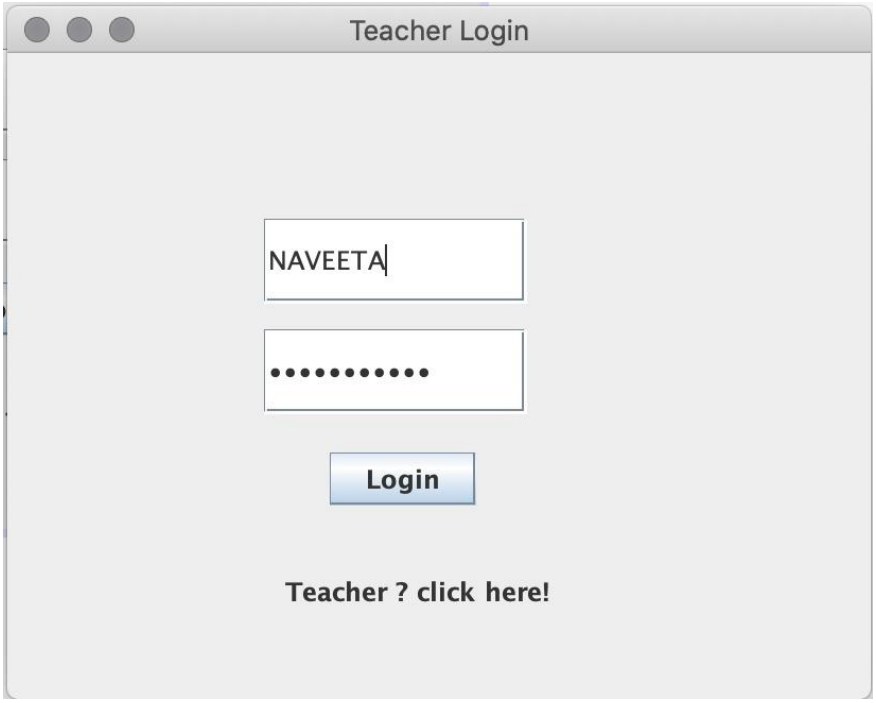
Empty login for the faculty:



Login details with my details:



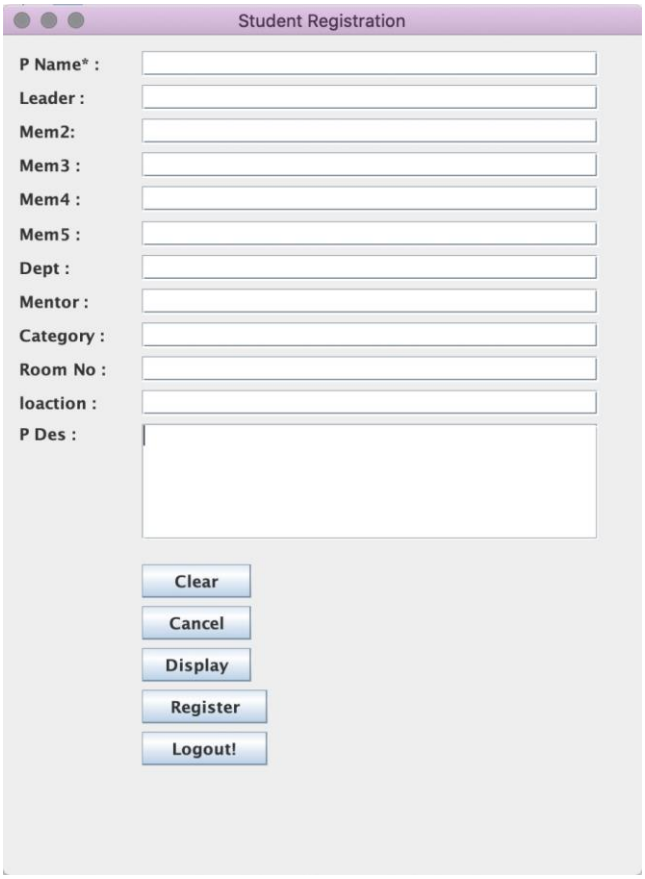
Login details with faculty:



Student registration details:

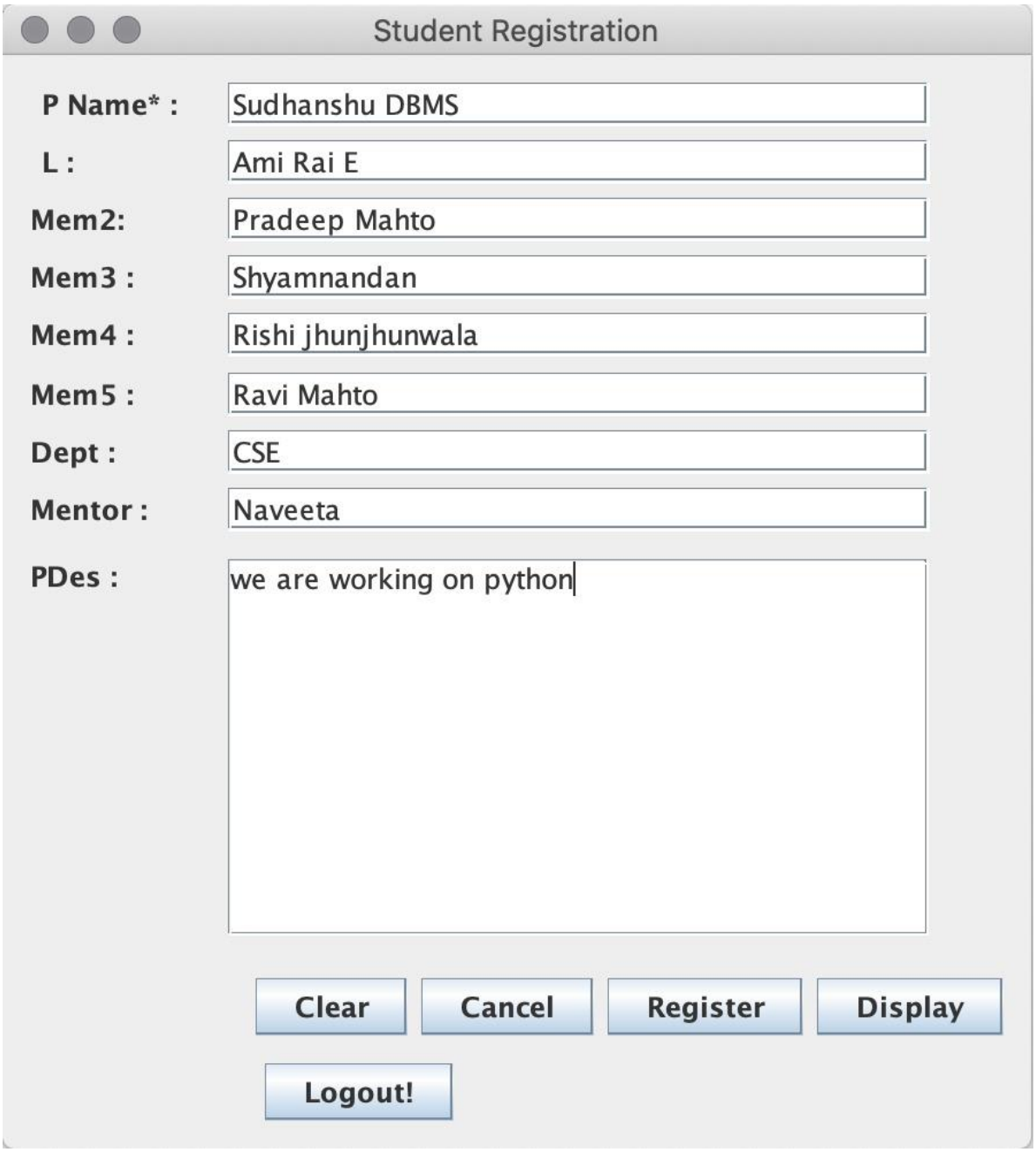
Registration for student empty:

The following are the empty table in which student are going to fill accordingly,They have to give the details like the name of the leader, total number of particepants in the task mentor for the project department they belongs to etc.



Registration for student:

When participant clicked the registration button, they can enter their details here, here we are creating a group project, in a group there is five members among them Sudhanshu DBMS is leader. For registrating another four members. I have designed GUI in so that one member can register the rest of the group member of group.



Database after being updated:

As from the above we have in sql if we will run the code from the database we will have the following tuple as show below.

#	pName	dept	pDes	gId	gPass
1	sudhanshu shekhar	CSE	we are working on python project	10	MSRUAS

I have shown here the update of group 1 in the database as shown in the above figure

A.4 Connection of front end with the database

Connection of the front end using the database is as follows:

```
1
2   package javaapplication10;
3
4   import java.sql.Connection;
5   import java.sql.DriverManager;
6
7   /**
8    *
9    * @author sudhanshu shekhar.
10   */
11  public class JavaApplication10 {
12
13      public static Connection getConnection(){
14          Connection con = null;
15          try {
16              Class.forName("com.mysql.jdbc.Driver");
17              con = DriverManager.getConnection(
18                      "jdbc:mysql://localhost:3306/ndb?useSSL=false",
19                      "root", "password sudhanshu");
20          }
            catch (Exception e) {
22              System.out.println(e.getMessage());
23          }
24          return con;
25      }
26      public static void main(String[] args) {
27          // TODO code application logic here
28      }
29
30  }
31
```

Code for the login of the student is as follows:

```java

package javaapplication10;

import java.sql.PreparedStatement;
import java.sql.ResultSet;
import javax.swing.JFrame;
import javax.swing.JOptionPane;

/**
 *
 * @author sudhanshu shekhar.
 * reg no 17ETCS002213
 */
public class studentlogin extends javax.swing.JFrame {

    public studentlogin() {
        initComponents();
        this.setLocationRelativeTo(null);
    }

    @SuppressWarnings("unchecked")
    Generated Code

    private void jLabelTeacherMouseClicked(java.awt.event.MouseEvent evt) {
        teacherLogin lgf = new teacherLogin();
        lgf.setVisible(true);
        lgf.pack();
        lgf.setLocationRelativeTo(null);
        lgf.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        this.dispose();
    }
```

```
91
92      private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
93          PreparedStatement ps;
94          ResultSet rs;
95          String usrName = jTextFieldUId.getText();
96          String pass = String.valueOf(jPasswordFieldPass.getPassword());
97          String query = "select gId,gPass from student ";
98              try {
99                  ps = JavaApplication10.getConnection().prepareStatement(query);
100                 rs = ps.executeQuery();          //process the result
101                 while (rs.next()) {
102                     String usr = rs.getString(1);
103                     String pwd = rs.getString(2);
104                     reg.func(usr);
105                     if (usr.equals(usrName) && pwd.equals(pass)) {
106                         reg sf = new reg();
107                         sf.setVisible(true);
108                         sf.pack();
109                         sf.setLocationRelativeTo(null);
110                         sf.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
111                         this.dispose();
112                     } else if(rs.isLast()) {
113                         JOptionPane.showMessageDialog(null, "Login Failed !!");
114                     }
115                 }
117             } catch (Exception e) {
118                 System.out.println(e.getMessage());
119             }
120     }
```

```
120
121     public static void main(String args[]) {
122         /* Set the Nimbus look and feel */
123         Look and feel setting code (optional)
144
145         /* Create and display the form */
148         java.awt.EventQueue.invokeLater(new Runnable() {
149             public void run() {
150                 new studentlogin().setVisible(true);
151             }
152         });
153     }
154
155     // Variables declaration - do not modify
156     private javax.swing.JButton jButton1;
157     private javax.swing.JLabel jLabelTeacher;
158     private javax.swing.JPasswordField jPasswordFieldPass;
159     private javax.swing.JTextField jTextFieldUId;
160     // End of variables declaration
}
```

CODE FOR FACALTY LOGIN IS AS FOLLOWS:

The following is the java program for the faculty login:

```
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import javax.swing.JFrame;
import javax.swing.JOptionPane;
import javax.swing.JTable;
import javax.swing.table.DefaultTableModel;
/**
 * @author SUDHANSHU SHEKHAR
 * REG NO - 17ETCS002213
 */
public class teacher extends javax.swing.JFrame {
    public teacher() {
        initComponents();
        fileCombo();
    }

    @SuppressWarnings("unchecked")
    Generated Code

    private void jButton3ActionPerformed(java.awt.event.ActionEvent evt) {
        teacherLogin lgf = new teacherLogin();
        lgf.setVisible(true);
        lgf.pack();
        lgf.setLocationRelativeTo(null);
        lgf.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        this.dispose();
    }
    public static void main(String args[]) {
        java.awt.EventQueue.invokeLater(new Runnable() {
            public void run() {
                new teacher().setVisible(true);
```

```
85              }
86          });
87      }
88      // Variables declaration - do not modify
89      private javax.swing.JButton jButton3;
90      private javax.swing.JScrollPane jScrollPanel;
91      private javax.swing.JTable jTableGrpRecord;
92      // End of variables declaration
93      DefaultTableModel t;
94      public static String userId;
95
96      public static void func(String arg) {
97          System.out.println(arg);
98          userId = arg;
99      }
100     private void fileCombo() {
101         PreparedStatement ps;
102         ResultSet rs;
103         t = new DefaultTableModel();
104         jTableGrpRecord.setSelectionMode(javax.swing.ListSelectionModel.SINGLE_SELECTION);
105         t.addColumn("PName");
106         t.addColumn("Deptt");
107         t.addColumn("Leader");
108         t.addColumn("mem2");
109         t.addColumn("mem3");
110         t.addColumn("mem4");
111         t.addColumn("mem5");
112         t.addColumn("Room");
113         t.addColumn("Table");
114         t.addColumn("PDes");
115         String query = "SELECT pName,dept,Leader,name2,name3,name4,name5,gRoom,gTable,pDes FROM student
```

```
116         try {
117             ps = JavaApplication10.getConnection().prepareStatement(query);
118             ps.setString(1, userId);
119             rs = ps.executeQuery();
120             while (rs.next()) {
121                 t.addRow(new Object[]{
122                     rs.getString(1),
123                     rs.getString(2),
124                     rs.getString(3),
125                     rs.getString(4),
126                     rs.getString(5),
127                     rs.getString(6),
128                     rs.getString(7),
129                     rs.getString(8),
130                     rs.getString(9),
131                     rs.getString(10)});
132             }
133             jTableGrpRecord.setModel(t);
134             jTableGrpRecord.setAutoResizeMode(JTable.AUTO_RESIZE_OFF);
135
136         } catch (Exception e) {
137             JOptionPane.showMessageDialog(null, e);
138         }
139     }
140 }
141
```

Registration code for student table:

```
public reg() {
    initComponents();
    PreparedStatement ps;
    ResultSet rs;
    String query = "SELECT gRoom,gTable FROM student where gId = ?;";
    try {
        ps = JavaApplication10.getConnection().prepareStatement(query);
        ps.setString(1, userId);
        rs = ps.executeQuery();
        while (rs.next()) {
            String roomNo = rs.getString(1);
            String tableNo = rs.getString(2);
            jLabel1.setText("Room No : " + roomNo+"  Table No : "+tableNo);
        }
    } catch (Exception e) {
        JOptionPane.showMessageDialog(null, e);
    }
}

    try {
        ps = JavaApplication10.getConnection().prepareStatement(query);
        ps.setString(1, userId);
        ps.executeUpdate();
    } catch (Exception e) {
        JOptionPane.showMessageDialog(null, e);
    }
    studentlogin lgf = new studentlogin();
    lgf.setVisible(true);
    lgf.pack();
    lgf.setLocationRelativeTo(null);
    lgf.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    this.dispose();
}
```

```
private void jButtonRegisterActionPerformed(java.awt.event.ActionEvent evt) {
    String projectName = jTextFieldProjectName.getText();
    String dept = jTextFieldDept.getText();
    String projDesc = jTextAreaProjectDescription.getText();
    String groupRoom = randomAlphaNumeric(1) + randomNumeric(3);
    String groupTable = randomNumeric(1);
    String mentor = jTextFieldMentor.getText();
    String grpLeader = jTextFieldLeaderName.getText();
    String mem2 = jTextFieldMem2.getText();
    String mem3 = jTextFieldMem3.getText();
    String mem4 = jTextFieldMem4.getText();
    String mem5 = jTextFieldMem5.getText();
    PreparedStatement ps;
    String query = ""
            + "update student set "
            + "pName = ?, "
            + "dept  = ?, "
            + "pDes  = ?, "
            + "gRoom       = ?, "
            + "gTable      = ?, "
            + "mentor      = ?, "
            + "Leader = ?, "
            + "name2      = ?, "
            + "name3      = ?, "
            + "name4      = ?, "
            + "name5      = ? "
            + "where gId   = ?;";

    try {
        ps = JavaApplication10.getConnection().prepareStatement(query);
        ps.setString(1, projectName);
        ps.setString(2, dept);
        ps.setString(3, projDesc);
        ps.setString(4, groupRoom);
        ps.setInt(5, Integer.valueOf(groupTable));
        ps.setString(6, mentor);
        ps.setString(7, grpLeader);
        ps.setString(8, mem2);
        ps.setString(9, mem3);
        ps.setString(10, mem4);
        ps.setString(11, mem5);
        ps.setString(12, userId);
        ps.executeUpdate();
    } catch (Exception e) {
        JOptionPane.showMessageDialog(null, e);
    }
}
```

Code for inputing the values for faculty:

The following is the code for the inputing the values for faculty:

```
    private void jButton3ActionPerformed(java.awt.event.ActionEvent evt) {
        teacherLogin lgf = new teacherLogin();
        lgf.setVisible(true);
        lgf.pack();
        lgf.setLocationRelativeTo(null);
        lgf.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        this.dispose();
    }
```

```
] private void fileCombo() {
        PreparedStatement ps;
        ResultSet rs;
        t = new DefaultTableModel();
        jTableGrpRecord.setSelectionMode(javax.swing.ListSelectionModel.SINGLE_SELECTION);
        //String queryl = "SELECT stationName FROM stations ";
        t.addColumn("PName");
        t.addColumn("Deptt");
        t.addColumn("Leader");
        t.addColumn("mem2");
        t.addColumn("mem3");
        t.addColumn("mem4");
        t.addColumn("mem5");
        t.addColumn("Room");
        t.addColumn("Table");
        t.addColumn("PDes");
        String query = "SELECT pName,dept,Leader,name2,name3,name4,name5,gRoom,gTable,pDes FROM student WHERE mentor=?;";
```

```
    while (rs.next()) {
        t.addRow(new Object[]{
            rs.getString(1),
            rs.getString(2),
            rs.getString(3),
            rs.getString(4),
            rs.getString(5),
            rs.getString(6),
            rs.getString(7),
            rs.getString(8),
            rs.getString(9),
            rs.getString(10)});
    }
    jTableGrpRecord.setModel(t);
    jTableGrpRecord.setAutoResizeMode(JTable.AUTO_RESIZE_OFF);
```

A.5 Concluding remarks (Summary, limitations, improvements)

SUMMARY

The GUI here is developed using the JFrame tool with Java. The GUI designed here allow us to perform insertion, deletion and user could also display the data in the database. The user can perform these operation any number of times. The exit button provides the user a way to quit the application. The execute button executes whichever query is currently there in the text area element. The queries are written in the Java back end and executed using the functions under SQL libraries imported. Connections are built by supplying an underlying driver or provider with a connection string, which is used to address a specific database or server and to provide instance and user authentication credentials. Once a connection has been built, it can be opened and closed at will, and properties can be set. The connection string consists of a set of key-value pairs, dictated by the data access interface of the data provider.

Limitations:

• Change of database software in the middle of project development it is so re develop JDBC persistence logic for another database software's.

• When inserting a tuple to check for the database a new command has to be passed every time.

Improvements:

The GUI can be made more colourful and innovative due to lack of time it can't be performed.