**A PROJECT REPORT**

**ON**

**AUTOMATIC TIMETABLE GENERATING SYSTEM** *Submitted to*

**Jawaharlal Nehru Technological University Anantapur, Ananthapuramu**

*In Partial Fulfillment of the Requirements for the Award of the Degree of*

**BACHELOR OF TECHNOLOGY**

**In**

**COMPUTER SCIENCE & ENGINEERING**

**Submitted By**

**B.K.MAMATHA 17691A0569**

**C.MOUNIKA 17691A0579**

**P.PRIYANKA 17691A0598**

Under the Guidance of

**Mr. C.NARASIMHA**

**Assistant Professor,**

**Department of Computer Science & Engineering**

****

**MADANAPALLE INSTITUTE OF TECHNOLOGY & SCIENCE**

**(UGC – AUTONOMOUS)**

**(Affiliated to JNTUA, Ananthapuramu, Approved by AICTE, New Delhi)**

**AN ISO 9001:2008 Certified Institution**

**P. B. No: 14, Angallu, Madanapalle – 517325**

**2018-2019**

**ABSTRACT**

Time table generation is tedious job for educationalist with respect to time and man power. Providing a automatic time table generator will help to generator will help to generate time table automatically proposed system of our project will help to save time. It avoids the complexity ofsetting and managing Timetable manually. The system will take various inputs like number of subjects, teachers, workload of a teacher, semester, priority of subject. By relying on these inputs, it will generate possible time tables for working days of the week for teaching faculty. The timetable is needed to be scheduled in such a way that the number of subjects in each, handled by a limited faculty provided with their slots and timings does not overlap.

**ACKNOWLEDGE**

There are many people who are associated with this project directly or indirectly whose help and timely suggestions are highly appreciable for completion of this project. First of all , we would like express my sincere gratitude to our course instructor Asst. Prof C. NARASIMHA for his supervision. It was an invaluable experience for us to be one of his students. Because of him, we have gained a careful research attitude.

We would like to thank our Prof. D Kalpana head of our department for her kind support and constant encouragements, valuable discussions which is highly commendable.

**INDEX**

**S.NO TITLE PAGENO**

**1 INTRODUCTION**

**3.1)PROBLEM STATEMENT 4**

**3.2)DESCRIPTION 4**

**3.2.1)BASIC STRUCTURE OF THE SYSTEM 4**

**2 SRS DOCUMENT**

**4.1)REQUIREMENTS 5**

**4.2)CONSTRAINTS 5**

**3 SYSTEM DESIGN**

**5.1)STATIC DIAGRAMS 6**

**5.1.1)CLASS DIAGRAMS 6**

**5.1.2)OBJECT DIAGRAMS 7**

**5.1.3)COMPONENT DIAGRAM 7**

**5.1.4)DEPLOYMENT DIAGRAM 8**

**5.2)DYNMAIC DIAGRAM 9**

**5.2.1)USECASE DIAGRAM 9**

**5.2.2)INTERACTION DIAGRAM 10**

**5.2.2.1)SEQUENCE DIAGRAM 10**

**5.2.2.2)COLLABORATION DIAGRAM 11**

**5.2.3)ACTIVITY DIAGRAM 12-13**

**5.2.4)STATE CHART DIAGRAM 14**

**4 SYSTEM IMPLEMENTATION 15**

**5 SYSTEM TESTING**

**7.1) BLACK BOX TESTING 15**

**7.2) WHITE BOX TESTING 16**

**6 RESULTS 17-23**

**7 CONCLUSION 24**

**8 APPENDEX 25-160**

**INTRODUCTION**

our project entitle ,"TIME TABLE GENERATOR AND MANAGEMENT SYSTEM" is meant to generating timetable scheduling process in colleges or in any other institutions which could minimize the human work and maximize the efficiency and the timetable was stored in a centralized server which could be easy to access everywhere**.**

**PROBLEM STATEMENT**

**.** In olden days, time tables are set manually.

**.** In this system also the time tables are set manually, but by using the application.

**.** Focuses on two modules class and faculty time tables.

**.** Detects the collisions occurred during the time table generation.

**.** The class time table module mainly focusing on the fields like year, branch, semester and section.

**.** The faculty time tables focusing on the fields on the fields like faculty id and faculty name**.**

**DESCRIPTION**

**.** Generate the timetable by satisfying all the constraints.

**.**  The admin was given with special privileges.

**.** Similarly, the faculty does have their own credentials to enter into the site and watch the class wise timetable.

**THE BASIC STRUCTURE OF THE SYSTEM**

1) Implement an effective intelligent system to generate and maintain the timetables.

2) Admin and faculty can login their own credentials and do their respective duties.

**SCOPE OF THE SYSTEM**

**.** Generating of the Timetable, Access the available Timetable(s), Modify/Delete an existing Timetable.

**.** Add/Modify the details of a department.

**.** Add/Modify time scheduling and period allocation process.

**2.SRS DOCUMENT**

**FUNCTIONAL REQUIREMENTS**

Structural things: class, interface

Annotation things: note

Classes identified: 1) Faculty

2)Section

3)Department

4)Subject

5)Admin

6)Time Scheduling

7)User ID

**HARDWARE REQUIREMENTS**

Pentium IV processor architecture with

1.512 MB RAM

2.80 GB Hard Disk Space

**SOFTWARE REQUIREMENTS**

**.** Operating System: Windows7/XP/2003 or Linux

**.** User Interface: Java

**.** Database: My SQL

**.** Designing: Rational Rose

**HARD CONSTRAINTS**

Hard constraints include "First Order Conflicts".

1. A Lecture cannot teach more than one class at the same time.

2.To generate the Timetable based on the number of period and the time schedule**.**

**SOFT CONSTRAINTS**

Some soft constraints are more important than others, and this is other specified with a priority value.

1. If possible, the lecture hours for a course should be scheduled consecutively.

2. A lunch break must be scheduled.

3. Lecturer's daily lecture hours should be restricted to be within the allowed maximum hours.

**3.SYSTEM DESIGN**

**3.1 STATIC DIAGRAMS**

**3.1.1 CLASS DIAGRAM**

A class diagrams is a type of static structure diagram that describes the structure of a system

by showing the system classes, their attributes and the relationship between the classes.

**CONTENTS**

classes, interfaces, collaborations, relationship, note, grouping things.

**REQUIRED CLASSES**

Staff

Admin

Timetable viewer

Timetable generator

Student

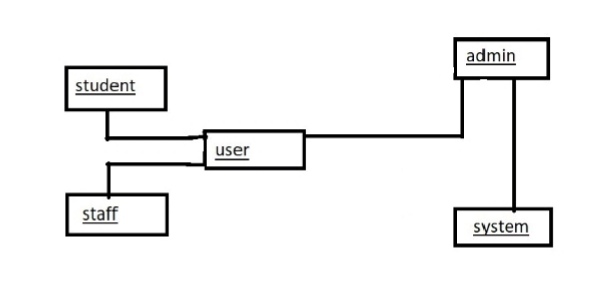


**3.1.2 OBJECT DIAGRAM**

Object diagram models the instance of classes from the class diagram.

Object diagram represent the static part of a system.

**CONTENTS**: Objects, links, annotations things, grouping things

****

**3.1.3 COMPONENT DIAGRAM**

A physical and replacable part of a system is called component. It is a represented using a tabbed rectangle.

**REQUIRED COMPONENTS**

User

Timetable generator

Admin



**3.1.4 DEPLOYMENT DIAGRAM**

It is used to represent the dynamic aspects of a system. It is represented using a node.

**REQUIRED NODES**

Admin

User

Timetable generator



**3.2 DYNAMIC DIAGRAMS**

**3.2.1 USECASE DIAGRAM**

Use case diagram are one of the five behavioural diagrams for modelling diagrams aspects of a system. It specifies what the system does.

**REQUIRED ACTORS**

Admin

Staff

Student

User



**3.2.2 INTERACTION DIAGRAM**

It is one of the dynamic behavioural diagram of UML. With which you are able to design the dynamic aspects of a system.

**INTERACTION**

An interaction is a behavioural that contains a set of messages exchanged between objects.

Interaction diagrams are represented in 2 ways.

1. Sequence diagram (Time ordering of messages)
2. Collaboration diagram (structural organization)

**3.2.2.1 SEQUENCE DIAGRAM**

The sequence diagram is one of the 2 interactions diagrams. It emphasizes on time ordering of messages.

**REQUIRED OBJECTS**

Admin

User  
Timetable generator



**3.2.2.2COLLABORATION DIAGRAM**

The collaboration diagram is one of the 2 interaction diagrams. The main focus is on structural organization of objects.

**REQUIRED THINGS**

Admin

User

Timetable generator



**3.2.3 ACTIVITY DIAGRAM**

It is similar like a flowchart. It can be used to represent the flow of execution from one activity to another activity. In case of interaction diagram, it can be represent the flow of control among objects. An activity diagram can be used to represent dynamic view of a system.

**REQUIRED SWIMLANES**

Admin

Timetable generator

User



**3.2.4 STATECHART DIAGRAM**

**STATE**

It is a behavioural thing, the group of interaction take place among objects with occurrence of events.

**STATECHART**

It is a state machine focusing on the flow of control from state to state on every occurrence of events

**REQUIRED TOOLS**

Initial state

Final state

State

Transition



**4.SYSTEM IMPLEMENTATION**

* 1. **Modules**

**Creating credentials**

If any user want to view the particular timetable they must have the valid credentials provided by admin of the institution. So the admin create credentials for the users.

**Adding classes**

To create a timetable for a particular class that class room number must to present in the database. So the admin adds the class room numbers into the database.

**Generating timetable**

Admin will generate the timetable for the added room numbers by using the available staff in the database.

He must check the generated timetable should satisfy all the constraints of the institution.

**Updating and deleting details**

If there are any updates in the user’s details, then the admin must update it time to time. For example, if any staff leaves the college no timetable must be generated by assigning a class for that faculty. So his details must be deleted in the database.

**Saving timetable in database**

The user must be able to view the required timetables saved in the database. So the admin must save the generated timetables in the database for further usage.

**SYSTEM TESTING**

**5.1 Black Box Testing**

Black Box Testing is a software Testing method in which testers evaluate the functionality of the software under test without looking at the internal code structure. This can be applied to every level of software testing such as Unit, Integration, System and Testing.

It is also called as Behavioral/Specification based/Input-output Testing

Black Box Testing Techniques:

1. Equivalence partitioning
2. Boundary Value Analysis
3. Decision Table
4. State Transition

**5.2 White Box Testing**

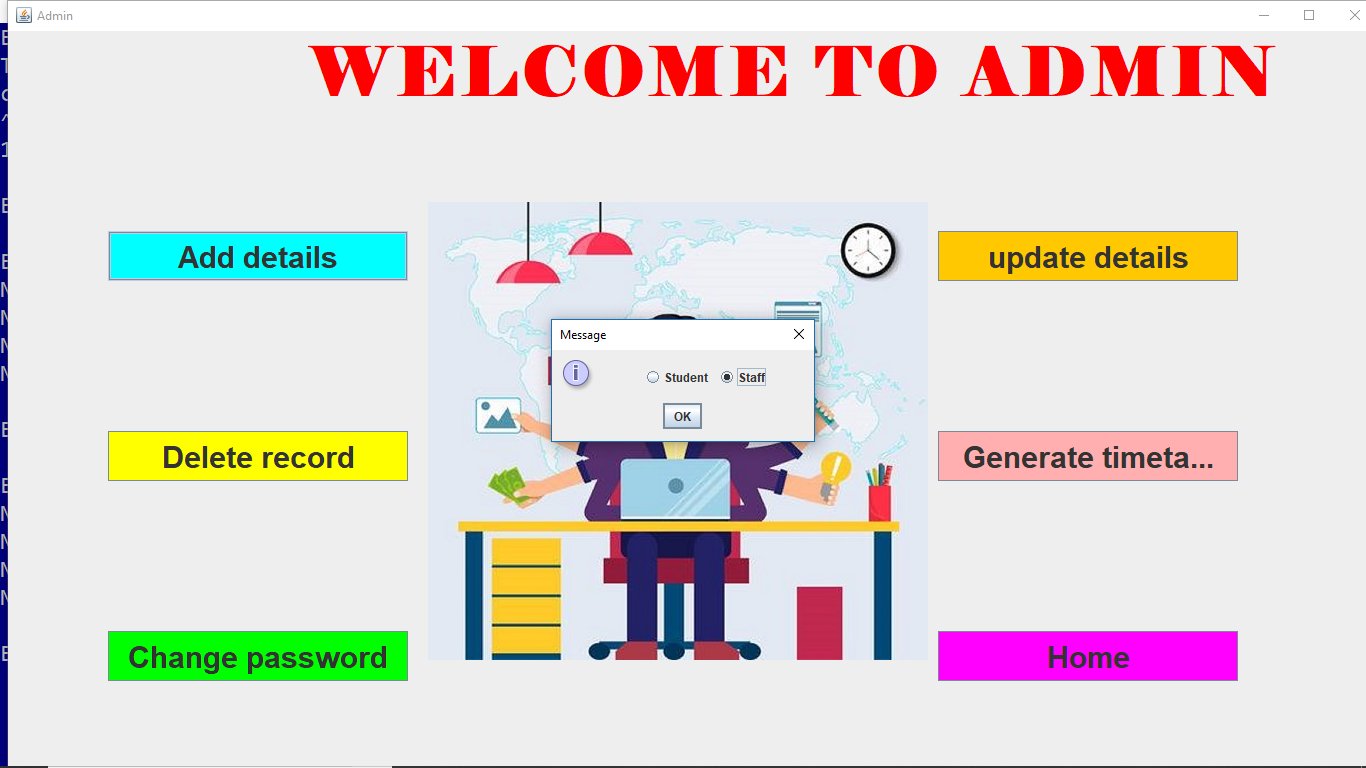
White Box Testing is based on applications internal code structure. In white-box Testing an internal perspective of the system, as well as programming skills, are used to design test cases. This testing usually of the system, as well as programming skills, are used to design test cases. This testing usually done at the unit level. It is also called as Glass Box, Clear Box, Structural Testing .

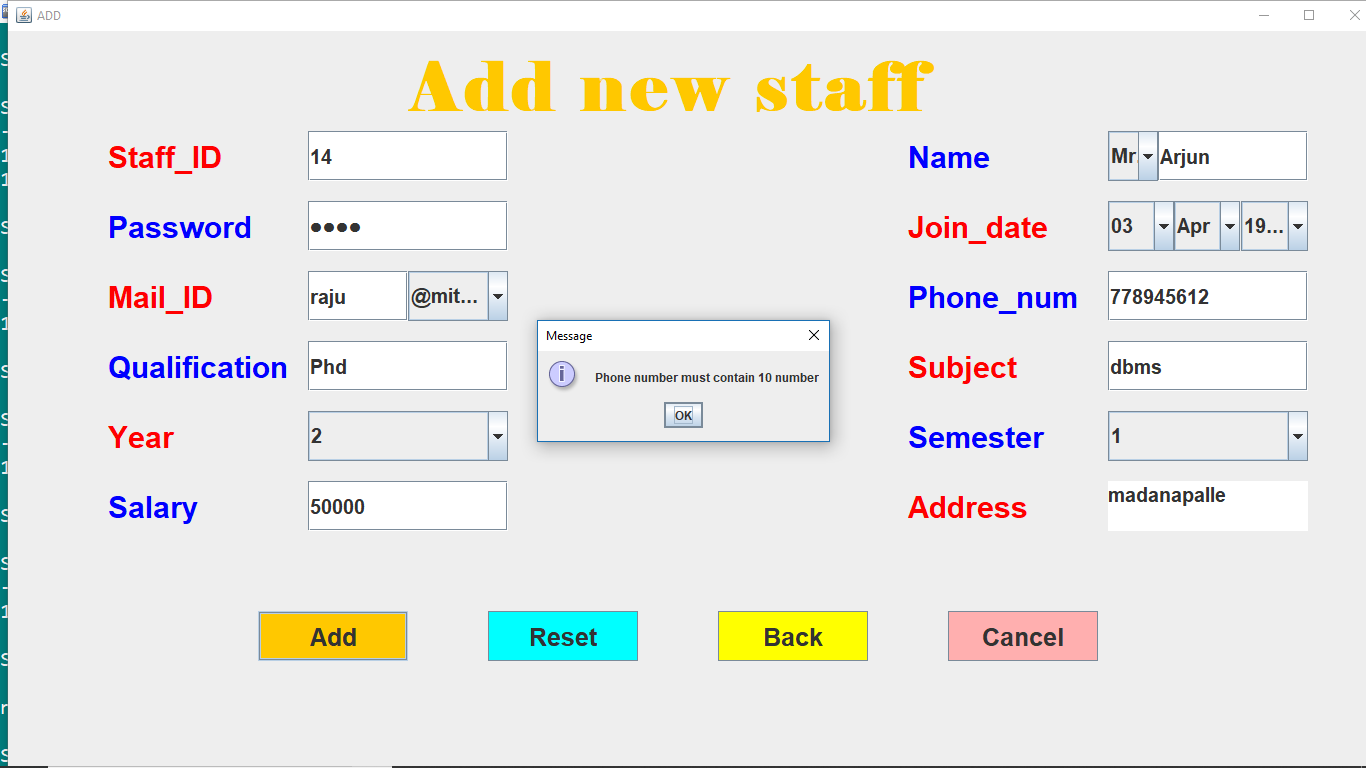
White Box Testing Techniques

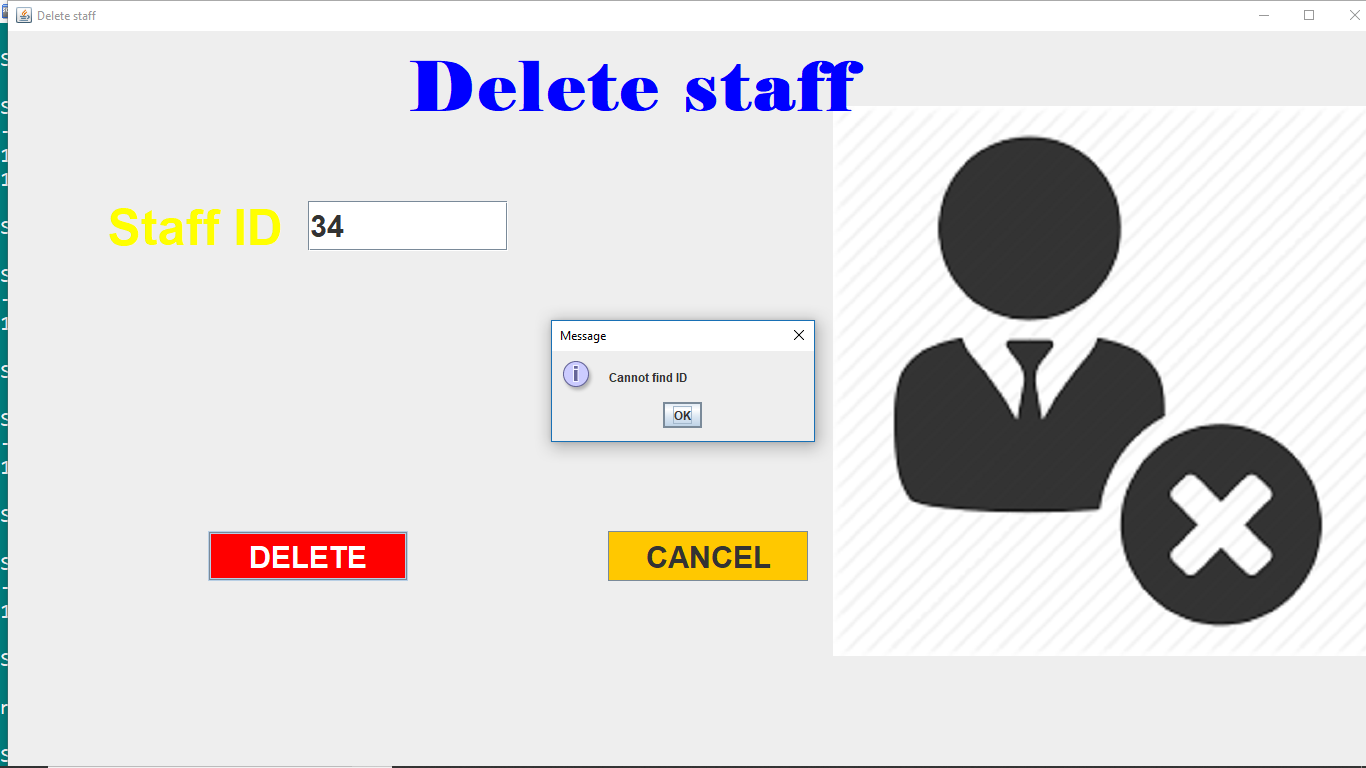
1. Statement Coverage
2. Branch Coverage
3. Path Coverage
4. **RESULTS AND ANALYSIS**

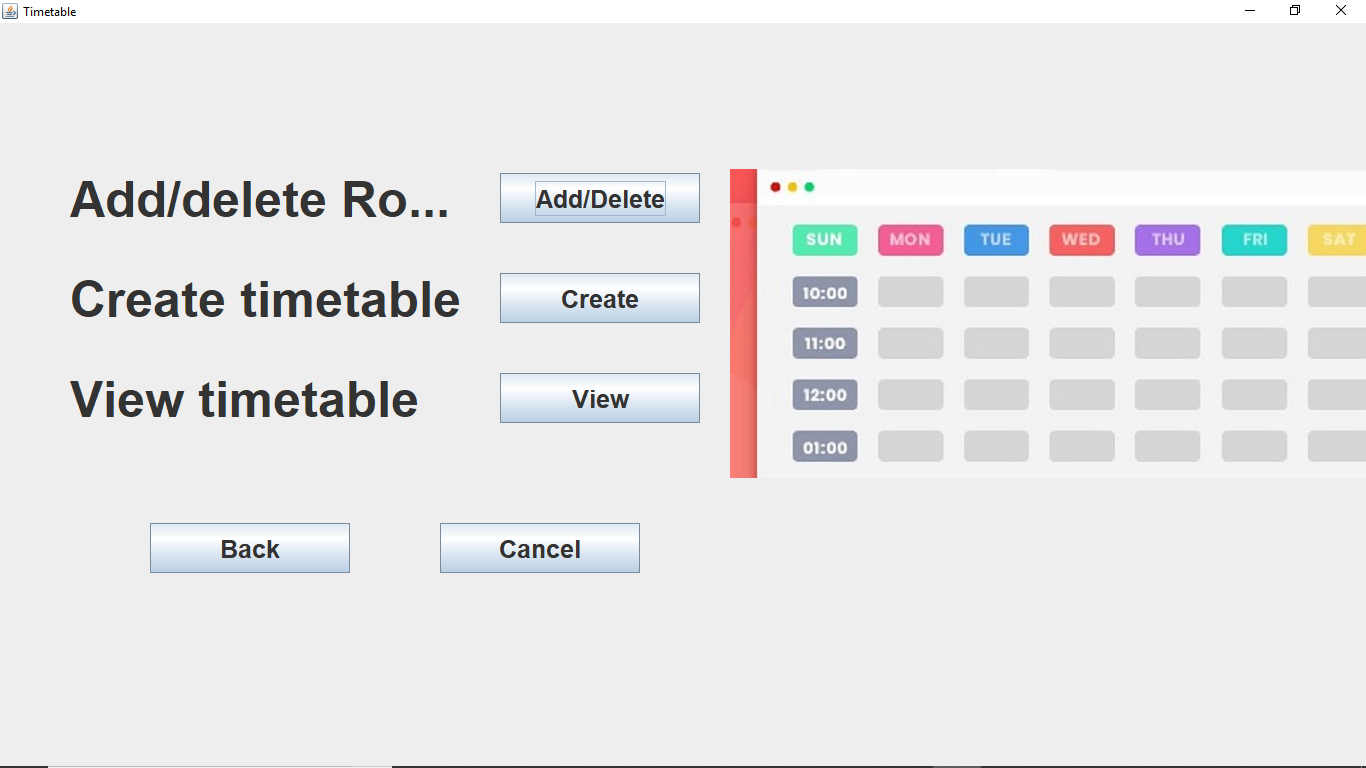


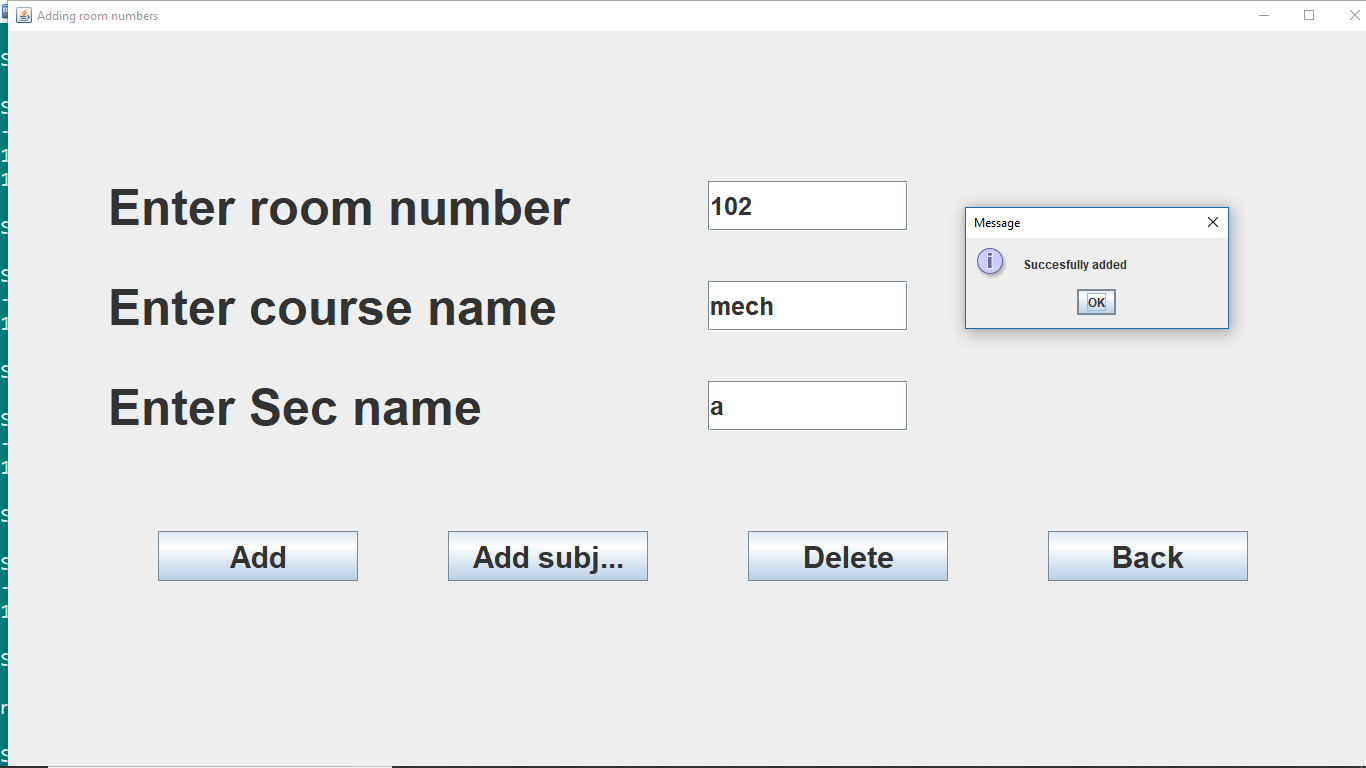


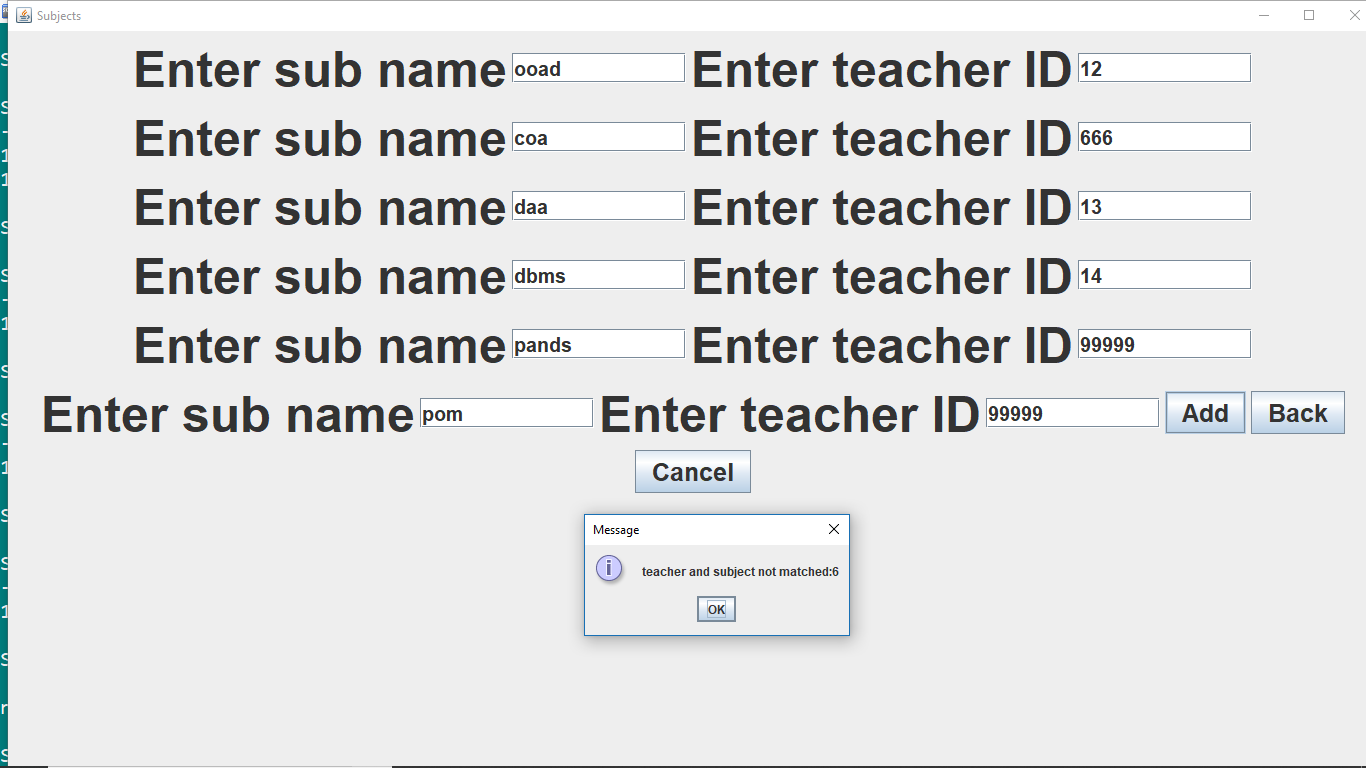


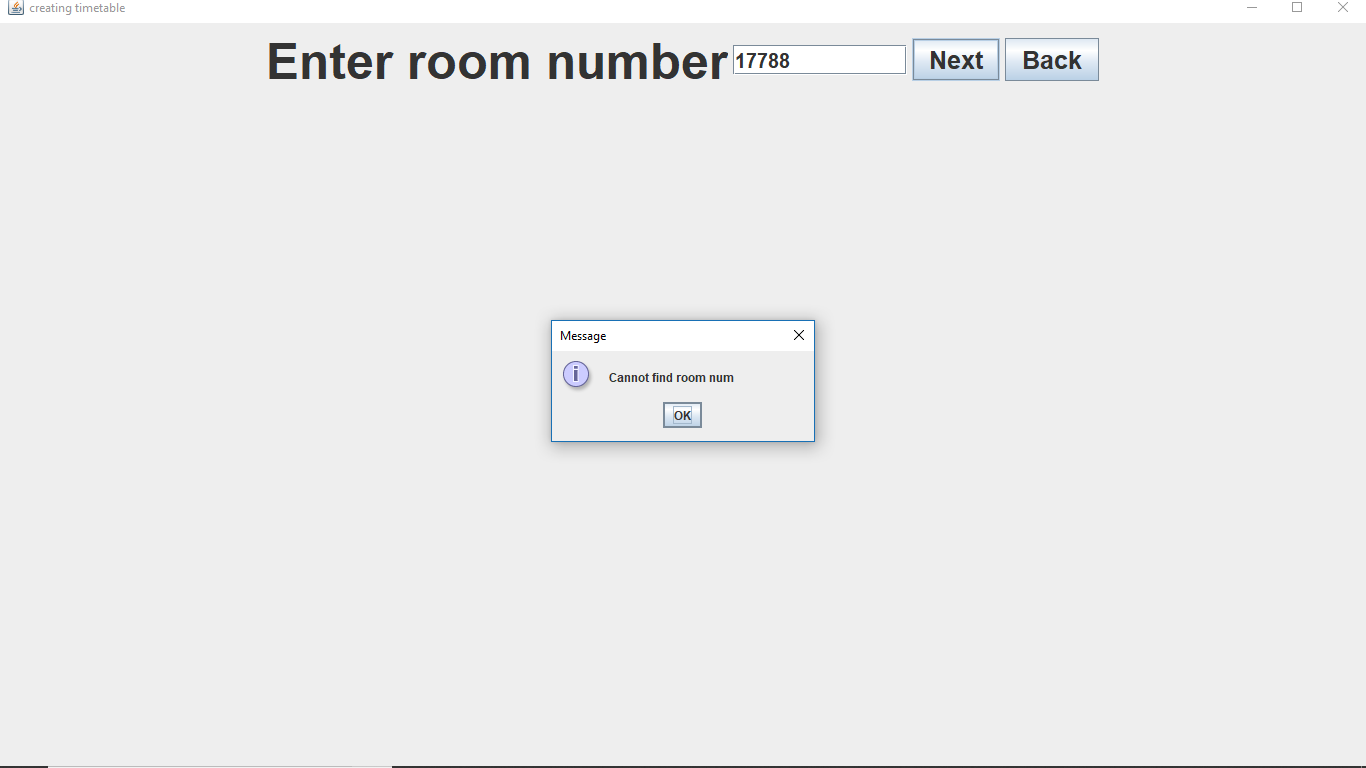


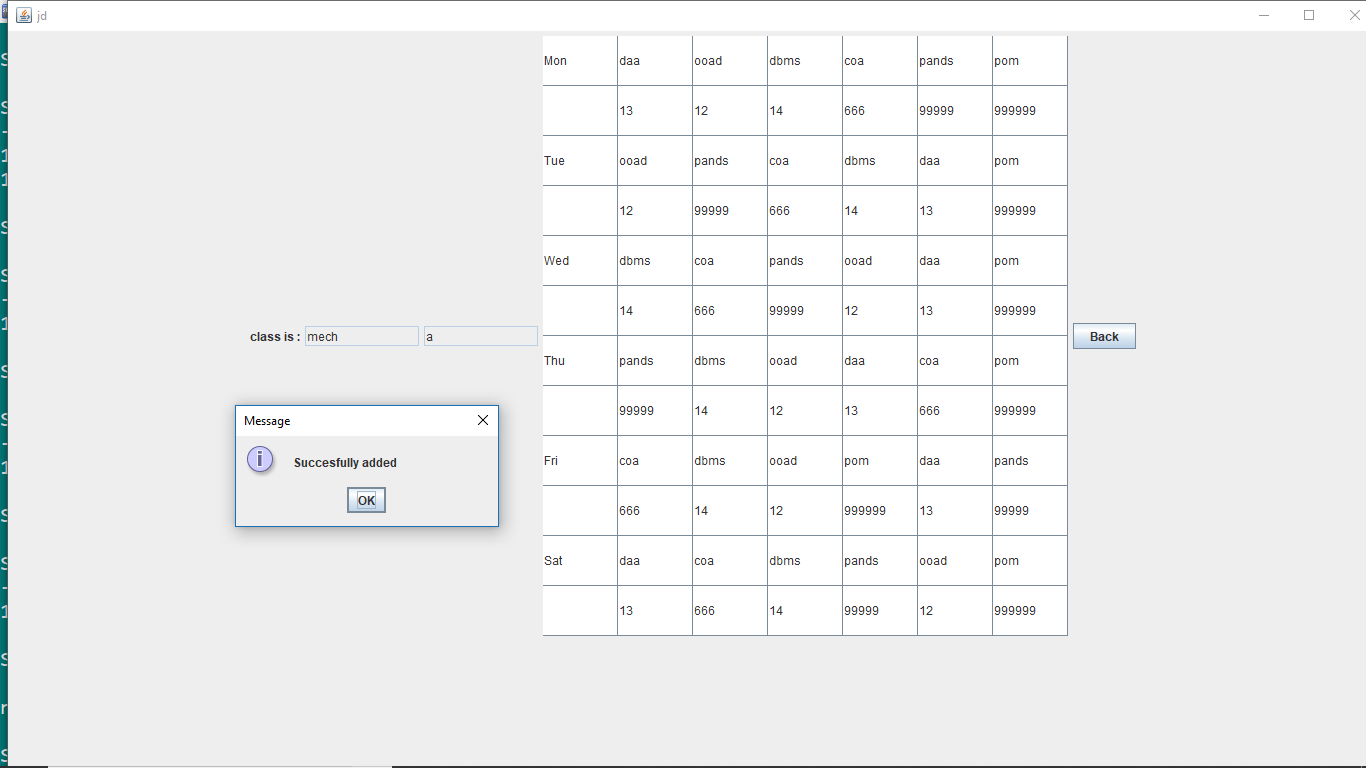


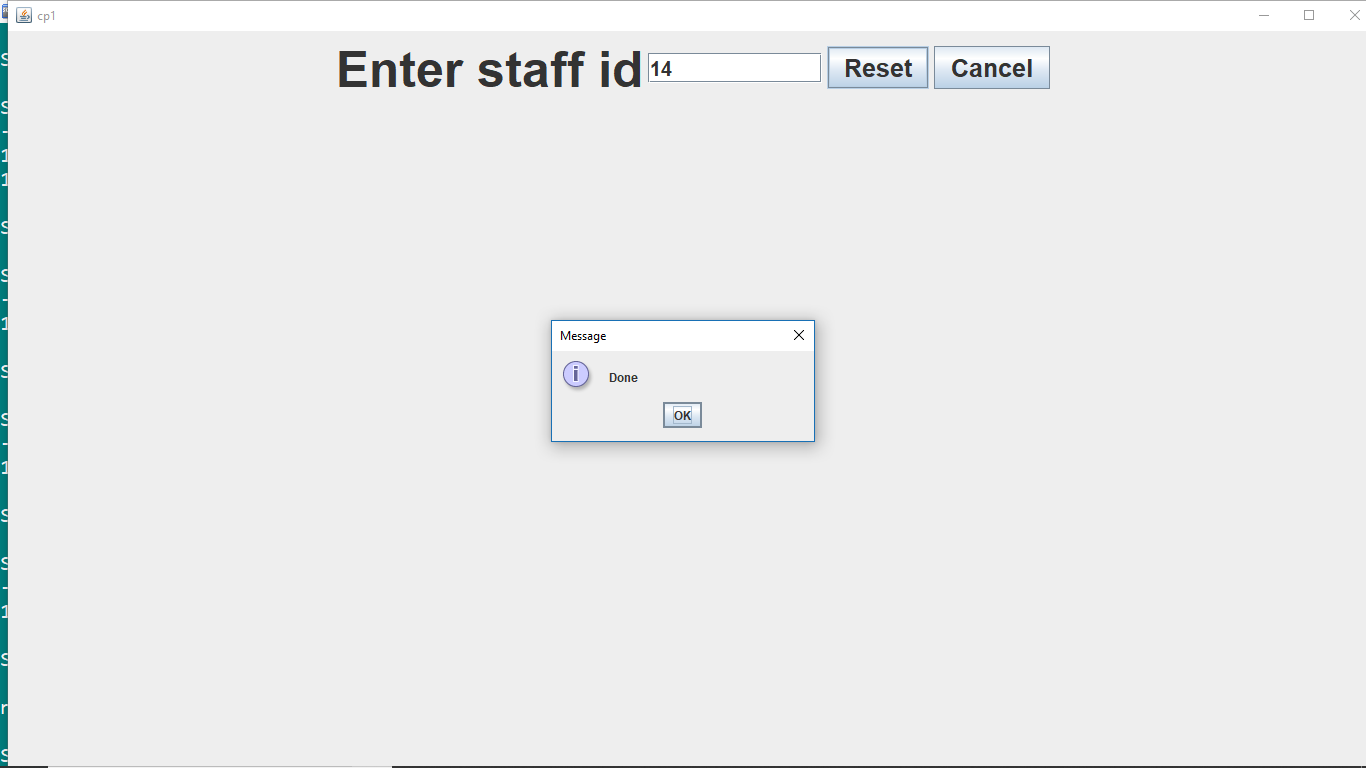


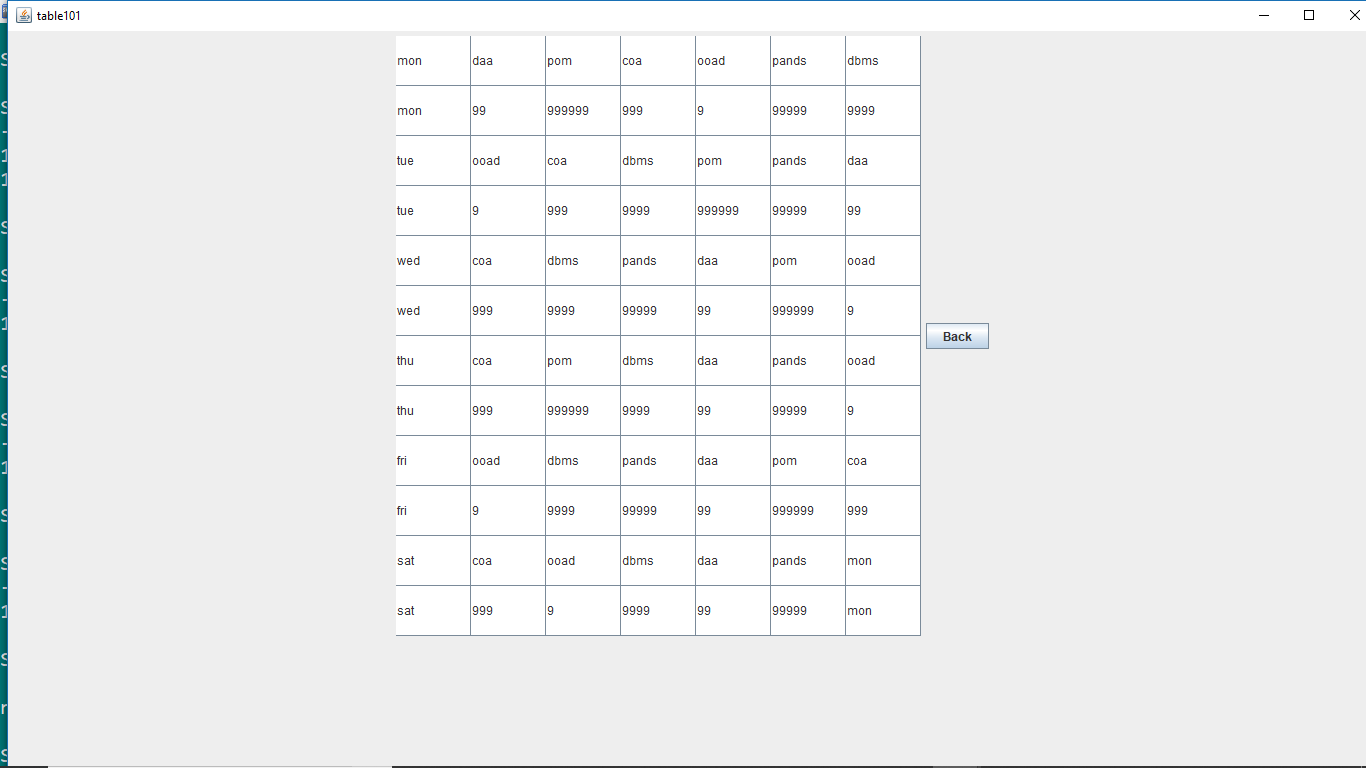




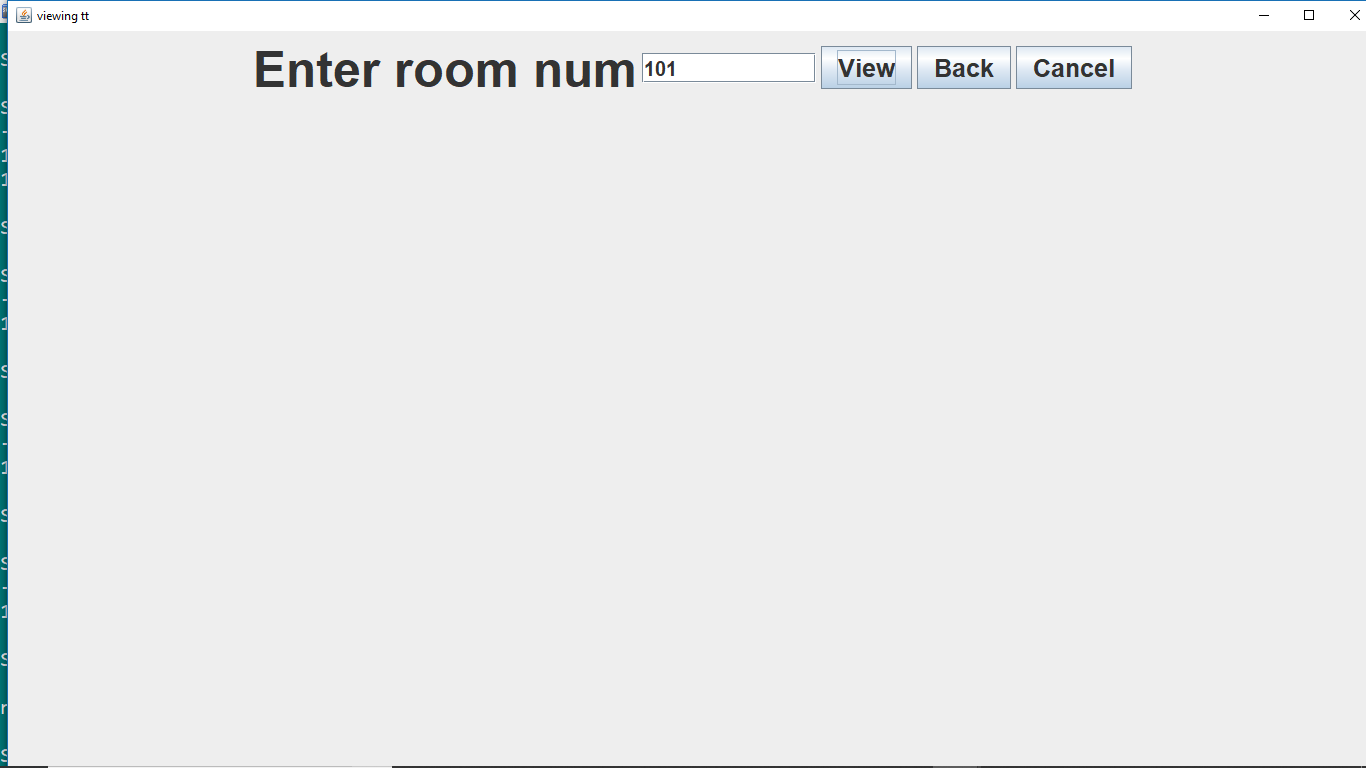


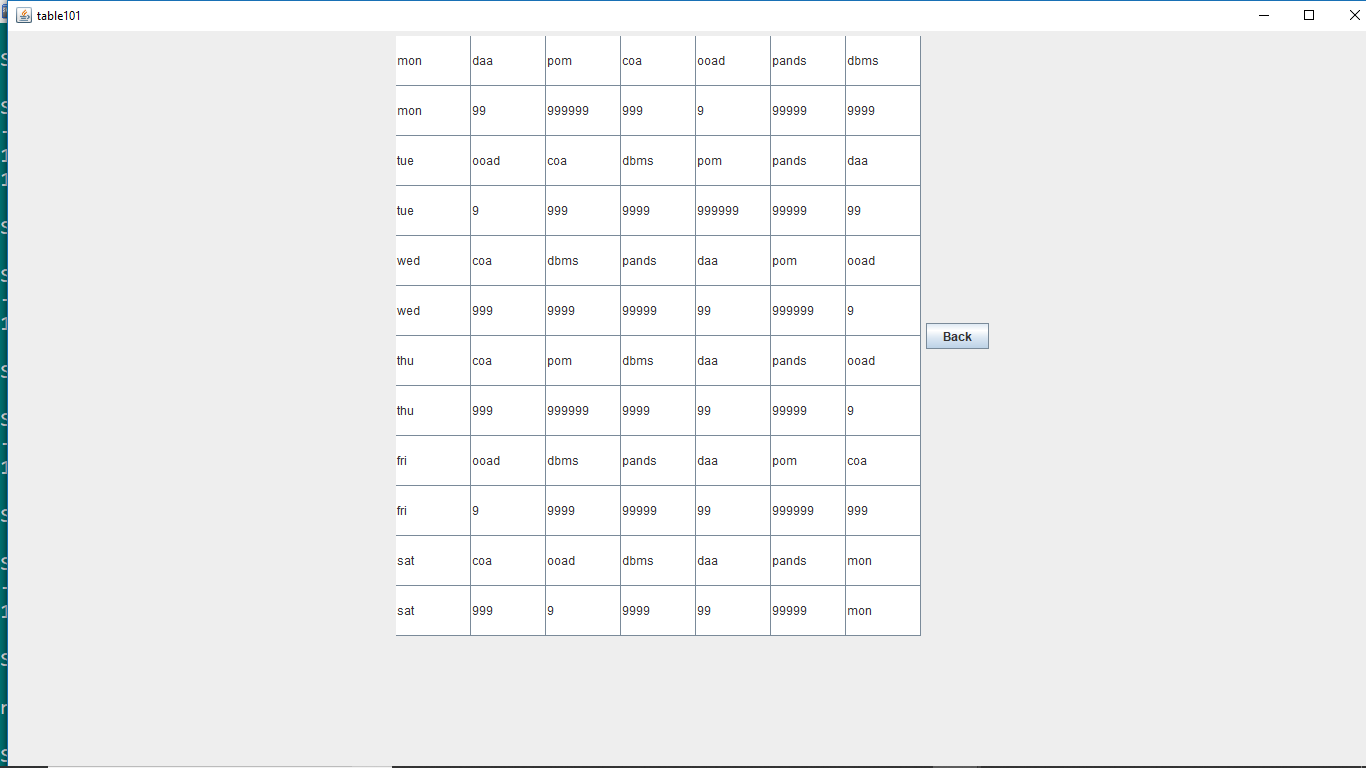












**7.CONCLUSION**

Now a days, timetable is being prepared manually in all schools and colleges which is a time waste process. Doing this manually may include many errors. To avoid this, we had built a new automatic time table generating system. Our project will generate a timetable automatically without an collisions which will minimize the human work and maximizes the efficiency. Our project is completely a time saving process.

This system can be used by any college or institution for generating a timetable.

**8.APPENDEX:**

import java.sql.\*;

import java.awt.\*;

import java.awt.event.\*;

import javax.swing.\*;

import java.util.\*;

import javax.swing.table.\*;

import java.io.\*;

import javax.swing.table.DefaultTableModel;

public class TTG implements ActionListener

{JFrame f;

JLabel l1,l2,l3,l4;

JButton b1,b2,b3,b4;

TTG(){

f=new JFrame("Timetable genrator");

f.setVisible(true);

f.setSize(1500,1500);

f.setLayout(null);

b1=new JButton("HOME");

b2=new JButton("ADMIN");

b3=new JButton("USER");

b4=new JButton("CANCEL");

l1=new JLabel("AUTOMATIC TIMETABLE GENERATOR");

ImageIcon x=new ImageIcon("ttg1.JPEG");

JLabel l=new JLabel(x);

f.setContentPane(l);

l1.setBounds(10,10,1500,70);

l2.setBounds(100,515,1000,50);

l3.setBounds(100,575,1000,50);

l4.setBounds(100,630,1500,50);

b1.setBounds(200,550,200,50);

b2.setBounds(425,550,200,50);

b3.setBounds(650,550,200,50);

b4.setBounds(875,550,200,50);

l.add(l1);l.add(b1);

l.add(b2);l.add(b3);

l.add(b4);

b1.addActionListener(this);

b2.addActionListener(this);

b3.addActionListener(this);

b4.addActionListener(this);}

public void actionPerformed(ActionEvent ae)

{if(ae.getSource().equals(b1)){

f.setVisible(true);}

if(ae.getSource().equals(b2))

{f.setVisible(false);

new Admin(f);}

if(ae.getSource().equals(b4))

{f.setVisible(false);}

if(ae.getSource().equals(b3)){

final JPanel panel1=new JPanel();

final JRadioButton button1=new JRadioButton("Student");

final JRadioButton button2=new JRadioButton("Staff");

ButtonGroup br=new ButtonGroup();

br.add(button1);

br.add(button2);

panel1.add(button1);

panel1.add(button2);

JOptionPane.showMessageDialog(null,panel1);

if(button2.isSelected()){

f.setVisible(false);

new user1(f);}

if(button1.isSelected())

{f.setVisible(false);

new user2(f);}}}

public static void main(String...aa)

{new TTG();}}

class Admin implements ActionListener

{JFrame f1,f2;

JLabel l,l1,l2,l3;

JTextField t1;

JPasswordField t2;

JButton b1,b2,b3;

String s1,s2;

Admin(JFrame f)

{try{f1=f;

f2=new JFrame("Admin homepage");

f2.setSize(1500,1500);

f2.setVisible(true);

f2.setLayout(null);

l1=new JLabel("ADMIN HOME PAGE");

l2=new JLabel("User ID");

l3=new JLabel("Password");

t1=new JTextField(20);

t2=new JPasswordField(20);

b1=new JButton("BACK");

b2=new JButton("LOGIN");

b3=new JButton("CANCEL");

ImageIcon x=new ImageIcon("admin3.JPEG");

l=new JLabel(x);

l1.setBounds(300,10,1000,60);

l2.setBounds(250,150,300,50);

l3.setBounds(250,250,300,50);

t1.setBounds(500,150,200,50);

t2.setBounds(500,250,200,50);

b1.setBounds(50,500,200,50);

b2.setBounds(270,500,200,50);

b3.setBounds(490,500,200,50);

l.setBounds(500,100,1000,500);

f2.add(l1);f2.add(l2);

f2.add(t1);f2.add(l3);

f2.add(t2);f2.add(b1);

f2.add(b2);f2.add(b3);

f2.add(l);

b1.addActionListener(this);

b2.addActionListener(this);

b3.addActionListener(this);}

catch(Exception j){}}

public void actionPerformed(ActionEvent ae)

{s1=t1.getText();

s2=t2.getText();

if(ae.getSource().equals(b1))

{f2.setVisible(false);

f1.setVisible(true);}

if(ae.getSource().equals(b3))

{f2.setVisible(false);}

if(ae.getSource().equals(b2))

{if(s1.equals("code")&&s2.equals("1234"))

{f2.setVisible(false);

new Hj(f1,f2);}

else

JOptionPane.showMessageDialog(null,"Invalid login");}}}

class Hj implements ActionListener

{JFrame f11,f22,f3;

JLabel l,l1;

JButton b1,b2,b3,b4,b5,b6;

JPanel panel;

Hj(JFrame f1,JFrame f2)

{f11=f1;f22=f2;

f3=new JFrame("Admin");

f3.setVisible(true);

f3.setLayout(null);

f3.setSize(1500,1500);

l1=new JLabel("WELCOME TO ADMIN");

b1=new JButton("Add details");

b2=new JButton("update details");

b3=new JButton("Delete record");

b4=new JButton("Generate timetable");

b5=new JButton("Change password");

b6=new JButton("Home");

ImageIcon x=new ImageIcon("admin4.JPEG");

l=new JLabel(x);

panel=new JPanel();

l.setBounds(170,150,1000,500);

b1.setBounds(100,200,300,50);

b2.setBounds(930,200,300,50);

b3.setBounds(100,400,300,50);

b4.setBounds(930,400,300,50);

b5.setBounds(100,600,300,50);

b6.setBounds(930,600,300,50);

l1.setBounds(300,10,1000,60);

f3.add(l);f3.add(l1);

f3.add(b1);f3.add(b2);

f3.add(b3);f3.add(b4);

f3.add(b5);f3.add(b6);

b1.addActionListener(this);

b2.addActionListener(this);

b3.addActionListener(this);

b4.addActionListener(this);

b5.addActionListener(this);

b6.addActionListener(this);}

public void actionPerformed(ActionEvent ae)

{if(ae.getSource().equals(b5))

{final JPanel panel1=new JPanel();

final JRadioButton button1=new JRadioButton("Student");

final JRadioButton button2=new JRadioButton("Staff");

ButtonGroup br=new ButtonGroup();

br.add(button1);

br.add(button2);

panel1.add(button1);

panel1.add(button2);

JOptionPane.showMessageDialog(null,panel1);

if(button2.isSelected())

{f3.setVisible(false);

new cp1(f11,f22,f3);}

if(button1.isSelected())

{f3.setVisible(false);

new cp2(f11,f22,f3);}}

if(ae.getSource().equals(b4))

{f3.setVisible(false);

new flex(f3);}

if(ae.getSource().equals(b6))

{f3.setVisible(false);

f11.setVisible(true);}

if(ae.getSource().equals(b1))

{final JPanel panel1=new JPanel();

final JRadioButton button1=new JRadioButton("Student");

final JRadioButton button2=new JRadioButton("Staff");

ButtonGroup br=new ButtonGroup();

br.add(button1);

br.add(button2);

panel1.add(button1);

panel1.add(button2);

JOptionPane.showMessageDialog(null,panel1);

if(button2.isSelected()){

f3.setVisible(false);

new Addstaff(f11,f22,f3);}

if(button1.isSelected())

{f3.setVisible(false);

new Addstudent(f11,f22,f3);}}

if(ae.getSource().equals(b3))

{final JPanel panel1=new JPanel();

final JRadioButton button1=new JRadioButton("Student");

final JRadioButton button2=new JRadioButton("Staff");

ButtonGroup br=new ButtonGroup();

br.add(button1);

br.add(button2);

panel1.add(button1);

panel1.add(button2);

JOptionPane.showMessageDialog(null,panel1);

if(button2.isSelected()){

f3.setVisible(false);

new Deletestaff(f11,f22,f3);}

if(button1.isSelected())

{f3.setVisible(false);

new Deletestudent(f11,f22,f3);}}

if(ae.getSource().equals(b2))

{final JPanel panel1=new JPanel();

final JRadioButton button1=new JRadioButton("Student");

final JRadioButton button2=new JRadioButton("Staff");

ButtonGroup br=new ButtonGroup();

br.add(button1);

br.add(button2);

panel1.add(button1);

panel1.add(button2);

JOptionPane.showMessageDialog(null,panel1);

if(button2.isSelected()){

f3.setVisible(false);

new Updatestaff1(f11,f22,f3);}

if(button1.isSelected())

{f3.setVisible(false);

new Updatestudent1(f11,f22,f3);}}}}

class Addstaff implements ActionListener

{JFrame f111,f222,f333,f4;

JLabel l1,l2,l3,l4,l5,l6,l7,l8,l9,l10,l11,l12,l13;

JTextField t1,t12,t2,t4,t5,t6,t7,t8,t9,t10,t11;

JPasswordField t3;

JButton b1,b2,b3,b4;

JTextArea txt;

JComboBox x,d1,d2,d3,mi,y,sem;

Addstaff(JFrame f11,JFrame f22,JFrame f3)

{f111=f11;f222=f22;f333=f3;

f4=new JFrame("ADD");

f4.setSize(1500,1500);

f4.setVisible(true);

f4.setLayout(null);

l1=new JLabel("Staff\_ID");

l2=new JLabel("Name");

l3=new JLabel("Password");

l4=new JLabel("Join\_date");

l5=new JLabel("Mail\_ID");

l6=new JLabel("Phone\_num");

l7=new JLabel("Qualification");

l8=new JLabel("Subject");

l9=new JLabel("Year");

l10=new JLabel("Semester");

l11=new JLabel("Salary");

l12=new JLabel("Address");

l13=new JLabel("Add new staff");

txt=new JTextArea(100,100);

x=new JComboBox();

mi=new JComboBox();

d1=new JComboBox();

d2=new JComboBox();

d3=new JComboBox();

y=new JComboBox();

sem=new JComboBox();

t1=new JTextField(20);

t12=new JTextField(20);

t2=new JTextField(20);

t4=new JTextField(20);

t5=new JTextField(20);

t6=new JTextField(20);

t7=new JTextField(20);

t8=new JTextField(20);

t9=new JTextField(20);

t10=new JTextField(20);

t11=new JTextField(20);

t3=new JPasswordField(20);

b1=new JButton("Add");

b2=new JButton("Reset");

b3=new JButton("Back");

b4=new JButton("Cancel");

x.addItem("Mr.");

x.addItem("Ms.");

x.addItem("Dr.");

x.addItem("Prof.");

d1.addItem("01");d1.addItem("02");

d1.addItem("03");d1.addItem("04");

d1.addItem("05");d1.addItem("06");

d1.addItem("07");d1.addItem("08");

d1.addItem("08");d1.addItem("10");

d1.addItem("11");d1.addItem("12");

d1.addItem("13");d1.addItem("14");

d1.addItem("15");d1.addItem("16");

d1.addItem("17");d1.addItem("18");

d1.addItem("19");d1.addItem("20");

d1.addItem("21");d1.addItem("22");

d1.addItem("23");d1.addItem("24");

d1.addItem("25");d1.addItem("26");

d1.addItem("27");d1.addItem("28");

d1.addItem("29");d1.addItem("30");

d1.addItem("31");d2.addItem("Jan");

d2.addItem("Feb");d2.addItem("Mar");

d2.addItem("Apr");d2.addItem("May");

d2.addItem("Jun");d2.addItem("Jul");

d2.addItem("Aug");d2.addItem("Sep");

d2.addItem("Oct");d2.addItem("Nov");

d2.addItem("Dec");d3.addItem("1998");

d3.addItem("1999");d3.addItem("2000");

d3.addItem("2001");d3.addItem("2002");

d3.addItem("2003");d3.addItem("2004");

d3.addItem("2005");d3.addItem("2006");

d3.addItem("2007");d3.addItem("2008");

d3.addItem("2009");d3.addItem("2010");

d3.addItem("2011");d3.addItem("2012");

d3.addItem("2013");d3.addItem("2014");

d3.addItem("2015");d3.addItem("2016");

d3.addItem("2017");d3.addItem("2018");

d3.addItem("2019");y.addItem("1");

y.addItem("2");y.addItem("3");

y.addItem("4");sem.addItem("1");

sem.addItem("2");mi.addItem("@gmail.com");

mi.addItem("@mits.ac.in");

l13.setBounds(400,5,1000,100);

l1.setBounds(100,100,200,50);

t1.setBounds(300,100,200,50);

l2.setBounds(900,100,200,50);

x.setBounds(1100,100,50,50);

t2.setBounds(1150,100,150,50);

l3.setBounds(100,170,200,50);

t3.setBounds(300,170,200,50);

l4.setBounds(900,170,200,50);

d1.setBounds(1100,170,66,50);

d2.setBounds(1166,170,66,50);

d3.setBounds(1233,170,67,50);

l5.setBounds(100,240,200,50);

t5.setBounds(300,240,100,50);

mi.setBounds(400,240,100,50);

l6.setBounds(900,240,200,50);

t6.setBounds(1100,240,200,50);

l7.setBounds(100,310,200,50);

t7.setBounds(300,310,200,50);

l8.setBounds(900,310,200,50);

t8.setBounds(1100,310,200,50);

l9.setBounds(100,380,200,50);

y.setBounds(300,380,200,50);

l10.setBounds(900,380,200,50);

sem.setBounds(1100,380,200,50);

l11.setBounds(100,450,200,50);

t11.setBounds(300,450,200,50);

l12.setBounds(900,450,200,50);

txt.setBounds(1100,450,200,50);

b1.setBounds(250,580,150,50);

b2.setBounds(480,580,150,50);

b3.setBounds(710,580,150,50);

b4.setBounds(940,580,150,50);

f4.add(txt);f4.add(y);

f4.add(sem);f4.add(mi);

f4.add(d1);f4.add(d2);

f4.add(d3);f4.add(x);

f4.add(l13);f4.add(l1);

f4.add(t12);f4.add(l2);

f4.add(t1);f4.add(l3);

f4.add(t2);

f4.add(l4);f4.add(t3);

f4.add(l5);f4.add(t4);

f4.add(l6);f4.add(t5);

f4.add(l7);f4.add(t6);

f4.add(l8);f4.add(t7);

f4.add(l9);f4.add(t8);

f4.add(l10);

f4.add(t9);f4.add(l11);

f4.add(t10);f4.add(l12);

f4.add(t11);f4.add(b1);

f4.add(b2);f4.add(b3);

f4.add(b4);b1.addActionListener(this);

b2.addActionListener(this);b3.addActionListener(this);

b4.addActionListener(this);}public void actionPerformed(ActionEvent ae)

{String s1=t1.getText();String s2=x.getSelectedItem().toString()+t2.getText();

String s3=t3.getText();

String s41=d1.getSelectedItem().toString();

String s42=d2.getSelectedItem().toString();

String s43=d3.getSelectedItem().toString();

String s5=t5.getText()+mi.getSelectedItem().toString();

String s6=t6.getText();

String s7=t7.getText();

String s8=t8.getText();

String s9=y.getSelectedItem().toString();

String s10=sem.getSelectedItem().toString();

String s11=t11.getText();

String s12=txt.getText();

String s4=s41+"-"+s42+"-"+s43;

if(ae.getSource().equals(b2))

{t1.setText("");

t2.setText("");

t3.setText("");

t4.setText("");

t5.setText("");

t6.setText("");

t7.setText("");

t8.setText("");

t9.setText("");

t10.setText("");

t11.setText("");

t12.setText("");

txt.setText("");}

if(ae.getSource().equals(b3))

{f4.setVisible(false);

f333.setVisible(true);}

if(ae.getSource().equals(b4))

{f4.setVisible(false);

f111.setVisible(true);}

if(ae.getSource().equals(b1))

{if(t1.getText().equals(""))

JOptionPane.showMessageDialog(null,"Fill the fields with \* mark:staff id");

else if(t2.getText().equals(""))

JOptionPane.showMessageDialog(null,"Fill the fields with \* mark:name");

else if(t3.getText().equals(""))

JOptionPane.showMessageDialog(null,"Fill the fields with \* mark:password");

else if(t5.getText().equals(""))

JOptionPane.showMessageDialog(null,"Fill the fields with \* mark:mail id");

else if(t6.getText().equals(""))

JOptionPane.showMessageDialog(null,"Fill the fields with \* mark:phone num");

else if(t7.getText().equals(""))

JOptionPane.showMessageDialog(null,"Fill the fields with \* mark:qualification");

else if(t8.getText().equals(""))

JOptionPane.showMessageDialog(null,"Fill the fields with \* mark:subject");

else if(s9.equals(""))

JOptionPane.showMessageDialog(null,"Fill the fields with \* mark:year");

else if(s10.equals(""))

JOptionPane.showMessageDialog(null,"Fill the field:semester");

else if(t11.getText().equals(""))

JOptionPane.showMessageDialog(null,"Fill the fields with \* mark:salary");

else if(txt.equals(""))

JOptionPane.showMessageDialog(null,"Fill the fields with \* mark:address");

else

{int phlen=s6.length();

int bbb=0;

int kk=0,lop=0;

int pp=0;

int ww=s11.length();

try

{if(s6.length()>10||s6.length()<10)

JOptionPane.showMessageDialog(null,"Phone number must contain 10 number");

else

{for(int i=0;i<s6.length();i++)

{if(s6.charAt(i)>=48&&s6.charAt(i)<=57)

bbb++;}

if(bbb<10)

JOptionPane.showMessageDialog(null,"Phone number doesnot allow characters");}}

catch(Exception aa){}

int lo=s3.length();

try

{for(int i=0;i<lo;i++)

{if(s3.charAt(i)>=65&&s3.charAt(i)<=90)

kk++;

if(s3.charAt(i)>=48&&s3.charAt(i)<=57)

lop++;}

if(kk==0)

JOptionPane.showMessageDialog(null,"Password must contain atleast one letter in uppercase");

else if(lop==0)

JOptionPane.showMessageDialog(null,"Password must contain atleast one number");

else{}}

catch(Exception kkkk){}

int ssss=Integer.parseInt(s43);

try

{if(s42.equals("Apr")||s42.equals("Jun")||s42.equals("Sep")||s42.equals("Nov"))

{if(s41.equals("31"))

JOptionPane.showMessageDialog(null,"Not valid date");}

if(ssss%4!=0)

{if(s42.equals("Feb"))

{if(s41.equals("31")||s41.equals("30")||s41.equals("29"))

JOptionPane.showMessageDialog(null,"Not valid date");}}

if(ssss%4==0)

{if(s42.equals("Feb"))

{if(s41.equals("31")||s41.equals("30"))

JOptionPane.showMessageDialog(null,"Not valid date");}}}

catch(Exception klop){}

try

{for(int o=0;o<ww;o++)

{if(s11.charAt(o)>=48&&s11.charAt(o)<=57)

{pp++;}}

if(ww==pp){}

else

JOptionPane.showMessageDialog(null,"Salary allows only numeric");}

catch(Exception kjolm){}

try

{DriverManager.registerDriver(new oracle.jdbc.OracleDriver());

Connection c1=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","system","password");

PreparedStatement s=c1.prepareStatement("select sid from addstaff ");

ResultSet rs=s.executeQuery();

String s111;

int op=0;

while(rs.next())

{s111=rs.getString("sid");

if(s1.equals(s111))

op=1;}

if(op==1)

{JOptionPane.showMessageDialog(null,"Already existing ID");}

else

{try

{if(phlen==10&&bbb==10&&kk!=0&&lop!=0&&ww==pp)

{DriverManager.registerDriver(new oracle.jdbc.OracleDriver());

Connection c=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","system","password");

PreparedStatement ps=c.prepareStatement("insert into addstaff values(?,?,?,?,?,?,?,?,?,?,?,?)");

ps.setString(1,s1);

ps.setString(2,s2);

ps.setString(3,s3);

ps.setString(4,s4);

ps.setString(5,s5);

ps.setString(6,s6);

ps.setString(7,s7);

ps.setString(8,s8);

ps.setString(9,s9);

ps.setString(10,s10);

ps.setString(11,s11);

ps.setString(12,s12);

int m=ps.executeUpdate();

if(m==1)

{JOptionPane.showMessageDialog(null,"Succesfully added");

t1.setText("");

t2.setText("");

t3.setText("");

t4.setText("");

t5.setText("");

t6.setText("");

t7.setText("");

t8.setText("");

t9.setText("");

t10.setText("");

t11.setText("");

txt.setText("");}

else

JOptionPane.showMessageDialog(null,"not added");

ps.close();

c.close();}}

catch(Exception jklo){}}

rs.close();

c1.close();}

catch(Exception jj){JOptionPane.showMessageDialog(null,jj);}}}}}

class Addstudent implements ActionListener

{JFrame f111,f222,f333,f4;

JLabel l1,l2,l3,l4,l5,l6,l7,l8,l9,l10,l11,l12,l13;

JTextField t1,t12,t2,t4,t5,t6,t7,t8,t9,t10,t11;

JPasswordField t3;

JButton b1,b2,b3,b4;

JComboBox x,d1,d2,d3,mi,y,sem,br,cat,bl;

Addstudent(JFrame f11,JFrame f22,JFrame f3)

{f111=f11;f222=f22;f333=f3;

f4=new JFrame("ADD");

f4.setSize(1500,1500);

f4.setVisible(true);

f4.setLayout(null);

l1=new JLabel("Register\_num");

l2=new JLabel("Name");

l3=new JLabel("Password");

l4=new JLabel("Join\_date");

l5=new JLabel("Mail\_ID");

l6=new JLabel("Phone\_num");

l7=new JLabel("Branch");

l8=new JLabel("Category");

l9=new JLabel("Year");

l10=new JLabel("Semester");

l11=new JLabel("Backlogs");

l12=new JLabel("Parents\_num");

l13=new JLabel("Add new student");

br=new JComboBox();

cat=new JComboBox();

bl=new JComboBox();

x=new JComboBox();

mi=new JComboBox();

d1=new JComboBox();

d2=new JComboBox();

d3=new JComboBox();

y=new JComboBox();

sem=new JComboBox();

t1=new JTextField(20);

t12=new JTextField(20);

t2=new JTextField(20);

t4=new JTextField(20);

t5=new JTextField(20);

t6=new JTextField(20);

t7=new JTextField(20);

t8=new JTextField(20);

t9=new JTextField(20);

t10=new JTextField(20);

t11=new JTextField(20);

t12=new JTextField(20);

t3=new JPasswordField(20);

b1=new JButton("Add");

b2=new JButton("Reset");

b3=new JButton("Back");

b4=new JButton("Cancel");

br.addItem("CSE");

br.addItem("EEE");

br.addItem("MECH");

br.addItem("ECE");

br.addItem("CIVIL");

br.addItem("CSIT");

cat.addItem("CONVENER");

cat.addItem("MANAGEMENT");

bl.addItem("0");

bl.addItem("1");

bl.addItem("2");

bl.addItem("3");

bl.addItem("4");

bl.addItem("5");

bl.addItem("6");

x.addItem("Mr.");

x.addItem("Ms.");

d1.addItem("01");

d1.addItem("02");

d1.addItem("03");

d1.addItem("04");

d1.addItem("05");

d1.addItem("06");

d1.addItem("07");

d1.addItem("08");

d1.addItem("08");

d1.addItem("10");

d1.addItem("11");

d1.addItem("12");

d1.addItem("13");

d1.addItem("14");

d1.addItem("15");

d1.addItem("16");

d1.addItem("17");

d1.addItem("18");

d1.addItem("19");

d1.addItem("20");

d1.addItem("21");

d1.addItem("22");

d1.addItem("23");

d1.addItem("24");

d1.addItem("25");

d1.addItem("26");

d1.addItem("27");

d1.addItem("28");

d1.addItem("29");

d1.addItem("30");

d1.addItem("31");

d2.addItem("Jan");

d2.addItem("Feb");

d2.addItem("Mar");

d2.addItem("Apr");

d2.addItem("May");

d2.addItem("Jun");

d2.addItem("Jul");

d2.addItem("Aug");

d2.addItem("Sep");

d2.addItem("Oct");

d2.addItem("Nov");

d2.addItem("Dec");

d3.addItem("1998");

d3.addItem("1999");

d3.addItem("2000");

d3.addItem("2001");

d3.addItem("2002");

d3.addItem("2003");

d3.addItem("2004");

d3.addItem("2005");

d3.addItem("2006");

d3.addItem("2007");

d3.addItem("2008");

d3.addItem("2009");

d3.addItem("2010");

d3.addItem("2011");

d3.addItem("2012");

d3.addItem("2013");

d3.addItem("2014");

d3.addItem("2015");

d3.addItem("2016");

d3.addItem("2017");

d3.addItem("2018");

d3.addItem("2019");

y.addItem("1");

y.addItem("2");

y.addItem("3");

y.addItem("4");

sem.addItem("1");

sem.addItem("2");

mi.addItem("@gmail.com");

mi.addItem("@mits.ac.in");

l13.setBounds(400,5,1000,100);

l1.setBounds(100,100,200,50);

t1.setBounds(300,100,200,50);

l2.setBounds(900,100,200,50);

x.setBounds(1100,100,50,50);

t2.setBounds(1150,100,150,50);

l3.setBounds(100,170,200,50);

t3.setBounds(300,170,200,50);

l4.setBounds(900,170,200,50);

d1.setBounds(1100,170,66,50);

d2.setBounds(1166,170,66,50);

d3.setBounds(1233,170,67,50);

l5.setBounds(100,240,200,50);

t5.setBounds(300,240,100,50);

mi.setBounds(400,240,100,50);

l6.setBounds(900,240,200,50);

t6.setBounds(1100,240,200,50);

l7.setBounds(100,310,200,50);

br.setBounds(300,310,200,50);

l8.setBounds(900,310,200,50);

cat.setBounds(1100,310,200,50);

l9.setBounds(100,380,200,50);

y.setBounds(300,380,200,50);

l10.setBounds(900,380,200,50);

sem.setBounds(1100,380,200,50);

l11.setBounds(100,450,200,50);

bl.setBounds(300,450,200,50);

l12.setBounds(900,450,200,50);

t12.setBounds(1100,450,200,50);

b1.setBounds(250,580,150,50);

b2.setBounds(480,580,150,50);

b3.setBounds(710,580,150,50);

b4.setBounds(940,580,150,50);

f4.add(y);

f4.add(br);

f4.add(bl);

f4.add(cat);

f4.add(sem);

f4.add(mi);

f4.add(d1);

f4.add(d2);

f4.add(d3);

f4.add(x);

f4.add(l13);

f4.add(l1);

f4.add(t12);

f4.add(l2);

f4.add(t1);

f4.add(l3);

f4.add(t2);

f4.add(l4);

f4.add(t3);

f4.add(l5);

f4.add(t4);

f4.add(l6);

f4.add(t5);

f4.add(l7);

f4.add(t6);

f4.add(l8);

f4.add(t7);

f4.add(l9);

f4.add(t8);

f4.add(l10);

f4.add(t9);

f4.add(l11);

f4.add(t10);

f4.add(l12);

f4.add(t11);

f4.add(b1);

f4.add(b2);

f4.add(b3);

f4.add(b4);

b1.addActionListener(this);

b2.addActionListener(this);

b3.addActionListener(this);

b4.addActionListener(this);}

public void actionPerformed(ActionEvent ae)

{String s1=t1.getText();

String s2=x.getSelectedItem().toString()+t2.getText();

String s3=t3.getText();

String s41=d1.getSelectedItem().toString();

String s42=d2.getSelectedItem().toString();

String s43=d3.getSelectedItem().toString();

String s5=t5.getText()+mi.getSelectedItem().toString();

String s6=t6.getText();

String s7=br.getSelectedItem().toString();

String s8=cat.getSelectedItem().toString();

String s9=y.getSelectedItem().toString();

String s10=sem.getSelectedItem().toString();

String s11=bl.getSelectedItem().toString();

String s12=t12.getText();

String s4=s41+"-"+s42+"-"+s43;

if(ae.getSource().equals(b2))

{t1.setText("");

t2.setText("");

t3.setText("");

t4.setText("");

t5.setText("");

t6.setText("");

t7.setText("");

t8.setText("");

t9.setText("");

t10.setText("");

t11.setText("");

t12.setText("");}

if(ae.getSource().equals(b3))

{f4.setVisible(false);

f333.setVisible(true);}

if(ae.getSource().equals(b4))

{f4.setVisible(false);

f111.setVisible(true);}

if(ae.getSource().equals(b1))

{if(t1.getText().equals(""))

JOptionPane.showMessageDialog(null,"Fill the fields with \* mark:student registered number");

else if(t2.getText().equals(""))

JOptionPane.showMessageDialog(null,"Fill the fields with \* mark:name");

else if(t3.getText().equals(""))

JOptionPane.showMessageDialog(null,"Fill the fields with \* mark:password");

else if(t5.getText().equals(""))

JOptionPane.showMessageDialog(null,"Fill the fields with \* mark:mail id");

else if(t6.getText().equals(""))

JOptionPane.showMessageDialog(null,"Fill the fields with \* mark:phone num");

else if(s7.equals(""))

JOptionPane.showMessageDialog(null,"Fill the fields with \* mark:branch");

else if(s8.equals(""))

JOptionPane.showMessageDialog(null,"Fill the fields with \* mark:category");

else if(s9.equals(""))

JOptionPane.showMessageDialog(null,"Fill the fields with \* mark:year");

else if(s10.equals(""))

JOptionPane.showMessageDialog(null,"Fill the field:semester");

else if(s11.equals(""))

JOptionPane.showMessageDialog(null,"Fill the fields with \* mark:backlogs");

else if(t12.getText().equals(""))

JOptionPane.showMessageDialog(null,"Fill the fields with \* mark:parents number");

else

{int phlen=s6.length();

int phlen1=s12.length();

int bbb=0,bbb1=0;

int kk=0,lop=0;

int pp=0;

try

{if(s6.length()>10||s6.length()<10||s12.length()>10||s12.length()<10)

JOptionPane.showMessageDialog(null,"Phone number must contain 10 number");

else

{for(int i=0;i<s6.length()||i<s12.length();i++)

{if(s6.charAt(i)>=48&&s6.charAt(i)<=57)

bbb++;

if(s12.charAt(i)>=48&&s12.charAt(i)<=57)

bbb1++;}

if(bbb<10||bbb1<10)

JOptionPane.showMessageDialog(null,"Phone number doesnot allow characters");}}

catch(Exception aa){}

int lo=s3.length();

try

{for(int i=0;i<lo;i++)

{if(s3.charAt(i)>=65&&s3.charAt(i)<=90)

kk++;

if(s3.charAt(i)>=48&&s3.charAt(i)<=57)

lop++;}

if(kk==0)

JOptionPane.showMessageDialog(null,"Password must contain atleast one letter in uppercase");

else if(lop==0)

JOptionPane.showMessageDialog(null,"Password must contain atleast one number");

else{}}

catch(Exception kkkk){}

int ssss=Integer.parseInt(s43);

try

{if(s42.equals("Apr")||s42.equals("Jun")||s42.equals("Sep")||s42.equals("Nov"))

{if(s41.equals("31"))

JOptionPane.showMessageDialog(null,"Not valid date");}

if(ssss%4!=0)

{if(s42.equals("Feb"))

{if(s41.equals("31")||s41.equals("30")||s41.equals("29"))

JOptionPane.showMessageDialog(null,"Not valid date");}}

if(ssss%4==0)

{if(s42.equals("Feb"))

{if(s41.equals("31")||s41.equals("30"))

JOptionPane.showMessageDialog(null,"Not valid date");}}}

catch(Exception klop){}

try

{DriverManager.registerDriver(new oracle.jdbc.OracleDriver());

Connection c1=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","system","password");

PreparedStatement s=c1.prepareStatement("select s\_id from addstudents");

ResultSet rs=s.executeQuery();

String s111;

int op=0;

while(rs.next())

{s111=rs.getString("s\_id");

if(s1.equals(s111))

op=1;}

if(op==1)

{JOptionPane.showMessageDialog(null,"Existing ID");}

else

{try

{if(phlen==10&&phlen1==10&&bbb==10&&bbb1==10&&kk!=0&&lop!=0)

{DriverManager.registerDriver(new oracle.jdbc.OracleDriver());

Connection c=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","system","password");

PreparedStatement ps=c.prepareStatement("insert into addstudents values(?,?,?,?,?,?,?,?,?,?,?,?)");

ps.setString(1,s1);

ps.setString(2,s2);

ps.setString(3,s3);

ps.setString(4,s4);

ps.setString(5,s5);

ps.setString(6,s6);

ps.setString(7,s7);

ps.setString(8,s8);

ps.setString(9,s9);

ps.setString(10,s10);

ps.setString(11,s11);

ps.setString(12,s12);

int m=ps.executeUpdate();

if(m==1)

{JOptionPane.showMessageDialog(null,"Succesfully added");

t1.setText("");

t2.setText("");

t3.setText("");

t4.setText("");

t5.setText("");

t6.setText("");

t7.setText("");

t8.setText("");

t9.setText("");

t10.setText("");

t11.setText("");

t12.setText("");}

else

JOptionPane.showMessageDialog(null,"not added");

ps.close();

c.close();}}

catch(Exception jklo){JOptionPane.showMessageDialog(null,jklo);}}

rs.close();

c1.close();}

catch(Exception jj){JOptionPane.showMessageDialog(null,jj);}}}}}

class Deletestaff implements ActionListener

{JFrame f111,f222,f333,f4;

JLabel l1,l2;

JTextField t1;

JButton b1,b2;

Deletestaff(JFrame f11,JFrame f22,JFrame f3)

{f111=f11;f222=f22;f333=f3;

f4=new JFrame("Delete staff");

f4.setVisible(true);

f4.setSize(1500,1500);

f4.setLayout(null);

l1=new JLabel("Delete staff");

l2=new JLabel("Staff ID");

ImageIcon x=new ImageIcon("ds.png");

JLabel l=new JLabel(x);

t1=new JTextField();

b1=new JButton("DELETE");

b2=new JButton("CANCEL");

l1.setFont(new Font("elephant",Font.BOLD,70));

l2.setFont(new Font("gill sans",Font.BOLD,50));

b1.setFont(new Font("gill sans",Font.BOLD,30));

b2.setFont(new Font("gill sans",Font.BOLD,30));

t1.setFont(new Font("gill sans",Font.BOLD,30));

b1.setBackground(Color.red);

b1.setForeground(Color.white);

b2.setBackground(Color.orange);

l2.setForeground(Color.red);

l1.setForeground(Color.blue);

l1.setBounds(400,5,1000,100);

l2.setBounds(100,170,200,50);

t1.setBounds(300,170,200,50);

l.setBounds(600,-150,1000,1000);

b1.setBounds(200,500,200,50);

b2.setBounds(600,500,200,50);

f4.add(l1);

f4.add(l2);

f4.add(t1);

f4.add(b1);

f4.add(b2);

f4.add(l);

b1.addActionListener(this);

b2.addActionListener(this);}

public void actionPerformed(ActionEvent ae)

{if(ae.getSource().equals(b2))

{f4.setVisible(false);

f333.setVisible(true);}

if(ae.getSource().equals(b1))

{Connection c=null;

PreparedStatement s=null;

try

{String s1=t1.getText();

DriverManager.registerDriver(new oracle.jdbc.OracleDriver());

c=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","system","password");

s=c.prepareStatement("select SID from addstaff ");

ResultSet rs=s.executeQuery();

String s11;

int op=0;

while(rs.next())

{s11=rs.getString("sid");

if(s1.equals(s11))

op=1;}

if(op==1)

{String query="delete from addstaff where SID=?";

try

{DriverManager.registerDriver(new oracle.jdbc.OracleDriver());

Connection c1=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","system","password");

PreparedStatement s111=c1.prepareStatement(query);

s111.setString(1,s1);

ResultSet rs11=s111.executeQuery();

JOptionPane.showMessageDialog(null,"ID deleted from DB");

t1.setText("");

s111.close();

c1.close();}

catch(Exception kl){JOptionPane.showMessageDialog(null,"GGG "+kl);}}

else

JOptionPane.showMessageDialog(null,"Cannot find ID");

c.close();

s.close();

rs.close();}

catch(Exception jj){JOptionPane.showMessageDialog(null,"EXCEPTION IS "+jj);}}}}

class Deletestudent implements ActionListener

{JFrame f111,f222,f333,f4;

JLabel l1,l2;

JTextField t1;

JButton b1,b2;

Deletestudent(JFrame f11,JFrame f22,JFrame f3)

{f111=f11;f222=f22;f333=f3;

f4=new JFrame("Delete staff");

f4.setVisible(true);

f4.setSize(1500,1500);

f4.setLayout(null);

l1=new JLabel("Delete staff");

l2=new JLabel("Student ID");

ImageIcon x=new ImageIcon("ds.png");

JLabel l=new JLabel(x);

t1=new JTextField();

b1=new JButton("DELETE");

b2=new JButton("CANCEL");

l1.setFont(new Font("elephant",Font.BOLD,70));

l2.setFont(new Font("gill sans",Font.BOLD,50));

b1.setFont(new Font("gill sans",Font.BOLD,30));

b2.setFont(new Font("gill sans",Font.BOLD,30));

t1.setFont(new Font("gill sans",Font.BOLD,30));

b1.setBackground(Color.red);

b1.setForeground(Color.white);

b2.setBackground(Color.orange);

l2.setForeground(Color.red);

l1.setForeground(Color.blue);

l1.setBounds(400,5,1000,100);

l2.setBounds(100,170,300,50);

t1.setBounds(400,170,200,50);

l.setBounds(600,-150,1000,1000);

b1.setBounds(200,500,200,50);

b2.setBounds(600,500,200,50);

f4.add(l1);

f4.add(l2);

f4.add(t1);

f4.add(b1);

f4.add(b2);

f4.add(l);

b1.addActionListener(this);

b2.addActionListener(this);}

public void actionPerformed(ActionEvent ae)

{if(ae.getSource().equals(b2))

{f4.setVisible(false);

f333.setVisible(true);}

if(ae.getSource().equals(b1))

{Connection c=null;

PreparedStatement s=null;

try

{String s1=t1.getText();

DriverManager.registerDriver(new oracle.jdbc.OracleDriver());

c=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","system","password");

s=c.prepareStatement("select S\_ID from addstudents ");

ResultSet rs=s.executeQuery();

String s11;

int op=0;

while(rs.next())

{s11=rs.getString("s\_id");

if(s1.equals(s11))

op=1;}

if(op==1)

{String query="delete from addstudents where S\_ID=?";

try

{DriverManager.registerDriver(new oracle.jdbc.OracleDriver());

Connection c1=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","system","password");

PreparedStatement s111=c1.prepareStatement(query);

s111.setString(1,s1);

ResultSet rs11=s111.executeQuery();

JOptionPane.showMessageDialog(null,"ID deleted from DB");

t1.setText("");

s111.close();

c1.close();}

catch(Exception kl){JOptionPane.showMessageDialog(null,"GGG "+kl);}}

else

JOptionPane.showMessageDialog(null,"Cannot find ID");

c.close();

s.close();

rs.close();}

catch(Exception jj){JOptionPane.showMessageDialog(null,"EXCEPTION IS "+jj);}}}

class Updatestaff1 implements ActionListener

{JFrame f111,f222,f333,f41;

JLabel l1,l2;

JTextField t1;

JButton b1,b2;

Updatestaff1(JFrame f11,JFrame f22,JFrame f3)

{f111=f11;

f222=f22;

f333=f3;

f41=new JFrame("Update staff");

f41.setVisible(true);

f41.setSize(1500,1500);

f41.setLayout(null);

l1=new JLabel("Update staff");

l2=new JLabel("Staff ID");

ImageIcon x=new ImageIcon("ds1.png");

JLabel l=new JLabel(x);

t1=new JTextField();

b1=new JButton("UPDATE");

b2=new JButton("CANCEL");

l1.setFont(new Font("elephant",Font.BOLD,70));

l2.setFont(new Font("gill sans",Font.BOLD,50));

b1.setFont(new Font("gill sans",Font.BOLD,30));

b2.setFont(new Font("gill sans",Font.BOLD,30));

t1.setFont(new Font("gill sans",Font.BOLD,30));

b1.setBackground(Color.red);

b1.setForeground(Color.white);

b2.setBackground(Color.orange);

l2.setForeground(Color.yellow);

l1.setForeground(Color.blue);

l1.setBounds(400,5,1000,100);

l2.setBounds(100,170,200,50);

t1.setBounds(300,170,200,50);

l.setBounds(500,-190,1000,1000);

b1.setBounds(200,500,200,50);

b2.setBounds(600,500,200,50);

f41.getContentPane().setBackground(Color.pink);

f41.add(l1);

f41.add(l2);

f41.add(t1);

f41.add(b1);

f41.add(b2);

f41.add(l);

b1.addActionListener(this);

b2.addActionListener(this);}

public void actionPerformed(ActionEvent ae)

{if(ae.getSource().equals(b2))

{f41.setVisible(false);

f333.setVisible(true);}

if(ae.getSource().equals(b1))

{Connection c=null;

PreparedStatement s=null;

try

{String s1=t1.getText();

DriverManager.registerDriver(new oracle.jdbc.OracleDriver());

c=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","system","password");

s=c.prepareStatement("select SID from addstaff");

ResultSet rs=s.executeQuery();

String s11;

int op=0;

while(rs.next())

{s11=rs.getString("sid");

if(s1.equals(s11))

op=1;}

if(op==1)

{f41.setVisible(false);

new Updatestaff2(f111,f222,f333,f41,s1);}

else

JOptionPane.showMessageDialog(null,"ID not found");}

catch(Exception jj){};}}}

class Updatestaff2 implements ActionListener

{JFrame f111,f222,f333,f444,f4;

JLabel l1,l2,l3,l4,l5,l6,l7,l8,l9,l10,l11,l12,l13;

JTextField t1,t12,t2,t4,t5,t6,t7,t8,t9,t10,t11;

JPasswordField t3;

JButton b1,b2,b3,b4;

JTextArea txt;

JComboBox x,d1,d2,d3,mi,y,sem;

String pp1;

Updatestaff2(JFrame f11,JFrame f22,JFrame f3,JFrame f41,String pp)

{f111=f11;f222=f22;

f333=f3;f444=f41;

pp1=pp;

f4=new JFrame("Updating staff");

f4.setVisible(true);

f4.setSize(1500,1500);

f4.setLayout(null);

l1=new JLabel("Staff\_ID");

l2=new JLabel("Name");

l3=new JLabel("Password");

l4=new JLabel("Join\_date");

l5=new JLabel("Mail\_ID");

l6=new JLabel("Phone\_num");

l7=new JLabel("Qualification");

l8=new JLabel("Subject");

l9=new JLabel("Year");

l10=new JLabel("Semester");

l11=new JLabel("Salary");

l12=new JLabel("Address");

l13=new JLabel("Add new staff");

txt=new JTextArea(100,100);

x=new JComboBox();

mi=new JComboBox();

d1=new JComboBox();

d2=new JComboBox();

d3=new JComboBox();

y=new JComboBox();

sem=new JComboBox();

t1=new JTextField(20);

t12=new JTextField(20);

t2=new JTextField(20);

t4=new JTextField(20);

t5=new JTextField(20);

t6=new JTextField(20);

t7=new JTextField(20);

t8=new JTextField(20);

t9=new JTextField(20);

t10=new JTextField(20);

t11=new JTextField(20);

t3=new JPasswordField(20);

b1=new JButton("Update");

b2=new JButton("Reset");

b3=new JButton("Back");

b4=new JButton("Cancel");

x.addItem("Mr.");

x.addItem("Ms.");

x.addItem("Dr.");

x.addItem("Prof.");

d1.addItem("01");

d1.addItem("02");

d1.addItem("03");

d1.addItem("04");

d1.addItem("05");

d1.addItem("06");

d1.addItem("07");

d1.addItem("08");

d1.addItem("08");

d1.addItem("10");

d1.addItem("11");

d1.addItem("12");

d1.addItem("13");

d1.addItem("14");

d1.addItem("15");

d1.addItem("16");

d1.addItem("17");

d1.addItem("18");

d1.addItem("19");

d1.addItem("20");

d1.addItem("21");

d1.addItem("22");

d1.addItem("23");

d1.addItem("24");

d1.addItem("25");

d1.addItem("26");

d1.addItem("27");

d1.addItem("28");

d1.addItem("29");

d1.addItem("30");

d1.addItem("31");

d2.addItem("Jan");

d2.addItem("Feb");

d2.addItem("Mar");

d2.addItem("Apr");

d2.addItem("May");

d2.addItem("Jun");

d2.addItem("Jul");

d2.addItem("Aug");

d2.addItem("Sep");

d2.addItem("Oct");

d2.addItem("Nov");

d2.addItem("Dec");

d3.addItem("1998");

d3.addItem("1999");

d3.addItem("2000");

d3.addItem("2001");

d3.addItem("2002");

d3.addItem("2003");

d3.addItem("2004");

d3.addItem("2005");

d3.addItem("2006");

d3.addItem("2007");

d3.addItem("2008");

d3.addItem("2009");

d3.addItem("2010");

d3.addItem("2011");

d3.addItem("2012");

d3.addItem("2013");

d3.addItem("2014");

d3.addItem("2015");

d3.addItem("2016");

d3.addItem("2017");

d3.addItem("2018");

d3.addItem("2019");

y.addItem("1");

y.addItem("2");

y.addItem("3");

y.addItem("4");

sem.addItem("1");

sem.addItem("2");

mi.addItem("@gmail.com");

mi.addItem("@mits.ac.in");

l13.setFont(new Font("elephant",Font.BOLD,70));

l2.setFont(new Font("gill sans",Font.BOLD,30));

l3.setFont(new Font("gill sans",Font.BOLD,30));

l4.setFont(new Font("gill sans",Font.BOLD,30));

l5.setFont(new Font("gill sans",Font.BOLD,30));

l6.setFont(new Font("gill sans",Font.BOLD,30));

l7.setFont(new Font("gill sans",Font.BOLD,30));

l8.setFont(new Font("gill sans",Font.BOLD,30));

l9.setFont(new Font("gill sans",Font.BOLD,30));

l10.setFont(new Font("gill sans",Font.BOLD,30));

l11.setFont(new Font("gill sans",Font.BOLD,30));

l12.setFont(new Font("gill sans",Font.BOLD,30));

l1.setFont(new Font("gill sans",Font.BOLD,30));

b2.setFont(new Font("gill sans",Font.BOLD,25));

b3.setFont(new Font("gill sans",Font.BOLD,25));

b4.setFont(new Font("gill sans",Font.BOLD,25));

b1.setFont(new Font("gill sans",Font.BOLD,25));

t1.setFont(new Font("gill sans",Font.BOLD,20));

t2.setFont(new Font("gill sans",Font.BOLD,20));

t3.setFont(new Font("gill sans",Font.BOLD,20));

t4.setFont(new Font("gill sans",Font.BOLD,20));

t5.setFont(new Font("gill sans",Font.BOLD,20));

t6.setFont(new Font("gill sans",Font.BOLD,20));

t7.setFont(new Font("gill sans",Font.BOLD,20));

t8.setFont(new Font("gill sans",Font.BOLD,20));

t9.setFont(new Font("gill sans",Font.BOLD,20));

t10.setFont(new Font("gill sans",Font.BOLD,20));

t11.setFont(new Font("gill sans",Font.BOLD,20));

txt.setFont(new Font("gill sans",Font.BOLD,20));

b2.setBackground(Color.cyan);

b1.setBackground(Color.orange);

b4.setBackground(Color.pink);

b3.setBackground(Color.yellow);

l13.setForeground(Color.orange);

l1.setForeground(Color.red);

l2.setForeground(Color.blue);

l3.setForeground(Color.blue);

l4.setForeground(Color.red);

l5.setForeground(Color.red);

l6.setForeground(Color.blue);

l7.setForeground(Color.blue);

l8.setForeground(Color.red);

l9.setForeground(Color.red);

l10.setForeground(Color.blue);

l11.setForeground(Color.blue);

l12.setForeground(Color.red);

l13.setBounds(400,5,1000,100);

l1.setBounds(100,100,200,50);

t1.setBounds(300,100,200,50);

l2.setBounds(900,100,200,50);

t2.setBounds(1100,100,200,50);

l3.setBounds(100,170,200,50);

t3.setBounds(300,170,200,50);

l4.setBounds(900,170,200,50);

t4.setBounds(1100,170,200,50);

l5.setBounds(100,240,200,50);

t5.setBounds(300,240,200,50);

l6.setBounds(900,240,200,50);

t6.setBounds(1100,240,200,50);

l7.setBounds(100,310,200,50);

t7.setBounds(300,310,200,50);

l8.setBounds(900,310,200,50);

t8.setBounds(1100,310,200,50);

l9.setBounds(100,380,200,50);

t9.setBounds(300,380,200,50);

l10.setBounds(900,380,200,50);

t10.setBounds(1100,380,200,50);

l11.setBounds(100,450,200,50);

t11.setBounds(300,450,200,50);

l12.setBounds(900,450,200,50);

txt.setBounds(1100,450,200,50);

b1.setBounds(250,580,150,50);

b2.setBounds(480,580,150,50);

b3.setBounds(710,580,150,50);

b4.setBounds(940,580,150,50);

f4.add(txt);

f4.add(y);

f4.add(sem);

f4.add(mi);

f4.add(d1);

f4.add(d2);

f4.add(d3);

f4.add(x);

f4.add(l13);

f4.add(l1);

f4.add(t12);

f4.add(l2);

f4.add(t1);

f4.add(l3);

f4.add(t2);

f4.add(l4);

f4.add(t3);

f4.add(l5);

f4.add(t4);

f4.add(l6);

f4.add(t5);

f4.add(l7);

f4.add(t6);

f4.add(l8);

f4.add(t7);

f4.add(l9);

f4.add(t8);

f4.add(l10);

f4.add(t9);

f4.add(l11);

f4.add(t10);

f4.add(l12);

f4.add(t11);

f4.add(b1);

f4.add(b2);

f4.add(b3);

f4.add(b4);

b1.addActionListener(this);

b2.addActionListener(this);

b3.addActionListener(this);

b4.addActionListener(this);

t1.setText(pp1);

t1.setEditable(false);

try

{DriverManager.registerDriver(new oracle.jdbc.OracleDriver());

Connection c=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","system","password");

PreparedStatement s=c.prepareStatement("select \* from addstaff where sid=?");

s.setString(1,pp1);

ResultSet rs=s.executeQuery();

while(rs.next())

{t2.setText(rs.getString("name"));

t3.setText(rs.getString("pw"));

t4.setText(rs.getString("jd"));

t5.setText(rs.getString("mi"));

t6.setText(rs.getString("ph"));

t7.setText(rs.getString("qf"));

t8.setText(rs.getString("sub"));

t9.setText(rs.getString("year"));

t10.setText(rs.getString("sem"));

t11.setText(rs.getString("sal"));

txt.setText(rs.getString("ad"));}

c.close();

s.close();}

catch(Exception jk){}}

public void actionPerformed(ActionEvent ae)

{if(ae.getSource().equals(b3))

{f4.setVisible(false);

f333.setVisible(true);}

if(ae.getSource().equals(b4))

{f4.setVisible(false);

f333.setVisible(true);}

if(ae.getSource().equals(b1))

{String z2=t2.getText();

String z3=t3.getText();

String z4=t4.getText();

String z5=t5.getText();

String z6=t6.getText();

String z7=t7.getText();

String z8=t8.getText();

String z9=t9.getText();

String z10=t10.getText();

String z11=t11.getText();

String z12=txt.getText();

if(z2.equals("")||z3.equals("")||z4.equals("")||z5.equals("")||z6.equals("")||z7.equals("")||z8.equals("")||z9.equals("")||z10.equals("")||z11.equals("")||z12.equals(""))

{JOptionPane.showMessageDialog(null,"Fill all the required fields");}

int ka=0;

if( z5.contains("@gmail.com")||z5.contains("@mits.ac.in")) {ka++;}

else

{ JOptionPane.showMessageDialog(null,"Maild id must contain @gmail.com or @mits.ac.in"); }

int ka1=0;

if( z2.contains("Mr.")||z2.contains("Ms.")||z2.contains("Dr.")||z2.contains("Prof.")) {ka1++;}

else

{ JOptionPane.showMessageDialog(null,"Name must contain either Mr. r Ms"); }

int kk=0,lop=0;

int lo=z3.length();

try

{for(int i=0;i<lo;i++)

{if(z3.charAt(i)>=65&&z3.charAt(i)<=90)

kk++;

if(z3.charAt(i)>=48&&z3.charAt(i)<=57)

lop++;}

if(kk==0)

JOptionPane.showMessageDialog(null,"Password must contain atleast one letter in uppercase");

else if(lop==0)

JOptionPane.showMessageDialog(null,"Password must contain atleast one number");

else{}}

catch(Exception kkkk){}

int bbb=0;

try

{if(z6.length()>10||z6.length()<10)

JOptionPane.showMessageDialog(null,"Phone number must contain 10 number");

else

{for(int i=0;i<z6.length();i++)

{if(z6.charAt(i)>=48&&z6.charAt(i)<=57)

bbb++;}

if(bbb<10)

JOptionPane.showMessageDialog(null,"Phone number doesnot allow characters");}}

catch(Exception aa){}

int pp=0,ww=z11.length();

try

{for(int o=0;o<ww;o++){

if(z11.charAt(o)>=48&&z11.charAt(o)<=57)

{pp++;}}

if(ww==pp){}

else

JOptionPane.showMessageDialog(null,"Salary allows only numeric");}

catch(Exception kjolm){}

int ggh=0;

int ho=z9.length();

try

{if(ho==1)

{if(z9.equals("1")||z9.equals("2")||z9.equals("3")||z9.equals("4")){ggh=1;}

else

JOptionPane.showMessageDialog(null,"Year is within 1 and 4");}

else

JOptionPane.showMessageDialog(null,"Year allows only single number");}

catch(Exception klo){}

int ddt=0;

int ho1=z10.length();

try

{if(ho1==1)

{if(z10.equals("1")||z10.equals("2")){ddt=1;}

else

JOptionPane.showMessageDialog(null,"sem is either 1 or 2");}

else

JOptionPane.showMessageDialog(null,"sem allows only single number");}

catch(Exception klod){}

int php=0;

String kl1="";

int ll=z5.length();

String z1=t1.getText();

if(kk!=0&&lop!=0&&bbb==10&&ww==pp&&ho==1&&ho1==1&&php!=ll&&ka==1&&ka1==1&&ddt==1&&ggh==1)

{try

{DriverManager.registerDriver(new oracle.jdbc.OracleDriver());

Connection con=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","system","password");

Statement st=con.createStatement();

st.executeUpdate("update addstaff set name='"+z2+"',pw='"+z3+"',jd='"+z4+"',mi='"+z5+"', ph='"+z6+"',qf='"+z7+"',sub='"+z8+"',year='"+z9+"',sem='"+z10+"',sal='"+z11+"',ad='"+z12+"' where sid='"+z1+"'");

JOptionPane.showMessageDialog(null,"Updated successfully");

con.close();

f4.setVisible(false);

f333.setVisible(true);}

catch(Exception kl){JOptionPane.showMessageDialog(null,"GGG "+kl);}}}

if(ae.getSource().equals(b2))

{try

{DriverManager.registerDriver(new oracle.jdbc.OracleDriver());

Connection c=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","system","password");

PreparedStatement s=c.prepareStatement("select \* from addstaff where sid=?");

s.setString(1,pp1);

ResultSet rs=s.executeQuery();

while(rs.next())

{t2.setText(rs.getString("name"));

t3.setText(rs.getString("pw"));

t4.setText(rs.getString("jd"));

t5.setText(rs.getString("mi"));

t6.setText(rs.getString("ph"));

t7.setText(rs.getString("qf"));

t8.setText(rs.getString("sub"));

t9.setText(rs.getString("year"));

t10.setText(rs.getString("sem"));

t11.setText(rs.getString("sal"));

txt.setText(rs.getString("ad"));}

c.close();

s.close();}

catch(Exception jk){}}}}

class Updatestudent1 implements ActionListener

{JFrame f111,f222,f333,f41;

JLabel l1,l2;

JTextField t1;

JButton b1,b2;

Updatestudent1(JFrame f11,JFrame f22,JFrame f3)

{f111=f11;

f222=f22;

f333=f3;

f41=new JFrame("Update student");

f41.setVisible(true);

f41.setSize(1500,1500);

f41.setLayout(null);

l1=new JLabel("Update student");

l2=new JLabel("Student ID");

ImageIcon x=new ImageIcon("ds1.png");

JLabel l=new JLabel(x);

t1=new JTextField();

b1=new JButton("UPDATE");

b2=new JButton("CANCEL");

l1.setFont(new Font("elephant",Font.BOLD,70));

l2.setFont(new Font("gill sans",Font.BOLD,50));

b1.setFont(new Font("gill sans",Font.BOLD,30));

b2.setFont(new Font("gill sans",Font.BOLD,30));

t1.setFont(new Font("gill sans",Font.BOLD,30));

b1.setBackground(Color.red);

b1.setForeground(Color.white);

b2.setBackground(Color.orange);

l2.setForeground(Color.yellow);

l1.setForeground(Color.blue);

l1.setBounds(400,5,1000,100);

l2.setBounds(100,170,300,50);

t1.setBounds(400,170,200,50);

l.setBounds(500,-190,1000,1000);

b1.setBounds(200,500,200,50);

b2.setBounds(600,500,200,50);

f41.getContentPane().setBackground(Color.pink);

f41.add(l1);

f41.add(l2);

f41.add(t1);

f41.add(b1);

f41.add(b2);

f41.add(l);

b1.addActionListener(this);

b2.addActionListener(this);}

public void actionPerformed(ActionEvent ae)

{if(ae.getSource().equals(b2))

{f41.setVisible(false);

f333.setVisible(true);}

if(ae.getSource().equals(b1))

{Connection c=null;

PreparedStatement s=null;

try

{String s1=t1.getText();

DriverManager.registerDriver(new oracle.jdbc.OracleDriver());

c=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","system","password");

s=c.prepareStatement("select S\_ID from addstudents");

ResultSet rs=s.executeQuery();

String s11;

int op=0;

while(rs.next())

{s11=rs.getString("s\_id");

if(s1.equals(s11))

op=1;}

if(op==1)

{f41.setVisible(false);

new Updatestudent2(f111,f222,f333,f41,s1);}

else

JOptionPane.showMessageDialog(null,"ID not found");}

catch(Exception jj){};}}}

class Updatestudent2 implements ActionListener

{JFrame f111,f222,f333,f444,f4;

JLabel l1,l2,l3,l4,l5,l6,l7,l8,l9,l10,l11,l12,l13;

JTextField t1,t12,t2,t4,t5,t6,t7,t8,t9,t10,t11;

JPasswordField t3;

JButton b1,b2,b3,b4;

JTextArea txt;

JComboBox x,d1,d2,d3,mi,y,sem;

String pp1;

Updatestudent2(JFrame f11,JFrame f22,JFrame f3,JFrame f41,String pp)

{f111=f11;f222=f22;

f333=f3;f444=f41;

pp1=pp;

f4=new JFrame("Updating student");

f4.setVisible(true);

f4.setSize(1500,1500);

f4.setLayout(null);

l1=new JLabel("Register\_num");

l2=new JLabel("Name");

l3=new JLabel("Password");

l4=new JLabel("Join\_date");

l5=new JLabel("Mail\_ID");

l6=new JLabel("Phone\_num");

l7=new JLabel("Branch");

l8=new JLabel("Category");

l9=new JLabel("Year");

l10=new JLabel("Semester");

l11=new JLabel("Backlogs");

l12=new JLabel("Parents\_num");

l13=new JLabel("Add new student");

txt=new JTextArea(100,100);

x=new JComboBox();

mi=new JComboBox();

d1=new JComboBox();

d2=new JComboBox();

d3=new JComboBox();

y=new JComboBox();

sem=new JComboBox();

t1=new JTextField(20);

t12=new JTextField(20);

t2=new JTextField(20);

t4=new JTextField(20);

t5=new JTextField(20);

t6=new JTextField(20);

t7=new JTextField(20);

t8=new JTextField(20);

t9=new JTextField(20);

t10=new JTextField(20);

t11=new JTextField(20);

t3=new JPasswordField(20);

b1=new JButton("Update");

b2=new JButton("Reset");

b3=new JButton("Back");

b4=new JButton("Cancel");

l13.setFont(new Font("elephant",Font.BOLD,70));

l2.setFont(new Font("gill sans",Font.BOLD,30));

l3.setFont(new Font("gill sans",Font.BOLD,30));

l4.setFont(new Font("gill sans",Font.BOLD,30));

l5.setFont(new Font("gill sans",Font.BOLD,30));

l6.setFont(new Font("gill sans",Font.BOLD,30));

l7.setFont(new Font("gill sans",Font.BOLD,30));

l8.setFont(new Font("gill sans",Font.BOLD,30));

l9.setFont(new Font("gill sans",Font.BOLD,30));

l10.setFont(new Font("gill sans",Font.BOLD,30));

l11.setFont(new Font("gill sans",Font.BOLD,30));

l12.setFont(new Font("gill sans",Font.BOLD,30));

l1.setFont(new Font("gill sans",Font.BOLD,30));

b2.setFont(new Font("gill sans",Font.BOLD,25));

b3.setFont(new Font("gill sans",Font.BOLD,25));

b4.setFont(new Font("gill sans",Font.BOLD,25));

b1.setFont(new Font("gill sans",Font.BOLD,25));

t1.setFont(new Font("gill sans",Font.BOLD,20));

t2.setFont(new Font("gill sans",Font.BOLD,20));

t3.setFont(new Font("gill sans",Font.BOLD,20));

t4.setFont(new Font("gill sans",Font.BOLD,20));

t5.setFont(new Font("gill sans",Font.BOLD,20));

t6.setFont(new Font("gill sans",Font.BOLD,20));

t7.setFont(new Font("gill sans",Font.BOLD,20));

t8.setFont(new Font("gill sans",Font.BOLD,20));

t9.setFont(new Font("gill sans",Font.BOLD,20));

t10.setFont(new Font("gill sans",Font.BOLD,20));

t11.setFont(new Font("gill sans",Font.BOLD,20));

txt.setFont(new Font("gill sans",Font.BOLD,20));

b2.setBackground(Color.cyan);

b1.setBackground(Color.orange);

b4.setBackground(Color.pink);

b3.setBackground(Color.yellow);

l13.setForeground(Color.orange);

l1.setForeground(Color.red);

l2.setForeground(Color.blue);

l3.setForeground(Color.blue);

l4.setForeground(Color.red);

l5.setForeground(Color.red);

l6.setForeground(Color.blue);

l7.setForeground(Color.blue);

l8.setForeground(Color.red);

l9.setForeground(Color.red);

l10.setForeground(Color.blue);

l11.setForeground(Color.blue);

l12.setForeground(Color.red);

l13.setBounds(400,5,1000,100);

l1.setBounds(100,100,200,50);

t1.setBounds(300,100,200,50);

l2.setBounds(900,100,200,50);

t2.setBounds(1100,100,200,50);

l3.setBounds(100,170,200,50);

t3.setBounds(300,170,200,50);

l4.setBounds(900,170,200,50);

t4.setBounds(1100,170,200,50);

l5.setBounds(100,240,200,50);

t5.setBounds(300,240,200,50);

l6.setBounds(900,240,200,50);

t6.setBounds(1100,240,200,50);

l7.setBounds(100,310,200,50);

t7.setBounds(300,310,200,50);

l8.setBounds(900,310,200,50);

t8.setBounds(1100,310,200,50);

l9.setBounds(100,380,200,50);

t9.setBounds(300,380,200,50);

l10.setBounds(900,380,200,50);

t10.setBounds(1100,380,200,50);

l11.setBounds(100,450,200,50);

t11.setBounds(300,450,200,50);

l12.setBounds(900,450,200,50);

txt.setBounds(1100,450,200,50);

b1.setBounds(250,580,150,50);

b2.setBounds(480,580,150,50);

b3.setBounds(710,580,150,50);

b4.setBounds(940,580,150,50);

f4.add(txt);

f4.add(y);

f4.add(sem);

f4.add(mi);

f4.add(d1);

f4.add(d2);

f4.add(d3);

f4.add(x);

f4.add(l13);

f4.add(l1);

f4.add(t12);

f4.add(l2);

f4.add(t1);

f4.add(l3);

f4.add(t2);

f4.add(l4);

f4.add(t3);

f4.add(l5);

f4.add(t4);

f4.add(l6);

f4.add(t5);

f4.add(l7);

f4.add(t6);

f4.add(l8);

f4.add(t7);

f4.add(l9);

f4.add(t8);

f4.add(l10);

f4.add(t9);

f4.add(l11);

f4.add(t10);

f4.add(l12);

f4.add(t11);

f4.add(b1);

f4.add(b2);

f4.add(b3);

f4.add(b4);

b1.addActionListener(this);

b2.addActionListener(this);

b3.addActionListener(this);

b4.addActionListener(this);

t1.setText(pp1);

t1.setEditable(false);

try

{DriverManager.registerDriver(new oracle.jdbc.OracleDriver());

Connection c=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","system","password");

PreparedStatement s=c.prepareStatement("select \* from addstudents where s\_id=?");

s.setString(1,pp1);

ResultSet rs=s.executeQuery();

while(rs.next())

{t2.setText(rs.getString("names"));

t3.setText(rs.getString("pws"));

t4.setText(rs.getString("jds"));

t5.setText(rs.getString("mis"));

t6.setText(rs.getString("phs"));

t7.setText(rs.getString("branchs"));

t8.setText(rs.getString("categorys"));

t9.setText(rs.getString("years"));

t10.setText(rs.getString("sems"));

t11.setText(rs.getString("backlogs"));

txt.setText(rs.getString("phps"));}

c.close();

s.close();}

catch(Exception jk){}}

public void actionPerformed(ActionEvent ae)

{if(ae.getSource().equals(b3))

{f4.setVisible(false);

f333.setVisible(true);}

if(ae.getSource().equals(b4))

{f4.setVisible(false);

f333.setVisible(true);}

if(ae.getSource().equals(b1))

{String z2=t2.getText();

String z3=t3.getText();

String z4=t4.getText();

String z5=t5.getText();

String z6=t6.getText();

String z7=t7.getText();

String z8=t8.getText();

String z9=t9.getText();

String z10=t10.getText();

String z11=t11.getText();

String z12=txt.getText();

if(z2.equals("")||z3.equals("")||z4.equals("")||z5.equals("")||z6.equals("")||z7.equals("")||z8.equals("")||z9.equals("")||z10.equals("")||z11.equals("")||z12.equals(""))

{JOptionPane.showMessageDialog(null,"Fill all the required fields");}

int ka=0;

if( z5.contains("@gmail.com")||z5.contains("@mits.ac.in")) {ka++;}

else

{ JOptionPane.showMessageDialog(null,"Maild id must contain @gmail.com or @mits.ac.in"); }

int ka1=0;

if( z2.contains("Mr.")||z2.contains("Ms.")) {ka1++;}

else

{ JOptionPane.showMessageDialog(null,"Name must contain either MR r Ms"); }

int kk=0,lop=0;

int lo=z3.length();

try

{for(int i=0;i<lo;i++)

{if(z3.charAt(i)>=65&&z3.charAt(i)<=90)

kk++;

if(z3.charAt(i)>=48&&z3.charAt(i)<=57)

lop++;}

if(kk==0)

JOptionPane.showMessageDialog(null,"Password must contain atleast one letter in uppercase");

else if(lop==0)

JOptionPane.showMessageDialog(null,"Password must contain atleast one number");

else{}}

catch(Exception kkkk){}

int bbb=0,bbb1=0;

try

{if(z6.length()>10||z6.length()<10||z12.length()<10||z12.length()>10)

JOptionPane.showMessageDialog(null,"Phone number must contain 10 number");

else

{for(int i=0;i<z6.length()||i<z12.length();i++)

{if(z6.charAt(i)>=48&&z6.charAt(i)<=57)

bbb++;

if(z12.charAt(i)>=48&&z12.charAt(i)<=57)

bbb1++;}

if(bbb<10||bbb1<10)

JOptionPane.showMessageDialog(null,"Phone number doesnot allow characters");}}

catch(Exception aa){}

int ggh=0;

int ho=z9.length();

try

{if(ho==1)

{if(z9.equals("1")||z9.equals("2")||z9.equals("3")||z9.equals("4")){ggh=1;}

else

JOptionPane.showMessageDialog(null,"Year is within 1 and 4");}

else

JOptionPane.showMessageDialog(null,"Year allows only single number");}

catch(Exception klo){}

int ddt=0;

int ho1=z10.length();

try

{if(ho1==1)

{if(z10.equals("1")||z10.equals("2")){ddt=1;}

else

JOptionPane.showMessageDialog(null,"sem is either 1 or 2");}

else

JOptionPane.showMessageDialog(null,"sem allows only single number");}

catch(Exception klod){}

int php=0;

String kl1="";

int ll=z5.length();

String z1=t1.getText();

if(kk!=0&&lop!=0&&bbb==10&&bbb1==10&&ho==1&&ho1==1&&php!=ll&&ka==1&&ka1==1&&ddt==1&&ggh==1)

{try

{DriverManager.registerDriver(new oracle.jdbc.OracleDriver());

Connection con=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","system","password");

Statement st=con.createStatement();

st.executeUpdate("update addstudents set names='"+z2+"',pws='"+z3+"',jds='"+z4+"',mis='"+z5+"', phs='"+z6+"',branchs='"+z7+"',categorys='"+z8+"',years='"+z9+"',sems='"+z10+"',backlogs='"+z11+"',phps='"+z12+"' where s\_id='"+z1+"'");

JOptionPane.showMessageDialog(null,"Updated successfully");

con.close();

f4.setVisible(false);

f333.setVisible(true);}

catch(Exception kl){JOptionPane.showMessageDialog(null,"GGG "+kl);}}}

if(ae.getSource().equals(b2))

{try

{DriverManager.registerDriver(new oracle.jdbc.OracleDriver());

Connection c=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","system","password");

PreparedStatement s=c.prepareStatement("select \* from addstaff where sid=?");

s.setString(1,pp1);

ResultSet rs=s.executeQuery();

while(rs.next())

{t2.setText(rs.getString("names"));

t3.setText(rs.getString("pws"));

t4.setText(rs.getString("jds"));

t5.setText(rs.getString("mis"));

t6.setText(rs.getString("phs"));

t7.setText(rs.getString("branchs"));

t8.setText(rs.getString("categorys"));

t9.setText(rs.getString("years"));

t10.setText(rs.getString("sems"));

t11.setText(rs.getString("backlogs"));

txt.setText(rs.getString("phps"));}

c.close();

s.close();}

catch(Exception jk){}}}}

class flex implements ActionListener

{JFrame p1,po;

JLabel l1,l2,l3;

JButton b1,b2,b3,b4,b5;

flex(JFrame f3)

{po=f3;

p1=new JFrame("Timetable");

p1.setSize(1300,1300);

p1.setVisible(true);

p1.setLayout(null);

l1=new JLabel("Add/delete Room");

l2=new JLabel("View timetable ");

l3=new JLabel("Create timetable");

b1=new JButton("Add/Delete");

b2=new JButton("View");

b3=new JButton("Create");

b4=new JButton("Back");

b5=new JButton("Cancel");

ImageIcon x=new ImageIcon("flex1.jpg");

JLabel l=new JLabel(x);

l1.setFont(new Font("gill sans",Font.BOLD,50));

l2.setFont(new Font("gill sans",Font.BOLD,50));

l3.setFont(new Font("gill sans",Font.BOLD,50));

b1.setFont(new Font("gill sans",Font.BOLD,25));

b2.setFont(new Font("gill sans",Font.BOLD,25));

b3.setFont(new Font("gill sans",Font.BOLD,25));

b4.setFont(new Font("gill sans",Font.BOLD,25));

b5.setFont(new Font("gill sans",Font.BOLD,25));

l.setBounds(580,50,1000,500);

l1.setBounds(70,150,400,50);

l3.setBounds(70,250,400,50);

l2.setBounds(70,350,400,50);

b1.setBounds(500,150,200,50);

b3.setBounds(500,250,200,50);

b2.setBounds(500,350,200,50);

b4.setBounds(150,500,200,50);

b5.setBounds(440,500,200,50);

p1.add(l1);

p1.add(b1);

p1.add(l);

p1.add(l2);

p1.add(b2);

p1.add(l3);

p1.add(b3);

p1.add(b4);

p1.add(b5);

b1.addActionListener(this);

b2.addActionListener(this);

b3.addActionListener(this);

b4.addActionListener(this);

b5.addActionListener(this);}

public void actionPerformed(ActionEvent ae)

{if(ae.getSource().equals(b5))

{p1.setVisible(false);}

if(ae.getSource().equals(b1))

{p1.setVisible(false);

new room(p1);}

if(ae.getSource().equals(b2))

{p1.setVisible(false);

new view(p1);}

if(ae.getSource().equals(b3))

{p1.setVisible(false);

new tt(p1);}

if(ae.getSource().equals(b4))

{p1.setVisible(false);

po.setVisible(true);}}}

class room implements ActionListener

{JFrame p2,p11;

JLabel l1,l2,l3;

JTextField t1,t2,t3;

JButton b1,b2,b3,b4;

room(JFrame p1)

{p11=p1;

p2=new JFrame("Adding room numbers");

p2.setVisible(true);

p2.setSize(1500,1500);

p2.setLayout(null);

l1=new JLabel("Enter room number");

t1=new JTextField(10);

l2=new JLabel("Enter course name");

t2=new JTextField(10);

l3=new JLabel("Enter Sec name");

t3=new JTextField(10);

b1=new JButton("Add");

b3=new JButton("Add subjects");

b4=new JButton("Delete");

b2=new JButton("Back");

l1.setFont(new Font("gill sans",Font.BOLD,50));

t1.setFont(new Font("gill sans",Font.BOLD,25));

l2.setFont(new Font("gill sans",Font.BOLD,50));

t2.setFont(new Font("gill sans",Font.BOLD,25));

l3.setFont(new Font("gill sans",Font.BOLD,50));

t3.setFont(new Font("gill sans",Font.BOLD,25));

b1.setFont(new Font("gill sans",Font.BOLD,30));

b2.setFont(new Font("gill sans",Font.BOLD,30));

b3.setFont(new Font("gill sans",Font.BOLD,30));

b4.setFont(new Font("gill sans",Font.BOLD,30));

l1.setBounds(100,150,600,50);

t1.setBounds(700,150,200,50);

l2.setBounds(100,250,600,50);

t2.setBounds(700,250,200,50);

l3.setBounds(100,350,600,50);

t3.setBounds(700,350,200,50);

b1.setBounds(150,500,200,50);

b3.setBounds(440,500,200,50);

b2.setBounds(1040,500,200,50);

b4.setBounds(740,500,200,50);

p2.add(l1);

p2.add(t1);

p2.add(l2);

p2.add(t2);

p2.add(l3);

p2.add(t3);

p2.add(b1);

p2.add(b2);

p2.add(b3);

p2.add(b4);

b1.addActionListener(this);

b3.addActionListener(this);

b4.addActionListener(this);

b2.addActionListener(this);}

public void actionPerformed(ActionEvent ae)

{String n1=t1.getText();

String n3=t2.getText();

String n5=t3.getText();

if(ae.getSource().equals(b2))

{p2.setVisible(false);

p11.setVisible(true);}

if(ae.getSource().equals(b1))

{if(n1.equals("")||n3.equals("")||n5.equals(""))

{JOptionPane.showMessageDialog(null,"Empty fields not allowed");}

else

{try

{DriverManager.registerDriver(new oracle.jdbc.OracleDriver());

Connection c1=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","system","password");

PreparedStatement s=c1.prepareStatement("select num,course,sec from room ");

ResultSet rs=s.executeQuery();

String n2,n4,n6;

int op=0;

while(rs.next())

{n2=rs.getString("num");

n4=rs.getString("course");

n6=rs.getString("sec");

if(n1.equals(n2)&&n3.equals(n4)&&n5.equals(n6))

op=1;}

if(op==1)

{JOptionPane.showMessageDialog(null,"Already added");

t1.setText("");

t2.setText("");

t3.setText("");}

else

{try

{DriverManager.registerDriver(new oracle.jdbc.OracleDriver());

Connection c=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","system","password");

PreparedStatement ps=c.prepareStatement("insert into room values(?,?,?)");

ps.setString(1,n1);

ps.setString(2,n3);

ps.setString(3,n5);

int m=ps.executeUpdate();

if(m==1)

{JOptionPane.showMessageDialog(null,"Succesfully added");}

else

JOptionPane.showMessageDialog(null,"Not added");

ps.close();

c.close();}

catch(Exception jkl){}}}

catch(Exception klo){}}}

if(ae.getSource().equals(b3))

{if(n1.equals("")||n3.equals("")||n5.equals(""))

{JOptionPane.showMessageDialog(null,"Empty fields not allowed");}

else

{p2.setVisible(false);

new sub(p11,n3,n1);}}

if(ae.getSource().equals(b4))

{if(n1.equals(""))

{JOptionPane.showMessageDialog(null,"Empty room num not allowed");}

else

{String query1="delete from xx1 where roomnum=?";String query="delete from room where num=?";

Connection cz=null;

PreparedStatement sz=null;

try

{DriverManager.registerDriver(new oracle.jdbc.OracleDriver());

cz=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","system","password");

sz=cz.prepareStatement("select \* from room ");

ResultSet rsz=sz.executeQuery();

String s11;

int opq=0;

while(rsz.next())

{s11=rsz.getString("num");

if(n1.equals(s11))

opq=1;}

if(opq==1)

{try

{DriverManager.registerDriver(new oracle.jdbc.OracleDriver());

Connection cz1=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","system","password");

PreparedStatement sz111=cz1.prepareStatement(query);

sz111.setString(1,n1);

ResultSet rsz11=sz111.executeQuery();

JOptionPane.showMessageDialog(null,"Room num deleted from DB");

sz111.close();

cz1.close();}

catch(Exception kl){JOptionPane.showMessageDialog(null,"GGG "+kl);}

try

{DriverManager.registerDriver(new oracle.jdbc.OracleDriver());

Connection cza1=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","system","password");

PreparedStatement sza111=cza1.prepareStatement(query1);

sza111.setString(14,n1);

ResultSet rsza11=sza111.executeQuery();

JOptionPane.showMessageDialog(null,"Room num deleted from DB");

sza111.close();

cza1.close();}

catch(Exception kql){JOptionPane.showMessageDialog(null,"GGG "+kql);}}

else

JOptionPane.showMessageDialog(null,"Room num not found in DB");

cz.close();

sz.close();}

catch(Exception dd){}}}}}

class sub implements ActionListener

{JFrame p4,p111;

JLabel l1,l2,l3,l4,l5,l6,l11,l22,l33,l44,l55,l66;

JTextField t1,t2,t3,t4,t5,t6,t11,t22,t33,t44,t55,t66;

JButton b1,b2,b3;

String branch,room;

sub(JFrame p11,String n3,String n1)

{room=n1;

branch=n3;

p111=p11;

p4=new JFrame("Subjects");

p4.setSize(1500,1500);

p4.setVisible(true);

p4.setLayout(new FlowLayout());

l1=new JLabel("Enter sub name");

l2=new JLabel("Enter sub name");

l3=new JLabel("Enter sub name");

l4=new JLabel("Enter sub name");

l5=new JLabel("Enter sub name");

l6=new JLabel("Enter sub name");

l11=new JLabel("Enter teacher ID");

l22=new JLabel("Enter teacher ID");

l33=new JLabel("Enter teacher ID");

l44=new JLabel("Enter teacher ID");

l55=new JLabel("Enter teacher ID");

l66=new JLabel("Enter teacher ID");

t1=new JTextField(10);

t2=new JTextField(10);

t3=new JTextField(10);

t4=new JTextField(10);

t5=new JTextField(10);

t6=new JTextField(10);

t11=new JTextField(10);

t22=new JTextField(10);

t33=new JTextField(10);

t44=new JTextField(10);

t55=new JTextField(10);

t66=new JTextField(10);

b1=new JButton("Add");

b2=new JButton("Back");

b3=new JButton("Cancel");

l1.setFont(new Font("gill sans",Font.BOLD,50));

l2.setFont(new Font("gill sans",Font.BOLD,50));

l3.setFont(new Font("gill sans",Font.BOLD,50));

l4.setFont(new Font("gill sans",Font.BOLD,50));

l5.setFont(new Font("gill sans",Font.BOLD,50));

l6.setFont(new Font("gill sans",Font.BOLD,50));

l11.setFont(new Font("gill sans",Font.BOLD,50));

l22.setFont(new Font("gill sans",Font.BOLD,50));

l33.setFont(new Font("gill sans",Font.BOLD,50));

l44.setFont(new Font("gill sans",Font.BOLD,50));

l55.setFont(new Font("gill sans",Font.BOLD,50));

l66.setFont(new Font("gill sans",Font.BOLD,50));

t1.setFont(new Font("gill sans",Font.BOLD,20));

t2.setFont(new Font("gill sans",Font.BOLD,20));

t3.setFont(new Font("gill sans",Font.BOLD,20));

t4.setFont(new Font("gill sans",Font.BOLD,20));

t5.setFont(new Font("gill sans",Font.BOLD,20));

t6.setFont(new Font("gill sans",Font.BOLD,20));

t11.setFont(new Font("gill sans",Font.BOLD,20));

t22.setFont(new Font("gill sans",Font.BOLD,20));

t33.setFont(new Font("gill sans",Font.BOLD,20));

t44.setFont(new Font("gill sans",Font.BOLD,20));

t55.setFont(new Font("gill sans",Font.BOLD,20));

t66.setFont(new Font("gill sans",Font.BOLD,20));

b1.setFont(new Font("gill sans",Font.BOLD,25));

b2.setFont(new Font("gill sans",Font.BOLD,25));

b3.setFont(new Font("gill sans",Font.BOLD,25));

p4.add(l1);

p4.add(t1);

p4.add(l11);

p4.add(t11);

p4.add(l2);

p4.add(t2);

p4.add(l22);

p4.add(t22);

p4.add(l3);

p4.add(t3);

p4.add(l33);

p4.add(t33);

p4.add(l4);

p4.add(t4);

p4.add(l44);

p4.add(t44);

p4.add(l5);

p4.add(t5);

p4.add(l55);

p4.add(t55);

p4.add(l6);

p4.add(t6);

p4.add(l66);

p4.add(t66);

p4.add(b1);

p4.add(b2);

p4.add(b3);

b1.addActionListener(this);

b2.addActionListener(this);

b3.addActionListener(this);}

public void actionPerformed(ActionEvent ae)

{String s1=t1.getText();

String s11=t11.getText();

String s2=t2.getText();

String s22=t22.getText();

String s3=t3.getText();

String s33=t33.getText();

String s4=t4.getText();

String s44=t44.getText();

String s5=t5.getText();

String s55=t55.getText();

String s6=t6.getText();

String s66=t66.getText();

if(ae.getSource().equals(b3))

{p4.setVisible(false);}

if(ae.getSource().equals(b2))

{p4.setVisible(false);

p111.setVisible(true);}

if(ae.getSource().equals(b1))

{if(s1.equals("")||s2.equals("")||s3.equals("")||s4.equals("")||s5.equals("")||s6.equals("")||s11.equals("")||s22.equals("")||s33.equals("")||s44.equals("")||s55.equals("")||s66.equals(""))

{JOptionPane.showMessageDialog(null,"Fill the fields correctly");}

else

{try

{

DriverManager.registerDriver(new oracle.jdbc.OracleDriver());

Connection c1=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","system","password");

PreparedStatement s=c1.prepareStatement("select sid,sub from addstaff ");

ResultSet rs=s.executeQuery();

String a1="",a2="";

int op=0,xy=0;

while(rs.next())

{a1=rs.getString("sid");

a2=rs.getString("sub");

if(s11.equals(a1))

{if(s1.equals(a2))

op++;

else

JOptionPane.showMessageDialog(null,"teacher and subject not matched:1");}

else

xy++;

if(s22.equals(a1))

{if(s2.equals(a2))

op++;

else

JOptionPane.showMessageDialog(null,"teacher and subject not matched:2");}

else

xy++;

if(s33.equals(a1))

{if(s3.equals(a2))

op++;

else

JOptionPane.showMessageDialog(null,"teacher and subject not matched:3");}

else

xy++;

if(s44.equals(a1))

{if(s4.equals(a2))

op++;

else

JOptionPane.showMessageDialog(null,"teacher and subject not matched:4");}

else

xy++;

if(s55.equals(a1))

{if(s5.equals(a2))

op++;

else

JOptionPane.showMessageDialog(null,"teacher and subject not matched:5");}

else

xy++;

if(s66.equals(a1))

{if(s6.equals(a2))

op++;

else

JOptionPane.showMessageDialog(null,"teacher and subject not matched:6");}

else

xy++;

}if(op==6)

{JOptionPane.showMessageDialog(null,"Done");

DriverManager.registerDriver(new oracle.jdbc.OracleDriver());

Connection c=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","system","password");

PreparedStatement ps=c.prepareStatement("insert into xxx values(?,?,?,?,?,?,?,?,?,?,?,?,?,?)");

ps.setString(1,s1);

ps.setString(2,s11);

ps.setString(3,s2);

ps.setString(4,s22);

ps.setString(5,s3);

ps.setString(6,s33);

ps.setString(7,s4);

ps.setString(8,s44);

ps.setString(9,s5);

ps.setString(10,s55);

ps.setString(11,s6);

ps.setString(12,s66);

ps.setString(13,branch);

ps.setString(14,room);

int m=ps.executeUpdate();

if(m==1)

{JOptionPane.showMessageDialog(null,"Succesfully added");

p111.setVisible(true);}

else

{JOptionPane.showMessageDialog(null,"Not added");}}

else

JOptionPane.showMessageDialog(null,"Not Done");}

catch(Exception oo)

{JOptionPane.showMessageDialog(null,"Exception is "+oo);}}}}}

class tt implements ActionListener

{JFrame p6,p11;

JLabel l1;

JTextField t1;

JButton b1,b2;

tt(JFrame p1)

{p11=p1;

p6=new JFrame("creating timetable");

p6.setSize(1500,1500);

p6.setVisible(true);

p6.setLayout(new FlowLayout());

l1=new JLabel("Enter room number");

t1=new JTextField(10);

b1=new JButton("Next");

b2=new JButton("Back");

l1.setFont(new Font("gill sans",Font.BOLD,50));

t1.setFont(new Font("gill sans",Font.BOLD,20));

b1.setFont(new Font("gill sans",Font.BOLD,25));

b2.setFont(new Font("gill sans",Font.BOLD,25));

p6.add(l1);

p6.add(t1);

p6.add(b1);

p6.add(b2);

b1.addActionListener(this);

b2.addActionListener(this);}

public void actionPerformed(ActionEvent ae)

{String sy11="";

String jyo="";

String sec="";

String st1=t1.getText();

if(ae.getSource().equals(b2))

{p6.setVisible(false);p11.setVisible(true);}

if(ae.getSource().equals(b1))

{Connection c=null;

PreparedStatement sw=null;

try

{DriverManager.registerDriver(new oracle.jdbc.OracleDriver());

c=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","system","password");

sw=c.prepareStatement("select num,course,sec from room ");

ResultSet rsw=sw.executeQuery();

int opw=0;

while(rsw.next())

{sy11=rsw.getString("num");

if(st1.equals(sy11))

{opw=1;

jyo=rsw.getString("course");

sec=rsw.getString("sec");}}

if(opw==1)

{Random D=new Random();

int yu=0;

int k=0;

int a[]=new int[36];

String b[]=new String[36];

String c1w[]=new String[36];

for(int i=0;i<6;i++)

{yu=0;

int r1=D.nextInt(6);

a[k]=r1;

for(int j=0;j<k;j++)

{if(a[k]==a[j])

{yu++;}}

if(yu>0)

{i--;}

else{k++;}}

for(int i=6;i<12;i++)

{yu=0;

int r11=D.nextInt(6);

a[k]=r11;

for(int j=6;j<k;j++)

{if(a[k]==a[j])

{yu++;}}

if(yu>0)

{i--;}

else{k++;}}

for(int i=12;i<18;i++)

{yu=0;

int r11=D.nextInt(6);

a[k]=r11;

for(int j=12;j<k;j++)

{if(a[k]==a[j])

{yu++;}}

if(yu>0)

{i--;}

else{k++;}}

for(int i=18;i<24;i++)

{yu=0;

int r11=D.nextInt(6);

a[k]=r11;

for(int j=18;j<k;j++)

{if(a[k]==a[j])

{yu++;}}

if(yu>0)

{i--;}

else{k++;}}

for(int i=24;i<30;i++)

{yu=0;

int r11=D.nextInt(6);

a[k]=r11;

for(int j=24;j<k;j++)

{if(a[k]==a[j])

{yu++;}}

if(yu>0)

{i--;}

else{k++;}}

for(int i=30;i<36;i++)

{yu=0;

int r11=D.nextInt(6);

a[k]=r11;

for(int j=30;j<k;j++)

{if(a[k]==a[j])

{yu++;}}

if(yu>0)

{i--;}

else{k++;}}

int yup=0,ru=0;

try

{DriverManager.registerDriver(new oracle.jdbc.OracleDriver());

Connection c1=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","system","password");

PreparedStatement s=c1.prepareStatement("select \* from xxx");

ResultSet rs=s.executeQuery();

String s1,s2,s3,s4,s5,s6;String sq1,sq2,sq3,sq4,sq5,sq6;

int op=0,op1=1,op2=2,op3=3,op4=4,op5=5;

int c11=0,c2=0,c3=0,c4=0,c5=0,c6=0;

while(rs.next())

{String hq1=rs.getString("roomnum");

if(hq1.equals(st1))

{s1=rs.getString("T1");

s2=rs.getString("T2");

s3=rs.getString("T3");

s4=rs.getString("T4");

s5=rs.getString("T5");

s6=rs.getString("T6");

sq1=rs.getString("T11");

sq2=rs.getString("T22");

sq3=rs.getString("T33");

sq4=rs.getString("T44");

sq5=rs.getString("T55");

sq6=rs.getString("T66");

try

{for(int q=0;q<k;q++)

{if(a[q]==op)

{b[yup]=s1;yup++;c1w[ru]=sq1;ru++;}

if(a[q]==op1)

{b[yup]=s2;yup++;c1w[ru]=sq2;ru++;}

if(a[q]==op2)

{b[yup]=s3;yup++;c1w[ru]=sq3;ru++;}

if(a[q]==op3)

{b[yup]=s4;yup++;c1w[ru]=sq4;ru++;}

if(a[q]==op4)

{b[yup]=s5;yup++;c1w[ru]=sq5;ru++;}

if(a[q]==op5)

{b[yup]=s6;yup++;c1w[ru]=sq6;ru++;}}}

catch(Exception jkl){JOptionPane.showMessageDialog(null,"FFF"+jkl);}

c1.close();

s.close();}}}

catch(Exception kl){}

new cha(p6,jyo,sec,yup,b,ru,c1w,sy11);}

else

JOptionPane.showMessageDialog(null,"Cannot find room num");

c.close();

sw.close();

rsw.close();}

catch(Exception jj){JOptionPane.showMessageDialog(null,"EXCEPTION IS "+jj);}}}

class cha implements ActionListener

{String pen;

JFrame u,p6612;

JTable j;

JLabel l1;

JTextField t1,t2;

JButton b1,b2;

int yup1,ru1;

String qua[]=new String[100];

String p11[]=new String[100];

String p22[]=new String[100];

String p33[]=new String[100];

String p44[]=new String[100];

String p55[]=new String[100];

String p66[]=new String[100];

String pq11[]=new String[100];

String pq22[]=new String[100];

String pq33[]=new String[100];

String pq44[]=new String[100];

String pq55[]=new String[100];

String pq66[]=new String[100];

String pu11[]=new String[100];

String pu22[]=new String[100];

String pu33[]=new String[100];

String pu44[]=new String[100];

String pu55[]=new String[100];

String pu66[]=new String[100];

String pv11[]=new String[100];

String pv22[]=new String[100];

String pv33[]=new String[100];

String pv44[]=new String[100];

String pv55[]=new String[100];

String pv66[]=new String[100];

String pw11[]=new String[100];

String pw22[]=new String[100];

String pw33[]=new String[100];

String pw44[]=new String[100];

String pw55[]=new String[100];

String pw66[]=new String[100];

String px11[]=new String[100];

String px22[]=new String[100];

String px33[]=new String[100];

String px44[]=new String[100];

String px55[]=new String[100];

String px66[]=new String[100];

int qua1=0,p555=0,p666=0,p111=0,p222=0,p333=0,p444=0,pq555=0,pq666=0,pq111=0,pq222=0,pq333=0,pq444=0,pu555=0,pu666=0,pu111=0,pu222=0,pu333=0,pu444=0;

int pv555=0,pv666=0,pv111=0,pv222=0,pv333=0,pv444=0,pw555=0,pw666=0,pw111=0,pw222=0,pw333=0,pw444=0,px555=0,px666=0,px111=0,px222=0,px333=0,px444=0;

String g11,g12,g13,g14,g15,g16,sec1;

String gu11,gu12,gu13,gu14,gu15,gu16;

String gq11,gq12,gq13,gq14,gq15,gq16;

String gw11,gw12,gw13,gw14,gw15,gw16;

String gz11,gz12,gz13,gz14,gz15,gz16;

String gt11,gt12,gt13,gt14,gt15,gt16,jyo1;

String[] columnNames = {"Day","I", "II", "III","IV","V","VI"};

cha(JFrame p6,String jyo,String sec,int yup,String b[],int ru,String c1w[],String sy11)

{jyo1=jyo;sec1=sec;pen=sy11;

yup1=yup;ru1=ru;

String h[]=new String[yup1];

p6612=p6;

u=new JFrame("jd");

u.setSize(1500,1500);

u.setVisible(true);

u.setLayout(new FlowLayout());

l1=new JLabel("class is:");

t1=new JTextField(10);

t2=new JTextField(5);

String[][] data = {{ "Mon",b[0],b[1],b[2],b[3],b[4],b[5] },

{" ",c1w[0],c1w[1],c1w[2],c1w[3],c1w[4],c1w[5]},

{ "Tue", b[6],b[7],b[8],b[9],b[10],b[11]},

{" ",c1w[6],c1w[7],c1w[8],c1w[9],c1w[10],c1w[11]},

{ "Wed",b[12],b[13],b[14],b[15],b[16],b[17] },

{" ",c1w[12],c1w[13],c1w[14],c1w[15],c1w[16],c1w[17] },

{ "Thu", b[18],b[19],b[20],b[21],b[22],b[23]},

{" ",c1w[18],c1w[19],c1w[20],c1w[21],c1w[22],c1w[23]},

{ "Fri",b[24],b[25],b[26],b[27],b[28],b[29]},

{" ",c1w[24],c1w[25],c1w[26],c1w[27],c1w[28],c1w[29]},

{ "Sat", b[30],b[31],b[32],b[33],b[34],b[35]},

{" ",c1w[30],c1w[31],c1w[32],c1w[33],c1w[34],c1w[35]}};

j = new JTable(data, columnNames);

j.setRowHeight( 0,50);

j.setRowHeight( 1,50);

j.setRowHeight( 2,50);

j.setRowHeight( 3,50);

j.setRowHeight( 4,50);

j.setRowHeight(5,50);

j.setRowHeight( 6,50);

j.setRowHeight( 7,50);

j.setRowHeight( 8,50);

j.setRowHeight( 9,50);

j.setRowHeight( 10,50);

j.setRowHeight(11,50);

j.getColumnModel().getColumn(1).setWidth(300 );

j.getColumnModel().getColumn(2).setWidth(300 );

j.getColumnModel().getColumn(3).setWidth(300 );

j.getColumnModel().getColumn(4).setWidth(300 );

j.getColumnModel().getColumn(5).setWidth(300 );

j.getColumnModel().getColumn(0).setWidth(300 );

b1=new JButton("Back");

b2=new JButton("Create");

l1.setFont(new Font("gill sans",Font.BOLD,50));

t1.setFont(new Font("gill sans",Font.BOLD,20));

t2.setFont(new Font("gill sans",Font.BOLD,20));

b1.setFont(new Font("gill sans",Font.BOLD,25));

b2.setFont(new Font("gill sans",Font.BOLD,25));

u.add(l1);

u.add(t1);

u.add(t2);

u.add(j);

u.add(b1);

t1.setText(jyo1);

t2.setText(sec1);

t1.setEditable(false);

t2.setEditable(false);

b1.addActionListener(this);

String mon="mon";String tue="tue";String wed="wed";String thu="thu";String fri="fri";String sat="sat";

String zp1=b[0];String zp2=b[1];String zp3=b[2];String zp4=b[3];String zp5=b[4];String zp6=b[5];

String zp7=b[6];String zp8=b[7];String zp9=b[8];String zp10=b[9];String zp11=b[10];String zp12=b[11];

String zp13=b[12];String zp14=b[13];String zp15=b[14];String zp16=b[15];String zp17=b[16];String zp18=b[17];

String zp19=b[18];String zp20=b[19];String zp21=b[20];String zp22=b[21];String zp23=b[22];String zp24=b[23];

String zp25=b[24];String zp26=b[25];String zp27=b[26];String zp28=b[27];String zp29=b[28];String zp30=b[29];

String zp31=b[30];String zp32=b[31];String zp33=b[32];String zp34=b[33];String zp35=b[34];String zp36=b[35];

String zpq1=c1w[0];String zpq2=c1w[1];String zpq3=c1w[2];String zpq4=c1w[3];String zpq5=c1w[4];String zpq6=c1w[5];

String zpq7=c1w[6];String zpq8=c1w[7];String zpq9=c1w[8];String zpq10=c1w[9];String zpq11=c1w[10];String zpq12=c1w[11];

String zpq13=c1w[12];String zpq14=c1w[13];String zpq15=c1w[14];String zpq16=c1w[15];String zpq17=c1w[16];String zpq18=c1w[17];

String zpq19=c1w[18];String zpq20=c1w[19];String zpq21=c1w[20];String zpq22=c1w[21];String zpq23=c1w[22];String zpq24=c1w[23];

String zpq25=c1w[24];String zpq26=c1w[25];String zpq27=c1w[26];String zpq28=c1w[27];String zpq29=c1w[28];String zpq30=c1w[29];

String zpq31=c1w[30];String zpq32=c1w[31];String zpq33=c1w[32];String zpq34=c1w[33];String zpq35=c1w[34];String zpq36=c1w[35];

try

{int yu=1;

int water=0;

try

{DriverManager.registerDriver(new oracle.jdbc.OracleDriver());

Connection cwq=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","system","password");

Statement swq=cwq.createStatement();

ResultSet rs12=swq.executeQuery("select \* from ids11");

while(rs12.next())

{String qw=rs12.getString(45);

if(jyo1.equals(qw))

{p11[p111]=rs12.getString(2);

p22[p222]=rs12.getString(3);

p33[p333]=rs12.getString(4);

p44[p444]=rs12.getString(5);

p55[p555]=rs12.getString(6);

p66[p666]=rs12.getString(7);

pq11[pq111]=rs12.getString(9);

pq22[pq222]=rs12.getString(10);

pq33[pq333]=rs12.getString(11);

pq44[pq444]=rs12.getString(12);

pq55[pq555]=rs12.getString(13);

pq66[pq666]=rs12.getString(14);

pu11[pu111]=rs12.getString(16);

pu22[pu222]=rs12.getString(17);

pu33[pu333]=rs12.getString(18);

pu44[pu444]=rs12.getString(19);

pu55[pu555]=rs12.getString(20);

pu66[pu666]=rs12.getString(21);

pv11[pv111]=rs12.getString(23);

pv22[pv222]=rs12.getString(24);

pv33[pv333]=rs12.getString(25);

pv44[pv444]=rs12.getString(26);

pv55[pv555]=rs12.getString(27);

pv66[pv666]=rs12.getString(28);

pw11[pw111]=rs12.getString(30);

pw22[pw222]=rs12.getString(31);

pw33[pw333]=rs12.getString(32);

pw44[pw444]=rs12.getString(33);

pw55[pw555]=rs12.getString(34);

pw66[pw666]=rs12.getString(35);

px11[px111]=rs12.getString(37);

px22[px222]=rs12.getString(38);

px33[px333]=rs12.getString(39);

px44[px444]=rs12.getString(40);

px55[px555]=rs12.getString(41);

px66[px666]=rs12.getString(42);

qua[qua1]=rs12.getString(43);

qua1++;

p111++;

p222++;

p333++;

p444++;

p555++;

p666++;

pq111++;

pq222++;

pq333++;

pq444++;

pq555++;

pq666++;

pu111++;

pu222++;

pu333++;

pu444++;

pu555++;

pu666++;

pv111++;

pv222++;

pv333++;

pv444++;

pv555++;

pv666++;

pw111++;

pw222++;

pw333++;

pw444++;

pw555++;

pw666++;

px111++;

px222++;

px333++;

px444++;

px555++;

px666++;}}

for(int qwe=0;qwe<qua1;qwe++){

if(qua[qwe].equals(jyo1)){

for(int gh=0;gh<p111;gh++){

if(p11[gh].equals(zpq1)||p22[gh].equals(zpq2)||p33[gh].equals(zpq3)||p44[gh].equals(zpq4)||p55[gh].equals(zpq5)||p66[gh].equals(zpq6)||pq11[gh].equals(zpq7)||pq22[gh].equals(zpq8)||pq33[gh].equals(zpq9)||pq44[gh].equals(zpq10)||pq55[gh].equals(zpq11)||pq66[gh].equals(zpq12)){

water++;}}}}

cwq.close();

swq.close();}

catch(Exception jweqk){JOptionPane.showMessageDialog(null,"gttt"+jweqk);}

if(water==0)

{try

{DriverManager.registerDriver(new oracle.jdbc.OracleDriver());

Connection cwe=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","system","password");

PreparedStatement psr=cwe.prepareStatement("insert into ttgmm11 values(?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?)");

psr.setString(1,mon);

psr.setString(2,zp1);

psr.setString(3,zp2);

psr.setString(4,zp3);

psr.setString(5,zp4);

psr.setString(6,zp5);

psr.setString(7,zp6);

psr.setString(8,tue);

psr.setString(9,zp7);

psr.setString(10,zp8);

psr.setString(11,zp9);

psr.setString(12,zp10);

psr.setString(13,zp11);

psr.setString(14,zp12);

psr.setString(15,wed);

psr.setString(16,zp13);

psr.setString(17,zp14);

psr.setString(18,zp15);

psr.setString(19,zp16);

psr.setString(20,zp17);

psr.setString(21,zp18);

psr.setString(22,thu);

psr.setString(23,zp19);

psr.setString(24,zp20);

psr.setString(25,zp21);

psr.setString(26,zp22);

psr.setString(27,zp23);

psr.setString(28,zp24);

psr.setString(29,fri);

psr.setString(30,zp25);

psr.setString(31,zp26);

psr.setString(32,zp27);

psr.setString(33,zp28);

psr.setString(34,zp29);

psr.setString(35,zp30);

psr.setString(36,sat);

psr.setString(37,zp31);

psr.setString(38,zp32);

psr.setString(39,zp33);

psr.setString(40,zp34);

psr.setString(41,zp35);

psr.setString(42,zp36);

psr.setString(43,jyo1);

psr.setString(44,sec1);

psr.setString(45,sy11);

int m=psr.executeUpdate();

if(m==1)

{JOptionPane.showMessageDialog(null,"Succesfully added");}

else

JOptionPane.showMessageDialog(null,"not added");

cwe.close();

psr.close();}

catch(Exception jkooyy){}

try

{DriverManager.registerDriver(new oracle.jdbc.OracleDriver());

Connection cwe1=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","system","password");

PreparedStatement psr1=cwe1.prepareStatement("insert into ids11 values(?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?,?)");

psr1.setString(1,mon);

psr1.setString(2,zpq1);

psr1.setString(3,zpq2);

psr1.setString(4,zpq3);

psr1.setString(5,zpq4);

psr1.setString(6,zpq5);

psr1.setString(7,zpq6);

psr1.setString(8,tue);

psr1.setString(9,zpq7);

psr1.setString(10,zpq8);

psr1.setString(11,zpq9);

psr1.setString(12,zpq10);

psr1.setString(13,zpq11);

psr1.setString(14,zpq12);

psr1.setString(15,wed);

psr1.setString(16,zpq13);

psr1.setString(17,zpq14);

psr1.setString(18,zpq15);

psr1.setString(19,zpq16);

psr1.setString(20,zpq17);

psr1.setString(21,zpq18);

psr1.setString(22,thu);

psr1.setString(23,zpq19);

psr1.setString(24,zpq20);

psr1.setString(25,zpq21);

psr1.setString(26,zpq22);

psr1.setString(27,zpq23);

psr1.setString(28,zpq24);

psr1.setString(29,fri);

psr1.setString(30,zpq25);

psr1.setString(31,zpq26);

psr1.setString(32,zpq27);

psr1.setString(33,zpq28);

psr1.setString(34,zpq29);

psr1.setString(35,zpq30);

psr1.setString(36,sat);

psr1.setString(37,zpq31);

psr1.setString(38,zpq32);

psr1.setString(39,zpq33);

psr1.setString(40,zpq34);

psr1.setString(41,zpq35);

psr1.setString(42,zpq36);

psr1.setString(43,jyo1);

psr1.setString(44,sec1);

psr1.setString(45,pen);

int m1=psr1.executeUpdate();

if(m1==1)

{JOptionPane.showMessageDialog(null,"Succesfully added");}

else

JOptionPane.showMessageDialog(null,"not added");

cwe1.close();

psr1.close();}

catch(Exception jkire){}}

else

{JOptionPane.showMessageDialog(null,"collision occured cant add to DB");}}

catch(Exception jklpqwer){JOptionPane.showMessageDialog(null,"oo"+jklpqwer);}}

public void actionPerformed(ActionEvent ae)

{if(ae.getSource().equals(b1))

{u.setVisible(false);

p6612.setVisible(true);}}}}

class view implements ActionListener

{JFrame po,v1;

JLabel l1,l2,l3,l4;

JButton b1,b2,b3,b4,b5,b6,b7,b8;

JTextField t1,t2,t3;

view(JFrame p1)

{po=p1;

v1=new JFrame("viewing tt");

v1.setLayout(new FlowLayout());

v1.setVisible(true);

v1.setSize(1500,1500);

b4=new JButton("Back");

b5=new JButton("Cancel");

b6=new JButton("View/delete");

l2=new JLabel("Enter room num");

t1=new JTextField(10);

v1.add(l2);

v1.add(t1);

v1.add(b6);

v1.add(b4);

v1.add(b5);

l2.setFont(new Font("gill sans",Font.BOLD,50));

t1.setFont(new Font("gill sans",Font.BOLD,20));

b4.setFont(new Font("gill sans",Font.BOLD,25));

b5.setFont(new Font("gill sans",Font.BOLD,25));

b6.setFont(new Font("gill sans",Font.BOLD,25));

b4.addActionListener(this);

b5.addActionListener(this);

b6.addActionListener(this);}

public void actionPerformed(ActionEvent ae)

{String t11=t1.getText();

if(ae.getSource().equals(b5))

{v1.setVisible(false);}

if(ae.getSource().equals(b4))

{v1.setVisible(false);

po.setVisible(true);}

if(ae.getSource().equals(b6))

{final JPanel panel1=new JPanel();

final JRadioButton button1=new JRadioButton("view");

final JRadioButton button2=new JRadioButton("delete");

ButtonGroup br=new ButtonGroup();

br.add(button1);

br.add(button2);

panel1.add(button1);

panel1.add(button2);

JOptionPane.showMessageDialog(null,panel1);

if(button2.isSelected())

{if(t11.equals(""))

{JOptionPane.showMessageDialog(null,"Empty fields not allowed");}

else

{try

{DriverManager.registerDriver(new oracle.jdbc.OracleDriver());

Connection c=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","system","password");

PreparedStatement s=c.prepareStatement("select \* from ttgmm11");

ResultSet rs=s.executeQuery();

String s11;

int op=0;

while(rs.next())

{s11=rs.getString("roomnum");

if(t11.equals(s11))

op=1;}

String rt="mon";

if(op==1)

{JOptionPane.showMessageDialog(null,"Deleted Successfully");

String query="delete from ttgmm11 where roomnum=?";

try

{DriverManager.registerDriver(new oracle.jdbc.OracleDriver());

Connection cwqa=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","system","password");

Statement swqa=cwqa.createStatement();

ResultSet rs12=swqa.executeQuery("select \* from ttgmm11");

while(rs12.next())

{String qw=rs12.getString(45);

if(t11.equals(qw))

{try

{DriverManager.registerDriver(new oracle.jdbc.OracleDriver());

Connection c1=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","system","password");

PreparedStatement s111=c1.prepareStatement(query);

s111.setString(45,t11);

ResultSet rs11=s111.executeQuery();

JOptionPane.showMessageDialog(null,"ID deleted from DB");

t1.setText("");

s111.close();

c1.close();}

catch(Exception kl){JOptionPane.showMessageDialog(null,"GGG "+kl);}

String query1="delete from ids11 where roomnum=?";

try

{DriverManager.registerDriver(new oracle.jdbc.OracleDriver());

Connection cwqr=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","system","password");

Statement swqr=cwqr.createStatement();

ResultSet rs12r=swqr.executeQuery("select \* from ids11");

while(rs12r.next())

{String qwq=rs12.getString(45);

if(t11.equals(qwq))

{try

{DriverManager.registerDriver(new oracle.jdbc.OracleDriver());

Connection cq1=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","system","password");

PreparedStatement sq111=cq1.prepareStatement(query1);

sq111.setString(45,t11);

ResultSet rsq11=sq111.executeQuery();

JOptionPane.showMessageDialog(null,"ID deleted from DB");

t1.setText("");

sq111.close();

cq1.close();}

catch(Exception klq){JOptionPane.showMessageDialog(null,"GGG12 "+klq);}}}

cwqr.close();

swqr.close();}

catch(Exception uio){}}}

s.close();

c.close();}

catch(Exception jui){}}

else

JOptionPane.showMessageDialog(null,"Not Deleted");}

catch(Exception hy){}}}

if(button1.isSelected())

{int op=0;

if(t11.equals(""))

{JOptionPane.showMessageDialog(null,"Empty fields not allowed");}

else

{int yto=0;

String q[]=new String[42];

try

{DriverManager.registerDriver(new oracle.jdbc.OracleDriver());

Connection c=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","system","password");

PreparedStatement s=c.prepareStatement("select \* from ttgmm11");

ResultSet rs=s.executeQuery();

String s11;

while(rs.next())

{op=1;

s11=rs.getString("roomnum");

if(t11.equals(s11))

{q[yto]=rs.getString(1);

yto++;

q[yto]=rs.getString(2);

yto++;

q[yto]=rs.getString(3);

yto++;

q[yto]=rs.getString(4);

yto++;

q[yto]=rs.getString(5);

yto++;

q[yto]=rs.getString(6);

yto++;

q[yto]=rs.getString(7);

yto++;

q[yto]=rs.getString(8);

yto++;

q[yto]=rs.getString(9);

yto++;

q[yto]=rs.getString(10);

yto++;

q[yto]=rs.getString(11);

yto++;

q[yto]=rs.getString(12);

yto++;

q[yto]=rs.getString(13);

yto++;

q[yto]=rs.getString(14);

yto++;

q[yto]=rs.getString(15);

yto++;

q[yto]=rs.getString(16);

yto++;

q[yto]=rs.getString(17);

yto++;

q[yto]=rs.getString(18);

yto++;

q[yto]=rs.getString(19);

yto++;

q[yto]=rs.getString(20);

yto++;

q[yto]=rs.getString(21);

yto++;

q[yto]=rs.getString(22);

yto++;

q[yto]=rs.getString(23);

yto++;

q[yto]=rs.getString(24);

yto++;

q[yto]=rs.getString(25);

yto++;

q[yto]=rs.getString(26);

yto++;

q[yto]=rs.getString(27);

yto++;

q[yto]=rs.getString(28);

yto++;

q[yto]=rs.getString(29);

yto++;

q[yto]=rs.getString(30);

yto++;

q[yto]=rs.getString(31);

yto++;

q[yto]=rs.getString(32);

yto++;

q[yto]=rs.getString(33);

yto++;

q[yto]=rs.getString(34);

yto++;

q[yto]=rs.getString(35);

yto++;

q[yto]=rs.getString(36);

yto++;

q[yto]=rs.getString(37);

yto++;

q[yto]=rs.getString(38);

yto++;

q[yto]=rs.getString(39);

yto++;

q[yto]=rs.getString(40);

yto++;

q[yto]=rs.getString(41);

yto++;

q[yto]=rs.getString(42);

}}}

catch(Exception jklo){}

int yto1=0;

String q1[]=new String[42];

try

{DriverManager.registerDriver(new oracle.jdbc.OracleDriver());

Connection c=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","system","password");

PreparedStatement s=c.prepareStatement("select \* from ids11");

ResultSet rs=s.executeQuery();

String s11;

while(rs.next())

{s11=rs.getString("roomnum");

if(t11.equals(s11))

{q1[yto1]=rs.getString(1);

yto1++;

q1[yto1]=rs.getString(2);

yto1++;

q1[yto1]=rs.getString(3);

yto1++;

q1[yto1]=rs.getString(4);

yto1++;

q1[yto1]=rs.getString(5);

yto1++;

q1[yto1]=rs.getString(6);

yto1++;

q1[yto1]=rs.getString(7);

yto1++;

q1[yto1]=rs.getString(8);

yto1++;

q1[yto1]=rs.getString(9);

yto1++;

q1[yto1]=rs.getString(10);

yto1++;

q1[yto1]=rs.getString(11);

yto1++;

q1[yto1]=rs.getString(12);

yto1++;

q1[yto1]=rs.getString(13);

yto1++;

q1[yto1]=rs.getString(14);

yto1++;

q1[yto1]=rs.getString(15);

yto1++;

q1[yto1]=rs.getString(16);

yto1++;

q1[yto1]=rs.getString(17);

yto1++;

q1[yto1]=rs.getString(18);

yto1++;

q1[yto1]=rs.getString(19);

yto1++;

q1[yto1]=rs.getString(20);

yto1++;

q1[yto1]=rs.getString(21);

yto1++;

q1[yto1]=rs.getString(22);

yto1++;

q1[yto1]=rs.getString(23);

yto1++;

q1[yto1]=rs.getString(24);

yto1++;

q1[yto1]=rs.getString(25);

yto1++;

q1[yto1]=rs.getString(26);

yto1++;

q1[yto1]=rs.getString(27);

yto1++;

q1[yto1]=rs.getString(28);

yto1++;

q1[yto1]=rs.getString(29);

yto1++;

q1[yto1]=rs.getString(30);

yto1++;

q1[yto1]=rs.getString(31);

yto1++;

q1[yto1]=rs.getString(32);

yto1++;

q1[yto1]=rs.getString(33);

yto1++;

q1[yto1]=rs.getString(34);

yto1++;

q1[yto1]=rs.getString(35);

yto1++;

q1[yto1]=rs.getString(36);

yto1++;

q1[yto1]=rs.getString(37);

yto1++;

q1[yto1]=rs.getString(38);

yto1++;

q1[yto1]=rs.getString(39);

yto1++;

q1[yto1]=rs.getString(40);

yto1++;

q1[yto1]=rs.getString(41);

yto1++;

q1[yto1]=rs.getString(42);

}}}

catch(Exception jklo){}

if(op==1)

{new min(v1,yto,q,yto1,q1,t11);}

else

JOptionPane.showMessageDialog(null,"No timetable created with this details");}}}}}

class min implements ActionListener

{JFrame v11,v2;

JButton b1;

JTable j;

String we="";

min(JFrame v1,int yto,String q[],int yto1,String q1[],String t11)

{we=t11;

v11=v1;

v2=new JFrame("table"+we);

v2.setVisible(true);

v2.setSize(1500,1500);

v2.setLayout(new FlowLayout());

b1=new JButton("Back");

String col[]={"Day","I","II","III","IV","V","VI"};

String row[][]={

{q[0],q[1],q[2],q[3],q[4],q[5],q[6]},

{q1[0],q1[1],q1[2],q1[3],q1[4],q1[5],q1[6]},

{q[7],q[8],q[9],q[10],q[11],q[12],q[13]},

{q1[7],q1[8],q1[9],q1[10],q1[11],q1[12],q1[13]},

{q[14],q[15],q[16],q[17],q[18],q[19],q[20]},

{q1[14],q1[15],q1[16],q1[17],q1[18],q1[19],q1[20]},

{q[21],q[22],q[23],q[24],q[25],q[26],q[27]},

{q1[21],q1[22],q1[23],q1[24],q1[25],q1[26],q1[27]},

{q[28],q[29],q[30],q[31],q[32],q[33],q[34]},

{q1[28],q1[29],q1[30],q1[31],q1[32],q1[33],q1[34]},

{q[35],q[36],q[37],q[38],q[39],q[40],q[41]},

{q1[35],q1[36],q1[37],q1[38],q1[39],q1[40],q1[41]}

};

j=new JTable(row,col);

j.setRowHeight( 0,50);

j.setRowHeight( 1,50);

j.setRowHeight( 2,50);

j.setRowHeight( 3,50);

j.setRowHeight( 4,50);

j.setRowHeight(5,50);

j.setRowHeight( 6,50);

j.setRowHeight( 7,50);

j.setRowHeight( 8,50);

j.setRowHeight( 9,50);

j.setRowHeight( 10,50);

j.setRowHeight(11,50);

j.getColumnModel().getColumn(1).setWidth(300 );

j.getColumnModel().getColumn(2).setWidth(300 );

j.getColumnModel().getColumn(3).setWidth(300 );

j.getColumnModel().getColumn(4).setWidth(300 );

j.getColumnModel().getColumn(5).setWidth(300 );

j.getColumnModel().getColumn(0).setWidth(300 );

v2.add(j);

v2.add(b1);

b1.addActionListener(this);}

public void actionPerformed(ActionEvent ae)

{if(ae.getSource().equals(b1))

{v2.setVisible(false);

v11.setVisible(true);}}}

class user1 implements ActionListener

{JFrame f1,f2;

JLabel l,l1,l2,l3;

JTextField t1;

JPasswordField t2;

JButton b1,b2,b3;

String s1,s2;

user1(JFrame f)

{try

{f1=f;

f2=new JFrame("user homepage");

f2.setSize(1500,1500);

f2.setVisible(true);

f2.setLayout(null);

l1=new JLabel("USER HOME PAGE");

l2=new JLabel("User ID");

l3=new JLabel("Password");

t1=new JTextField(20);

t2=new JPasswordField(20);

b1=new JButton("BACK");

b2=new JButton("LOGIN");

b3=new JButton("CANCEL");

ImageIcon x=new ImageIcon("admin3.JPEG");

l=new JLabel(x);

l1.setBounds(300,10,1000,60);

l2.setBounds(250,150,300,50);

l3.setBounds(250,250,300,50);

t1.setBounds(500,150,200,50);

t2.setBounds(500,250,200,50);

b1.setBounds(50,500,200,50);

b2.setBounds(270,500,200,50);

b3.setBounds(490,500,200,50);

l.setBounds(500,100,1000,500);

l1.setFont(new Font("Elephant",Font.BOLD,70));

l2.setFont(new Font("gill sans",Font.BOLD,50));

l3.setFont(new Font("gill sans",Font.BOLD,50));

b1.setFont(new Font("gill sans",Font.BOLD,25));

b2.setFont(new Font("gill sans",Font.BOLD,25));

b3.setFont(new Font("gill sans",Font.BOLD,25));

t1.setFont(new Font("gill sans",Font.BOLD,25));

t2.setFont(new Font("gill sans",Font.BOLD,25));

f2.add(l1);

f2.add(l2);

f2.add(t1);

f2.add(l3);

f2.add(t2);

f2.add(b1);

f2.add(b2);

f2.add(b3);

b1.addActionListener(this);

b2.addActionListener(this);

b3.addActionListener(this);}

catch(Exception j){}}

public void actionPerformed(ActionEvent ae)

{s1=t1.getText();

s2=t2.getText();

if(ae.getSource().equals(b1))

{f2.setVisible(false);

f1.setVisible(true);}

if(ae.getSource().equals(b3))

{f2.setVisible(false);}

if(ae.getSource().equals(b2))

{try

{DriverManager.registerDriver(new oracle.jdbc.OracleDriver());

Connection c1=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","system","password");

PreparedStatement s=c1.prepareStatement("select sid,pw from addstaff ");

ResultSet rs=s.executeQuery();

String n2,n4;

int op=0;

while(rs.next())

{n2=rs.getString("sid");

n4=rs.getString("pw");

if(s1.equals(n2)&&s2.equals(n4))

op=1;}

if(op==1)

{JOptionPane.showMessageDialog(null,"Login Successfull");

new view1(f2);}

else

JOptionPane.showMessageDialog(null,"Invalid login");}

catch(Exception klo){}}}}

class view1 implements ActionListener

{JFrame po,v1;

JLabel l1,l2,l3,l4;

JButton b1,b2,b3,b4,b5,b6,b7,b8;

JTextField t1,t2,t3;

view1(JFrame f2)

{po=f2;

v1=new JFrame("viewing tt");

v1.setLayout(new FlowLayout());

v1.setVisible(true);

v1.setSize(1500,1500);

b4=new JButton("Back");

b5=new JButton("Cancel");

b6=new JButton("View");

l2=new JLabel("Enter room num");

t1=new JTextField(10);

v1.add(l2);

v1.add(t1);

v1.add(b6);

v1.add(b4);

v1.add(b5);

l2.setFont(new Font("gill sans",Font.BOLD,50));

t1.setFont(new Font("gill sans",Font.BOLD,20));

b4.setFont(new Font("gill sans",Font.BOLD,25));

b5.setFont(new Font("gill sans",Font.BOLD,25));

b6.setFont(new Font("gill sans",Font.BOLD,25));

b4.addActionListener(this);

b5.addActionListener(this);

b6.addActionListener(this);}

public void actionPerformed(ActionEvent ae)

{String t11=t1.getText();

if(ae.getSource().equals(b5))

{v1.setVisible(false);}

if(ae.getSource().equals(b4))

{v1.setVisible(false);

po.setVisible(true);}

if(ae.getSource().equals(b6))

{int op=0;

if(t11.equals(""))

{JOptionPane.showMessageDialog(null,"Empty fields not allowed");}

else

{int yto=0;

String q[]=new String[42];

try

{DriverManager.registerDriver(new oracle.jdbc.OracleDriver());

Connection c=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","system","password");

PreparedStatement s=c.prepareStatement("select \* from ttgmm11");

ResultSet rs=s.executeQuery();

String s11;

while(rs.next())

{op=1;

s11=rs.getString("roomnum");

if(t11.equals(s11))

{q[yto]=rs.getString(1);

yto++;

q[yto]=rs.getString(2);

yto++;

q[yto]=rs.getString(3);

yto++;

q[yto]=rs.getString(4);

yto++;

q[yto]=rs.getString(5);

yto++;

q[yto]=rs.getString(6);

yto++;

q[yto]=rs.getString(7);

yto++;

q[yto]=rs.getString(8);

yto++;

q[yto]=rs.getString(9);

yto++;

q[yto]=rs.getString(10);

yto++;

q[yto]=rs.getString(11);

yto++;

q[yto]=rs.getString(12);

yto++;

q[yto]=rs.getString(13);

yto++;

q[yto]=rs.getString(14);

yto++;

q[yto]=rs.getString(15);

yto++;

q[yto]=rs.getString(16);

yto++;

q[yto]=rs.getString(17);

yto++;

q[yto]=rs.getString(18);

yto++;

q[yto]=rs.getString(19);

yto++;

q[yto]=rs.getString(20);

yto++;

q[yto]=rs.getString(21);

yto++;

q[yto]=rs.getString(22);

yto++;

q[yto]=rs.getString(23);

yto++;

q[yto]=rs.getString(24);

yto++;

q[yto]=rs.getString(25);

yto++;

q[yto]=rs.getString(26);

yto++;

q[yto]=rs.getString(27);

yto++;

q[yto]=rs.getString(28);

yto++;

q[yto]=rs.getString(29);

yto++;

q[yto]=rs.getString(30);

yto++;

q[yto]=rs.getString(31);

yto++;

q[yto]=rs.getString(32);

yto++;

q[yto]=rs.getString(33);

yto++;

q[yto]=rs.getString(34);

yto++;

q[yto]=rs.getString(35);

yto++;

q[yto]=rs.getString(36);

yto++;

q[yto]=rs.getString(37);

yto++;

q[yto]=rs.getString(38);

yto++;

q[yto]=rs.getString(39);

yto++;

q[yto]=rs.getString(40);

yto++;

q[yto]=rs.getString(41);

yto++;

q[yto]=rs.getString(42);

}}}

catch(Exception jklo){}

int yto1=0;

String q1[]=new String[42];

try

{DriverManager.registerDriver(new oracle.jdbc.OracleDriver());

Connection c=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","system","password");

PreparedStatement s=c.prepareStatement("select \* from ids11");

ResultSet rs=s.executeQuery();

String s11;

while(rs.next())

{s11=rs.getString("roomnum");

if(t11.equals(s11))

{q1[yto1]=rs.getString(1);

yto1++;

q1[yto1]=rs.getString(2);

yto1++;

q1[yto1]=rs.getString(3);

yto1++;

q1[yto1]=rs.getString(4);

yto1++;

q1[yto1]=rs.getString(5);

yto1++;

q1[yto1]=rs.getString(6);

yto1++;

q1[yto1]=rs.getString(7);

yto1++;

q1[yto1]=rs.getString(8);

yto1++;

q1[yto1]=rs.getString(9);

yto1++;

q1[yto1]=rs.getString(10);

yto1++;

q1[yto1]=rs.getString(11);

yto1++;

q1[yto1]=rs.getString(12);

yto1++;

q1[yto1]=rs.getString(13);

yto1++;

q1[yto1]=rs.getString(14);

yto1++;

q1[yto1]=rs.getString(15);

yto1++;

q1[yto1]=rs.getString(16);

yto1++;

q1[yto1]=rs.getString(17);

yto1++;

q1[yto1]=rs.getString(18);

yto1++;

q1[yto1]=rs.getString(19);

yto1++;

q1[yto1]=rs.getString(20);

yto1++;

q1[yto1]=rs.getString(21);

yto1++;

q1[yto1]=rs.getString(22);

yto1++;

q1[yto1]=rs.getString(23);

yto1++;

q1[yto1]=rs.getString(24);

yto1++;

q1[yto1]=rs.getString(25);

yto1++;

q1[yto1]=rs.getString(26);

yto1++;

q1[yto1]=rs.getString(27);

yto1++;

q1[yto1]=rs.getString(28);

yto1++;

q1[yto1]=rs.getString(29);

yto1++;

q1[yto1]=rs.getString(30);

yto1++;

q1[yto1]=rs.getString(31);

yto1++;

q1[yto1]=rs.getString(32);

yto1++;

q1[yto1]=rs.getString(33);

yto1++;

q1[yto1]=rs.getString(34);

yto1++;

q1[yto1]=rs.getString(35);

yto1++;

q1[yto1]=rs.getString(36);

yto1++;

q1[yto1]=rs.getString(37);

yto1++;

q1[yto1]=rs.getString(38);

yto1++;

q1[yto1]=rs.getString(39);

yto1++;

q1[yto1]=rs.getString(40);

yto1++;

q1[yto1]=rs.getString(41);

yto1++;

q1[yto1]=rs.getString(42);

}}}

catch(Exception jklo){}

if(op==1)

{new min1(v1,yto,q,yto1,q1,t11);}

else

JOptionPane.showMessageDialog(null,"No timetable created with this details");}}}}

class min1 implements ActionListener

{JFrame v11,v2;

JButton b1;

JTable j;

String we="";

min1(JFrame v1,int yto,String q[],int yto1,String q1[],String t11)

{we=t11;

v11=v1;

v2=new JFrame("table"+we);

v2.setVisible(true);

v2.setSize(1500,1500);

v2.setLayout(new FlowLayout());

b1=new JButton("Back");

String col[]={"Day","I","II","III","IV","V","VI"};

String row[][]={{q[0],q[1],q[2],q[3],q[4],q[5],q[6]},

{q1[0],q1[1],q1[2],q1[3],q1[4],q1[5],q1[6]},

{q[7],q[8],q[9],q[10],q[11],q[12],q[13]},

{q1[7],q1[8],q1[9],q1[10],q1[11],q1[12],q1[13]},

{q[14],q[15],q[16],q[17],q[18],q[19],q[20]},

{q1[14],q1[15],q1[16],q1[17],q1[18],q1[19],q1[20]},

{q[21],q[22],q[23],q[24],q[25],q[26],q[27]},

{q1[21],q1[22],q1[23],q1[24],q1[25],q1[26],q1[27]},

{q[28],q[29],q[30],q[31],q[32],q[33],q[34]},

{q1[28],q1[29],q1[30],q1[31],q1[32],q1[33],q1[34]},

{q[35],q[36],q[37],q[38],q[39],q[40],q[41]},

{q1[35],q1[36],q1[37],q1[38],q1[39],q1[40],q1[41]}};

j=new JTable(row,col);

j.setRowHeight( 0,50);

j.setRowHeight( 1,50);

j.setRowHeight( 2,50);

j.setRowHeight( 3,50);

j.setRowHeight( 4,50);

j.setRowHeight(5,50);

j.setRowHeight( 6,50);

j.setRowHeight( 7,50);

j.setRowHeight( 8,50);

j.setRowHeight( 9,50);

j.setRowHeight( 10,50);

j.setRowHeight(11,50);

j.getColumnModel().getColumn(1).setWidth(300 );

j.getColumnModel().getColumn(2).setWidth(300 );

j.getColumnModel().getColumn(3).setWidth(300 );

j.getColumnModel().getColumn(4).setWidth(300 );

j.getColumnModel().getColumn(5).setWidth(300 );

j.getColumnModel().getColumn(0).setWidth(300 );

v2.add(j);

v2.add(b1);

b1.addActionListener(this);}

public void actionPerformed(ActionEvent ae)

{if(ae.getSource().equals(b1))

{v2.setVisible(false);

v11.setVisible(true);}}}

class user2 implements ActionListener

{JFrame f1,f2;

JLabel l,l1,l2,l3;

JTextField t1;

JPasswordField t2;

JButton b1,b2,b3;

String s1,s2;

user2(JFrame f)

{try

{f1=f;

f2=new JFrame("user homepage");

f2.setSize(1500,1500);

f2.setVisible(true);

f2.setLayout(null);

l1=new JLabel("USER HOME PAGE");

l2=new JLabel("User ID");

l3=new JLabel("Password");

t1=new JTextField(20);

t2=new JPasswordField(20);

b1=new JButton("BACK");

b2=new JButton("LOGIN");

b3=new JButton("CANCEL");

ImageIcon x=new ImageIcon("admin3.JPEG");

l=new JLabel(x);

l1.setBounds(300,10,1000,60);

l2.setBounds(250,150,300,50);

l3.setBounds(250,250,300,50);

t1.setBounds(500,150,200,50);

t2.setBounds(500,250,200,50);

b1.setBounds(50,500,200,50);

b2.setBounds(270,500,200,50);

b3.setBounds(490,500,200,50);

l.setBounds(500,100,1000,500);

l1.setFont(new Font("Elephant",Font.BOLD,70));

l2.setFont(new Font("gill sans",Font.BOLD,50));

l3.setFont(new Font("gill sans",Font.BOLD,50));

b1.setFont(new Font("gill sans",Font.BOLD,25));

b2.setFont(new Font("gill sans",Font.BOLD,25));

b3.setFont(new Font("gill sans",Font.BOLD,25));

t1.setFont(new Font("gill sans",Font.BOLD,25));

t2.setFont(new Font("gill sans",Font.BOLD,25));

f2.add(l1);

f2.add(l2);

f2.add(t1);

f2.add(l3);

f2.add(t2);

f2.add(b1);

f2.add(b2);

f2.add(b3);

b1.addActionListener(this);

b2.addActionListener(this);

b3.addActionListener(this);}

catch(Exception j){}}

public void actionPerformed(ActionEvent ae)

{s1=t1.getText();

s2=t2.getText();

if(ae.getSource().equals(b1))

{f2.setVisible(false);

f1.setVisible(true);}

if(ae.getSource().equals(b3))

{f2.setVisible(false);}

if(ae.getSource().equals(b2))

{try

{DriverManager.registerDriver(new oracle.jdbc.OracleDriver());

Connection c1=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","system","password");

PreparedStatement s=c1.prepareStatement("select s\_id,pws from addstudents ");

ResultSet rs=s.executeQuery();

String n2,n4;

int op=0;

while(rs.next())

{n2=rs.getString("s\_id");

n4=rs.getString("pws");

if(s1.equals(n2)&&s2.equals(n4))

op=1;}

if(op==1)

{JOptionPane.showMessageDialog(null,"Login Successfull");

new view2(f2);}

else

JOptionPane.showMessageDialog(null,"Invalid login");}

catch(Exception klo){}}}}

class view2 implements ActionListener

{JFrame po,v1;

JLabel l1,l2,l3,l4;

JButton b1,b2,b3,b4,b5,b6,b7,b8;

JTextField t1,t2,t3;

view2(JFrame f2)

{po=f2;

v1=new JFrame("viewing tt");

v1.setLayout(new FlowLayout());

v1.setVisible(true);

v1.setSize(1500,1500);

b4=new JButton("Back");

b5=new JButton("Cancel");

b6=new JButton("View");

l2=new JLabel("Enter room num");

t1=new JTextField(10);

v1.add(l2);

v1.add(t1);

v1.add(b6);

v1.add(b4);

v1.add(b5);

l2.setFont(new Font("gill sans",Font.BOLD,50));

t1.setFont(new Font("gill sans",Font.BOLD,20));

b4.setFont(new Font("gill sans",Font.BOLD,25));

b5.setFont(new Font("gill sans",Font.BOLD,25));

b6.setFont(new Font("gill sans",Font.BOLD,25));

b4.addActionListener(this);

b5.addActionListener(this);

b6.addActionListener(this);}

public void actionPerformed(ActionEvent ae)

{String t11=t1.getText();

if(ae.getSource().equals(b5))

{v1.setVisible(false);}

if(ae.getSource().equals(b4))

{v1.setVisible(false);

po.setVisible(true);}

if(ae.getSource().equals(b6))

{int op=0;

if(t11.equals(""))

{JOptionPane.showMessageDialog(null,"Empty fields not allowed");}

else

{int yto=0;

String q[]=new String[42];

try

{DriverManager.registerDriver(new oracle.jdbc.OracleDriver());

Connection c=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","system","password");

PreparedStatement s=c.prepareStatement("select \* from ttgmm11");

ResultSet rs=s.executeQuery();

String s11;

while(rs.next())

{op=1;

s11=rs.getString("roomnum");

if(t11.equals(s11))

{q[yto]=rs.getString(1);

yto++;

q[yto]=rs.getString(2);

yto++;

q[yto]=rs.getString(3);

yto++;

q[yto]=rs.getString(4);

yto++;

q[yto]=rs.getString(5);

yto++;

q[yto]=rs.getString(6);

yto++;

q[yto]=rs.getString(7);

yto++;

q[yto]=rs.getString(8);

yto++;

q[yto]=rs.getString(9);

yto++;

q[yto]=rs.getString(10);

yto++;

q[yto]=rs.getString(11);

yto++;

q[yto]=rs.getString(12);

yto++;

q[yto]=rs.getString(13);

yto++;

q[yto]=rs.getString(14);

yto++;

q[yto]=rs.getString(15);

yto++;

q[yto]=rs.getString(16);

yto++;

q[yto]=rs.getString(17);

yto++;

q[yto]=rs.getString(18);

yto++;

q[yto]=rs.getString(19);

yto++;

q[yto]=rs.getString(20);

yto++;

q[yto]=rs.getString(21);

yto++;

q[yto]=rs.getString(22);

yto++;

q[yto]=rs.getString(23);

yto++;

q[yto]=rs.getString(24);

yto++;

q[yto]=rs.getString(25);

yto++;

q[yto]=rs.getString(26);

yto++;

q[yto]=rs.getString(27);

yto++;

q[yto]=rs.getString(28);

yto++;

q[yto]=rs.getString(29);

yto++;

q[yto]=rs.getString(30);

yto++;

q[yto]=rs.getString(31);

yto++;

q[yto]=rs.getString(32);

yto++;

q[yto]=rs.getString(33);

yto++;

q[yto]=rs.getString(34);

yto++;

q[yto]=rs.getString(35);

yto++;

q[yto]=rs.getString(36);

yto++;

q[yto]=rs.getString(37);

yto++;

q[yto]=rs.getString(38);

yto++;

q[yto]=rs.getString(39);

yto++;

q[yto]=rs.getString(40);

yto++;

q[yto]=rs.getString(41);

yto++;

q[yto]=rs.getString(42);}}}

catch(Exception jklo){}

int yto1=0;

String q1[]=new String[42];

try

{DriverManager.registerDriver(new oracle.jdbc.OracleDriver());

Connection c=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","system","password");

PreparedStatement s=c.prepareStatement("select \* from ids11");

ResultSet rs=s.executeQuery();

String s11;

while(rs.next())

{s11=rs.getString("roomnum");

if(t11.equals(s11))

{q1[yto1]=rs.getString(1);

yto1++;

q1[yto1]=rs.getString(2);

yto1++;

q1[yto1]=rs.getString(3);

yto1++;

q1[yto1]=rs.getString(4);

yto1++;

q1[yto1]=rs.getString(5);

yto1++;

q1[yto1]=rs.getString(6);

yto1++;

q1[yto1]=rs.getString(7);

yto1++;

q1[yto1]=rs.getString(8);

yto1++;

q1[yto1]=rs.getString(9);

yto1++;

q1[yto1]=rs.getString(10);

yto1++;

q1[yto1]=rs.getString(11);

yto1++;

q1[yto1]=rs.getString(12);

yto1++;

q1[yto1]=rs.getString(13);

yto1++;

q1[yto1]=rs.getString(14);

yto1++;

q1[yto1]=rs.getString(15);

yto1++;

q1[yto1]=rs.getString(16);

yto1++;

q1[yto1]=rs.getString(17);

yto1++;

q1[yto1]=rs.getString(18);

yto1++;

q1[yto1]=rs.getString(19);

yto1++;

q1[yto1]=rs.getString(20);

yto1++;

q1[yto1]=rs.getString(21);

yto1++;

q1[yto1]=rs.getString(22);

yto1++;

q1[yto1]=rs.getString(23);

yto1++;

q1[yto1]=rs.getString(24);

yto1++;

q1[yto1]=rs.getString(25);

yto1++;

q1[yto1]=rs.getString(26);

yto1++;

q1[yto1]=rs.getString(27);

yto1++;

q1[yto1]=rs.getString(28);

yto1++;

q1[yto1]=rs.getString(29);

yto1++;

q1[yto1]=rs.getString(30);

yto1++;

q1[yto1]=rs.getString(31);

yto1++;

q1[yto1]=rs.getString(32);

yto1++;

q1[yto1]=rs.getString(33);

yto1++;

q1[yto1]=rs.getString(34);

yto1++;

q1[yto1]=rs.getString(35);

yto1++;

q1[yto1]=rs.getString(36);

yto1++;

q1[yto1]=rs.getString(37);

yto1++;

q1[yto1]=rs.getString(38);

yto1++;

q1[yto1]=rs.getString(39);

yto1++;

q1[yto1]=rs.getString(40);

yto1++;

q1[yto1]=rs.getString(41);

yto1++;

q1[yto1]=rs.getString(42);}}}

catch(Exception jklo){}

if(op==1)

{new min2(v1,yto,q,yto1,q1,t11);}

else

JOptionPane.showMessageDialog(null,"No timetable created with this details");}}}}

class min2 implements ActionListener

{JFrame v11,v2;

JButton b1;

JTable j;

String we="";

min2(JFrame v1,int yto,String q[],int yto1,String q1[],String t11)

{we=t11;

v11=v1;

v2=new JFrame("table"+we);

v2.setVisible(true);

v2.setSize(1500,1500);

v2.setLayout(new FlowLayout());

b1=new JButton("Back");

String col[]={"Day","I","II","III","IV","V","VI"};

String row[][]={{q[0],q[1],q[2],q[3],q[4],q[5],q[6]},

{q1[0],q1[1],q1[2],q1[3],q1[4],q1[5],q1[6]},

{q[7],q[8],q[9],q[10],q[11],q[12],q[13]},

{q1[7],q1[8],q1[9],q1[10],q1[11],q1[12],q1[13]},

{q[14],q[15],q[16],q[17],q[18],q[19],q[20]},

{q1[14],q1[15],q1[16],q1[17],q1[18],q1[19],q1[20]},

{q[21],q[22],q[23],q[24],q[25],q[26],q[27]},

{q1[21],q1[22],q1[23],q1[24],q1[25],q1[26],q1[27]},

{q[28],q[29],q[30],q[31],q[32],q[33],q[34]},

{q1[28],q1[29],q1[30],q1[31],q1[32],q1[33],q1[34]},

{q[35],q[36],q[37],q[38],q[39],q[40],q[41]},

{q1[35],q1[36],q1[37],q1[38],q1[39],q1[40],q1[41]}};

j=new JTable(row,col);

j.setRowHeight( 0,50);

j.setRowHeight( 1,50);

j.setRowHeight( 2,50);

j.setRowHeight( 3,50);

j.setRowHeight( 4,50);

j.setRowHeight(5,50);

j.setRowHeight( 6,50);

j.setRowHeight( 7,50);

j.setRowHeight( 8,50);

j.setRowHeight( 9,50);

j.setRowHeight( 10,50);

j.setRowHeight(11,50);

j.getColumnModel().getColumn(1).setWidth(300 );

j.getColumnModel().getColumn(2).setWidth(300 );

j.getColumnModel().getColumn(3).setWidth(300 );

j.getColumnModel().getColumn(4).setWidth(300 );

j.getColumnModel().getColumn(5).setWidth(300 );

j.getColumnModel().getColumn(0).setWidth(300 );

v2.add(j);

v2.add(b1);

b1.addActionListener(this);}

public void actionPerformed(ActionEvent ae)

{if(ae.getSource().equals(b1))

{v2.setVisible(false);

v11.setVisible(true);}}}

class cp1 implements ActionListener

{JFrame d1,d2,d3,d4;

JTextField t1;

JButton b1,b2;

JLabel l1;

cp1(JFrame f11,JFrame f22,JFrame f3)

{d1=f11;d2=f22;d3=f3;

d4=new JFrame("cp1");

d4.setVisible(true);

d4.setSize(1500,1500);

d4.setLayout(new FlowLayout());

l1=new JLabel("Enter staff id");

t1=new JTextField(10);

b1=new JButton("Reset");

b2=new JButton("Cancel");

l1.setFont(new Font("gill sans",Font.BOLD,50));

t1.setFont(new Font("gill sans",Font.BOLD,20));

b1.setFont(new Font("gill sans",Font.BOLD,25));

b2.setFont(new Font("gill sans",Font.BOLD,25));

d4.add(l1);

d4.add(t1);

d4.add(b1);

d4.add(b2);

b1.addActionListener(this);

b2.addActionListener(this);}

public void actionPerformed(ActionEvent ae)

{String s1=t1.getText();

if(ae.getSource().equals(b2))

{d4.setVisible(false);d3.setVisible(true);}

if(ae.getSource().equals(b1))

{try

{DriverManager.registerDriver(new oracle.jdbc.OracleDriver());

Connection c1=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","system","password");

PreparedStatement s=c1.prepareStatement("select sid,pw from addstaff ");

ResultSet rs=s.executeQuery();

String n2,n4;

int op=0;

while(rs.next())

{n2=rs.getString("sid");

n4=rs.getString("pw");

if(s1.equals(n2))

{op=1;

String dog="Mits@1234";

try

{DriverManager.registerDriver(new oracle.jdbc.OracleDriver());

Connection con=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","system","password");

Statement st=con.createStatement();

st.executeUpdate("update addstaff set pw='"+dog+"' where sid='"+s1+"'");}

catch(Exception monk){}}}

if(op==1)

JOptionPane.showMessageDialog(null,"Done");

else

JOptionPane.showMessageDialog(null,"Invalid ID");}

catch(Exception klo){}}}}

class cp2 implements ActionListener

{JFrame d1,d2,d3,d4;

JTextField t1;

JButton b1,b2;

JLabel l1;

cp2(JFrame f11,JFrame f22,JFrame f3)

{d1=f11;d2=f22;d3=f3;

d4=new JFrame("cp2");

d4.setVisible(true);

d4.setSize(1500,1500);

d4.setLayout(new FlowLayout());

l1=new JLabel("Enter student id");

t1=new JTextField(10);

b1=new JButton("Reset");

b2=new JButton("Cancel");

l1.setFont(new Font("gill sans",Font.BOLD,50));

t1.setFont(new Font("gill sans",Font.BOLD,20));

b1.setFont(new Font("gill sans",Font.BOLD,25));

b2.setFont(new Font("gill sans",Font.BOLD,25));

d4.add(l1);

d4.add(t1);

d4.add(b1);

d4.add(b2);

b1.addActionListener(this);

b2.addActionListener(this);}

public void actionPerformed(ActionEvent ae)

{String s1=t1.getText();

if(ae.getSource().equals(b2))

{d4.setVisible(false);d3.setVisible(true);}

if(ae.getSource().equals(b1))

{try

{DriverManager.registerDriver(new oracle.jdbc.OracleDriver());

Connection c1=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","system","password");

PreparedStatement s=c1.prepareStatement("select s\_id,pws from addstudents ");

ResultSet rs=s.executeQuery();

String n2,n4;

int op=0;

while(rs.next())

{n2=rs.getString("s\_id");

n4=rs.getString("pws");

if(s1.equals(n2))

{op=1;

String dog="Mits@1234";

try

{DriverManager.registerDriver(new oracle.jdbc.OracleDriver());

Connection con=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","system","password");

Statement st=con.createStatement();

st.executeUpdate("update addstudents set pws='"+dog+"' where s\_id='"+s1+"'");}

catch(Exception monk){}}}

if(op==1)

JOptionPane.showMessageDialog(null,"Done");

else

JOptionPane.showMessageDialog(null,"Invalid ID");}

catch(Exception klo){}}}}