Kendriya Vidyalaya Mandsaur

2017

Library Management System

Informatics Practices Project

Submitted By- Harshit Singh(1716444)

Yash Parmar(1716458)

[Type the company address]

**Certificate**

This is to certify that the project report entitled “School Management System ” which is being submitted by Ma. Pratik Parmar of class 12“A”, and as a partial fulfillment of the laboratory work for the award of class 12 practical examination for the academic year 2014-2015 has been duly completed as a record of confide work in the informatics practices.

**The report is up to the standard both in respect of its contents and its literary presentation for being referred to the examiner.**

**Mr. Hitesh Bhabhiwal**

**[Principal & PGT C.S.]**

**Acknowledgment**

I'm a student of class 12th of Informatics Practices know that our institute is one of the best institute of the city. The environment is such a vivacious which provoked me to develop this project.

        I am really indebted to our school principal “Mr.H.K.Bhabhiwal”. I am also thankful for the guidance & encouragement that helped me to work. Before attempting to design a project it is important to have clear in-depth, understanding of the dynamic character of the program doing a project in Java (net beans) was not an easy task. In between was a setback in my project design due to various reasons. But there was always an upsurge of interest and enthusiasm that helped me in laying strategies for tackling the problem raised.

                 Finally my acknowledgment will be incomplete if I do not mention all those who have generally given helpful comment, idea & support.

Pratik Parmar

XII SCI

K.V.Mandsaur

Contents

1. Introduction of Project
2. Aim of Project
3. Hardware and Software Requirement
4. Database Design
5. Frames and Source Code
6. System Installation
7. References

**Introduction**

The solo purpose of the software is to meet the intended users’ need. As it reflect from the statement “ SOFTWARE DEVELOPMENT IS THE TRANSLATION OF A USER. NEED”. If all the requirements of user are served by computerized solution, the user is satisfied enough .Thus this software has been designed effectively so as to meet its objective of user satisfaction Lawrence Peter has stated “SOFTWARE DESIGN IS A SYSTEM NOT A SECRET’. The designing equally applies to database design and development. The most important things keep in mind throughout the design process, is integrity of the data, so that in future the data should be retrieved. To avoid these bugs, a systematic approach is followed. This involves lots of steps and phases.

**Aim of Project**

This project titled “School Management System” has been developed with an aim to help in managing school’s record, handle and retrieve information about the students in an organized way.

**Minimum Hardware Requirment**

1. Hard Disk 20 GB
2. 256 MB RAM
3. P3 Processor

**Software Requirment**

1. Netbeans IDE 6.5 or Higher
2. MySQL 5.0 or Higher

**Minimum Hardware Requirment**

1. Windows XP
2. Windows 7
3. Windows 8

**Database Design**

**mysql> CREATE TABLE LOGIN**

**-> ( USERNAME VARCHAR(20) NOT NULL,**

**-> PASSWORD VARCHAR(10) NOT NULL,**

**-> PRIMARY KEY(USERNAME,PASSWORD));**

|  |  |  |  |
| --- | --- | --- | --- |
| Field | Type | Null | Key |
| Username | Varchar(20) | No | Pri |
| Password | Varchar(10) | No | Pri |

mysql> CREATE TABLE STUDENT

-> ( ADMNO INT(11) PRIMARY KEY,

-> NAME VARCHAR(30) NOT NULL,

-> CLASS INT(11) NOT NULL,

-> GENDER CHAR NOT NULL,

-> DOB DATE NOT NULL,

-> CATEGORY VARCHAR(15) NOT NULL);

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Field | Type | Null | Key | Default |
| Admno | Int(11) | No | Pri | Null |
| Name | Varchar(30) | No |  | Null |
| Class | Int(11) | No |  | Null |
| Gender | Char(1) | No |  | Null |
| Dob | Date | No |  | Null |
| Category | Varchar(15) | No |  | Null |

mysql> CREATE TABLE STUDENT1

-> ( ADMNO INT(11) PRIMARY KEY,

-> BLOOD VARCHAR(4) NOT NULL,

-> NATIONALITY VARCHAR(15) NOT NULL DEFAULT “INDIA”,

-> LAST\_RESULT INT ,

-> FOREIGN KEY(ADMNO) REFERENCES STUDENT(ADMNO));

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Field | Type | Null | Key | Default |
| Admno | Int(11) | No | Mul | Null |
| Blood | Varchar(4) | No |  | Null |
| Nationality | Varchar(15) | No |  | India |
| Last\_result | Int(11) | Yes |  | Null |

mysql> CREATE TABLE STUDENT2

-> ( ADMNO INT(11),

-> FATHER VARCHAR(30) NOT NULL,

-> F\_OCCUPATION VARCHAR(30) NOT NULL,

-> MOTHER VARCHAR(30) NOT NULL,

-> M\_OCCUPATION VARCHAR(30) NOT NULL,

-> BASIC\_PAY INT(11) NOT NULL,

-> ADDRESS VARCHAR(50) NOT NULL,

-> MOBILE VARCHAR(15) NOT NULL,

-> FOREIGN KEY(ADMNO) REFERENCES STUDENT(ADMNO));

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Field | Type | Null | Key | Default |
| Admno | Int(11) | No | Mul | Null |
| Father | Varchar(30) | No |  | Null |
| F\_occupation | Varchar(30) | No |  | Null |
| Mother | Varchar(30) | No |  | Null |
| M\_occupation | Varchar(30) | No |  | Null |
| Basic\_pay | Int(11) | Yes |  | Null |
| Address | Varchar(50) | No |  | Null |
| Mobile | Varchar(15) | Yes |  | Null |

mysql> CREATE TABLE STUDENT\_DATA

-> ( ADMNO INT(11) PRIMARY KEY,

-> NAME VARCHAR(25) NOT NULL,

-> CLASS INT NOT NULL,

-> GENDER CHAR NOT NULL,

-> DOB DATE NOT NULL,

-> CATEGORY VARCHAR(20) NOT NULL,

-> BLOOD VARCHAR(5) NOT NULL,

-> NATIONALITY VARCHAR(15) NOT NULL,

-> LAST\_RESULT INT NOT NULL,

-> FATHER VARCHAR(30) NOT NULL,

-> F\_OCCUPATION VARCHAR(20) NOT NULL,

-> MOTHER VARCHAR(30) NOT NULL,

-> M\_OCCUPATION VARCHAR(20) NOT NULL,

-> BASIC\_PAY INT NOT NULL,

-> ADDRESS VARCHAR(50) NOT NULL,

-> MOBILE VARCHAR(15) NOT NULL,

-> TC VARCHAR(5) NOT NULL);

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Field | Type | Null | Key | Default |
| Admno | Int(11) | No | PRI | Null |
| Name | Varchar(30) | No |  | Null |
| Class | Int(11) | No |  | Null |
| Gender | Char(1) | No |  | Null |
| Dob | Date | No |  | Null |
| Category | Varchar(15) | No |  | Null |
| Blood | Varchar(4) | No |  | Null |
| Nationality | Varchar(15) | No |  | Null |
| Last\_result | Int(11) | No |  | Null |
| Father | Varchar(30) | No |  | Null |
| F\_occupation | Varchar(20) | No |  | Null |
| Mother | Varchar(30) | No |  | Null |
| M\_occupation | Varchar(20) | No |  | Null |
| Basic\_pay | Int(11) | No |  | Null |
| Address | Varchar(50) | No |  | Null |
| Mobile | Varchar(15) | No |  | Null |
| Tc | Varchar(5) | No |  | Null |

**Frames and Source Code**

class 1:

package school;

import java.sql.Connection;

import java.sql.DriverManager;

import java.sql.ResultSet;

import java.sql.Statement;

import javax.swing.JOptionPane;

**public class School**

{

public static int tc=0;

public static String url = "jdbc:mysql://localhost:3306/school";

public static String pswd = "1234";

public static String username = "root";

static int engut=0,chemut=0,phyut=0,iput=0,hinut=0;

static int mthsut=0,biout=0,accut=0, ecout=0,bsut=0;

static int enghy=0,chemhy=0,phyhy=0,iphy=0,hinhy=0,mthshy=0,biohy=0;

static int acchy=0,ecohy=0,bshy=0;

static int engse=0,chemse=0,physe=0,ipse=0,hinse=0,mthsse=0;

static int biose=0,accse=0,ecose=0,bsse=0;

static int hyprac\_chem=0,hyprac\_phy=0,hyprac\_ip=0,hyprac\_bio=0;

static int seprac\_chem=0,seprac\_phy=0,seprac\_ip=0,seprac\_bio=0;

static int A\_eng=0,A\_chem=0,A\_phy=0,A\_ip\_hin=0,A\_bio\_mths=0,A\_acc=0,A\_eco=0,A\_bs=0,A\_ip=0,A\_hin=0,A\_bio=0,A\_mths=0;

static int P\_eng=0,P\_chem=0,P\_phy=0,P\_ip\_hin=0,P\_bio\_mths=0,P\_acc=0,P\_eco=0,P\_bs=0,P\_ip=0,P\_hin=0,P\_bio=0,P\_mths=0;

static long Teng=0,Tchem=0,Tphy=0,Tip=0,Thin=0,Tmths=0,Tbio=0,Tacc=0,Teco=0,Tbs=0;

static String name,roll,cls,session,admno;

static long Rengut=0,Rchemut=0,Rphyut=0,Riput=0,Rhinut=0,Rmthsut=0,Rbiout=0,Raccut=0,Recout=0,Rbsut=0;

static long Renghy=0,Rchemhy=0,Rphyhy=0,Riphy=0,Rhinhy=0,Rmthshy=0,Rbiohy=0,Racchy=0,Recohy=0,Rbshy=0;

static long Rengse=0,Rchemse=0,Rphyse=0,Ripse=0,Rhinse=0,Rmthsse=0,Rbiose=0,Raccse=0,Recose=0,Rbsse=0;

static long Rhyprac\_chem=0,Rhyprac\_phy=0,Rhyprac\_ip=0,Rhyprac\_bio=0;

static long Rseprac\_chem=0,Rseprac\_phy=0,Rseprac\_ip=0,Rseprac\_bio=0;

static long RA\_eng=0,RA\_chem=0,RA\_phy=0,RA\_ip\_hin=0,RA\_bio\_mths=0,RA\_acc=0,RA\_eco=0,RA\_bs=0,RA\_ip=0,RA\_hin=0,RA\_bio=0,RA\_mths=0;

static long RP\_eng=0,RP\_chem=0,RP\_phy=0,RP\_ip\_hin=0,RP\_bio\_mths=0,RP\_acc=0,RP\_eco=0,RP\_bs=0,RP\_ip=0,RP\_hin=0,RP\_bio=0,RP\_mths=0;

static String strm;

static String sub1,sub2;

static int clsA;

static void getClass(int a)

{

clsA=a;

}

static int setClass()

{

return clsA;

}

static int rollno;

static void getRollno(int a)

{

rollno=a;

}

static int setRollno()

{

return rollno;

}

}

class 2:

**public class Grade extends School** {

static char grade(long a)

{ char c='G';

if(a>=90)

{

c='A';

}

else

{

if(a>=80)

{

c='B';

}

else

{

if(a>=70)

{

c='C';

}

else

{

if(a>=60)

{

c='D';

}

else

{

if(a>=45)

{

c='E';

}

else

{

if(a>=33)

{

c='F';

}

else

{

if(a==0)

{

c='G';

}

}

}

}

}

}

}

return c;

}

static char grade(double a)

{ char c='G';

if(a>=90)

{

c='A';

}

else

{

if(a>=80)

{

c='B';

}

else

{

if(a>=70)

{

c='C';

}

else

{

if(a>=60)

{

c='D';

}

else

{

if(a>=45)

{

c='E';

}

else

{

if(a>=33)

{

c='F';

}

else

{

if(a==0)

{

c='-';

}

}

}

}

}

}

}

return c;

}

}

class 3:

**public class SEARCH\_DATABASE**

{

public static String search\_by="admno";

public static int admno=0,name=0,cls=0,gender=0,dob=0,category=0;

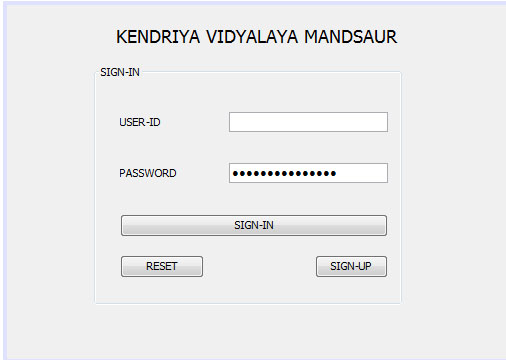
public static int blood=0,nationality=0,f\_occupation=0,f\_name=0,

basic\_pay=0,address=0,mobile=0;

public static String search=null;

}

**Frame 1:**



**private void jSignInBtnActionPerformed(java.awt.event.ActionEvent evt)** {

try{

String userid,password;

userid=jTextField1.getText().trim(); password=jPasswordField1.getText().trim();

Class.forName("java.sql.DriverManager");

Connection conn=(Connection)DriverManager.getConnection(School.url,School.username,School.pswd);

Statement st=(Statement)conn.createStatement();

ResultSet rs;

String sql="Select \* from login";

JOptionPane.showMessageDialog(rootPane, sql);

rs=st.executeQuery(sql);

int s=0;

while(rs.next())

{

if(userid.equals(rs.getString(1))&& password.equals(rs.getString(2)))

{

s=1;

JOptionPane.showMessageDialog(rootPane, "Successful Login....");

dispose();

new HOME().setVisible(true);

}

}

if(s==0)

{

JOptionPane.showMessageDialog(rootPane, "Wrong userid or password");

}

}

catch(Exception e)

{

JOptionPane.showInputDialog(e.getMessage());

}

}

**private void jSignUpBtnActionPerformed(java.awt.event.ActionEvent evt)**

{

new SIGNUP().setVisible(true);

new LoginIn().setVisible(false);

}

**private void jPasswordField1MouseClicked(java.awt.event.MouseEvent evt)**

{

jPasswordField1.setText(null);

}

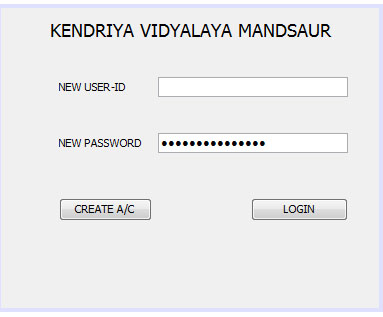
**private void jResetBtnActionPerformed(java.awt.event.ActionEvent evt)** {

jTextField1.setText(null);

jPasswordField1.setText(null);

}

**Frame 2:**



**private void jCreateACBtnActionPerformed(java.awt.event.ActionEvent evt)** {

try

{

String userid,password;

userid=jTextField1.getText().trim();

password=jPasswordField1.getText().trim();

Class.forName("java.sql.DriverManager");

Connection conn=(Connection)DriverManager.getConnection(School.url,School.username,School.pswd);

Statement st=(Statement)conn.createStatement();

String sql="insert into login values('"+userid+"','"+password+"')";

int k=st.executeUpdate(sql);

if(k>0)

{

JOptionPane.showMessageDialog(rootPane, "Record Added Successfully....");

}

else

{

JOptionPane.showMessageDialog(null, "Record Not Added");

}

st.close();

conn.close();

}

catch(Exception e)

{

JOptionPane.showMessageDialog(rootPane, e.getMessage());

}

}

**private void jLoginBtnActionPerformed(java.awt.event.ActionEvent evt)**

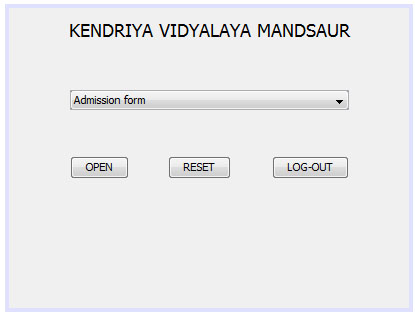
{

new LoginIn().setVisible(true);

dispose();

}

**Frame 3:**



**private void jResetBtnActionPerformed(java.awt.event.ActionEvent evt)**

{

jComboBox1.setSelectedIndex(0);

}

**private void jLogOutBtnActionPerformed(java.awt.event.ActionEvent evt)**

{

new LoginIn().setVisible(true);

dispose();

}

**private void jOpenBtnActionPerformed(java.awt.event.ActionEvent evt)** {

try

{

int index=jComboBox1.getSelectedIndex();

switch(index)

{

case 0: new Admit().setVisible(true);

break;

case 1:new Result().setVisible(true);

break;

case 5:new UPDATE().setVisible(true);

break;

case 2:new TC().setVisible(true);

break;

case 3:new RecordSearch().setVisible(true);

break;

}

dispose();

}

catch(Exception e)

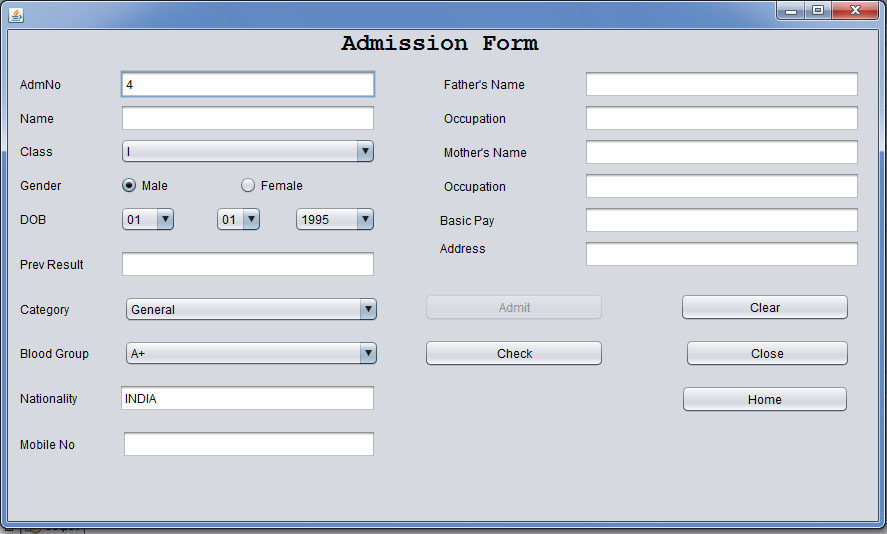
{

JOptionPane.showMessageDialog(rootPane, e.getMessage());

}

}

**Frame 3:**



private void jClearBtnActionPerformed(java.awt.event.ActionEvent evt) {

try

{ Class.forName("java.sql.DriverManager");

Connection conn=(Connection)DriverManager.getConnection(School.url,School.username,School.pswd);

Statement st=(Statement)conn.createStatement();

int admno=1;

String s="Select \* from student\_data";

ResultSet r=st.executeQuery(s);

if(r.last())

{

admno=r.getInt(1)+1;

}

jTextField1.setText(admno+"");

}

catch(Exception e)

{

JOptionPane.showMessageDialog(rootPane, e.getMessage());

}

jNameTF.setText(null);

jBasic\_PayTF.setText(null);

jAddressTF.setText(null);

jMobileTF.setText(null);

jTextField9.setText(null);

jTextField10.setText(null);

jF\_nameTF.setText(null);

jTextField12.setText(null);

jF\_OccupationTF.setText(null);

jM\_NameTF.setText(null);

jM\_Occupation.setText(null);

jNationalityTF.setText("INDIA");

jResultTF.setText(null);

jMaleRB.setSelected(true);

jClassCB.setSelectedIndex(0);

jDayCB.setSelectedIndex(0);

jMonthCB.setSelectedIndex(0);

jYearCB.setSelectedIndex(0);

jCategoryCB.setSelectedIndex(0);

jBloodCB.setSelectedIndex(0);

}

private void jMobileTFKeyTyped(java.awt.event.KeyEvent evt) {

if(!(evt.getKeyChar()>='0'&&evt.getKeyChar()<='9'))

{

evt.consume();

}

try

{

String Mobile=jMobileTF.getText();

int l=Mobile.length();

if(l>10)

{

evt.consume();

}

}

catch(Exception e)

{

JOptionPane.showMessageDialog(null,e.getMessage());

}// TODO add your handling code here:

}

private void jBasic\_PayTFKeyTyped(java.awt.event.KeyEvent evt) {

if(!(evt.getKeyChar()>='0'&&evt.getKeyChar()<='9'))

{

evt.consume();

}

try

{

String Basic\_Pay=jBasic\_PayTF.getText();

int l=Basic\_Pay.length();

if(l>11)

{

evt.consume();

}

}

catch(Exception e)

{

JOptionPane.showMessageDialog(null,e.getMessage());

}

// TODO add your handling code here:

}

private void jCloseBtnActionPerformed(java.awt.event.ActionEvent evt) {

System.exit(0);// TODO add your handling code here:

}

private void jHomeBtnActionPerformed(java.awt.event.ActionEvent evt) {

dispose();

new HOME().setVisible(true); // TODO add your handling code here:

}

private void jAdmitBtnActionPerformed(java.awt.event.ActionEvent evt) {

try

{

int cls,basic\_pay,prev\_result;

int admno=Integer.parseInt(jTextField1.getText());

String name,dob,gender,category,nation,blood,f\_name,m\_name,f\_occupation,m\_occupation,address,mob;

name=jNameTF.getText().trim().toUpperCase();

cls=jClassCB.getSelectedIndex()+1;

dob=""+jYearCB.getSelectedItem()+jMonthCB.getSelectedItem()+jDayCB.getSelectedItem();

if(jMaleRB.isSelected())

{

gender=jMaleRB.getText().toUpperCase().charAt(0)+"";

}

else

{

gender=jFemaleRB.getText().toUpperCase().charAt(0)+"";

}

category=jCategoryCB.getSelectedItem().toString().toUpperCase();

blood=jBloodCB.getSelectedItem().toString().toUpperCase();

f\_name=jF\_nameTF.getText().trim().toUpperCase();

f\_occupation=jF\_OccupationTF.getText().trim().toUpperCase();

m\_name=jM\_NameTF.getText().trim().toUpperCase();

m\_occupation=jM\_Occupation.getText().trim().toUpperCase();

address=jAddressTF.getText().trim().toUpperCase();

mob=jMobileTF.getText();

nation=jNationalityTF.getText().toUpperCase();

basic\_pay=Integer.parseInt(jBasic\_PayTF.getText().trim());

prev\_result=Integer.parseInt(jResultTF.getText().trim());

Class.forName("java.sql.DriverManager");

Connection conn=(Connection)DriverManager.getConnection(School.url,School.username,School.pswd);

Statement st=(Statement)conn.createStatement();

String sql="insert into student values("+admno+",'"+name+"',"+cls+",'"+gender+"','"+dob+"','"+category+"')";

String sql1="insert into student1 values("+admno+",'"+blood+"','"+nation+"',"+prev\_result+")";

String sql2="insert into student2 values("+admno+",'"+f\_name+"','"+f\_occupation+"','"+m\_name+"','"+m\_occupation+"',"+basic\_pay+",'"+address+"',"+mob+")";

String sql4="select \* from cls"+cls+"record";

String sql5="insert into student\_data values("+admno+",'"+name+"',"+cls+",'"+gender+"','"+dob+"','"+category+"','"+blood+"','"+nation+"',"+prev\_result+",'"+f\_name+"','"+f\_occupation+"','"+m\_name+"','"+m\_occupation+"',"+basic\_pay+",'"+address+"',"+mob+",NO)";

ResultSet r=st.executeQuery(sql4);

int rollno=0;

if(r.last())

{

rollno=r.getRow()+1;

}

String sql3="insert into cls"+cls+"record values("+rollno+","+admno+",'"+name+"','"+dob+"','"+category+"')";

st.executeUpdate(sql);

st.executeUpdate(sql1);

st.executeUpdate(sql2);

st.executeUpdate(sql3);

st.executeUpdate(sql5);

st.close();

conn.close();

JOptionPane.showMessageDialog(rootPane, "Record added successfully...");

new Admit1().setVisible(true);

dispose();

}

catch(Exception e)

{

JOptionPane.showMessageDialog(rootPane, e.getMessage());

}// TODO add your handling code here:

}

private void jResultTFKeyTyped(java.awt.event.KeyEvent evt) {

if(!((evt.getKeyChar()>='0'&& evt.getKeyChar()<='9')||evt.getKeyChar()=='.'))

{

evt.consume();

}

else

{

String res=jResultTF.getText();

int l\_res=res.length();

for(int i=0;i<l\_res;i++)

{

if((evt.getKeyChar()=='.'&& res.charAt(i)=='.')||(evt.getKeyChar()=='.'&& res.equals("")))

{

evt.consume();

}

}

}// TODO add your handling code here:

}

private void jNameTFFocusLost(java.awt.event.FocusEvent evt) {

String name=jNameTF.getText().trim();

int l=name.length();

int c=0;

for(int n=0;n<l;n++)

{

if(name.charAt(n)==' ')

{

c=100;

break;

}

}

if(c==0)

{

JOptionPane.showMessageDialog(rootPane,"Include Last Name Also");

}

// TODO add your handling code here:

}

private void jNameTFKeyTyped(java.awt.event.KeyEvent evt) {

try

{

String name=jNameTF.getText();

int l=name.length();

if(l>30)

{

evt.consume();

}

}

catch(Exception e)

{

JOptionPane.showMessageDialog(null,e.getMessage());

}// TODO add your handling code here:

}

private void jNationalityTFKeyTyped(java.awt.event.KeyEvent evt) {

try

{

String nation=jNationalityTF.getText();

int l=nation.length();

if(l>15)

{

evt.consume();

}

}

catch(Exception e)

{

JOptionPane.showMessageDialog(null,e.getMessage());

} // TODO add your handling code here:

}

private void jF\_nameTFKeyTyped(java.awt.event.KeyEvent evt) {

try

{

String F\_name=jF\_nameTF.getText();

int l=F\_name.length();

if(l>30)

{

evt.consume();

}

}

catch(Exception e)

{

JOptionPane.showMessageDialog(null,e.getMessage());

} // TODO add your handling code here:

}

private void jF\_OccupationTFKeyTyped(java.awt.event.KeyEvent evt) {

try

{

String F\_Occupation=jF\_OccupationTF.getText();

int l=F\_Occupation.length();

if(l>30)

{

evt.consume();

}

}

catch(Exception e)

{

JOptionPane.showMessageDialog(null,e.getMessage());

} // TODO add your handling code here:

}

private void jM\_NameTFKeyTyped(java.awt.event.KeyEvent evt) {

try

{

String M\_name=jM\_NameTF.getText();

int l=M\_name.length();

if(l>30)

{

evt.consume();

}

}

catch(Exception e)

{

JOptionPane.showMessageDialog(null,e.getMessage());

} // TODO add your handling code here:

}

private void jM\_OccupationKeyTyped(java.awt.event.KeyEvent evt) {

try

{

String M\_Occupation=jM\_Occupation.getText();

int l=M\_Occupation.length();

if(l>30)

{

evt.consume();

}

}

catch(Exception e)

{

JOptionPane.showMessageDialog(null,e.getMessage());

} // TODO add your handling code here:

}

private void jAddressTFKeyTyped(java.awt.event.KeyEvent evt) {

try

{

String Address=jAddressTF.getText();

int l=Address.length();

if(l>50)

{

evt.consume();

}

}

catch(Exception e)

{

JOptionPane.showMessageDialog(null,e.getMessage());

} // TODO add your handling code here:

}

private void jCheckBtnActionPerformed(java.awt.event.ActionEvent evt) {

try

{ int check=0;

if(jNameTF.getText().equals("")||jF\_nameTF.getText().equals("")||jM\_NameTF.getText().equals("")||jF\_OccupationTF.getText().equals("")||jAddressTF.getText().equals("")||jBasic\_PayTF.getText().equals("")||jM\_Occupation.getText().equals("")||jMobileTF.getText().equals("")||jNameTF.getText().equals("")||jNationalityTF.getText().equals(""))

{

check++;

}

if(check==0)

{

jAdmitBtn.setEnabled(true);

JOptionPane.showMessageDialog(rootPane, "Egligible for admission.....");

}

else

{

JOptionPane.showMessageDialog(rootPane,"All fields are mandadory!!!!!!!!!");

}

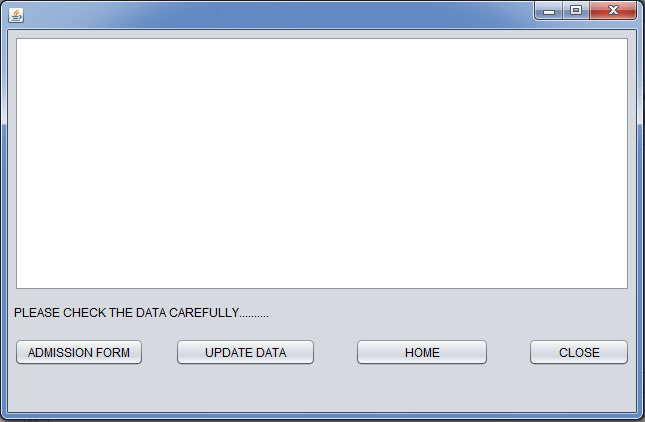
}

catch(Exception e)

{ JOptionPane.showMessageDialog(null,e.getMessage());}

}

**Frame 4:**

****

private void formWindowOpened(java.awt.event.WindowEvent evt) {

try

{

Class.forName("java.sql.DriverManager");

Connection conn=(Connection)DriverManager.getConnection(School.url,School.username,School.pswd);

Statement st=(Statement)conn.createStatement();

String sql="Select \* from student";

ResultSet rs=st.executeQuery(sql);

if(rs.last())

{

jOutputArea.append("ADMNO: "+rs.getString(1)+"\n");

jOutputArea.append("NAME: "+rs.getString(2)+"\n");

jOutputArea.append("CLASS: "+rs.getString(3)+"\n");

jOutputArea.append("GENDER: "+rs.getString(4)+"\n");

jOutputArea.append("DOB: "+rs.getString(5)+"\n");

jOutputArea.append("CATEGORY: "+rs.getString(6)+"\n");

}

rs=st.executeQuery("select \* from student1");

if(rs.last())

{

jOutputArea.append("BLODD GROUP: "+rs.getString(2)+"\n");

jOutputArea.append("NATIONALITY: "+rs.getString(3)+"\n");

jOutputArea.append("PREVIOUS RESULT: "+rs.getString(4)+"\n");

}

rs=st.executeQuery("select \* from student2");

if(rs.last())

{

jOutputArea.append("FATHER'S NAME: "+rs.getString(2)+"\n");

jOutputArea.append("MOTHER'S NAME: "+rs.getString(4)+"\n");

jOutputArea.append("FATHER'S OCCUPATION: "+rs.getString(3)+"\n");

jOutputArea.append("MOTHER'S OCCUPATION: "+rs.getString(5)+"\n");

jOutputArea.append("BASIC PAY: "+rs.getString(6)+"\n");

jOutputArea.append("ADDRESS: "+rs.getString(7)+"\n");

jOutputArea.append("MOBILE: "+rs.getString(8)+"\n");

}

}

catch(Exception p)

{

JOptionPane.showMessageDialog(rootPane, p.getMessage());

}// TODO add your handling code here:

}

private void jCloseBtnActionPerformed(java.awt.event.ActionEvent evt) {

System.exit(0);// TODO add your handling code here:

}

private void jHomeBtnActionPerformed(java.awt.event.ActionEvent evt) {

new HOME().setVisible(true);

dispose(); // TODO add your handling code here:

}

private void jUpdateDataBtnActionPerformed(java.awt.event.ActionEvent evt) {

new $UPDATE().setVisible(true);

dispose();// TODO add your handling code here:

}

private void jAdmFormBtnActionPerformed(java.awt.event.ActionEvent evt) {

new Admit().setVisible(true);

dispose();// TODO add your handling code here:

}

**Frame 5:**



private void jMobileTFKeyTyped(java.awt.event.KeyEvent evt) {

if(!(evt.getKeyChar()>='0'&&evt.getKeyChar()<='9'))

{

evt.consume();

}// TODO add your handling code here:

}

private void jBasic\_PayTFKeyTyped(java.awt.event.KeyEvent evt) {

if(!(evt.getKeyChar()>='0'&&evt.getKeyChar()<='9'))

{

evt.consume();

}

// TODO add your handling code here:

}

private void formWindowOpened(java.awt.event.WindowEvent evt) {

try {

String adm=JOptionPane.showInputDialog("Admission No......");

int admno=Integer.parseInt(adm);

Class.forName("java.sql.DriverManager");

Connection conn=(Connection)DriverManager.getConnection(School.url,School.username,School.pswd);

Statement st=(Statement)conn.createStatement();

ResultSet rs=st.executeQuery("select \* from student where admno="+admno+"");

if(rs.next())

{

jTextField1.setText(rs.getString(1));

jNameTF.setText(rs.getString(2));

jClassCB.setSelectedIndex(rs.getInt(3)-1);

if(rs.getString(4).equals("M"))

{

jMaleRB.setSelected(true);

}

else

{

jFemaleRB.setSelected(true);

}

jYearCB.setSelectedItem(rs.getString(5).substring(0,4));

jMonthCB.setSelectedItem(rs.getString(5).substring(5,7));

jDayCB.setSelectedItem(rs.getString(5).substring(8,10));

jCategoryCB.setSelectedItem(rs.getString(6));

}

rs=st.executeQuery("select \* from student1 where admno="+admno+"");

if(rs.next())

{

jBloodCB.setSelectedItem(rs.getString(2));

jNationalityTF.setText(rs.getString(3));

jResultTF.setText(rs.getString(4));

}

rs=st.executeQuery("select \* from student2 where admno="+admno+"");

if(rs.last())

{

jF\_nameTF.setText(rs.getString(2));

jF\_OccupationTF.setText(rs.getString(3));

jM\_NameTF.setText(rs.getString(4));

jM\_Occupation.setText(rs.getString(5));

jBasic\_PayTF.setText((rs.getString(6)));

jAddressTF.setText(rs.getString(7));

jMobileTF.setText(rs.getString(8));

}

st.close();

conn.close();

}

catch(Exception e)

{;}// TODO add your handling code here:

// TODO add your handling code here:

}

private void jUpdateBtnActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

try

{

int cls,basic\_pay,prev\_result;

int admno=Integer.parseInt(jTextField1.getText());

String name,dob,gender,category,nation,blood,f\_name,m\_name,f\_occupation,m\_occupation,address,mob;

name=jNameTF.getText().trim().toUpperCase();

cls=jClassCB.getSelectedIndex()+1;

dob=""+jYearCB.getSelectedItem()+jMonthCB.getSelectedItem()+jDayCB.getSelectedItem();

if(jMaleRB.isSelected())

{

gender=jMaleRB.getText().toUpperCase().charAt(0)+"";

}

else

{

gender=jFemaleRB.getText().toUpperCase().charAt(0)+"";

}

category=jCategoryCB.getSelectedItem().toString().toUpperCase();

blood=jBloodCB.getSelectedItem().toString().toUpperCase();

f\_name=jF\_nameTF.getText().trim().toUpperCase();

f\_occupation=jF\_OccupationTF.getText().trim().toUpperCase();

m\_name=jM\_NameTF.getText().trim().toUpperCase();

m\_occupation=jM\_Occupation.getText().trim().toUpperCase();

address=jAddressTF.getText().trim().toUpperCase();

mob=jMobileTF.getText();

nation=jNationalityTF.getText().toUpperCase();

basic\_pay=Integer.parseInt(jBasic\_PayTF.getText().trim());

prev\_result=Integer.parseInt(jResultTF.getText().trim());

Class.forName("java.sql.DriverManager");

Connection conn=(Connection)DriverManager.getConnection(School.url,School.username,School.pswd);

Statement st=(Statement)conn.createStatement();

String sql="update student set NAME='"+name+"',CLASS="+cls+",GENDER='"+gender+"',DOB='"+dob+"',CATEGORY='"+category+"' where admno="+admno+"";

int k=st.executeUpdate(sql);

if(k>0)

{

JOptionPane.showMessageDialog(rootPane, "Record Updated successfully...");

}

else

{

JOptionPane.showMessageDialog(rootPane, "Record Updated unsuccessfully...");

}

String sql1="update student1 set BLOOD='"+blood+"',NATIONALITY='"+nation+"',LAST\_RESULT="+prev\_result+" where admno="+admno+"";

int m=st.executeUpdate(sql1);

String sql2="update student2 set father='"+f\_name+"',f\_occupation='"+f\_occupation+"',mother='"+m\_name+"',m\_occupation='"+m\_occupation+"',basic\_pay="+basic\_pay+",address='"+address+"',mobile='"+mob+"' where admno="+admno+"";

int n=st.executeUpdate(sql2);

}

catch(Exception e)

{

JOptionPane.showMessageDialog(rootPane, e.getMessage());

}

}

private void jCloseBtnActionPerformed(java.awt.event.ActionEvent evt) {

System.exit(0); // TODO add your handling code here:

}

private void jAdmFormActionPerformed(java.awt.event.ActionEvent evt) {

new Admit().setVisible(true);

dispose();

// TODO add your handling code here:

}

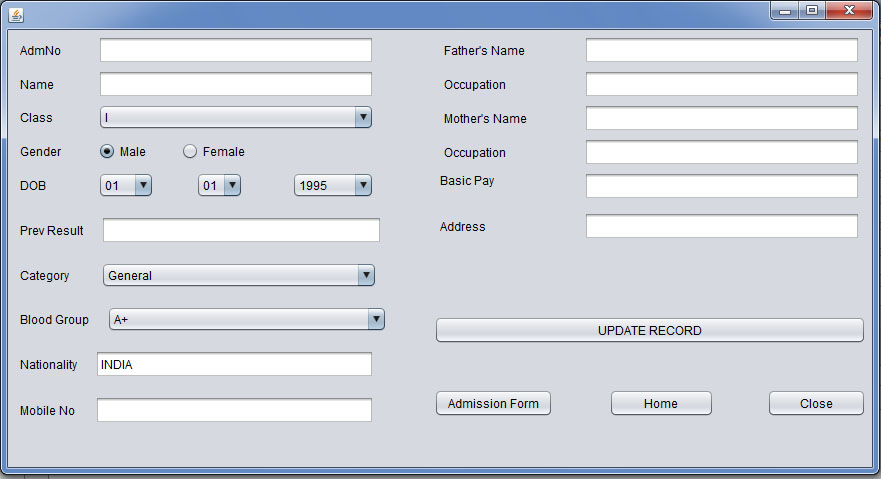
private void jHomeBtnActionPerformed(java.awt.event.ActionEvent evt) {

new HOME().setVisible(true);

dispose();// TODO add your handling code here:

}

**Frame 6:**



private void jBasic\_PayTFKeyTyped(java.awt.event.KeyEvent evt) {

if(!(evt.getKeyChar()>='0'&&evt.getKeyChar()<='9'))

{

evt.consume();

}

// TODO add your handling code here:

}

private void jMobileTFKeyTyped(java.awt.event.KeyEvent evt) {

if(!(evt.getKeyChar()>='0'&&evt.getKeyChar()<='9'))

{

evt.consume();

}// TODO add your handling code here:

}

private void formWindowOpened(java.awt.event.WindowEvent evt) {

try {

Class.forName("java.sql.DriverManager");

Connection conn=(Connection)DriverManager.getConnection(School.url,School.username,School.pswd);

Statement st=(Statement)conn.createStatement();

ResultSet rs=st.executeQuery("select \* from student");

if(rs.last())

{

jTextField1.setText(rs.getString(1));

jNameTF.setText(rs.getString(2));

jClassCB.setSelectedIndex(rs.getInt(3)-1);

if(rs.getString(4).equals("M"))

{

jMaleRB.setSelected(true);

}

else

{

jFemaleRB.setSelected(true);

}

jYearCB.setSelectedItem(rs.getString(5).substring(0,4));

jMonthCB.setSelectedItem(rs.getString(5).substring(5,7));

jDayCB.setSelectedItem(rs.getString(5).substring(8,10));

jCategoryCB.setSelectedItem(rs.getString(6));

}

rs=st.executeQuery("select \* from student1");

if(rs.last())

{

jBloodCB.setSelectedItem(rs.getString(2));

jNationalityTF.setText(rs.getString(3));

jResultTF.setText(rs.getString(4));

}

rs=st.executeQuery("select \* from student2");

if(rs.last())

{

jF\_nameTF.setText(rs.getString(2));

jF\_OccupationTF.setText(rs.getString(3));

jM\_NameTF.setText(rs.getString(4));

jM\_Occupation.setText(rs.getString(5));

jBasic\_PayTF.setText((rs.getString(6)));

jAddressTF.setText(rs.getString(7));

jMobileTF.setText(rs.getString(8));

}

st.close();

conn.close();

}

catch(Exception e)

{;}// TODO add your handling code here:

}

private void jUpdateRecBtnActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

try

{

int cls,basic\_pay,prev\_result;

int admno=Integer.parseInt(jTextField1.getText());

String name,dob,gender,category,nation,blood,f\_name,m\_name,f\_occupation,m\_occupation,address,mob;

name=jNameTF.getText().trim().toUpperCase();

cls=jClassCB.getSelectedIndex()+1;

dob=""+jYearCB.getSelectedItem()+jMonthCB.getSelectedItem()+jDayCB.getSelectedItem();

if(jMaleRB.isSelected())

{

gender=jMaleRB.getText().toUpperCase().charAt(0)+"";

}

else

{

gender=jFemaleRB.getText().toUpperCase().charAt(0)+"";

}

category=jCategoryCB.getSelectedItem().toString().toUpperCase();

blood=jBloodCB.getSelectedItem().toString().toUpperCase();

f\_name=jF\_nameTF.getText().trim().toUpperCase();

f\_occupation=jF\_OccupationTF.getText().trim().toUpperCase();

m\_name=jM\_NameTF.getText().trim().toUpperCase();

m\_occupation=jM\_Occupation.getText().trim().toUpperCase();

address=jAddressTF.getText().trim().toUpperCase();

mob=jMobileTF.getText();

nation=jNationalityTF.getText().toUpperCase();

basic\_pay=Integer.parseInt(jBasic\_PayTF.getText().trim());

prev\_result=Integer.parseInt(jResultTF.getText().trim());

Class.forName("java.sql.DriverManager");

Connection conn=(Connection)DriverManager.getConnection(School.url,School.username,School.pswd);

Statement st=(Statement)conn.createStatement();

String sql="update student set NAME='"+name+"',CLASS="+cls+",GENDER='"+gender+"',DOB='"+dob+"',CATEGORY='"+category+"' where admno="+admno+"";

int k=st.executeUpdate(sql);

if(k>0)

{

JOptionPane.showMessageDialog(rootPane, "Record Updated successfully...");

}

else

{

JOptionPane.showMessageDialog(rootPane, "Record Updated unsuccessfully...");

}

String sql1="update student1 set BLOOD='"+blood+"',NATIONALITY='"+nation+"',LAST\_RESULT="+prev\_result+" where admno="+admno+"";

int m=st.executeUpdate(sql1);

String sql2="update student2 set father='"+f\_name+"',f\_occupation='"+f\_occupation+"',mother='"+m\_name+"',m\_occupation='"+m\_occupation+"',basic\_pay="+basic\_pay+",address='"+address+"',mobile='"+mob+"' where admno="+admno+"";

int n=st.executeUpdate(sql2);

}

catch(Exception e)

{

JOptionPane.showMessageDialog(rootPane, e.getMessage());

}

}

private void jAdmFormActionPerformed(java.awt.event.ActionEvent evt) {

new Admit().setVisible(true);

dispose();// TODO add your handling code here:

}

private void jHomeBtnActionPerformed(java.awt.event.ActionEvent evt) {

new HOME().setVisible(true);

dispose(); // TODO add your handling code here:

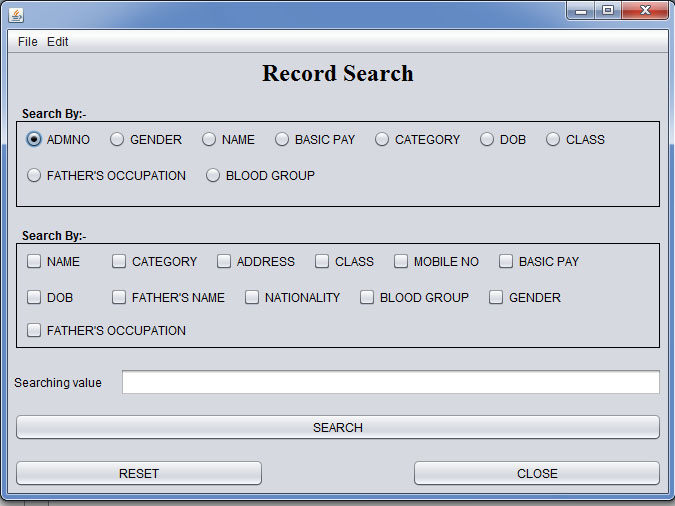
}

private void jCloseBtnActionPerformed(java.awt.event.ActionEvent evt) {

System.exit(0); // TODO add your handling code here:

}

**Frame 7:**



private void jBloodCBActionPerformed(java.awt.event.ActionEvent evt) {

SEARCH\_DATABASE.blood=1; // TODO add your handling code here:

}

private void jAdmRBActionPerformed(java.awt.event.ActionEvent evt) {

SEARCH\_DATABASE.search\_by="admno";

// TODO add your handling code here:

}

private void jGenderRBActionPerformed(java.awt.event.ActionEvent evt) {

SEARCH\_DATABASE.search\_by="gender"; // TODO add your handling code here:

}

private void jNameRBActionPerformed(java.awt.event.ActionEvent evt) {

SEARCH\_DATABASE.search\_by="name"; // TODO add your handling code here:

}

private void jBasicPayRBActionPerformed(java.awt.event.ActionEvent evt) {

SEARCH\_DATABASE.search\_by="basic\_pay"; // TODO add your handling code here:

}

private void jCategoryRBActionPerformed(java.awt.event.ActionEvent evt) {

SEARCH\_DATABASE.search\_by="category"; // TODO add your handling code here:

}

private void jDOBRBActionPerformed(java.awt.event.ActionEvent evt) {

SEARCH\_DATABASE.search\_by="dob"; // TODO add your handling code here:

}

private void jClassRBActionPerformed(java.awt.event.ActionEvent evt) {

SEARCH\_DATABASE.search\_by="class"; // TODO add your handling code here:

}

private void jFatherOccupRBActionPerformed(java.awt.event.ActionEvent evt) {

SEARCH\_DATABASE.search\_by="f\_occupation"; // TODO add your handling code here:

}

private void jBloodRBActionPerformed(java.awt.event.ActionEvent evt) {

SEARCH\_DATABASE.search\_by="blood"; // TODO add your handling code here:

}

private void jSearchBtnActionPerformed(java.awt.event.ActionEvent evt) {

try

{

SEARCH\_DATABASE.search=jSearchTF.getText().trim();

new $earch().setVisible(true);

dispose();

}

catch(Exception e)

{} // TODO add your handling code here:

}

private void jNameCBActionPerformed(java.awt.event.ActionEvent evt) {

SEARCH\_DATABASE.name=1; // TODO add your handling code here:

}

private void jCategoryCBActionPerformed(java.awt.event.ActionEvent evt) {

SEARCH\_DATABASE.category=1; // TODO add your handling code here:

}

private void jAddressCBActionPerformed(java.awt.event.ActionEvent evt) {

SEARCH\_DATABASE.address=1; // TODO add your handling code here:

}

private void jClassCBActionPerformed(java.awt.event.ActionEvent evt) {

SEARCH\_DATABASE.cls=1; // TODO add your handling code here:

}

private void jMobCBActionPerformed(java.awt.event.ActionEvent evt) {

SEARCH\_DATABASE.mobile=1; // TODO add your handling code here:

}

private void jBasicPayCBActionPerformed(java.awt.event.ActionEvent evt) {

SEARCH\_DATABASE.basic\_pay=1; // TODO add your handling code here:

}

private void jDOBCBActionPerformed(java.awt.event.ActionEvent evt) {

SEARCH\_DATABASE.dob=1; // TODO add your handling code here:

}

private void jFatherCBActionPerformed(java.awt.event.ActionEvent evt) {

SEARCH\_DATABASE.f\_name=1; // TODO add your handling code here:

}

private void jNAtionCBActionPerformed(java.awt.event.ActionEvent evt) {

SEARCH\_DATABASE.nationality=1; // TODO add your handling code here:

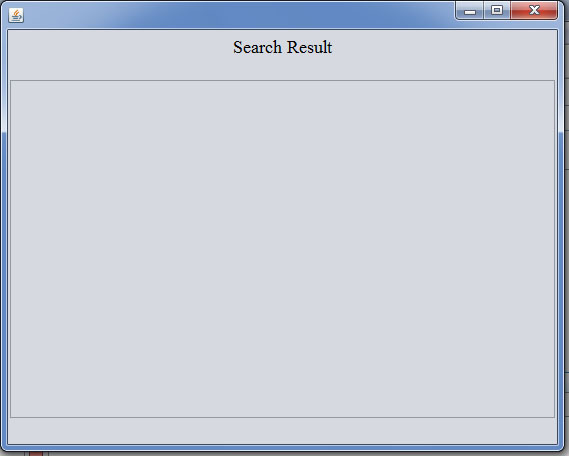
}

private void jGenderCBActionPerformed(java.awt.event.ActionEvent evt) {

SEARCH\_DATABASE.gender=1; // TODO add your handling code here:

}

**Frame 8:**



public class $earch extends javax.swing.JFrame {

int noc=0;

/\*\*

\* Creates new form $earch

\*/

public $earch() {

initComponents();

}

DefaultTableModel dtm;

String rw="admno,";

void checkpoint(int i,String s)

{

if(i==1)

{

dtm.addColumn(s);

rw+=s+",";

noc++;

}

}

private void formWindowOpened(java.awt.event.WindowEvent evt) {

// TODO add your handling code here:

try

{

dtm=(DefaultTableModel)jTable1.getModel();

dtm.addColumn("Admno");

checkpoint(SEARCH\_DATABASE.name,"NAME");

checkpoint(SEARCH\_DATABASE.cls,"CLASS");

checkpoint(SEARCH\_DATABASE.gender,"GENDER");

checkpoint(SEARCH\_DATABASE.dob,"DOB");

checkpoint(SEARCH\_DATABASE.category,"CATEGORY");

checkpoint(SEARCH\_DATABASE.blood,"BLOOD");

checkpoint(SEARCH\_DATABASE.address,"ADDRESS");

checkpoint(SEARCH\_DATABASE.f\_name,"FATHER");

checkpoint(SEARCH\_DATABASE.f\_occupation,"F\_OCCUPATION");

checkpoint(SEARCH\_DATABASE.mobile,"MOBILE");

checkpoint(SEARCH\_DATABASE.nationality,"NATIONALITY");

int l=rw.length();

String raw="";

for(int n=0;n<(l-1);n++)

{

raw=raw+rw.charAt(n);

}

Class.forName("java.sql.DriverManager");

Connection conn=(Connection)DriverManager.getConnection(School.url,School.username,School.pswd);

Statement st=(Statement)conn.createStatement();

String s="Select "+raw+" from student\_data where "+SEARCH\_DATABASE.search\_by+"= "+SEARCH\_DATABASE.search+"";

JOptionPane.showMessageDialog(rootPane, s);

ResultSet r=st.executeQuery(s);

while(r.next())

{

dtm.setRowCount(dtm.getRowCount()+1);

for(int i=1;i<=noc+1;i++)

{

dtm.setValueAt(r.getString(i), dtm.getRowCount()-1, i-1);

}

}

}

catch(Exception e)

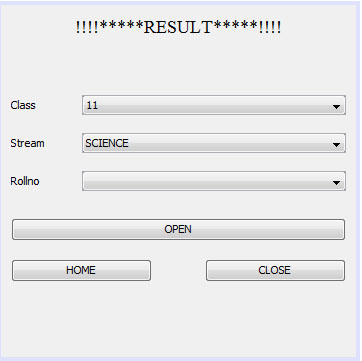
{

JOptionPane.showMessageDialog(rootPane, e.getMessage());

}

}

**Frame 9:**

****

private void jHomeBtnActionPerformed(java.awt.event.ActionEvent evt) {

new HOME().setVisible(true);

dispose();// TODO add your handling code here:

}

private void jOpenBtnActionPerformed(java.awt.event.ActionEvent evt) {

try

{

int cls=11;

String strm=jComboBox3.getSelectedItem().toString();

if(jComboBox3.getSelectedIndex()==0)

{

new RESULT1().setVisible(true);

dispose();

}

else

{

new resultcomm().setVisible(true);

dispose();

}

}

catch(Exception e)

{

JOptionPane.showMessageDialog(rootPane, e.getMessage());

}

// TODO add your handling code here:

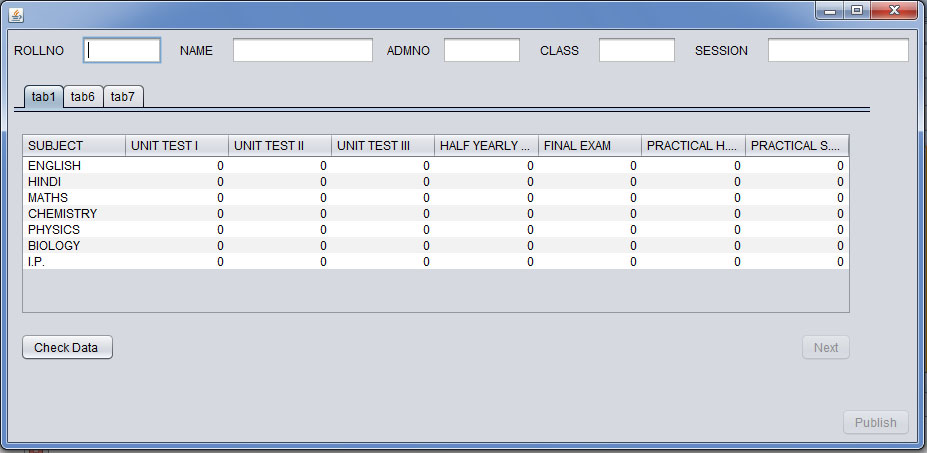
}

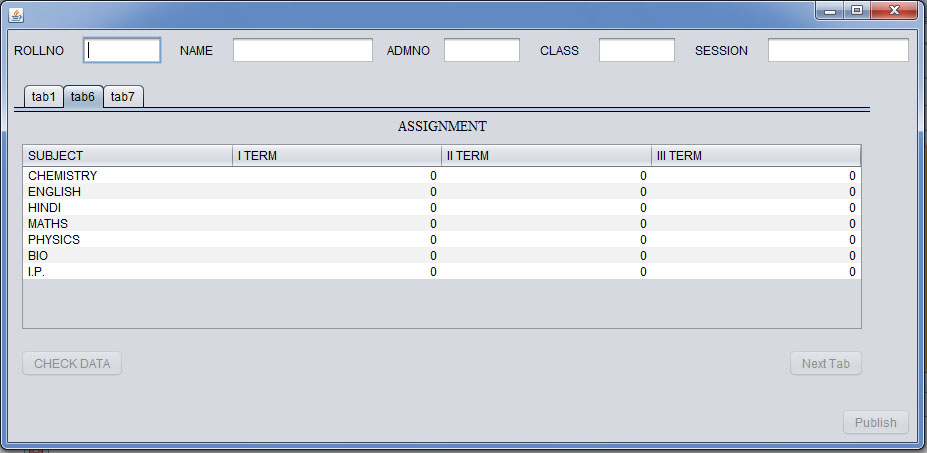
private void jCloseBtnActionPerformed(java.awt.event.ActionEvent evt) {

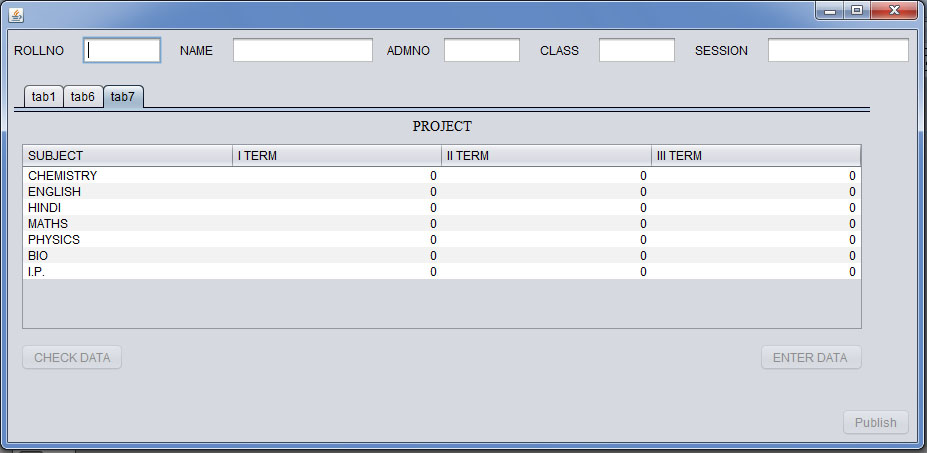
System.exit(0); // TODO add your handling code here:

}

**Frame 10:**

****

****

****

private void jNextBtn1ActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

try{

jAssignmentTable.nextFocus();

School.engut=(int)jMarksTable.getValueAt(0, 1)+(int)jMarksTable.getValueAt(0, 2)+(int)jMarksTable.getValueAt(0, 3);

School.hinut=(int)jMarksTable.getValueAt(1, 1)+(int)jMarksTable.getValueAt(1, 2)+(int)jMarksTable.getValueAt(1, 3);

School.mthsut=(int)jMarksTable.getValueAt(2, 1)+(int)jMarksTable.getValueAt(2, 2)+(int)jMarksTable.getValueAt(2, 3);

School.chemut=(int)jMarksTable.getValueAt(3, 1)+(int)jMarksTable.getValueAt(3, 2)+(int)jMarksTable.getValueAt(3, 3);

School.phyut=(int)jMarksTable.getValueAt(4, 1)+(int)jMarksTable.getValueAt(4, 2)+(int)jMarksTable.getValueAt(4, 3);

School.biout=(int)jMarksTable.getValueAt(5, 1)+(int)jMarksTable.getValueAt(5, 2)+(int)jMarksTable.getValueAt(5, 3);

School.iput=(int)jMarksTable.getValueAt(6, 1)+(int)jMarksTable.getValueAt(6, 2)+(int)jMarksTable.getValueAt(6, 3);

School.enghy=(int)jMarksTable.getValueAt(0, 4);

School.hinhy=(int)jMarksTable.getValueAt(1, 4);

School.mthshy=(int)jMarksTable.getValueAt(2,4);

School.chemhy=(int)jMarksTable.getValueAt(3, 4);

School.phyhy=(int)jMarksTable.getValueAt(4, 4);

School.biohy=(int)jMarksTable.getValueAt(5, 4);

School.iphy=(int)jMarksTable.getValueAt(6, 4);

School.engse=(int)jMarksTable.getValueAt(0, 5);

School.hinse=(int)jMarksTable.getValueAt(1, 5);

School.mthsse=(int)jMarksTable.getValueAt(2,5);

School.chemse=(int)jMarksTable.getValueAt(3, 5);

School.physe=(int)jMarksTable.getValueAt(4, 5);

School.biose=(int)jMarksTable.getValueAt(5, 5);

School.ipse=(int)jMarksTable.getValueAt(6, 5);

School.hyprac\_chem=(int)jMarksTable.getValueAt(3, 6);

School.hyprac\_phy=(int)jMarksTable.getValueAt(4, 6);

School.hyprac\_bio=(int)jMarksTable.getValueAt(5, 6);

School.hyprac\_ip=(int)jMarksTable.getValueAt(6, 6);

School.seprac\_chem=(int)jMarksTable.getValueAt(3, 7);

School.seprac\_phy=(int)jMarksTable.getValueAt(4, 7);

School.seprac\_bio=(int)jMarksTable.getValueAt(5, 7);

School.seprac\_ip=(int)jMarksTable.getValueAt(6, 7);

jCheckDataBtn2.setEnabled(true);

jTabbedPane2.setSelectedIndex(1);

}catch(Exception e)

{

JOptionPane.showMessageDialog(rootPane, e.getMessage());

}

}

private void jEnterDataBtnActionPerformed(java.awt.event.ActionEvent evt) {

try{School.P\_eng=(int)jProjectTable.getValueAt(1, 1)+(int)jProjectTable.getValueAt(1, 2)+(int)jProjectTable.getValueAt(1, 3);

School.P\_hin=(int)jProjectTable.getValueAt(2, 1)+(int)jProjectTable.getValueAt(2, 2)+(int)jProjectTable.getValueAt(2, 3);

School.P\_mths=(int)jProjectTable.getValueAt(3, 1)+(int)jProjectTable.getValueAt(3, 2)+(int)jProjectTable.getValueAt(3, 3);

School.P\_chem=(int)jProjectTable.getValueAt(0, 1)+(int)jProjectTable.getValueAt(0, 2)+(int)jProjectTable.getValueAt(0, 3);

School.P\_ip=(int)jProjectTable.getValueAt(6, 1)+(int)jProjectTable.getValueAt(6, 2)+(int)jProjectTable.getValueAt(6, 3);

School.P\_bio=(int)jProjectTable.getValueAt(5, 1)+(int)jProjectTable.getValueAt(5, 2)+(int)jProjectTable.getValueAt(5, 3);

School.P\_phy=(int)jProjectTable.getValueAt(4, 1)+(int)jProjectTable.getValueAt(4, 2)+(int)jProjectTable.getValueAt(4, 3);

jPublishBtn.setEnabled(true);

}catch(Exception e)

{}// TODO add your handling code here:

}

private void jNextBtn2ActionPerformed(java.awt.event.ActionEvent evt) {

try{

jProjectTable.nextFocus();

School.A\_eng=(int)jAssignmentTable.getValueAt(1, 1)+(int)jAssignmentTable.getValueAt(1, 2)+(int)jAssignmentTable.getValueAt(1, 3);

School.A\_hin=(int)jAssignmentTable.getValueAt(2, 1)+(int)jAssignmentTable.getValueAt(2, 2)+(int)jAssignmentTable.getValueAt(2, 3);

School.A\_mths=(int)jAssignmentTable.getValueAt(3, 1)+(int)jAssignmentTable.getValueAt(3, 2)+(int)jAssignmentTable.getValueAt(3, 3);

School.A\_ip=(int)jAssignmentTable.getValueAt(6, 1)+(int)jAssignmentTable.getValueAt(6, 2)+(int)jAssignmentTable.getValueAt(6, 3);

School.A\_bio=(int)jAssignmentTable.getValueAt(5, 1)+(int)jAssignmentTable.getValueAt(5, 2)+(int)jAssignmentTable.getValueAt(5, 3);

School.A\_chem=(int)jAssignmentTable.getValueAt(0, 1)+(int)jAssignmentTable.getValueAt(0, 2)+(int)jAssignmentTable.getValueAt(0, 3);

School.A\_phy=(int)jAssignmentTable.getValueAt(4, 1)+(int)jAssignmentTable.getValueAt(4, 2)+(int)jAssignmentTable.getValueAt(4, 3);

jCheckBtn3.setEnabled(true);

jTabbedPane2.setSelectedIndex(2);

} catch(Exception e)

{}// TODO add your handling code here:

}

private void jPublishBtnActionPerformed(java.awt.event.ActionEvent evt) {

try{

School.name=jNameTF.getText();

School.roll=jRollnoTF.getText();

School.admno=jAdmNoTF.getText();

School.cls=jCLassTF.getText();

School.session=jSessionTF.getText();

School.Rengut=Math.round(School.engut/8.0);

School.Rchemut=Math.round(School.chemut/8.0);

School.Rphyut=Math.round(School.phyut/8.0);

School.Rhinut=Math.round(School.hinut/8.0);

School.Rmthsut=Math.round(School.mthsut/8.0);

School.Rbiout=Math.round(School.biout/8.0);

School.Riput=Math.round(School.iput/8.0);

School.Renghy=Math.round(School.enghy/4.0);

School.Rchemhy=Math.round(School.chemhy/3.5);

School.Rphyhy=Math.round(School.phyhy/3.5);

School.Rhinhy=Math.round(School.hinhy/4.0);

School.Rmthshy=Math.round(School.mthshy/4.0);

School.Rbiohy=Math.round(School.biohy/3.5);

School.Riphy=Math.round(School.iphy/3.5);

School.Rengse=Math.round(School.engse/2.0);

School.Rchemse=Math.round(School.chemse/2.0);

School.Rphyse=Math.round(School.physe/2.0);

School.Rhinse=Math.round(School.hinse/2.0);

School.Rmthsse=Math.round(School.mthsse/2.0);

School.Rbiose=Math.round(School.biose/2.0);

School.Ripse=Math.round(School.ipse/2.0);

School.RA\_bio=Math.round(School.A\_bio/20.0);

School.RA\_mths=Math.round(School.A\_mths/20.0);

School.RA\_chem=Math.round(School.A\_chem/20.0);

School.RA\_phy=Math.round(School.A\_phy/20.0);

School.RA\_eng=Math.round(School.A\_eng/20.0);

School.RA\_ip=Math.round(School.A\_ip/20.0);

School.RA\_hin=Math.round(School.A\_hin/20.0);

School.RP\_bio=Math.round(School.P\_bio/20.0);

School.RP\_mths=Math.round(School.P\_mths/20.0);

School.RP\_chem=Math.round(School.P\_chem/20.0);

School.RP\_phy=Math.round(School.P\_phy/20.0);

School.RP\_eng=Math.round(School.P\_eng/20.0);

School.RP\_ip=Math.round(School.P\_ip/20.0);

School.RP\_hin=Math.round(School.P\_hin/20.0);

School.Rhyprac\_bio=Math.round(School.hyprac\_bio/6.0);

School.Rhyprac\_phy=Math.round(School.hyprac\_phy/6.0);

School.Rhyprac\_ip=Math.round(School.hyprac\_ip/6.0);

School.Rhyprac\_chem=Math.round(School.hyprac\_chem/6.0);

School.Rseprac\_bio=Math.round(School.hyprac\_bio/2.0);

School.Rseprac\_phy=Math.round(School.hyprac\_phy/2.0);

School.Rseprac\_ip=Math.round(School.hyprac\_ip/2.0);

School.Rseprac\_chem=Math.round(School.hyprac\_chem/2.0);

School.Teng=School.RP\_eng+School.RA\_eng+School.Rengut+School.Renghy+School.Rengse;

School.Thin=School.RP\_hin+School.RA\_hin+School.Rhinut+School.Rhinhy+School.Rhinse;

School.Tmths=School.RP\_mths+School.RA\_mths+School.Rmthsut+School.Rmthshy+School.Rmthsse;

School.Tchem=School.RP\_chem+School.RA\_chem+School.Rchemut+School.Rchemhy+School.Rchemse+School.Rhyprac\_chem+School.Rseprac\_chem;

School.Tphy=School.RP\_phy+School.RA\_phy+School.Rphyut+School.Rphyhy+School.Rphyse+School.Rhyprac\_phy+School.Rseprac\_phy;

School.Tbio=School.RP\_bio+School.RA\_bio+School.Rbiout+School.Rbiohy+School.Rbiose+School.Rhyprac\_bio+School.Rseprac\_bio;

School.Tip=School.RP\_ip+School.RA\_ip+School.Riput+School.Riphy+School.Ripse+School.Rhyprac\_ip+School.Rseprac\_ip;

new Reportcard().setVisible(true);

dispose();

}catch(Exception e)

{}

//jTabbedPane2.setSelectedIndex(1); // TODO add your handling code here:

}

private void jCheckDataBtn1ActionPerformed(java.awt.event.ActionEvent evt) {

try

{

jNextBtn1.nextFocus();

boolean b=((int)jMarksTable.getValueAt(0, 1)>40||(int)jMarksTable.getValueAt(0, 2)>40||(int)jMarksTable.getValueAt(0, 3)>40||(int)jMarksTable.getValueAt(1, 1)>40||(int)jMarksTable.getValueAt(1, 2)>40||(int)jMarksTable.getValueAt(1, 3)>40||(int)jMarksTable.getValueAt(2, 1)>40||(int)jMarksTable.getValueAt(2, 2)>40||(int)jMarksTable.getValueAt(2, 3)>40||(int)jMarksTable.getValueAt(3, 1)>40||(int)jMarksTable.getValueAt(3, 2)>40);//||jTable3.getValueAt(0, 2)>40||

boolean d=((int)jMarksTable.getValueAt(0, 4)>100||(int)jMarksTable.getValueAt(0, 5)>100||(int)jMarksTable.getValueAt(1, 4)>100||(int)jMarksTable.getValueAt(1, 5)>100||(int)jMarksTable.getValueAt(2, 4)>100||(int)jMarksTable.getValueAt(2, 5)>100||(int)jMarksTable.getValueAt(3, 4)>70||(int)jMarksTable.getValueAt(3, 5)>70||(int)jMarksTable.getValueAt(4, 4)>70||(int)jMarksTable.getValueAt(4, 5)>70||(int)jMarksTable.getValueAt(5, 4)>70);//||jTable3.getValueAt(0, 2)>40||

boolean e=((int)jMarksTable.getValueAt(5, 5)>70||(int)jMarksTable.getValueAt(6, 4)>70||(int)jMarksTable.getValueAt(6, 5)>70);//||(int)jTable3.getValueAt(1, 5)>100||(int)jTable3.getValueAt(2, 4)>100);

boolean c=((int)jMarksTable.getValueAt(3, 3)>40||(int)jMarksTable.getValueAt(4, 1)>40||(int)jMarksTable.getValueAt(4, 2)>40||(int)jMarksTable.getValueAt(4, 3)>40||(int)jMarksTable.getValueAt(5, 1)>40||(int)jMarksTable.getValueAt(5, 2)>40||(int)jMarksTable.getValueAt(5,3)>40||(int)jMarksTable.getValueAt(6, 1)>40||(int)jMarksTable.getValueAt(6, 2)>40||(int)jMarksTable.getValueAt(6, 3)>40);//||(int)jTable3.getValueAt(3, 2)>40);

boolean f=((int)jMarksTable.getValueAt(3, 6)>30||(int)jMarksTable.getValueAt(3, 7)>30||(int)jMarksTable.getValueAt(4, 6)>30);

boolean g=((int)jMarksTable.getValueAt(4, 7)>30||(int)jMarksTable.getValueAt(5, 7)>30||(int)jMarksTable.getValueAt(5, 6)>30);

boolean h=((int)jMarksTable.getValueAt(6, 6)>30||(int)jMarksTable.getValueAt(6, 7)>30||(int)jMarksTable.getValueAt(4, 6)>30);

for(int r=0;r<=6;r++)

{

for(int col=1;col<=7;col++)

{

if(jMarksTable.getValueAt(r, col).equals(null))

{

jMarksTable.setValueAt(0,r, col);

}

}

}

if(b==true || c==true||d==true || e==true||f==true || g==true||h==true)

{

JOptionPane.showMessageDialog(rootPane, "MARKS OVER THE LIMIT.....");

}

else

{

jNextBtn1.setEnabled(true);

}

}

catch(Exception e)

{

JOptionPane.showMessageDialog(rootPane, e.getMessage());

}// TODO add your handling code here:

}

private void jMarksTableKeyTyped(java.awt.event.KeyEvent evt) {

if(!(evt.getKeyChar()>='0'&&evt.getKeyChar()<='9'))

{

evt.consume();

}// TODO add your handling code here:

}

private void jAssignmentTableKeyTyped(java.awt.event.KeyEvent evt) {

if(!(evt.getKeyChar()>='0'&&evt.getKeyChar()<='9'))

{

evt.consume();

} // TODO add your handling code here:

}

private void jProjectTableKeyTyped(java.awt.event.KeyEvent evt) {

if(!(evt.getKeyChar()>='0'&&evt.getKeyChar()<='9'))

{

evt.consume();

} // TODO add your handling code here:

}

private void jRollnoTFKeyTyped(java.awt.event.KeyEvent evt) {

if(!(evt.getKeyChar()>='0'&&evt.getKeyChar()<='9'))

{

evt.consume();

} // TODO add your handling code here:

}

private void jAdmNoTFKeyTyped(java.awt.event.KeyEvent evt) {

if(!(evt.getKeyChar()>='0'&&evt.getKeyChar()<='9'))

{

evt.consume();

} // TODO add your handling code here:

}

private void jCLassTFKeyTyped(java.awt.event.KeyEvent evt) {

if(!(evt.getKeyChar()>='0'&&evt.getKeyChar()<='9'))

{

evt.consume();

} // TODO add your handling code here:

}

private void jSessionTFKeyTyped(java.awt.event.KeyEvent evt) {

if(!(evt.getKeyChar()>='0'&&evt.getKeyChar()<='9'&&evt.getKeyChar()=='-'))

{

evt.consume();

}

// TODO add your handling code here:

}

private void jCheckDataBtn2ActionPerformed(java.awt.event.ActionEvent evt) {

try{

jNextBtn2.nextFocus();

boolean b=((int)jAssignmentTable.getValueAt(0, 1)>30||(int)jAssignmentTable.getValueAt(0, 2)>30||(int)jAssignmentTable.getValueAt(0, 3)>40||(int)jAssignmentTable.getValueAt(1, 1)>30||(int)jAssignmentTable.getValueAt(1, 2)>30||(int)jAssignmentTable.getValueAt(1, 3)>40||(int)jAssignmentTable.getValueAt(2, 1)>30||(int)jAssignmentTable.getValueAt(2, 2)>30||(int)jAssignmentTable.getValueAt(2, 3)>40||(int)jAssignmentTable.getValueAt(3, 1)>30||(int)jAssignmentTable.getValueAt(3, 2)>30);//||jTable3.getValueAt(0, 2)>40||

// TODO add your handling code here:

boolean c=((int)jAssignmentTable.getValueAt(3, 3)>40||(int)jAssignmentTable.getValueAt(4, 1)>30||(int)jAssignmentTable.getValueAt(4, 2)>30||(int)jAssignmentTable.getValueAt(4, 3)>40||(int)jAssignmentTable.getValueAt(5, 1)>30||(int)jAssignmentTable.getValueAt(5, 2)>30||(int)jAssignmentTable.getValueAt(5, 3)>40||(int)jAssignmentTable.getValueAt(6, 1)>30||(int)jAssignmentTable.getValueAt(6, 2)>30||(int)jAssignmentTable.getValueAt(6, 3)>40);//||(int)jTable1.getValueAt(3, 2)>30);//||jTable3.getValueAt(0, 2)>40||

// TODO add your handling code here:

if(b==true||c==true)

{

JOptionPane.showMessageDialog(rootPane, "MARKS OVER THE LIMIT.....");

}

else

{

jNextBtn2.setEnabled(true);

}

}

catch(Exception e)

{

JOptionPane.showMessageDialog(rootPane, e.getMessage());

}

}

private void jCheckBtn3ActionPerformed(java.awt.event.ActionEvent evt) {

try{

jEnterDataBtn.nextFocus();

boolean b=((int)jAssignmentTable.getValueAt(0, 1)>30||(int)jAssignmentTable.getValueAt(0, 2)>30||(int)jAssignmentTable.getValueAt(0, 3)>40||(int)jAssignmentTable.getValueAt(1, 1)>30||(int)jAssignmentTable.getValueAt(1, 2)>30||(int)jAssignmentTable.getValueAt(1, 3)>40||(int)jAssignmentTable.getValueAt(2, 1)>30||(int)jAssignmentTable.getValueAt(2, 2)>30||(int)jAssignmentTable.getValueAt(2, 3)>40||(int)jAssignmentTable.getValueAt(3, 1)>30||(int)jAssignmentTable.getValueAt(3, 2)>30);//||jTable3.getValueAt(0, 2)>40||

// TODO add your handling code here:

boolean c=((int)jAssignmentTable.getValueAt(3, 3)>40||(int)jAssignmentTable.getValueAt(4, 1)>30||(int)jAssignmentTable.getValueAt(4, 2)>30||(int)jAssignmentTable.getValueAt(4, 3)>40||(int)jAssignmentTable.getValueAt(5, 1)>30||(int)jAssignmentTable.getValueAt(5, 2)>30||(int)jAssignmentTable.getValueAt(5, 3)>40||(int)jAssignmentTable.getValueAt(6, 1)>30||(int)jAssignmentTable.getValueAt(6, 2)>30||(int)jAssignmentTable.getValueAt(6, 3)>40);//||(int)jTable1.getValueAt(3, 2)>30);//||jTable3.getValueAt(0, 2)>40||

// TODO add your handling code here:

if(b==true||c==true)

{

JOptionPane.showMessageDialog(rootPane, "MARKS OVER THE LIMIT.....");

}

else

{

jEnterDataBtn.setEnabled(true);

}

}

catch(Exception e)

{

JOptionPane.showMessageDialog(rootPane, e.getMessage());

}// TODO add your handling code here:

}

private void formWindowOpened(java.awt.event.WindowEvent evt) {

try

{

School.sub1=JOptionPane.showInputDialog("HINDI OR IP????");

while(!(School.sub1.equalsIgnoreCase("hindi")||School.sub1.equalsIgnoreCase("ip")))

{

School.sub1=JOptionPane.showInputDialog("HINDI OR IP????");

}

School.sub2=JOptionPane.showInputDialog("BIO OR MATHS????");

while(!(School.sub2.equalsIgnoreCase("bio")||School.sub2.equalsIgnoreCase("maths")))

{

School.sub2=JOptionPane.showInputDialog("BIO OR MATHS????");

}

}

catch(Exception e)

{

JOptionPane.showMessageDialog(rootPane, e.getMessage());

}

// TODO add your handling code here:

}

private void jMarksTableFocusLost(java.awt.event.FocusEvent evt) {

if(School.sub1.equalsIgnoreCase("hindi"))

{

for(int col=1;col<=7;col++)

{

jMarksTable.setValueAt(0,6, col);

}

}

else

{

for(int col=1;col<=7;col++)

{

jMarksTable.setValueAt(0,1, col);

}

}

if(School.sub2.equalsIgnoreCase("bio"))

{

for(int col=1;col<=7;col++)

{

jMarksTable.setValueAt(0,2, col);

}

}

else

{

for(int col=1;col<=7;col++)

{

jMarksTable.setValueAt(0,5, col);

}

}// TODO add your handling code here:

}

private void jAssignmentTableFocusLost(java.awt.event.FocusEvent evt) {

try

{

if(School.sub1.equalsIgnoreCase("hindi"))

{

for(int col=1;col<=3;col++)

{

jAssignmentTable.setValueAt(0,6, col);

}

}

else

{

for(int col=1;col<=3;col++)

{

jAssignmentTable.setValueAt(0,2, col);

}

}

if(School.sub2.equalsIgnoreCase("bio"))

{

for(int col=1;col<=3;col++)

{

jAssignmentTable.setValueAt(0,3, col);

}

}

else

{

for(int col=1;col<=3;col++)

{

jAssignmentTable.setValueAt(0,5, col);

}

}

}

catch(Exception e)

{

JOptionPane.showMessageDialog(rootPane, e.getMessage());

} }

private void jProjectTableFocusLost(java.awt.event.FocusEvent evt) {

try

{

if(School.sub1.equalsIgnoreCase("hindi"))

{

for(int col=1;col<=3;col++)

{

jProjectTable.setValueAt(0,6, col);

}

}

else

{

for(int col=1;col<=3;col++)

{

jProjectTable.setValueAt(0,2, col);

}

}

if(School.sub2.equalsIgnoreCase("bio"))

{

for(int col=1;col<=3;col++)

{

jProjectTable.setValueAt(0,3, col);

}

}

else

{

for(int col=1;col<=3;col++)

{

jProjectTable.setValueAt(0,5, col);

}

}

}

catch(Exception e)

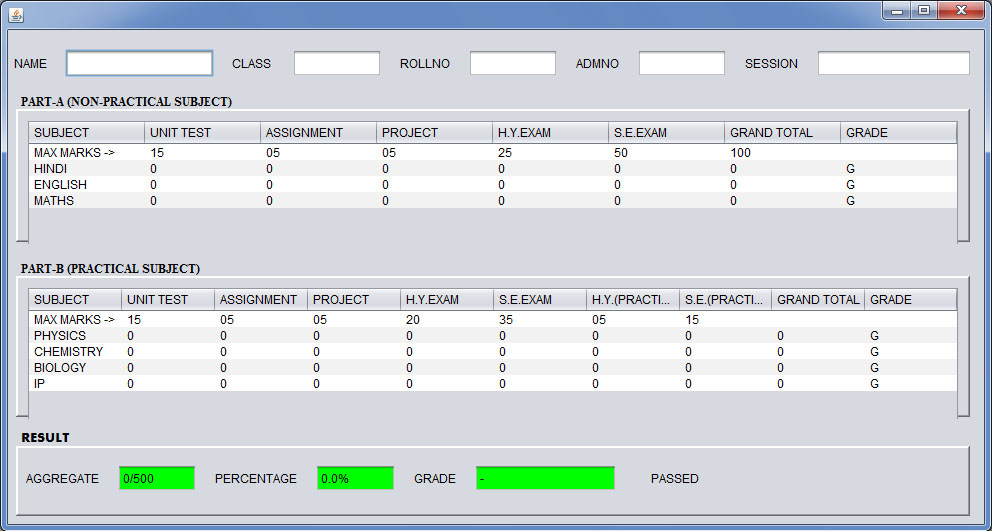
{

JOptionPane.showMessageDialog(rootPane, e.getMessage());

}// TODO add your handling code here:

}

**Frame 11:**



private void formWindowOpened(java.awt.event.WindowEvent evt) {

try

{

jAggregateTTF.setBackground(Color.red);

jPercentTF.setBackground(Color.red);

jGradeTF.setBackground(Color.red);

jNameTF.setText(School.name);

jClassTF.setText(School.cls);

jRollNoTF.setText(School.roll);

jSessionTF.setText(School.session);

jAdmNoTF.setText(School.admno);

jPartATable.setValueAt(School.Rhinut,1,1);

jPartATable.setValueAt(School.Rengut,2,1);

jPartATable.setValueAt(School.Rmthsut,3,1);

jPartATable.setValueAt(School.RA\_hin,1,2);

jPartATable.setValueAt(School.RA\_eng,2,2);

jPartATable.setValueAt(School.RA\_mths,3,2);

jPartATable.setValueAt(School.RP\_hin,1,3);

jPartATable.setValueAt(School.RP\_eng,2,3);

jPartATable.setValueAt(School.RP\_mths,3,3);

jPartATable.setValueAt(School.Rhinhy,1,4);

jPartATable.setValueAt(School.Renghy,2,4);

jPartATable.setValueAt(School.Rmthshy,3,4);

jPartATable.setValueAt(School.Rhinse,1,5);

jPartATable.setValueAt(School.Rengse,2,5);

jPartATable.setValueAt(School.Rmthsse,3,5);

jPartATable.setValueAt(School.Thin,1,6);

jPartATable.setValueAt(School.Teng,2,6);

jPartATable.setValueAt(School.Tmths,3,6);

jPartATable.setValueAt(Grade.grade(School.Thin),1,7);

jPartATable.setValueAt(Grade.grade(School.Teng),2,7);

jPartATable.setValueAt(Grade.grade(School.Tmths),3,7);

jPartBTable.setValueAt(School.Rphyut, 1, 1);

jPartBTable.setValueAt(School.Rchemut, 2, 1);

jPartBTable.setValueAt(School.Rbiout, 3, 1);

jPartBTable.setValueAt(School.Riput, 4, 1);

jPartBTable.setValueAt(School.RA\_phy, 1, 2);

jPartBTable.setValueAt(School.RA\_chem, 2, 2);

jPartBTable.setValueAt(School.RA\_bio, 3, 2);

jPartBTable.setValueAt(School.RA\_ip, 4, 2);

jPartBTable.setValueAt(School.RP\_phy, 1, 3);

jPartBTable.setValueAt(School.RP\_chem, 2, 3);

jPartBTable.setValueAt(School.RP\_bio, 3, 3);

jPartBTable.setValueAt(School.RP\_ip, 4, 3);

jPartBTable.setValueAt(School.Rphyhy, 1, 4);

jPartBTable.setValueAt(School.Rchemhy, 2, 4);

jPartBTable.setValueAt(School.Rbiohy, 3, 4);

jPartBTable.setValueAt(School.Riphy, 4, 4);

jPartBTable.setValueAt(School.Rphyse, 1, 5);

jPartBTable.setValueAt(School.Rchemse, 2, 5);

jPartBTable.setValueAt(School.Rbiose, 3, 5);

jPartBTable.setValueAt(School.Ripse, 4, 5);

jPartBTable.setValueAt(School.Rhyprac\_phy, 1, 6);

jPartBTable.setValueAt(School.Rhyprac\_chem, 2, 6);

jPartBTable.setValueAt(School.Rhyprac\_bio, 3, 6);

jPartBTable.setValueAt(School.Rhyprac\_ip, 4, 6);

jPartBTable.setValueAt(School.Rseprac\_phy, 1, 7);

jPartBTable.setValueAt(School.Rseprac\_chem, 2, 7);

jPartBTable.setValueAt(School.Rseprac\_bio, 3, 7);

jPartBTable.setValueAt(School.Rseprac\_ip, 4, 7);

jPartBTable.setValueAt(School.Tphy, 1, 8);

jPartBTable.setValueAt(School.Tchem, 2, 8);

jPartBTable.setValueAt(School.Tbio, 3, 8);

jPartBTable.setValueAt(School.Tip, 4, 8);

jPartBTable.setValueAt(Grade.grade(School.Tphy), 1, 9);

jPartBTable.setValueAt(Grade.grade(School.Tchem), 2, 9);

jPartBTable.setValueAt(Grade.grade(School.Tbio), 3, 9);

jPartBTable.setValueAt(Grade.grade(School.Tip), 4, 9);

long total=School.Tbio+School.Tphy+School.Tchem+School.Tip+School.Tmths+School.Teng+School.Thin;

double percnt=(School.Tbio+School.Tphy+School.Tchem+School.Tip+School.Tmths+School.Teng+School.Thin)/5.0;

jAggregateTTF.setText(total+"/500");

jPercentTF.setText(percnt+"%");

jGradeTF.setText(Grade.grade(percnt)+"");

System.out.print(Grade.grade(percnt));

int supp=0;

if(supp<=2&&supp!=0)

{

if(Grade.grade(School.Tip)=='G')

{supp++;

}

if(Grade.grade(School.Tphy)=='G')

{supp++;

}

if(Grade.grade(School.Tchem)=='G')

{supp++;

}

if(Grade.grade(School.Tbio)=='G')

{supp++;

}

if(Grade.grade(School.Tmths)=='G')

{supp++;

}

if(Grade.grade(School.Teng)=='G')

{supp++;

}

if(Grade.grade(School.Thin)=='G')

{supp++;

}

}

if(Grade.grade(percnt)=='G'||supp>2)

{

jAggregateTTF.setBackground(Color.red);

jPercentTF.setBackground(Color.red);

jGradeTF.setBackground(Color.red);

jResultLabel.setText("DETAINED");

}

else

{

if(supp!=0)

{

jAggregateTTF.setBackground(Color.yellow);

jPercentTF.setBackground(Color.yellow);

jGradeTF.setBackground(Color.yellow);

jResultLabel.setText("SUPPLYMENTARY");

}

else

{ jAggregateTTF.setBackground(Color.GREEN);

jPercentTF.setBackground(Color.GREEN);

jGradeTF.setBackground(Color.GREEN);

jResultLabel.setText("PASSED");

}

}

}

catch(Exception e)

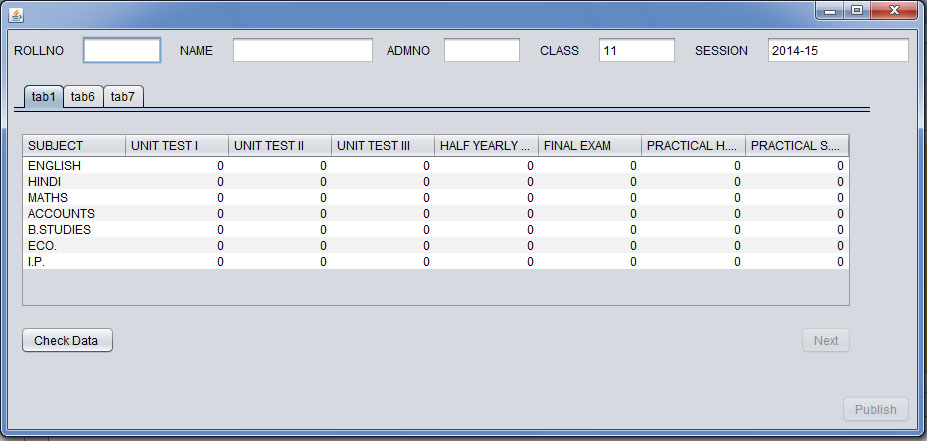
{

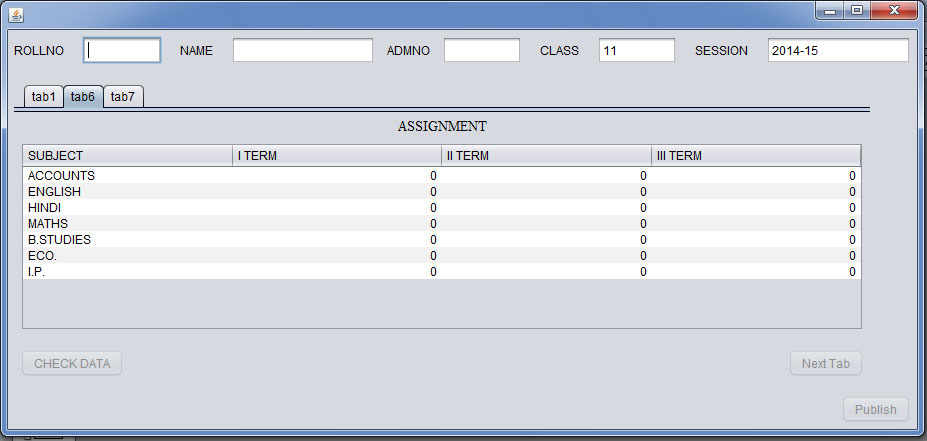
JOptionPane.showMessageDialog(rootPane, e.getMessage());

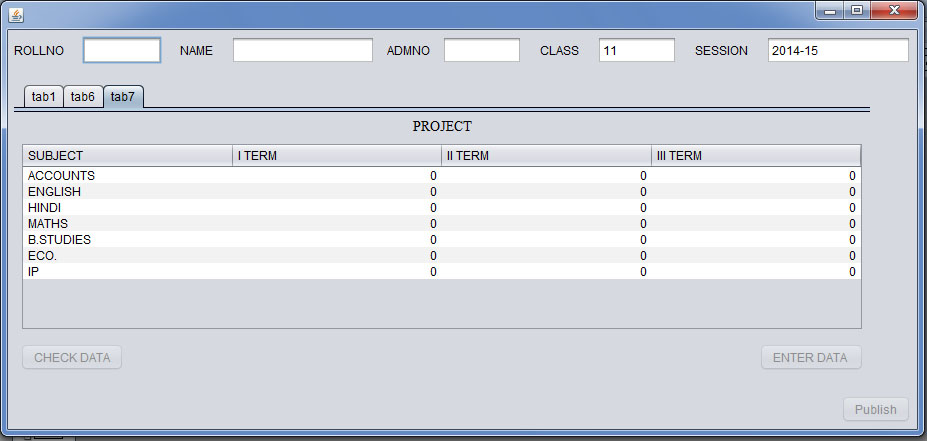
}// TODO add your handling code here:

}

**Frame 12:**







private void jPublishBtnActionPerformed(java.awt.event.ActionEvent evt) {

try{

School.name=jNameTf.getText();

School.roll=jRollnoTF.getText();

School.admno=jAdmnoTF.getText();

School.cls=jClassTF.getText();

School.session=jSessionTF.getText();

School.Rengut=Math.round(School.engut/8.0);

School.Raccut=Math.round(School.accut/8.0);

School.Rbsut=Math.round(School.bsut/8.0);

School.Rhinut=Math.round(School.hinut/8.0);

School.Rmthsut=Math.round(School.mthsut/8.0);

School.Recout=Math.round(School.ecout/8.0);

School.Riput=Math.round(School.iput/8.0);

School.Renghy=Math.round(School.enghy/4.0);

School.Racchy=Math.round(School.acchy/4.0);

School.Rbshy=Math.round(School.bshy/4.0);

School.Rhinhy=Math.round(School.hinhy/4.0);

School.Rmthshy=Math.round(School.mthshy/4.0);

School.Recohy=Math.round(School.ecohy/4.0);

School.Riphy=Math.round(School.iphy/3.5);

School.Rengse=Math.round(School.engse/2.0);

School.Raccse=Math.round(School.accse/2.0);

School.Rbsse=Math.round(School.bsse/2.0);

School.Rhinse=Math.round(School.hinse/2.0);

School.Rmthsse=Math.round(School.mthsse/2.0);

School.Recose=Math.round(School.ecose/2.0);

School.Ripse=Math.round(School.ipse/2.0);

School.RA\_mths=Math.round(School.A\_mths/20.0);

School.RA\_acc=Math.round(School.A\_acc/20.0);

School.RA\_bs=Math.round(School.A\_bs/20.0);

School.RA\_eng=Math.round(School.A\_eng/20.0);

School.RA\_ip=Math.round(School.A\_ip/20.0);

School.RA\_eco=Math.round(School.A\_eco/20.0);

School.RA\_hin=Math.round(School.A\_hin/20.0);

School.RP\_mths=Math.round(School.P\_mths/20.0);

School.RP\_acc=Math.round(School.P\_acc/20.0);

School.RP\_bs=Math.round(School.P\_bs/20.0);

School.RP\_eng=Math.round(School.P\_eng/20.0);

School.RP\_ip=Math.round(School.P\_ip/20.0);

School.RP\_eco=Math.round(School.P\_eco/20.0);

School.RP\_hin=Math.round(School.P\_hin/20.0);

School.Rhyprac\_ip=Math.round(School.hyprac\_ip/6.0);

School.Rseprac\_ip=Math.round(School.seprac\_ip/2.0);

School.Teng=School.RP\_eng+School.RA\_eng+School.Rengut+School.Renghy+School.Rengse;

School.Thin=School.RP\_hin+School.RA\_hin+School.Rhinut+School.Rhinhy+School.Rhinse;

School.Tmths=School.RP\_mths+School.RA\_mths+School.Rmthsut+School.Rmthshy+School.Rmthsse;

School.Tacc=School.RP\_acc+School.RA\_acc+School.Raccut+School.Racchy+School.Raccse;

School.Tbs=School.RP\_bs+School.RA\_bs+School.Rbsut+School.Rbshy+School.Rbsse;

School.Teco=School.RP\_eco+School.RA\_eco+School.Recout+School.Recohy+School.Recose;

School.Tip=School.RP\_ip+School.RA\_ip+School.Riput+School.Riphy+School.Ripse+School.Rhyprac\_ip+School.Rseprac\_ip;

new reportcardcomm().setVisible(true);

dispose();

}catch(Exception e)

{}

//jTabbedPane2.setSelectedIndex(1); // TODO add your handling code here:

}

private void jMarksTableKeyTyped(java.awt.event.KeyEvent evt) {

if(!(evt.getKeyChar()>='0'&&evt.getKeyChar()<='9'))

{

evt.consume();

}// TODO add your handling code here:

}

private void jCheckBtn1ActionPerformed(java.awt.event.ActionEvent evt) {

try

{

jNextBtn1.nextFocus();

boolean b=((int)jMarksTable.getValueAt(0, 1)>40||(int)jMarksTable.getValueAt(0, 2)>40||(int)jMarksTable.getValueAt(0, 3)>40||(int)jMarksTable.getValueAt(1, 1)>40||(int)jMarksTable.getValueAt(1, 2)>40||(int)jMarksTable.getValueAt(1, 3)>40||(int)jMarksTable.getValueAt(2, 1)>40||(int)jMarksTable.getValueAt(2, 2)>40||(int)jMarksTable.getValueAt(2, 3)>40||(int)jMarksTable.getValueAt(3, 1)>40||(int)jMarksTable.getValueAt(3, 2)>40);//||jTable3.getValueAt(0, 2)>40||

boolean d=((int)jMarksTable.getValueAt(0, 4)>100||(int)jMarksTable.getValueAt(0, 5)>100||(int)jMarksTable.getValueAt(1, 4)>100||(int)jMarksTable.getValueAt(1, 5)>100||(int)jMarksTable.getValueAt(2, 4)>100||(int)jMarksTable.getValueAt(2, 5)>100||(int)jMarksTable.getValueAt(3, 4)>100||(int)jMarksTable.getValueAt(3, 5)>100||(int)jMarksTable.getValueAt(4, 4)>100||(int)jMarksTable.getValueAt(4, 5)>100||(int)jMarksTable.getValueAt(5, 4)>100);//||jTable3.getValueAt(0, 2)>40||

boolean e=((int)jMarksTable.getValueAt(5, 5)>100||(int)jMarksTable.getValueAt(6, 4)>70||(int)jMarksTable.getValueAt(6, 5)>70);//||(int)jTable3.getValueAt(1, 5)>100||(int)jTable3.getValueAt(2, 4)>100);

boolean c=((int)jMarksTable.getValueAt(3, 3)>40||(int)jMarksTable.getValueAt(4, 1)>40||(int)jMarksTable.getValueAt(4, 2)>40||(int)jMarksTable.getValueAt(4, 3)>40||(int)jMarksTable.getValueAt(5, 1)>40||(int)jMarksTable.getValueAt(5, 2)>40||(int)jMarksTable.getValueAt(5,3)>40||(int)jMarksTable.getValueAt(6, 1)>40||(int)jMarksTable.getValueAt(6, 2)>40||(int)jMarksTable.getValueAt(6, 3)>40);//||(int)jTable3.getValueAt(3, 2)>40);

boolean f=((int)jMarksTable.getValueAt(3, 6)>0||(int)jMarksTable.getValueAt(3, 7)>0||(int)jMarksTable.getValueAt(4, 6)>0);

boolean g=((int)jMarksTable.getValueAt(4, 7)>0||(int)jMarksTable.getValueAt(5, 7)>0||(int)jMarksTable.getValueAt(5, 6)>0);

boolean h=((int)jMarksTable.getValueAt(6, 6)>30||(int)jMarksTable.getValueAt(6, 7)>30||(int)jMarksTable.getValueAt(4, 6)>0);

for(int r=0;r<=6;r++)

{

for(int col=1;col<=7;col++)

{

if(jMarksTable.getValueAt(r, col).equals(null))

{

jMarksTable.setValueAt(0,r, col);

}

}

}

if(b==true || c==true||d==true || e==true||f==true || g==true||h==true)

{

JOptionPane.showMessageDialog(rootPane, "MARKS OVER THE LIMIT.....");

}

else

{

jNextBtn1.setEnabled(true);

}

}

catch(Exception e)

{

JOptionPane.showMessageDialog(rootPane, e.getMessage());

}// TODO add your handling code here:

// TODO add your handling code here:

}

private void jNextBtn1ActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

try{

jCheckBtn2.setEnabled(true);

School.engut=(int)jMarksTable.getValueAt(0, 1)+(int)jMarksTable.getValueAt(0, 2)+(int)jMarksTable.getValueAt(0, 3);

School.hinut=(int)jMarksTable.getValueAt(1, 1)+(int)jMarksTable.getValueAt(1, 2)+(int)jMarksTable.getValueAt(1, 3);

School.mthsut=(int)jMarksTable.getValueAt(2, 1)+(int)jMarksTable.getValueAt(2, 2)+(int)jMarksTable.getValueAt(2, 3);

School.accut=(int)jMarksTable.getValueAt(3, 1)+(int)jMarksTable.getValueAt(3, 2)+(int)jMarksTable.getValueAt(3, 3);

School.bsut=(int)jMarksTable.getValueAt(4, 1)+(int)jMarksTable.getValueAt(4, 2)+(int)jMarksTable.getValueAt(4, 3);

School.ecout=(int)jMarksTable.getValueAt(5, 1)+(int)jMarksTable.getValueAt(5, 2)+(int)jMarksTable.getValueAt(5, 3);

School.iput=(int)jMarksTable.getValueAt(6, 1)+(int)jMarksTable.getValueAt(6, 2)+(int)jMarksTable.getValueAt(6, 3);

School.enghy=(int)jMarksTable.getValueAt(0, 4);

School.hinhy=(int)jMarksTable.getValueAt(1, 4);

School.mthshy=(int)jMarksTable.getValueAt(2,4);

School.acchy=(int)jMarksTable.getValueAt(3, 4);

School.bshy=(int)jMarksTable.getValueAt(4, 4);

School.ecohy=(int)jMarksTable.getValueAt(5, 4);

School.iphy=(int)jMarksTable.getValueAt(6, 4);

School.engse=(int)jMarksTable.getValueAt(0, 5);

School.hinse=(int)jMarksTable.getValueAt(1, 5);

School.mthsse=(int)jMarksTable.getValueAt(2,5);

School.accse=(int)jMarksTable.getValueAt(3, 5);

School.bsse=(int)jMarksTable.getValueAt(4, 5);

School.ecose=(int)jMarksTable.getValueAt(5, 5);

School.ipse=(int)jMarksTable.getValueAt(6, 5);

School.hyprac\_ip=(int)jMarksTable.getValueAt(6, 6);

School.seprac\_ip=(int)jMarksTable.getValueAt(6, 7);

jTabbedPane2.setSelectedIndex(1);

}catch(Exception e)

{

JOptionPane.showMessageDialog(rootPane, e.getMessage());

}

}

private void jAssignmentTableKeyTyped(java.awt.event.KeyEvent evt) {

if(!(evt.getKeyChar()>='0'&&evt.getKeyChar()<='9'))

{

evt.consume();

} // TODO add your handling code here:

}

private void jNextBtn2ActionPerformed(java.awt.event.ActionEvent evt) {

try{School.A\_eng=(int)jAssignmentTable.getValueAt(1, 1)+(int)jAssignmentTable.getValueAt(1, 2)+(int)jAssignmentTable.getValueAt(1, 3);

School.A\_hin=(int)jAssignmentTable.getValueAt(2, 1)+(int)jAssignmentTable.getValueAt(2, 2)+(int)jAssignmentTable.getValueAt(2, 3);

School.A\_mths=(int)jAssignmentTable.getValueAt(3, 1)+(int)jAssignmentTable.getValueAt(3, 2)+(int)jAssignmentTable.getValueAt(3, 3);

School.A\_acc=(int)jAssignmentTable.getValueAt(0, 1)+(int)jAssignmentTable.getValueAt(0, 2)+(int)jAssignmentTable.getValueAt(0, 3);

School.A\_bs=(int)jAssignmentTable.getValueAt(4, 1)+(int)jAssignmentTable.getValueAt(4, 2)+(int)jAssignmentTable.getValueAt(4, 3);

School.A\_ip=(int)jAssignmentTable.getValueAt(6, 1)+(int)jAssignmentTable.getValueAt(6, 2)+(int)jAssignmentTable.getValueAt(6, 3);

School.A\_eco=(int)jAssignmentTable.getValueAt(5, 1)+(int)jAssignmentTable.getValueAt(5, 2)+(int)jAssignmentTable.getValueAt(5, 3);

jTabbedPane2.setSelectedIndex(2);

jCheckBtn3.setEnabled(true);

} catch(Exception e)

{}// TODO add your handling code here:

}

private void jProjectTableKeyTyped(java.awt.event.KeyEvent evt) {

if(!(evt.getKeyChar()>='0'&&evt.getKeyChar()<='9'))

{

evt.consume();

} // TODO add your handling code here:

}

private void jEnterDataBtnActionPerformed(java.awt.event.ActionEvent evt) {

try{School.P\_eng=(int)jProjectTable.getValueAt(1, 1)+(int)jProjectTable.getValueAt(1, 2)+(int)jProjectTable.getValueAt(1, 3);

School.P\_hin=(int)jProjectTable.getValueAt(2, 1)+(int)jProjectTable.getValueAt(2, 2)+(int)jProjectTable.getValueAt(2, 3);

School.P\_mths=(int)jProjectTable.getValueAt(3, 1)+(int)jProjectTable.getValueAt(3, 2)+(int)jProjectTable.getValueAt(3, 3);

School.P\_acc=(int)jProjectTable.getValueAt(0, 1)+(int)jProjectTable.getValueAt(0, 2)+(int)jProjectTable.getValueAt(0, 3);

School.P\_bs=(int)jProjectTable.getValueAt(4, 1)+(int)jProjectTable.getValueAt(4, 2)+(int)jProjectTable.getValueAt(4, 3);

School.P\_ip=(int)jProjectTable.getValueAt(6, 1)+(int)jProjectTable.getValueAt(6, 2)+(int)jProjectTable.getValueAt(6, 3);

School.P\_eco=(int)jProjectTable.getValueAt(5, 1)+(int)jProjectTable.getValueAt(5, 2)+(int)jProjectTable.getValueAt(5, 3);

jPublishBtn.setEnabled(true);

}catch(Exception e)

{}// TODO add your handling code here:

}

private void jRollnoTFKeyTyped(java.awt.event.KeyEvent evt) {

if(!(evt.getKeyChar()>='0'&&evt.getKeyChar()<='9'))

{

evt.consume();

} // TODO add your handling code here:

}

private void jSessionTFKeyTyped(java.awt.event.KeyEvent evt) {

String session=jSessionTF.getText();

int l=session.length();

if(!(evt.getKeyChar()>='0'&&evt.getKeyChar()<='9'&&evt.getKeyChar()=='-'))

{

evt.consume();

}

else

{ for(int n=0;n<l;n++)

{

if(evt.getKeyChar()=='-'&&session.charAt(n)=='-')

{

evt.consume();

}

if(evt.getKeyChar()=='-' &&n==0 )

{

evt.consume();

}

}

// here i have to apply a condition check...if(sessio)

}// TODO add your handling code here:

}

private void jAdmnoTFKeyTyped(java.awt.event.KeyEvent evt) {

if(!(evt.getKeyChar()>='0'&&evt.getKeyChar()<='9'))

{

evt.consume();

} // TODO add your handling code here:

}

private void jClassTFKeyTyped(java.awt.event.KeyEvent evt) {

if(!(evt.getKeyChar()>='0'&&evt.getKeyChar()<='9'))

{

evt.consume();

} // TODO add your handling code here:

}

private void jMarksTableFocusLost(java.awt.event.FocusEvent evt) {

if(School.sub1.equalsIgnoreCase("hindi"))

{

for(int col=1;col<=7;col++)

{

jMarksTable.setValueAt(0,6, col);

}

}

else

{

for(int col=1;col<=7;col++)

{

jMarksTable.setValueAt(0,1, col);

}

}

if(School.sub2.equalsIgnoreCase("eco"))

{

for(int col=1;col<=7;col++)

{

jMarksTable.setValueAt(0,2, col);

}

}

else

{

for(int col=1;col<=7;col++)

{

jMarksTable.setValueAt(0,5, col);

}

} // TODO add your handling code here:

}

private void formWindowOpened(java.awt.event.WindowEvent evt) {

try

{

School.sub1=JOptionPane.showInputDialog("HINDI OR IP????");

while(!(School.sub1.equalsIgnoreCase("hindi")||School.sub1.equalsIgnoreCase("ip")))

{

School.sub1=JOptionPane.showInputDialog("HINDI OR IP????");

}

School.sub2=JOptionPane.showInputDialog("ECO OR MATHS????");

while(!(School.sub2.equalsIgnoreCase("eco")||School.sub2.equalsIgnoreCase("maths")))

{

School.sub2=JOptionPane.showInputDialog("ECO OR MATHS????");

}

}

catch(Exception e)

{

JOptionPane.showMessageDialog(rootPane, e.getMessage());

} // TODO add your handling code here:

}

private void jAssignmentTableFocusLost(java.awt.event.FocusEvent evt) {

try

{

int col;

if(School.sub1.equalsIgnoreCase("hindi"))

{

for(col=1;col<=3;col++)

{

jAssignmentTable.setValueAt(0,6, col);

}

}

else

{

for(col=1;col<=3;col++)

{

jAssignmentTable.setValueAt(0,2, col);

}

}

if(School.sub2.equalsIgnoreCase("eco"))

{

for(col=1;col<=3;col++)

{

jAssignmentTable.setValueAt(0,3, col);

}

}

else

{

for(col=1;col<=3;col++)

{

jAssignmentTable.setValueAt(0,5, col);

}

}

}catch(Exception e)

{

JOptionPane.showMessageDialog(null,e.getMessage());

}// TODO add your handling code here:

}

private void jProjectTableFocusLost(java.awt.event.FocusEvent evt) {

try

{

int col;

if(School.sub1.equalsIgnoreCase("hindi"))

{

for(col=1;col<=3;col++)

{

jProjectTable.setValueAt(0,6, col);

}

}

else

{

for(col=1;col<=3;col++)

{

jProjectTable.setValueAt(0,2, col);

}

}

if(School.sub2.equalsIgnoreCase("eco"))

{

for(col=1;col<=3;col++)

{

jProjectTable.setValueAt(0,3, col);

}

}

else

{

for(col=1;col<=3;col++)

{

jProjectTable.setValueAt(0,5, col);

}

}

}catch(Exception e)

{

JOptionPane.showMessageDialog(null,e.getMessage());

}// TODO add your handling code here:

}

private void jCheckBtn2ActionPerformed(java.awt.event.ActionEvent evt) {

try{

jNextBtn2.nextFocus();

boolean b=((int)jAssignmentTable.getValueAt(0, 1)>30||(int)jAssignmentTable.getValueAt(0, 2)>30||(int)jAssignmentTable.getValueAt(0, 3)>40||(int)jAssignmentTable.getValueAt(1, 1)>30||(int)jAssignmentTable.getValueAt(1, 2)>30||(int)jAssignmentTable.getValueAt(1, 3)>40||(int)jAssignmentTable.getValueAt(2, 1)>30||(int)jAssignmentTable.getValueAt(2, 2)>30||(int)jAssignmentTable.getValueAt(2, 3)>40||(int)jAssignmentTable.getValueAt(3, 1)>30||(int)jAssignmentTable.getValueAt(3, 2)>30);//||jTable3.getValueAt(0, 2)>40||

// TODO add your handling code here:

boolean c=((int)jAssignmentTable.getValueAt(3, 3)>40||(int)jAssignmentTable.getValueAt(4, 1)>30||(int)jAssignmentTable.getValueAt(4, 2)>30||(int)jAssignmentTable.getValueAt(4, 3)>40||(int)jAssignmentTable.getValueAt(5, 1)>30||(int)jAssignmentTable.getValueAt(5, 2)>30||(int)jAssignmentTable.getValueAt(5, 3)>40||(int)jAssignmentTable.getValueAt(6, 1)>30||(int)jAssignmentTable.getValueAt(6, 2)>30||(int)jAssignmentTable.getValueAt(6, 3)>40);//||(int)jTable1.getValueAt(3, 2)>30);//||jTable3.getValueAt(0, 2)>40||

// TODO add your handling code here:

if(b==true||c==true)

{

JOptionPane.showMessageDialog(rootPane, "MARKS OVER THE LIMIT.....");

}

else

{

jNextBtn2.setEnabled(true);

}

}

catch(Exception e)

{

JOptionPane.showMessageDialog(rootPane, e.getMessage());

}

// TODO add your handling code here:

}

private void jCheckBtn3ActionPerformed(java.awt.event.ActionEvent evt) {

try{

jEnterDataBtn.nextFocus();

boolean b=((int)jAssignmentTable.getValueAt(0, 1)>30||(int)jAssignmentTable.getValueAt(0, 2)>30||(int)jAssignmentTable.getValueAt(0, 3)>40||(int)jAssignmentTable.getValueAt(1, 1)>30||(int)jAssignmentTable.getValueAt(1, 2)>30||(int)jAssignmentTable.getValueAt(1, 3)>40||(int)jAssignmentTable.getValueAt(2, 1)>30||(int)jAssignmentTable.getValueAt(2, 2)>30||(int)jAssignmentTable.getValueAt(2, 3)>40||(int)jAssignmentTable.getValueAt(3, 1)>30||(int)jAssignmentTable.getValueAt(3, 2)>30);

boolean c=((int)jAssignmentTable.getValueAt(3, 3)>40||(int)jAssignmentTable.getValueAt(4, 1)>30||(int)jAssignmentTable.getValueAt(4, 2)>30||(int)jAssignmentTable.getValueAt(4, 3)>40||(int)jAssignmentTable.getValueAt(5, 1)>30||(int)jAssignmentTable.getValueAt(5, 2)>30||(int)jAssignmentTable.getValueAt(5, 3)>40||(int)jAssignmentTable.getValueAt(6, 1)>30||(int)jAssignmentTable.getValueAt(6, 2)>30||(int)jAssignmentTable.getValueAt(6, 3)>40);

if(b==true||c==true)

{

JOptionPane.showMessageDialog(rootPane, "MARKS OVER THE LIMIT.....");

}

else

{

jEnterDataBtn.setEnabled(true);

}

}

catch(Exception e)

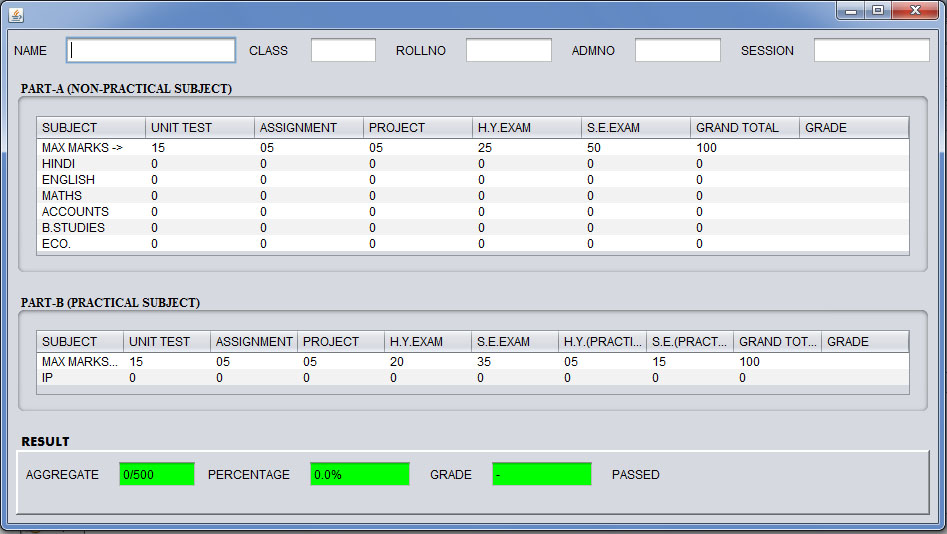
{

JOptionPane.showMessageDialog(rootPane, e.getMessage());

}

}

**Frame 13:**

****

private void formWindowOpened(java.awt.event.WindowEvent evt) {

try

{

jNameTF.setText(School.name);

jClassTF.setText(School.cls);

jRollnoTF.setText(School.roll);

jSessionTF.setText(School.session);

jAdmnoTF.setText(School.admno);

jPartATable.setValueAt(School.Rhinut,1,1);

jPartATable.setValueAt(School.Rengut,2,1);

jPartATable.setValueAt(School.Rmthsut,3,1);

jPartATable.setValueAt(School.RA\_hin,1,2);

jPartATable.setValueAt(School.RA\_eng,2,2);

jPartATable.setValueAt(School.RA\_mths,3,2);

jPartATable.setValueAt(School.RP\_hin,1,3);

jPartATable.setValueAt(School.RP\_eng,2,3);

jPartATable.setValueAt(School.RP\_mths,3,3);

jPartATable.setValueAt(School.Rhinhy,1,4);

jPartATable.setValueAt(School.Renghy,2,4);

jPartATable.setValueAt(School.Rmthshy,3,4);

jPartATable.setValueAt(School.Rhinse,1,5);

jPartATable.setValueAt(School.Rengse,2,5);

jPartATable.setValueAt(School.Rmthsse,3,5);

jPartATable.setValueAt(School.Thin,1,6);

jPartATable.setValueAt(School.Teng,2,6);

jPartATable.setValueAt(School.Tmths,3,6);

jPartATable.setValueAt(School.Rbsut, 5, 1);

jPartATable.setValueAt(School.Raccut, 4, 1);

jPartATable.setValueAt(School.Recout, 6, 1);

jPartBTable.setValueAt(School.Riput, 1, 1);

jPartATable.setValueAt(School.RA\_bs, 5, 2);

jPartATable.setValueAt(School.RA\_acc, 4, 2);

jPartATable.setValueAt(School.RA\_eco, 6, 2);

jPartBTable.setValueAt(School.RA\_ip, 1, 2);

jPartATable.setValueAt(School.RP\_bs, 5, 3);

jPartATable.setValueAt(School.RP\_acc, 4, 3);

jPartATable.setValueAt(School.RP\_eco, 6, 3);

jPartBTable.setValueAt(School.RP\_ip, 1, 3);

jPartATable.setValueAt(School.Rbshy, 5, 4);

jPartATable.setValueAt(School.Racchy, 4, 4);

jPartATable.setValueAt(School.Recohy, 6, 4);

jPartBTable.setValueAt(School.Riphy, 1, 4);

jPartATable.setValueAt(School.Rbsse, 5, 5);

jPartATable.setValueAt(School.Raccse, 4, 5);

jPartATable.setValueAt(School.Recose, 6, 5);

jPartBTable.setValueAt(School.Ripse, 1, 5);

jPartBTable.setValueAt(School.Rhyprac\_ip, 1, 6);

jPartBTable.setValueAt(School.Rseprac\_ip, 1, 7);

jPartATable.setValueAt(School.Tbs, 5, 6);

jPartATable.setValueAt(School.Tacc, 4, 6);

jPartATable.setValueAt(School.Teco, 6, 6);

jPartBTable.setValueAt(School.Tip, 1, 8);

long total=School.Teco+School.Tbs+School.Tacc+School.Tip+School.Tmths+School.Teng+School.Thin;

double percnt=(total)/5.0;

jAggregateTF.setText(total+"/500");

jPercentTF.setText(percnt+"%");

jGradeTF.setText(Grade.grade(percnt)+"");

System.out.print(Grade.grade(percnt));

int supp=0;

if(supp<=2 &&supp!=0){

if(Grade.grade(School.Tip)=='G')

{supp++;

}

if(Grade.grade(School.Tbs)=='G')

{supp++;

}

if(Grade.grade(School.Tacc)=='G')

{supp++;

}

if(Grade.grade(School.Teco)=='G')

{supp++;

}

if(Grade.grade(School.Tmths)=='G')

{supp++;

}

if(Grade.grade(School.Teng)=='G')

{supp++;

}

if(Grade.grade(School.Thin)=='G')

{supp++;

}

}

if(Grade.grade(percnt)=='G'||supp>2)

{

jAggregateTF.setBackground(Color.red);

jPercentTF.setBackground(Color.red);

jGradeTF.setBackground(Color.red);

jResultLabel.setText("DETAINED");

}

else

{

if(supp!=0)

{

jAggregateTF.setBackground(Color.yellow);

jPercentTF.setBackground(Color.yellow);

jGradeTF.setBackground(Color.yellow);

jResultLabel.setText("SUPPLYMENTARY");

}

else

{ jAggregateTF.setBackground(Color.GREEN);

jPercentTF.setBackground(Color.GREEN);

jGradeTF.setBackground(Color.GREEN);

jResultLabel.setText("PASSED");

}

}

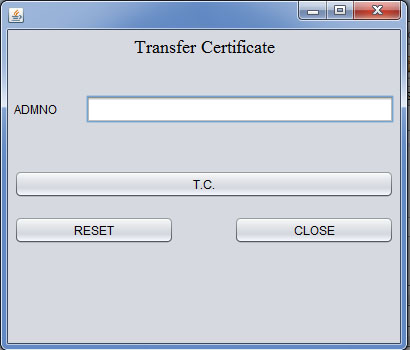
}

catch(Exception e)

{}

}

**Frame 14:**

****

private void jResetBtnActionPerformed(java.awt.event.ActionEvent evt) {

jAdmnoTF.setText(null);// TODO add your handling code here:

}

private void jCLoseBtnActionPerformed(java.awt.event.ActionEvent evt) {

System.exit(0); // TODO add your handling code here:

}

private void jTCBtnActionPerformed(java.awt.event.ActionEvent evt) {

try

{

Class.forName("java.sql.DriverManager");

Connection conn=(Connection)DriverManager.getConnection(School.url,School.username,School.pswd);

Statement st=(Statement)conn.createStatement();

String sql="Update student\_data set tc= '"+"Yes"+"' where admno="+Integer.parseInt(jAdmnoTF.getText())+"";

int k=st.executeUpdate(sql);

if(k>0)

{

JOptionPane.showMessageDialog(rootPane,"Record Updated Successfully...");

}

else

{

JOptionPane.showMessageDialog(rootPane,"Record Not Updated Successfully...");

}

st.close();

conn.close();

}

catch(Exception e)

{

JOptionPane.showMessageDialog(rootPane, e.getMessage());

}// TODO add your handling code here:

}

**Installation**

1. **Insert CD in your computer.**
2. **Open MySQL.**
3. **Create database(School).**
4. **Type command-**

* **Use school;**
* **Source <path of database from CD>**

1. **Open Netbeans.**
2. **Open Project(School Management).**
3. **Run Project.**
4. **SignUp as New user.**