



MINI PROJECT IN JAVA

TEAM MEMBERS

- 320126552001 A. Manasa
- 320126552002 B. Harika
- 320126552024 P. Jyonitha
- 320126552005 Ch.S.E Balaji
- 320126552009 G. Abhilash
- 320126552L01 K. GunaShankar

**BRANCH
CSM(AI&ML)**

**PROJECT ASSIGNED BY
V. SANGEETHA MADAM
DEPT OF COMPUTER SCIENCE**

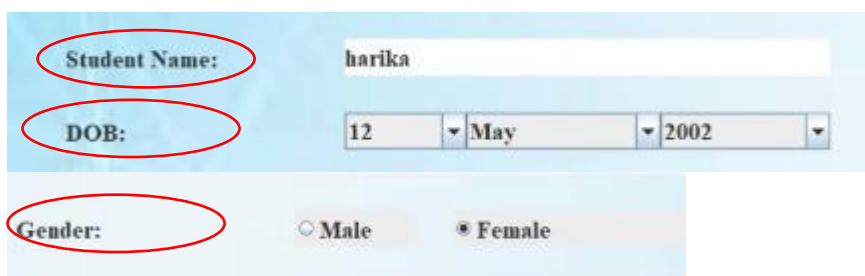
AIM: To create a registration form with details like Student name, DOB, Mail ID, Mobile number, Course, Father name, Branch etc.

COMPONENTS:

- i) JLabel
- ii) JTextField
- iii) JRadioButton
- iv) ButtonGroup
- v) JComboBox
- vi) JTextArea
- vii) JPasswordField
- viii) JButton

1. JLabel:

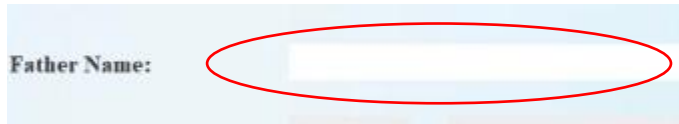
JLabel is used to display a single line of read only text. JLabel is used to display text, image or both.




JLabel is used in creating Student name, DOB, Gender.

2. JTextField:

JTextField allows the editing of a single line of text.



Father Name:

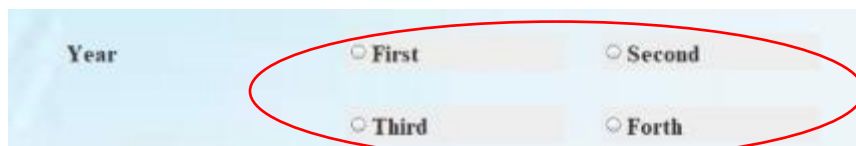


Mobile No:
Mail Id:

TextField is used in creating Mail ID, Mobile no etc. Where we can enter our preferred ID and number.

3.JRadioButton:

JRadioButton is used to choose one option from multiple options.



Year ☐ First ☐ Second
☐ Third ☐ Forth



Gender: ☐ Male ☒ Female

JRadioButton is used in creating gender and year of study. We can choose based on our preference.

4.ButtonGroup:

ButtonGroup are used in combination with radio buttons to ensure that only one radio button is selected.



Year ☐ First ☐ Second
☐ Third ☐ Forth

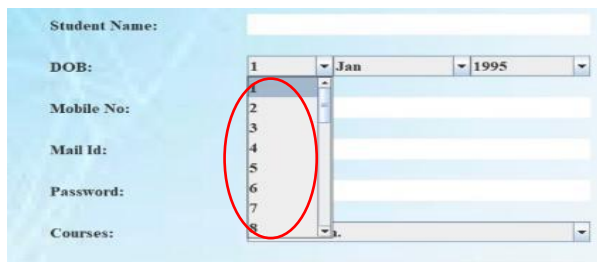


Gender: ☐ Male ☒ Female

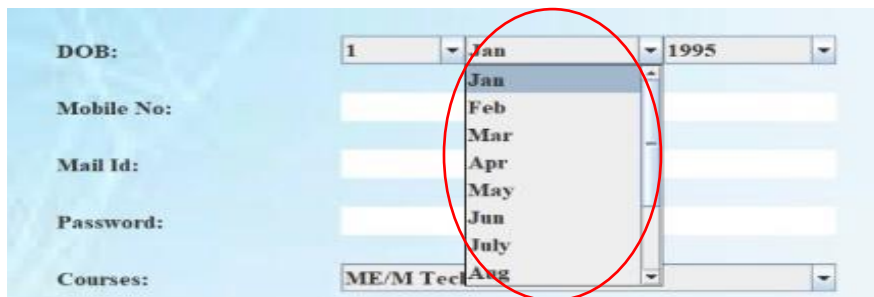
ButtonGroup is used in combination of year radiobutton and gender radiobutton to select one option.

5. JComboBox:

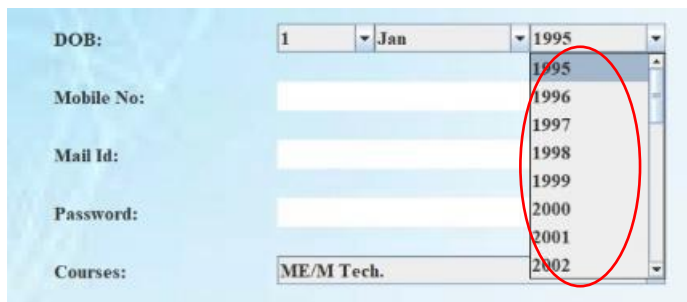
JComboBox can be editable or read only depending on the choice of the programmer.



A screenshot of a form with labels: Student Name, DOB, Mobile No, Mail Id, Password, and Courses. The DOB field is a JComboBox with three dropdowns: day (1), month (Jan), and year (1995). A red circle highlights the day dropdown menu, which is open and shows a list of numbers from 1 to 8.



A screenshot of a form with labels: DOB, Mobile No, Mail Id, Password, and Courses. The DOB field is a JComboBox with three dropdowns: day (1), month (Jan), and year (1995). A red circle highlights the month dropdown menu, which is open and shows a list of months from Jan to Aug.



A screenshot of a form with labels: DOB, Mobile No, Mail Id, Password, and Courses. The DOB field is a JComboBox with three dropdowns: day (1), month (Jan), and year (1995). A red circle highlights the year dropdown menu, which is open and shows a list of years from 1995 to 2002.

JComboBox is used to select the date, year and month according to user DOB.

6. JTextArea:

JTextArea is a multi-line area that displays plain text.



A screenshot of a form with a label: Address. The Address field is a JTextArea, which is a multi-line text area. A red circle highlights the JTextArea.

JTextArea is used to enter the address of the user.

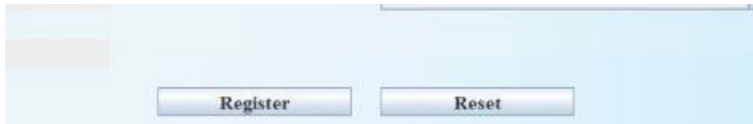
7. JPasswordField:

JPasswordField is used to provide specialized text fields for password entry.

It is used to enter respective password of the email ID.

8. JButton:

JButton is used to create a labeled button.



JButton is used in creating register and reset buttons.

SOURCE CODE:

```
import java.awt.Container;
import java.awt.Font;
import java.awt.event.*;
import javax.lang.model.util.ElementScanner14;
import javax.swing.*;
import java.util.regex.Matcher;
import java.util.regex.Pattern;
import javax.swing.border.Border;
import javax.swing.BorderFactory;
import java.awt.Color;
import java.io.*;
public class RegisterFrame extends JFrame implements
ActionListener
{
JLabel message;
JLabel nameLabel, dobLabel,
genderLabel, yearofstudyLabel, prevper;
JTextField nameField;
JRadioButton genderMale, genderFemale;
```

```
JRadioButton
yearofstudyFirst,yearofstudySecond,yearofstudyThird,yearof
studyForth;
ButtonGroup genderGroup;
ButtonGroup yearofstudyGroup;
JComboBox date;
JComboBox month;
JComboBox year;
JLabel mailIdLabel, mobileNoLabel,addressLabel;
JTextField mailIdField, mobileNoField, fatherField;
JTextArea addressarea,gpa;
JLabel fatherLabel, passwordLabel;
JPasswordField passwordField;
JLabel programLabel;
JComboBox<String> programList;
JLabel branchLabel, semesterLabel;
JComboBox<String> branchList;
JComboBox<Integer> semesterList;
JButton registerButton,resetButton;
Container container;
JPanel panel;
JLabel bg,img;
ImageIcon image1,image2;
private String dates[]
= { "1", "2", "3", "4", "5",
    "6", "7", "8", "9", "10",
    "11", "12", "13", "14", "15",
    "16", "17", "18", "19", "20",
    "21", "22", "23", "24", "25",
    "26", "27", "28", "29", "30",
    "31" };
private String months[]
= { "Jan", "Feb", "Mar", "Apr",
```

```
"May", "Jun", "July", "Aug",  
"Sep", "Oct", "Nov", "Dec" };  
private String years[]  
= { "1995", "1996", "1997", "1998",  
"1999", "2000", "2001", "2002",  
"2003", "2004", "2005", "2006",  
"2007", "2008", "2009", "2010",  
"2011", "2012", "2013", "2014",  
"2015", "2016", "2017", "2018",  
"2019", "2020", "2021", "2022" };  
public RegisterFrame()  
{  
    prevper = new JLabel();  
    // prevper label end  
    nameField = new JTextField();  
    // gpa textfield  
    gpa = new JTextArea();  
    // gpa textfield end  
    image1=new  
    ImageIcon("C:\\\\Users\\\\harikakushal\\\\OneDrive\\\\Desktop\\\\background1.jpg");  
    bg = new JLabel(image1);  
    bg.setSize(1700,1080);  
    panel = new JPanel();  
    image2=new  
    ImageIcon("C:\\\\Users\\\\harikakushal\\\\OneDrive\\\\Desktop\\\\anits22.jpg");  
    panel.setSize(1700,100);  
    img = new JLabel(image2);  
    Border eb = BorderFactory.createEmptyBorder();//For setting  
    the default border  
    message = new JLabel("STUDENT REGISTRATION FORM");  
    message.setFont(new Font("Courier", Font.BOLD, 50));
```

```

nameLabel = new JLabel("Student Name: ");
nameLabel.setFont(new Font("Serif", Font.BOLD, 20));
nameField = new JTextField();
nameField.setBorder(eb);
nameField.setFont(new Font("Serif", Font.BOLD, 20));
dobLabel = new JLabel("DOB: ");
dobLabel.setFont(new Font("Serif", Font.BOLD, 20));
date = new JComboBox(dates);
date.setFont(new Font("Serif", Font.BOLD, 20));
month = new JComboBox(months);
month.setFont(new Font("Serif", Font.BOLD, 20));
year = new JComboBox(years);
year.setFont(new Font("Serif", Font.BOLD, 20));
yearofstudyLabel=new JLabel("Year");
yearofstudyFirst=new JRadioButton(new
AbstractAction("First"){
public void actionPerformed(ActionEvent e){
prevper.setVisible(true);
prevper.setText("Inter Gpa: ");
gpa.setVisible(true);
}
});
prevper.setFont(new Font("Serif", Font.BOLD, 20));
gpa.setFont(new Font("Serif", Font.BOLD, 20));
yearofstudyLabel.setFont(new Font("Serif", Font.BOLD, 20));
// yearofstudyFirst=new JRadioButton("First");
yearofstudyFirst.setFont(new Font("Serif", Font.BOLD, 20));
// yearofstudySecond=new JRadioButton("Second");
yearofstudySecond.setFont(new Font("Serif", Font.BOLD,
20));
// yearofstudyThird=new JRadioButton("Third");
yearofstudyThird.setFont(new Font("Serif", Font.BOLD, 20));
// yearofstudyForth=new JRadioButton("Forth");

```



```
yearofstudyForth.setFont(new Font("Serif", Font.BOLD, 20));
yearofstudyGroup=new ButtonGroup();
yearofstudyGroup.add(yearofstudyFirst);
yearofstudyGroup.add(yearofstudySecond);
yearofstudyGroup.add(yearofstudyThird);
yearofstudyGroup.add(yearofstudyForth);
genderLabel = new JLabel("Gender:");
genderLabel.setFont(new Font("Serif", Font.BOLD, 20));
genderMale = new JRadioButton("Male", true);
genderMale.setFont(new Font("Serif", Font.BOLD, 20));
genderFemale = new JRadioButton("Female");
genderFemale.setFont(new Font("Serif", Font.BOLD,20));
genderGroup = new ButtonGroup();
genderGroup.add(genderMale);
genderGroup.add(genderFemale);
fatherLabel = new JLabel("Father Name:");
fatherLabel.setFont(new Font("Serif", Font.BOLD,20));
fatherField = new JTextField();
fatherField.setFont(new Font("Serif", Font.BOLD, 20));
fatherField.setBorder(eb);
mobileNoLabel = new JLabel("Mobile No:");
mobileNoLabel.setFont(new Font("Serif", Font.BOLD, 20));
mobileNoField = new JTextField();
mobileNoField.setFont(new Font("Serif", Font.BOLD, 20));
mobileNoField.setBorder(eb);
mailIdLabel = new JLabel("Mail Id:");
mailIdLabel.setFont(new Font("Serif", Font.BOLD, 20));
mailIdField = new JTextField();
mailIdField.setFont(new Font("Serif", Font.BOLD, 20));
mailIdField.setBorder(eb);
passwordLabel = new JLabel("Password:");
passwordLabel.setFont(new Font("Serif", Font.BOLD, 20));
passwordField = new JPasswordField();
```

```
passwordField.setFont(new Font("Serif", Font.BOLD, 20));
passwordField.setBorder(eb);
programLabel = new JLabel("Courses:");
programLabel.setFont(new Font("Serif", Font.BOLD, 20));
programList = new JComboBox<String>();
programList.setFont(new Font("Serif", Font.BOLD, 20));
programList.addItem("ME/M Tech.");
programList.addItem("BE/B Tech.");
branchLabel = new JLabel("Branch:");
branchLabel.setFont(new Font("Serif", Font.BOLD, 20));
branchList = new JComboBox<String>();
branchList.setFont(new Font("Serif", Font.BOLD, 20));
branchList.addItem("Computer Science and Engineering in AI
and ML");
branchList.addItem("Computer Science and Engineering in
Data Science");
branchList.addItem("Computer Science and Engineering");
branchList.addItem("Electronics and Communications
Engineering");
branchList.addItem("Information Technology");
branchList.addItem("Electrical and Electronics Engineering");
branchList.addItem("Civil Engineering");
branchList.addItem("Chemical Engineering");
semesterLabel = new JLabel("Semester:");
semesterLabel.setFont(new Font("Serif", Font.BOLD, 20));
semesterList = new JComboBox<>();
semesterList.setFont(new Font("Serif", Font.BOLD, 20));
semesterList.addItem(1);
semesterList.addItem(2);
addressLabel = new JLabel("Address:");
addressLabel.setFont(new Font("Serif", Font.BOLD, 20));
addressarea = new JTextArea();
addressarea.setFont(new Font("Serif", Font.BOLD, 20));
```

```
registerButton = new JButton("Register");
registerButton.setFont(new Font("Serif", Font.BOLD, 20));
registerButton.setBackground(Color.green);
resetButton = new JButton("Reset");
resetButton.setFont(new Font("Serif", Font.BOLD, 20));
resetButton.setBackground(Color.orange);
container = getContentPane();
container.setLayout(null);
setBounds();
addComponents();
}
public void setBounds()
{// prevper label bounds
prevper.setBounds(720,600,200,30);
// prevper label bounds end
// gpa textfield bounds
gpa.setBounds(950,600,400,30);
// gpa textfield bounds end
message.setBounds(400, 100, 1000, 50);
nameLabel.setBounds(50, 180, 200, 30);
nameField.setBounds(280, 180, 400, 30);
fatherLabel.setBounds(720, 180, 200, 30);
fatherField.setBounds(950, 180, 400, 30);
dobLabel.setBounds(50, 240, 200, 30);
date.setBounds(280,240,100,30);
month.setBounds(380,240,160,30);
year.setBounds(540,240,140,30);
genderLabel.setBounds(720, 240, 200, 30);
genderMale.setBounds(950, 240, 100, 30);
genderFemale.setBounds(1080, 240, 200, 30);
mobileNoLabel.setBounds(50, 300, 200, 30);
mobileNoField.setBounds(280, 300, 400, 30);
addressLabel.setBounds(720,300,200,30);
```

```
addressarea.setBounds(950,300,400,150);
mailIdLabel.setBounds(50, 360, 200, 30);
mailIdField.setBounds(280, 360, 400, 30);
passwordLabel.setBounds(50, 420, 200, 30);
passwordField.setBounds(280, 420, 400, 30);
programLabel.setBounds(50, 480, 200, 30);
programList.setBounds(280, 480, 400, 30);
branchLabel.setBounds(720, 480, 200, 30);
branchList.setBounds(950, 480, 400, 30);
yearofstudyLabel.setBounds(50,540,200,30);
yearofstudyFirst.setBounds(280,540,180,30);
yearofstudySecond.setBounds(490,540,180,30);
yearofstudyThird.setBounds(280,600,180,30);
yearofstudyForth.setBounds(490,600,180,30);
semesterLabel.setBounds(720, 540, 200, 30);
semesterList.setBounds(950, 540, 400, 30);
registerButton.setBounds(480, 680, 200, 30);
resetButton.setBounds(720, 680, 200, 30);
registerButton.addActionListener(this);
resetButton.addActionListener(this);
}

public void addComponents()
{
    // prevper label add
    container.add(prevper);
    // prevper lable add end
    // gpa field add
    container.add(gpa);
    gpa.setVisible(false);
    // gpa field add end
    panel.add(img);
}
```

```
bg.add(message);
bg.add(nameLabel);
bg.add(nameField);
bg.add(dobLabel);
bg.add(date);
bg.add(month);
bg.add(year);
bg.add(genderLabel);
bg.add(genderMale);
bg.add(genderFemale);
bg.add(fatherLabel);
bg.add(fatherField);
bg.add(mobileNoLabel);
bg.add(mobileNoField);
bg.add(mailIdLabel);
bg.add(mailIdField);
bg.add(passwordLabel);
bg.add(passwordField);
bg.add(programLabel);
bg.add(programList);
bg.add(branchLabel);
bg.add(branchList);
bg.add(yearofstudyLabel);
bg.add(yearofstudyFirst);
bg.add(yearofstudySecond);
```

```
bg.add(yearofstudyThird);
bg.add(yearofstudyForth);
bg.add(semesterLabel);
bg.add(semesterList);
bg.add(addressLabel);
bg.add(addressarea);
bg.add(registerButton);
bg.add(resetButton);
bg.add(panel);
container.add(bg);
}

//To verify Names
public boolean CheckName(String s){
if(s.length()==0)return false;
char[] c=s.toCharArray();
int d=0;
for(char i:c){
if(i=='.')d++;
if(!Character.isAlphabetic(i))return false;
}
if(d>1)return false;
return true;
}

//To verify Number
public boolean CheckNum(String s){
```

```

char[] c=s.toCharArray();
if(s.length()!=10)return false;
for (char i:c){
if(!Character.isDigit(i))return false;
}
return true;
}

//To verify Mail Id using REGEX expression
public boolean CheckMail(String email){
if(email.length()==0)return false;

else{    String  ePattern  =  "^[a-zA-Z0-9.!#$%&'*/=?^_`{|}~-
]+@((\\[[0-9]{1,3}\\.[0-9]{1,3}\\.[0-9]{1,3}\\.[0-9]{1,3}\\)|
([a-zA-Z\\-0-9]+\\.)+[a-zA-Z]{2,}))$";

java.util.regex.Pattern                                p                                =
java.util.regex.Pattern.compile(ePattern);

java.util.regex.Matcher m = p.matcher(email);

return m.matches();}

}

//To verify Password
public boolean CheckP(String s1) {
if(s1.length()<8)
return false;
else
return true;
}

```

```
// To reset Border color for correct Entries
public void resetB(){
    Border res = BorderFactory.createEmptyBorder();
    nameField.setBorder(res);
    fatherField.setBorder(res);
    mobileNoField.setBorder(res);
    mailIdField.setBorder(res);
    passwordField.setBorder(res);
    addressarea.setBorder(res);
}

public boolean Checkg(String gpa){
    if (gpa == null) {
        return false;
    }
    try {
        double d = Double.parseDouble(gpa);
        if(d<0 || d>10)return false;
    } catch (NumberFormatException nfe) {
        return false;
    }
    return true;
}

public void actionPerformed(ActionEvent e)
{
    // For creating Border for Invalid entries
```



```
Border border = BorderFactory.createLineBorder(Color.RED,2);
if (e.getSource() == registerButton)
{
String name=nameField.getText();
String fname=fatherField.getText();
String num=mobileNoField.getText();
String mail=mailIdField.getText();
String pass=new String (passwordField.getPassword());
String add=addressarea.getText();
String g=gpa.getText();
String h;
if(genderMale.isSelected())
h="male";
else
h="female";
String zf;
if( yearofstudyFirst.isSelected())
zf="1st year";
else if(yearofstudySecond.isSelected())
zf="2nd year";
else if(yearofstudyThird.isSelected())
zf="3rd year";
else
zf="4th year";
String str=branchList.getSelectedItem().toString();
```

```
String mtr= semesterList.getSelectedItemAt().toString();
String fan= date.getSelectedItemAt().toString();
String fin =month.getSelectedItemAt().toString();
String fen= year.getSelectedItemAt().toString();
String ntr= programList.getSelectedItemAt().toString();
if(!CheckName(name)){
JOptionPane.showMessageDialog(this ,"Invalid Name");
resetB();
nameField.setBorder(border);
nameField.requestFocusInWindow();
}else if(!CheckName(fname)){
JOptionPane.showMessageDialog(this ,"Invalid Father Name");
resetB();
fatherField.setBorder(border);
fatherField.requestFocusInWindow();
}else if(!CheckNum(num)){
JOptionPane.showMessageDialog(this ,"Invalid Mobile Number");
resetB();
mobileNoField.setBorder(border);
mobileNoField.requestFocusInWindow();
}else if(!CheckMail(mail)){
JOptionPane.showMessageDialog(this ,"Invalid email
Credentials");
resetB();
mailIdField.setBorder(border);
```

```
//passwordField.setBorder(border);
mailIdField.requestFocusInWindow();
}
else if(!CheckP(pass)){
JOptionPane.showMessageDialog(this , "Invalid password
Credentials");
resetB();
passwordField.setBorder(border);
passwordField.requestFocusInWindow();
}

else if(add.length()==0){
JOptionPane.showMessageDialog(this , "Invalid Address");
resetB();
addressarea.setBorder(border);
}
else if(!Checkg(g)){
JOptionPane.showMessageDialog(this , "Invalid Gpa");
resetB();
gpa.setBorder(border);
gpa.requestFocusInWindow();
}
else {
try{
FileWriter w= new FileWriter("Sam.txt",true);
```

```
w.write("Name:"+name+"\n") ;
w.write("Father Name:"+fname+"\n") ;
w.write("DOB:"+fan+"\\ "+fin+"\\ "+fen+"\n");
w.write("Gender:"+h+"\n");
w.write("Mobile number:"+num+"\n") ;
w.write("Address:"+add+"\n") ;
w.write("Mail id:"+mail+"\n") ;
w.write("Password"+pass+"\n") ;
w.write("Course name:"+ntr+"\n");
w.write("Branch name:"+str+"\n");
w.write("Year of Study:"+zf+"\n");
w.write("Semester:"+mtr+"\n");
w.write("GPA:"+g+"\n");
w.write("\n");
w.close();
JOptionPane.showMessageDialog(this ,"Data Saved Successfully");
resetB();
}
catch(Exception ae)
{
System.out.println(ae);
}
}
}
if(e.getSource() == resetButton)
```

```
{
resetB();
semesterList.setSelectedIndex(0);
branchList.setSelectedIndex(0);
programList.setSelectedIndex(0);
date.setSelectedIndex(0);
month.setSelectedIndex(0);
year.setSelectedIndex(0);
    genderGroup.clearSelection();
    yearofstudyGroup.clearSelection();
    nameField.setText("");
    mailIdField.setText("");
    addressarea.setText("");
    mobileNoField.setText("");
    fatherField.setText("");
    passwordField.setText("");
    gpa.setText("");
}
}

public static void main(String[] args)
{
    RegisterFrame frame = new RegisterFrame();
    frame.setTitle("Student Register Form");
    frame.setVisible(true);
}
```

```

frame.setBounds(500, 100, 500, 700);

frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

frame.setResizable(true);

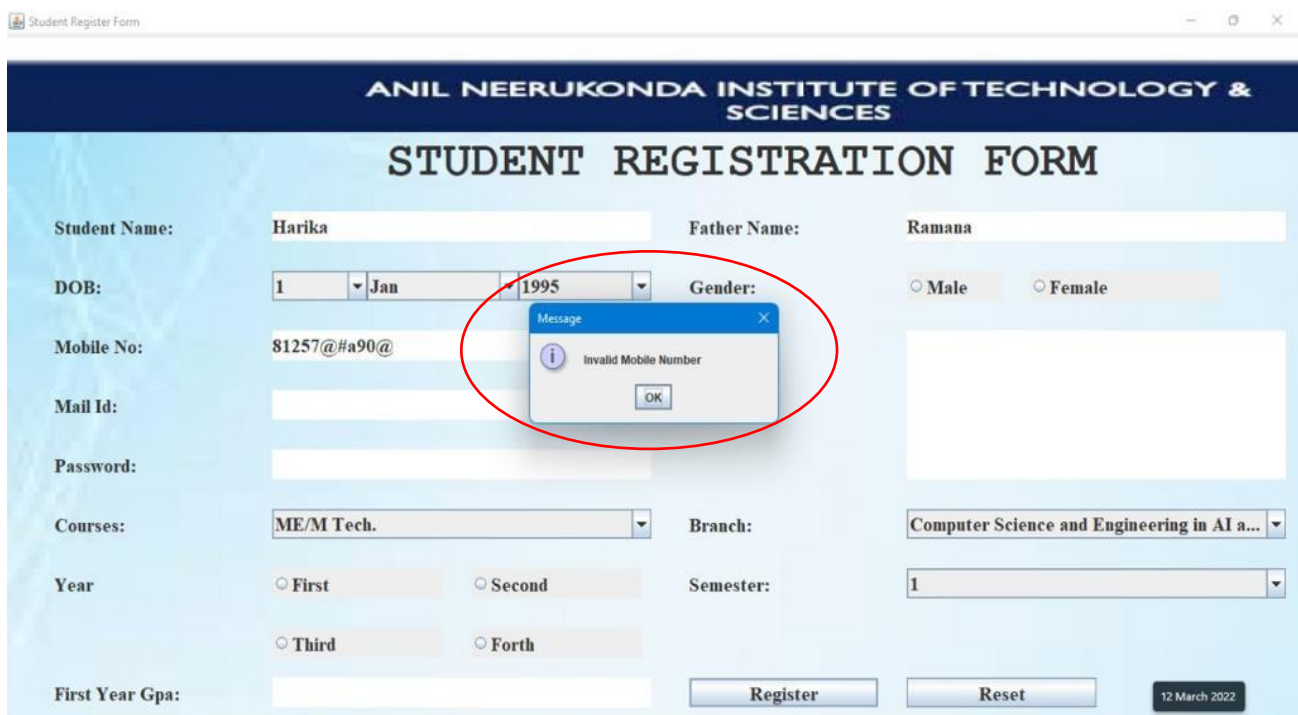
}

}

```

SPECIFICATIONS:

1. Displaying dialogue box:- whenever you click on register Button a display box pops up showing the status of the form whether completed filled or any invalid data entry.



The screenshot shows a web application titled "ANIL NEERUKONDA INSTITUTE OF TECHNOLOGY & SCIENCES" with a "STUDENT REGISTRATION FORM". The form contains several input fields: Student Name (Harika), Father Name (Ramana), DOB (1 Jan 1995), Gender (Male/Female), Mobile No (81257@#a90@), Mail Id, Password, Courses (ME/M Tech.), Branch (Computer Science and Engineering in AI a...), Year (First/Second/Third/Forth), Semester (1), and First Year Gpa. A red circle highlights the Mobile No field, which is displaying an error message: "Invalid Mobile Number". The message box is titled "Message" and has an "OK" button. The form also includes "Register" and "Reset" buttons at the bottom right, and a date stamp "12 March 2022".

2.Red colour border:- Whenever the user enters any invalid details in the given fields rather than the accepted one, then the field is highlighted with a red colour border.

ANIL NEERUKONDA INSTITUTE OF TECHNOLOGY & SCIENCES

STUDENT REGISTRATION FORM

Student Name: harika	Father Name: Ramana
DOB: 12 May 2002	Gender: <div style="border: 1px solid black; padding: 2px;">Computer Science and Engineering in AI and M Computer Science and Engineering in Data Scie Computer Science and Engineering Electronics and Communications Engineering Information Technology Electrical and Electronics Engineering Civil Engineering Chemical Engineering Computer Science and Engineering in AI a...</div>
Mobile No: 8125700902	Address:
Mail: harikabagadhi@gmail.com	
Password:	
Courses: ME/M Tech.	Branch: <div style="border: 1px solid black; padding: 2px;">Computer Science and Engineering in AI a...</div>
Year: <input type="radio"/> First <input checked="" type="radio"/> Second <input type="radio"/> Third <input type="radio"/> Forth	Semester: 1
First Year Gpa: 9.83	<div style="display: inline-block; border: 1px solid black; padding: 2px 10px;">Register</div> <div style="display: inline-block; border: 1px solid black; padding: 2px 10px; margin-left: 20px;">Reset</div>

3.Validity checking:-In order to check the given content of the fields are valid or not Regex or Regular expressions is an API that can be used to define the email password name areas of the string to define the constrains properly.

ANIL NEERUKONDA INSTITUTE OF TECHNOLOGY & SCIENCES

STUDENT REGISTRATION FORM

Student Name: Harika	Father Name: Ramana
DOB: 1 Jan 1995	Gender: <input type="radio"/> Male <input type="radio"/> Female
Mobile: 81257@#a90@	Address:
Mail Id:	
Password:	
Courses: ME/M Tech.	Branch: <div style="border: 1px solid black; padding: 2px;">Computer Science and Engineering in AI a...</div>
Year: <input type="radio"/> First <input type="radio"/> Second <input type="radio"/> Third <input type="radio"/> Forth	Semester: 1
First Year Gpa:	<div style="display: inline-block; border: 1px solid black; padding: 2px 10px;">Register</div> <div style="display: inline-block; border: 1px solid black; padding: 2px 10px; margin-left: 20px;">Reset</div>

4.Previous year percentage :-

ANIL NEERUKONDA INSTITUTE OF TECHNOLOGY & SCIENCES

STUDENT REGISTRATION FORM

Student Name: harika	Father Name: Ramana
DOB: 12 May 2002	Gender: <div style="border: 1px solid black; padding: 2px;">Computer Science and Engineering in AI and M Computer Science and Engineering in Data Sci Computer Science and Engineering Electronics and Communications Engineering Information Technology Electrical and Electronics Engineering Civil Engineering Chemical Engineering Computer Science and Engineering in AI a...</div>
Mobile No: 8125700902	Address:
Mail Id: harikabagadhi@gmail.com	
Password:	
Courses: ME/M Tech.	Branch: Computer Science and Engineering in AI a...
Year: <input type="radio"/> First <input checked="" type="radio"/> Second <input type="radio"/> Third <input type="radio"/> Forth	Semester: 1
First Year Gpa: 9.83	<div style="display: inline-block; border: 1px solid black; padding: 5px 10px;">Register</div> <div style="display: inline-block; border: 1px solid black; padding: 5px 10px; margin-left: 20px;">Reset</div>

Soon after clicking on the respective year of the student the corresponding previous year gpa field label is displayed along a text box .For example on clicking on the Second year First year gpa is displayed as shown in the above figure.

5.Reset:- The reset button is used to reset all the input fields. A new empty form is displayed on clicking on the Reset button.

Student Register Form

ANIL NEERUKONDA INSTITUTE OF TECHNOLOGY & SCIENCES

STUDENT REGISTRATION FORM

Student Name: <input type="text"/>	Father Name: <input type="text"/>
DOB: 1 Jan 1995	Gender: <input type="radio"/> Male <input type="radio"/> Female
Mobile No: <input type="text"/>	Address: <div style="border: 1px solid black; height: 40px;"></div>
Mail Id: <input type="text"/>	
Password: <input type="text"/>	
Courses: ME/M Tech.	Branch: Computer Science and Engineering in AI a...
Year: <input type="radio"/> First <input type="radio"/> Second <input type="radio"/> Third <input type="radio"/> Forth	Semester: 1
First Year Gpa: <input type="text"/>	

Register

Reset

6.Inserting the image :-

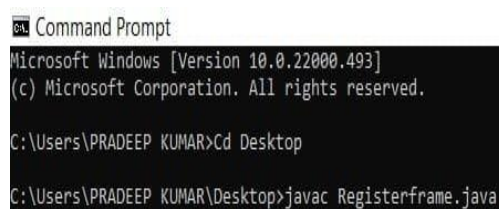
To embrace the look and feel of the registration form, Background image and a image containing the college name are inserted.

7.Transferring the details of the form to a Txt File:

After successful registration the details entered by the student are transferred to a txt file ,for readability purpose.

FINAL OUTCOME:-

Compiling the code :

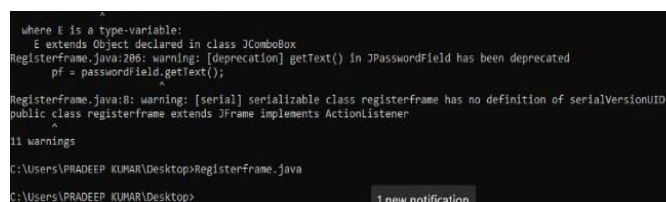


```
Command Prompt
Microsoft Windows [Version 10.0.22000.493]
(c) Microsoft Corporation. All rights reserved.

C:\Users\PRADEEP KUMAR>cd Desktop

C:\Users\PRADEEP KUMAR\Desktop>javac Registerframe.java
```

Running the code:



```
where E is a type-variable:
  E extends Object declared in class JComboBox
Registerframe.java:286: warning: [deprecation] getText() in JPasswordField has been deprecated
    pf = passwordField.getText();
                        ^
Registerframe.java:8: warning: [serial] serializable class registerframe has no definition of serialVersionUID
public class registerframe extends JFrame implements ActionListener
      ^
11 warnings

C:\Users\PRADEEP KUMAR\Desktop>Registerframe.java
C:\Users\PRADEEP KUMAR\Desktop>
```

1 new notification

Output:

Form is displayed in the below format

ANIL NEERUKONDA INSTITUTE OF TECHNOLOGY & SCIENCES

STUDENT REGISTRATION FORM

Student Name:	<input type="text"/>	Father Name:	<input type="text"/>
DOB:	<input type="text" value="1"/> <input type="text" value="Jan"/> <input type="text" value="1995"/>	Gender:	<input type="radio"/> Male <input type="radio"/> Female
Mobile No:	<input type="text"/>	Address:	<input type="text"/>
Mail Id:	<input type="text"/>		
Password:	<input type="text"/>		
Courses:	<input type="text" value="ME/M Tech."/>	Branch:	<input type="text" value="Computer Science and Engineering in AI a..."/>
Year	<input type="radio"/> First <input type="radio"/> Second <input type="radio"/> Third <input type="radio"/> Forth	Semester:	<input type="text" value="1"/>
		First Year Gpa:	<input type="text"/>

Register

Reset

After all the details are entered correctly ,then on clicking on the Register button a dialogue box pops up showing that the data is successfully saved ,then we proceed by click the ok button of the dialogue box.

ANIL NEERUKONDA INSTITUTE OF TECHNOLOGY & SCIENCES

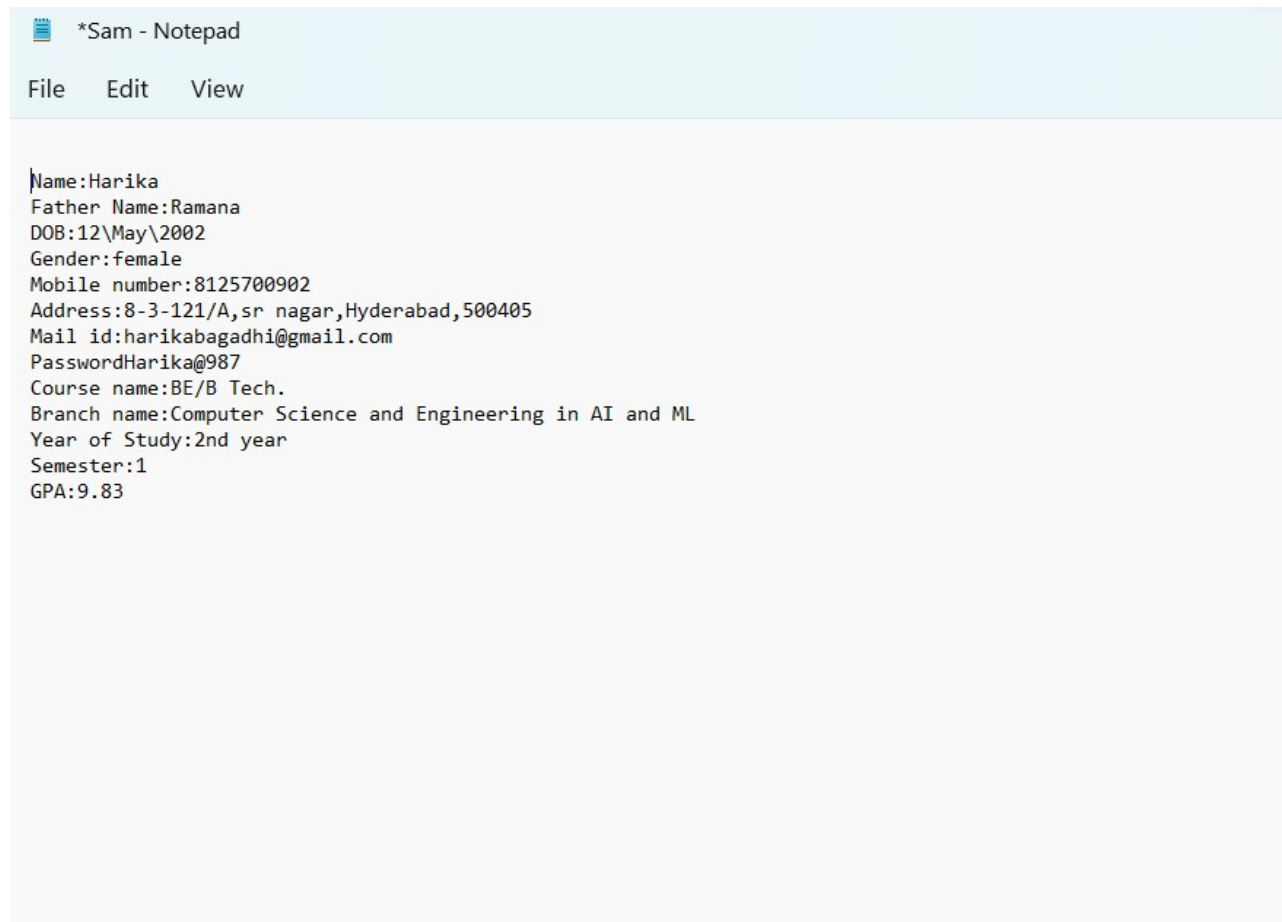
STUDENT REGISTRATION FORM

Student Name:	<input type="text" value="Harika"/>	Father Name:	<input type="text" value="Ramana"/>
DOB:	<input type="text" value="12"/> <input type="text" value="May"/> <input type="text" value="2002"/>	Gender:	<input type="radio"/> Male <input checked="" type="radio"/> Female
Mobile No:	<input type="text" value="8125700902"/>	Address:	<input type="text" value="8-3-121/A,sr nagar,Hyderabad,500405"/>
Mail Id:	<input type="text" value="harikabagadhi@gmail.com"/>		
Password:	<input type="text" value="*****"/>		
Courses:	<input type="text" value="BE/B Tech."/>	Branch:	<input type="text" value="Computer Science and Engineering in AI a..."/>
Year	<input type="radio"/> First <input checked="" type="radio"/> Second <input type="radio"/> Third <input type="radio"/> Forth	Semester:	<input type="text" value="1"/>
		First Year Gpa:	<input type="text" value="9.83"/>

Register

Reset

As a last step, we can view the details entered in the form in a file called Sam.txt. Thus the no of users visiting the form and getting registered is shown in the specified file as below.



```
*Sam - Notepad
File Edit View

Name:Harika
Father Name:Ramana
DOB:12\May\2002
Gender:female
Mobile number:8125700902
Address:8-3-121/A, sr nagar, Hyderabad, 500405
Mail id:harikabagadhi@gmail.com
PasswordHarika@987
Course name:BE/B Tech.
Branch name:Computer Science and Engineering in AI and ML
Year of Study:2nd year
Semester:1
GPA:9.83
```

*-----***THE END***-----*