



**JUBAIL UNIVERSITY COLLEGE**

**DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING**

**SEMESTER 431**

**GROUP ASSIGNMENT (6%)**

<b>Course Code</b>	CS231	<b>Course Title</b>	Programming II	<b>Section</b>	203
<b>Date Given</b>	23-11-2021	<b>Date of Submission</b>	<b>Tuesday 14-12-2021 ( at 11:59 pm)</b>		

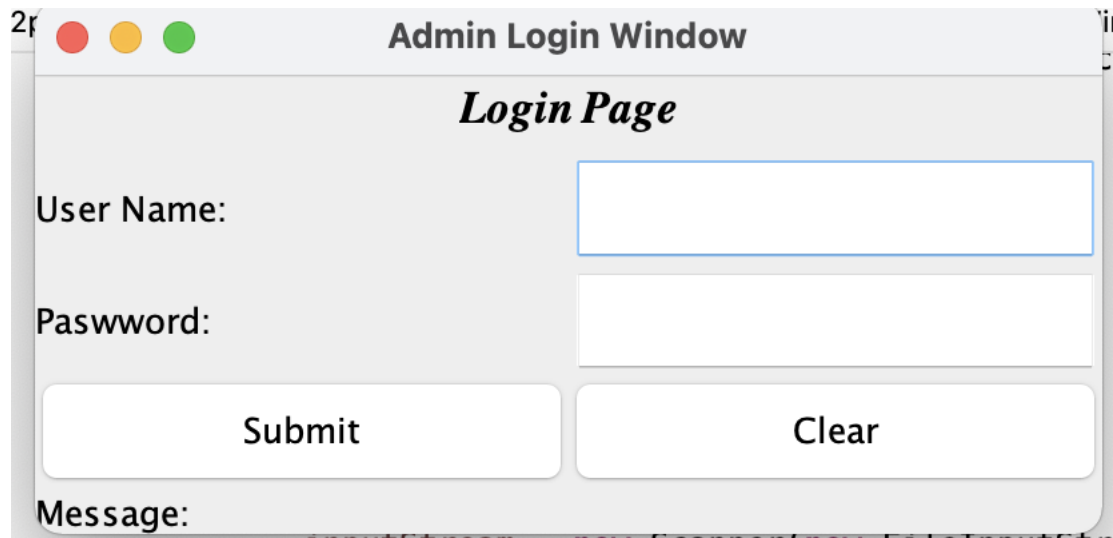
<b>PART I</b>	<b>TO BE FILLED BY THE STUDENT</b>		
<b>STUDENT'S NAME</b>	Raneem Alshehri	<b>ID. No.</b>	401200244
<b>STUDENT'S NAME</b>	Johrah Khabti	<b>ID. No.</b>	401200096
<b>STUDENT'S NAME</b>	Manar Alali	<b>ID. No.</b>	421200173

**TO BE FILLED BY THE CONCERNED DEPARTMENT**

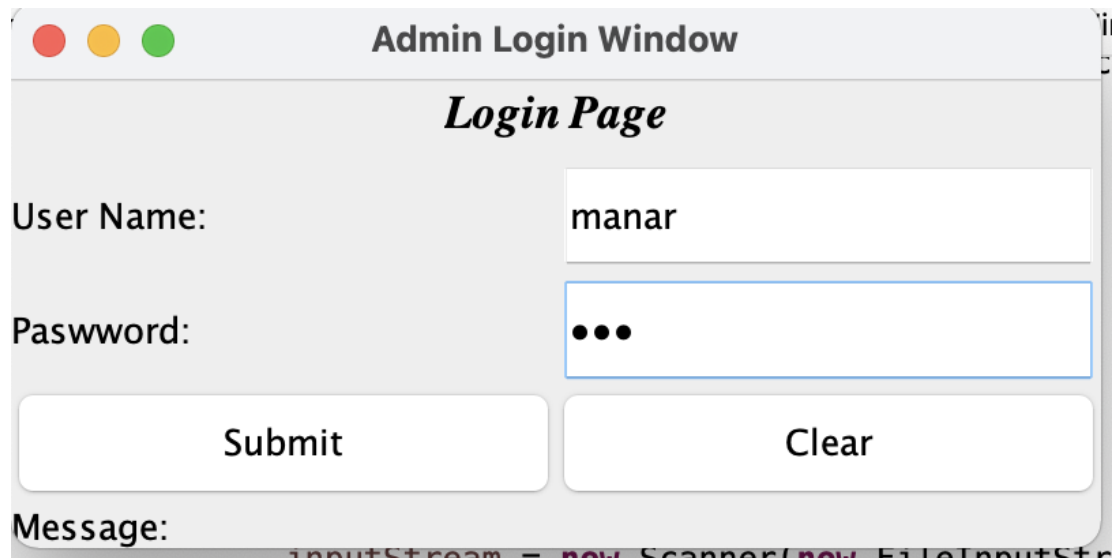
<b>PART II</b>		<b>1<sup>st</sup> Marker</b>	
<b>Area of Assessment</b>	<b>Max Marks</b>	<b>Actual Marks</b>	<b>Comments/Remarks</b>
<b>Total</b>	<b>100</b>		
<b>Percentage</b>	<b>6%</b>		
<b>Name:</b>			
<b>Signature:</b>			

This is a project which models a real-life Student Information System (SIS), though less complex. This paper will follow a step-by-step documentation of all the windows and interfaces made, as well as the files that were written and read to and from the program.

1. A login window was made for the admin to enter the System using their username and password

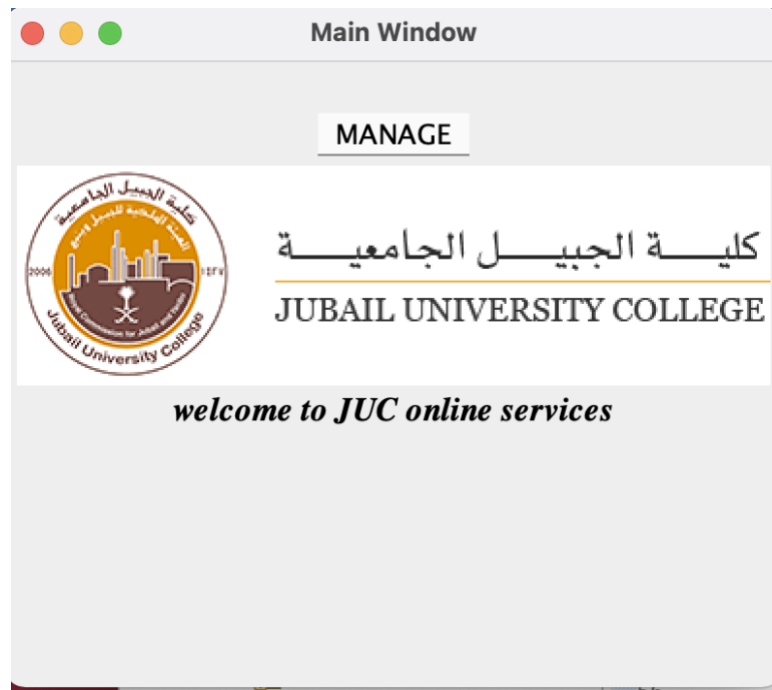


A screenshot of a Java Swing window titled "Admin Login Window". The window has a standard macOS-style title bar with red, yellow, and green buttons. The content area has a light gray background and is titled "Login Page" in a bold, italicized font. It contains two labels, "User Name:" and "Paswword:", each followed by a white text input field. Below the input fields are two white buttons labeled "Submit" and "Clear". At the bottom, there is a label "Message:" followed by a text area that is partially obscured by a code snippet: `InputStream = new Scanner(new File(InputStr`.

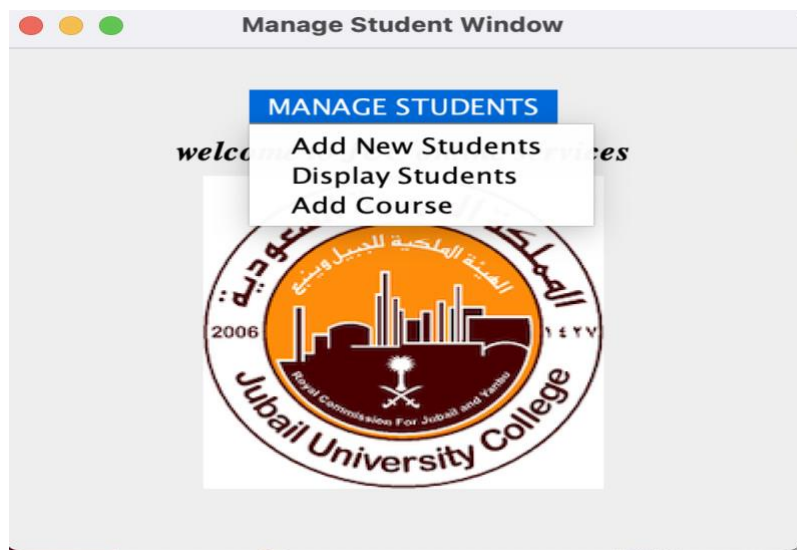
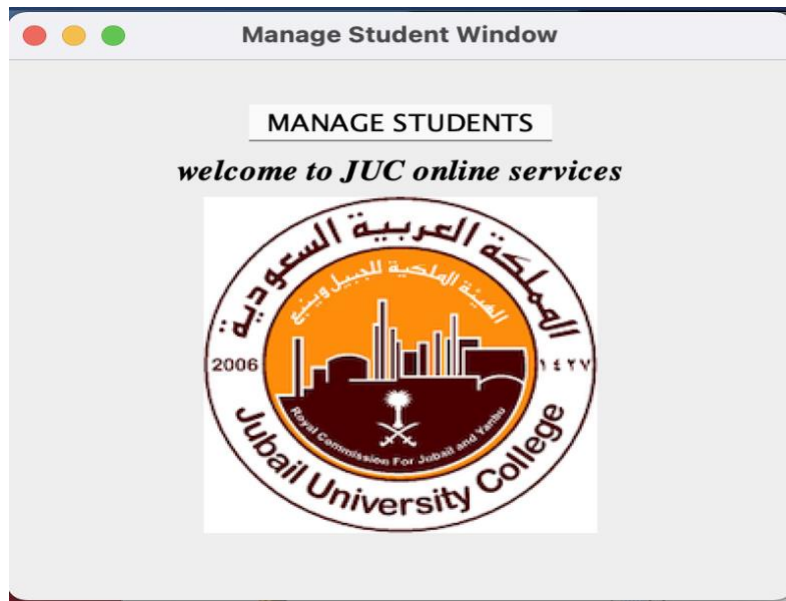


A second screenshot of the "Admin Login Window" showing the same interface as the first, but with the input fields filled. The "User Name:" field contains the text "manar". The "Paswword:" field contains three black dots, indicating a password. The "Submit" and "Clear" buttons remain visible below the fields. The "Message:" label and the partially visible code snippet are also present at the bottom.

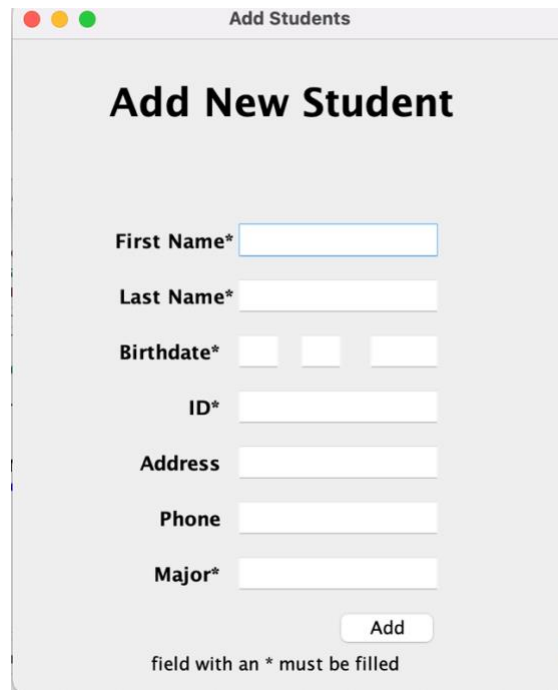
2. After the Submit button is clicked, and data is valid. The main window will pop, it has a welcome message with the JUC icon. And a menu called Manage, containing 2 menu items: Students and Courses.



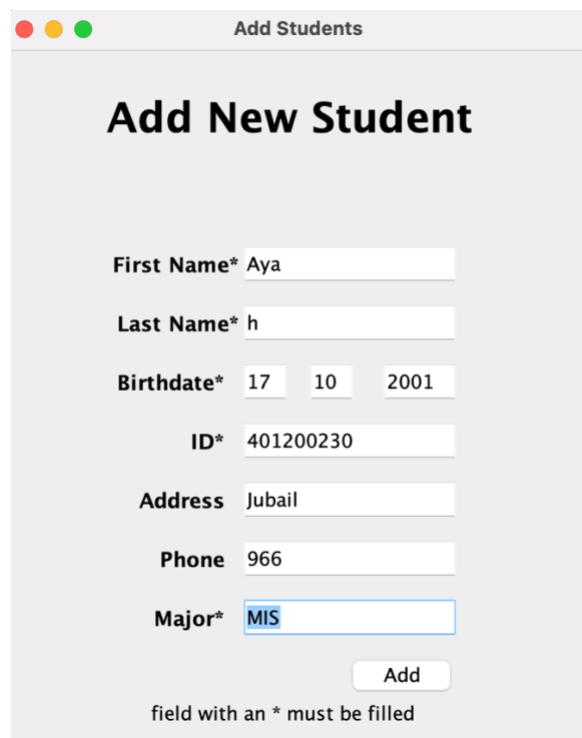
3. When the Students option is clicked, it'll take you to a Mange Students Window that has 3 functions: Add students, Display students, and Add Courses. Depending on what is chosen a new window will pop. But all these options are interconnected to each other and related to the Manage Courses menu.



4. If Add New Students is clicked, then it will redirect you to another window. It has fields with information like First Name, ID, etc..., If the field is labeled with an asterisk (\*), then it is a must to fill it, otherwise a warning message will be seen.

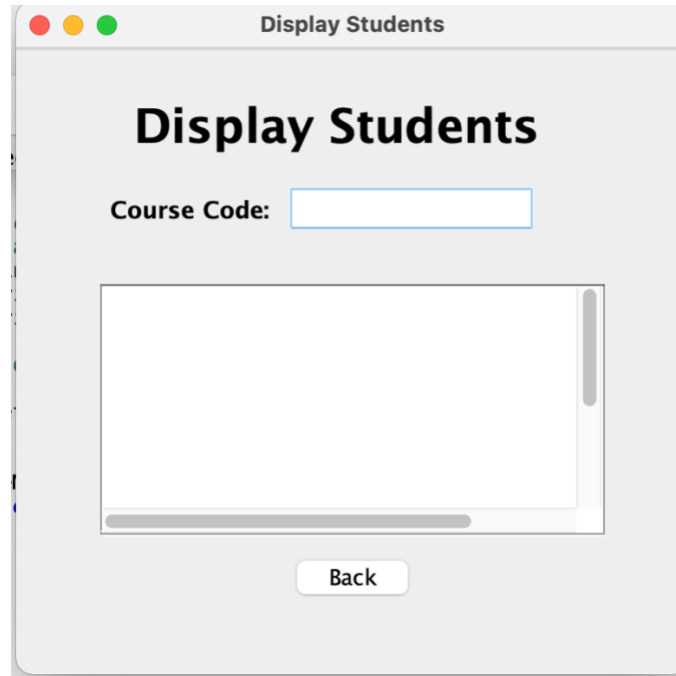


A screenshot of a web application window titled "Add Students". The main heading is "Add New Student". Below the heading are several input fields: "First Name\*", "Last Name\*", "Birthdate\*" (with three separate boxes for day, month, and year), "ID\*", "Address", "Phone", and "Major\*". Each field with an asterisk is followed by a small red asterisk icon. At the bottom right is an "Add" button. At the bottom left, a message reads "field with an \* must be filled".

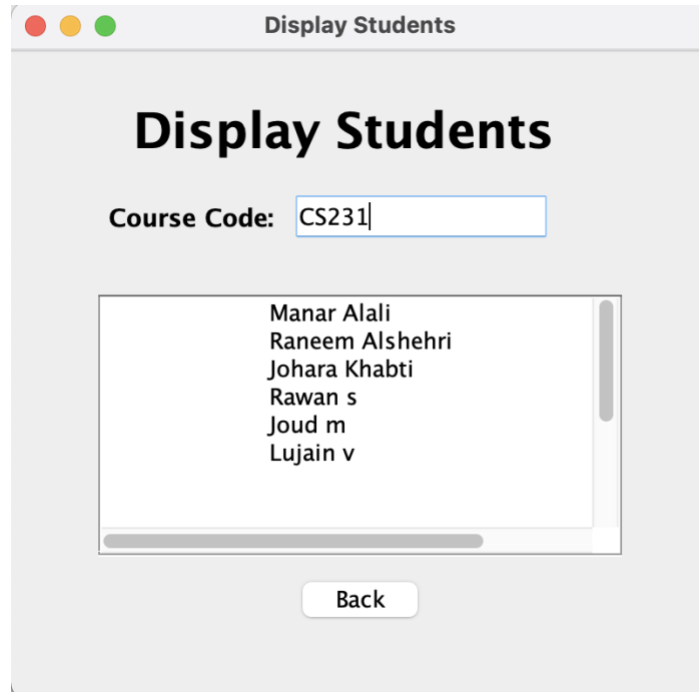


A screenshot of the same "Add New Student" form, but now with data entered into the fields. The "First Name\*" field contains "Aya", "Last Name\*" contains "h", "Birthdate\*" contains "17", "10", and "2001", "ID\*" contains "401200230", "Address" contains "Jubail", "Phone" contains "966", and "Major\*" contains "MIS". The "Add" button is still present at the bottom right, and the message "field with an \* must be filled" is still at the bottom left.

5. If Display Students is clicked, then a new Window will be seen. It needs just the Course code entered to list all the students entered in a certain course.

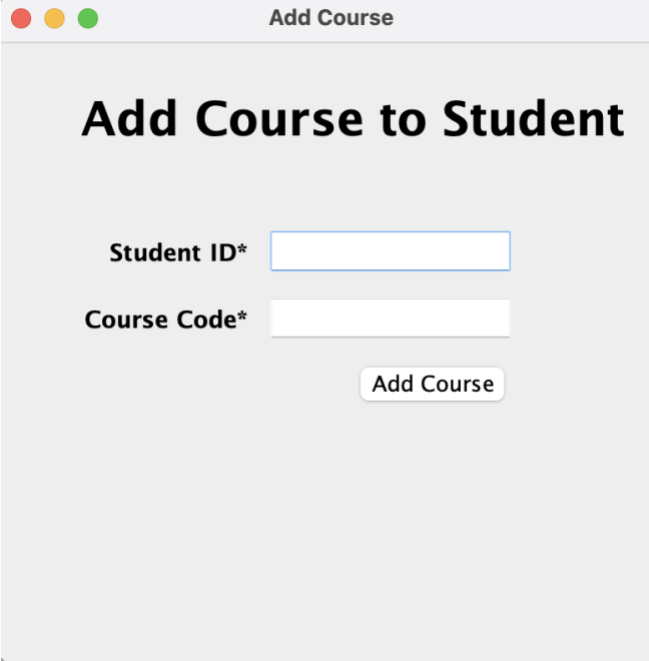


The screenshot shows a window titled "Display Students". It has a title bar with three colored buttons (red, yellow, green). The main content area has a large heading "Display Students". Below the heading is a label "Course Code:" followed by an empty text input field. Below the input field is a large, empty rectangular area with a vertical scrollbar on the right and a horizontal scrollbar at the bottom. At the bottom center of the window is a button labeled "Back".

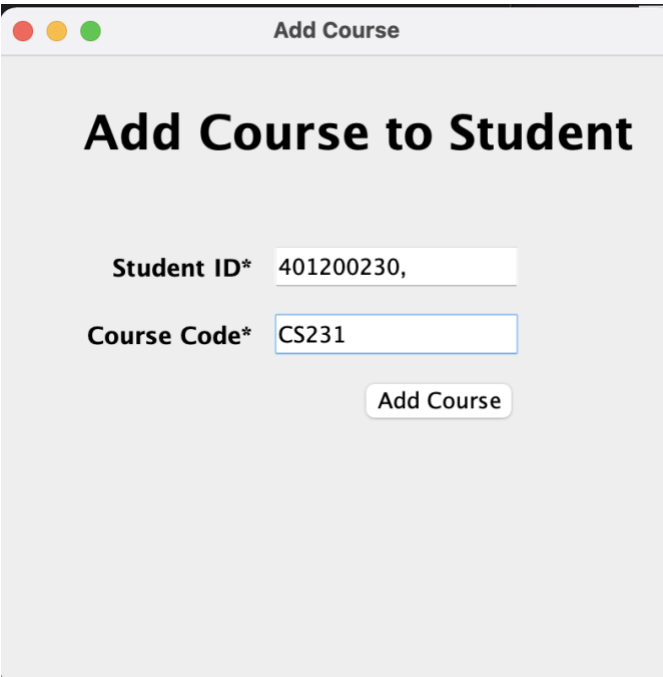


The screenshot shows the same "Display Students" window, but now the text input field contains "CS231". The large rectangular area below the input field now displays a list of student names: Manar Alali, Raneem Alshehri, Johara Khabti, Rawan s, Joud m, and Lujain v. The "Back" button remains at the bottom center.

- When the Add Course option is chosen, then the ID of the student must be entered with the course code they will study, to add the course to their system.

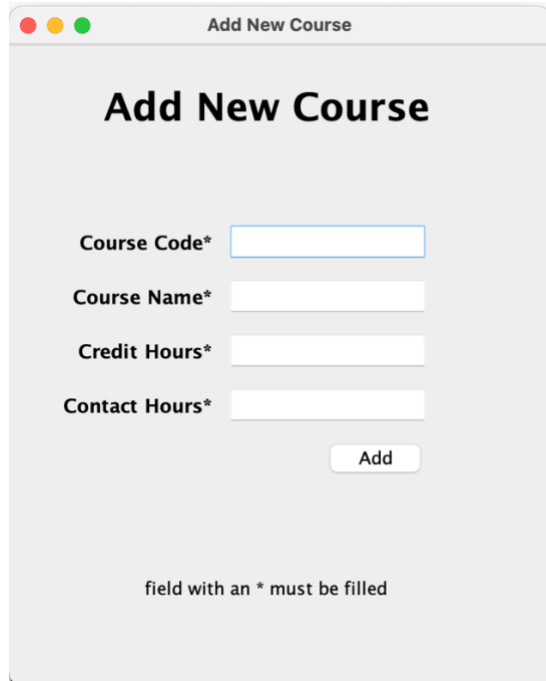


A screenshot of a macOS-style dialog box titled "Add Course". The dialog has a light gray background and a title bar with three colored window control buttons (red, yellow, green) on the left. The main heading "Add Course to Student" is centered in a large, bold, black font. Below the heading, there are two input fields. The first is labeled "Student ID\*" and is empty. The second is labeled "Course Code\*" and is also empty. Below these fields is a rounded rectangular button labeled "Add Course".

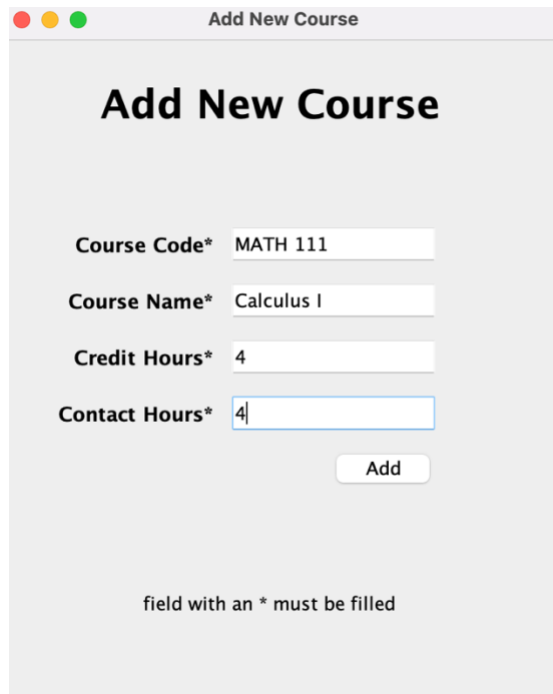


A second screenshot of the same "Add Course" dialog box. In this version, the "Student ID\*" field contains the text "401200230," and the "Course Code\*" field contains the text "CS231". The "Add Course" button remains at the bottom.

7. This concludes the Manage Student Window, and now a further look on Manage Courses will be done. The first option in the menu: Add new Courses. This has 4 fields that create a new course in the system. And could be used in assigning each student their course.



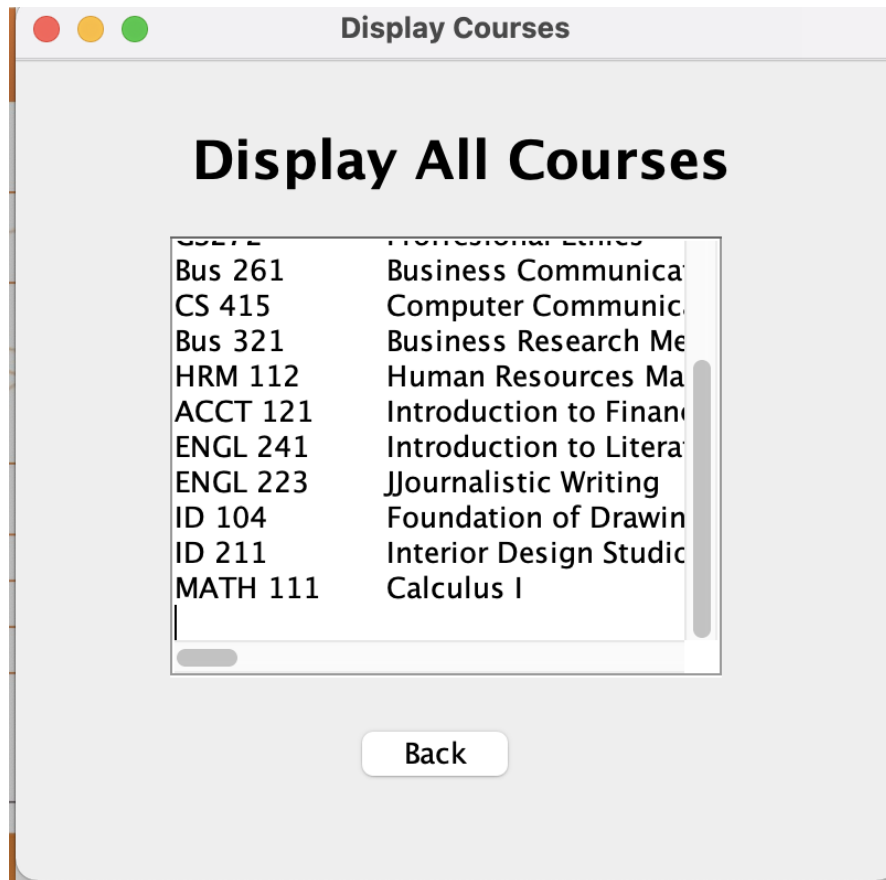
The screenshot shows a window titled "Add New Course" with a light gray background. At the top, there are three colored window control buttons (red, yellow, green) and the title "Add New Course". Below the title is a large heading "Add New Course". Underneath, there are four labeled input fields: "Course Code\*", "Course Name\*", "Credit Hours\*", and "Contact Hours\*". Each field is currently empty. To the right of the "Contact Hours\*" field is a button labeled "Add". At the bottom of the window, there is a small text label that reads "field with an \* must be filled".



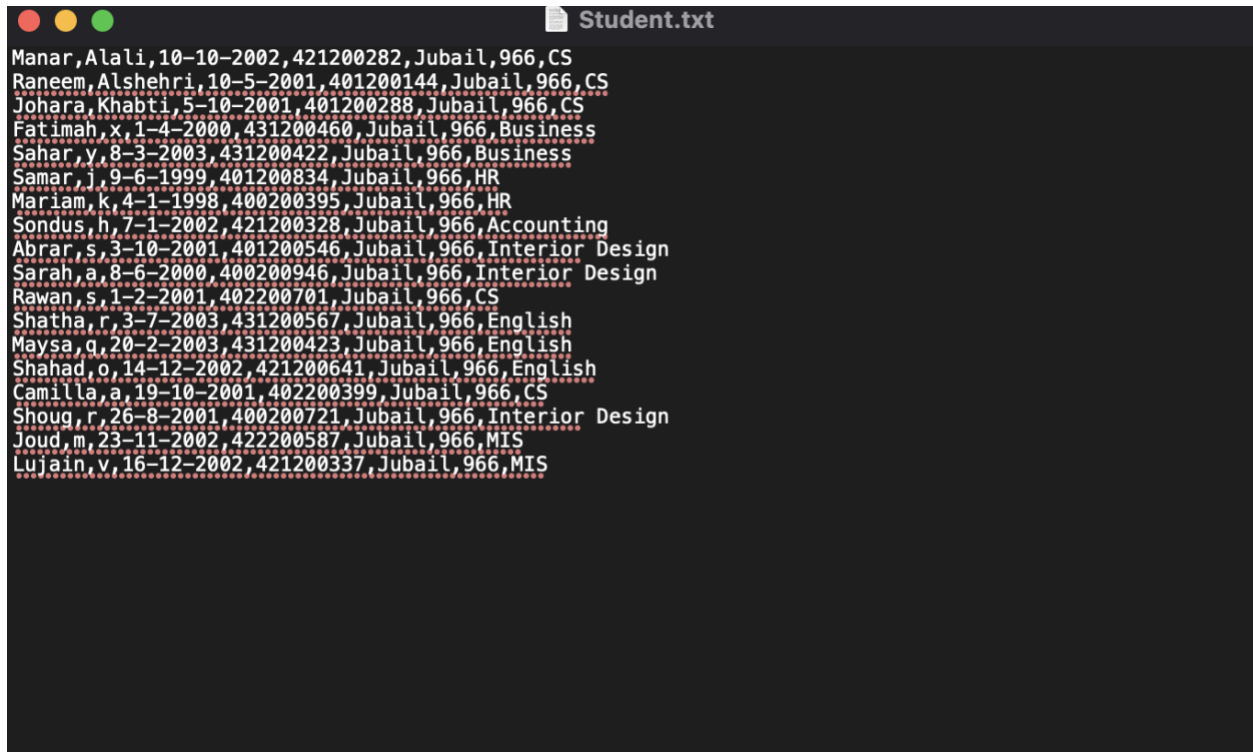
The screenshot shows the same "Add New Course" window, but now the input fields are filled with data. The "Course Code\*" field contains "MATH 111", the "Course Name\*" field contains "Calculus I", the "Credit Hours\*" field contains "4", and the "Contact Hours\*" field contains "4". The "Add" button remains to the right of the "Contact Hours\*" field. The text label at the bottom still reads "field with an \* must be filled".



8. The Display Courses option displays all courses added to the system in a JTextArea with both vertical and horizontal scrollbars.



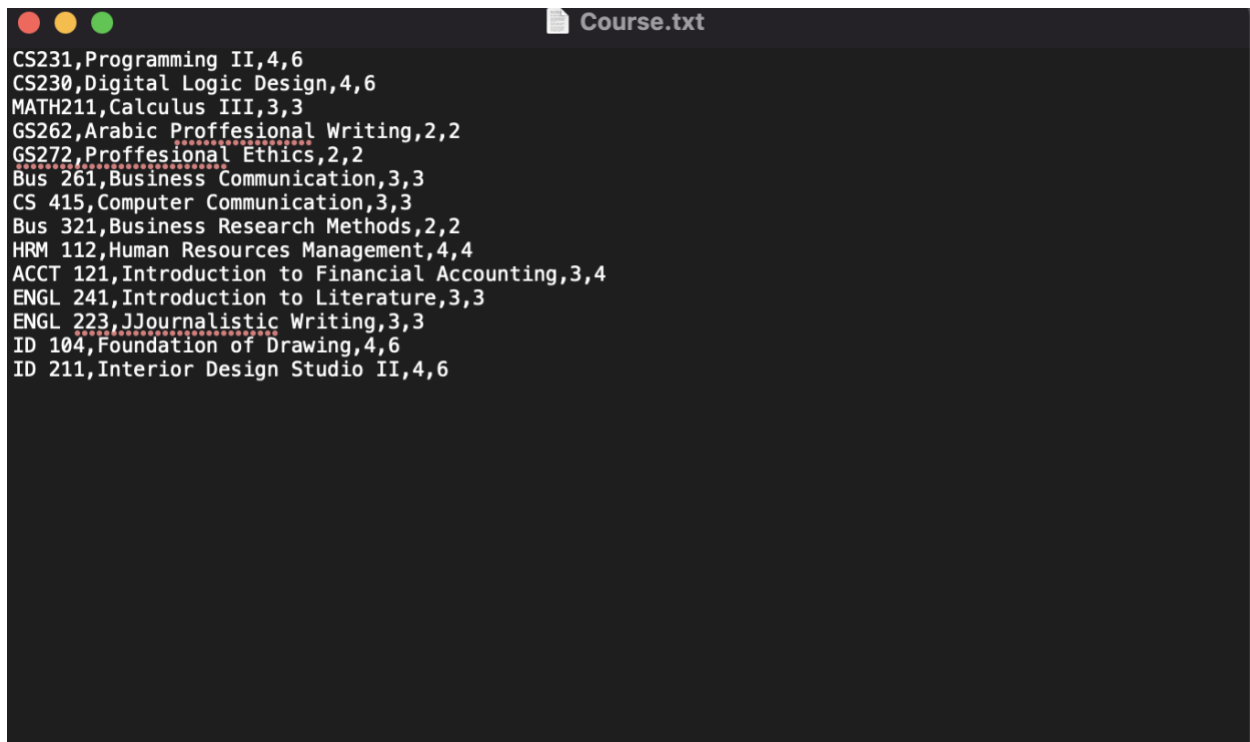
9. The “Student.txt” is the file that is being written on by the program. It takes in information about the student and stores it in the file. It is later read and displays all the students



The image shows a screenshot of a text file named "Student.txt". The file contains a list of student records, each consisting of a name, date of birth, ID number, location, and major. The records are as follows:

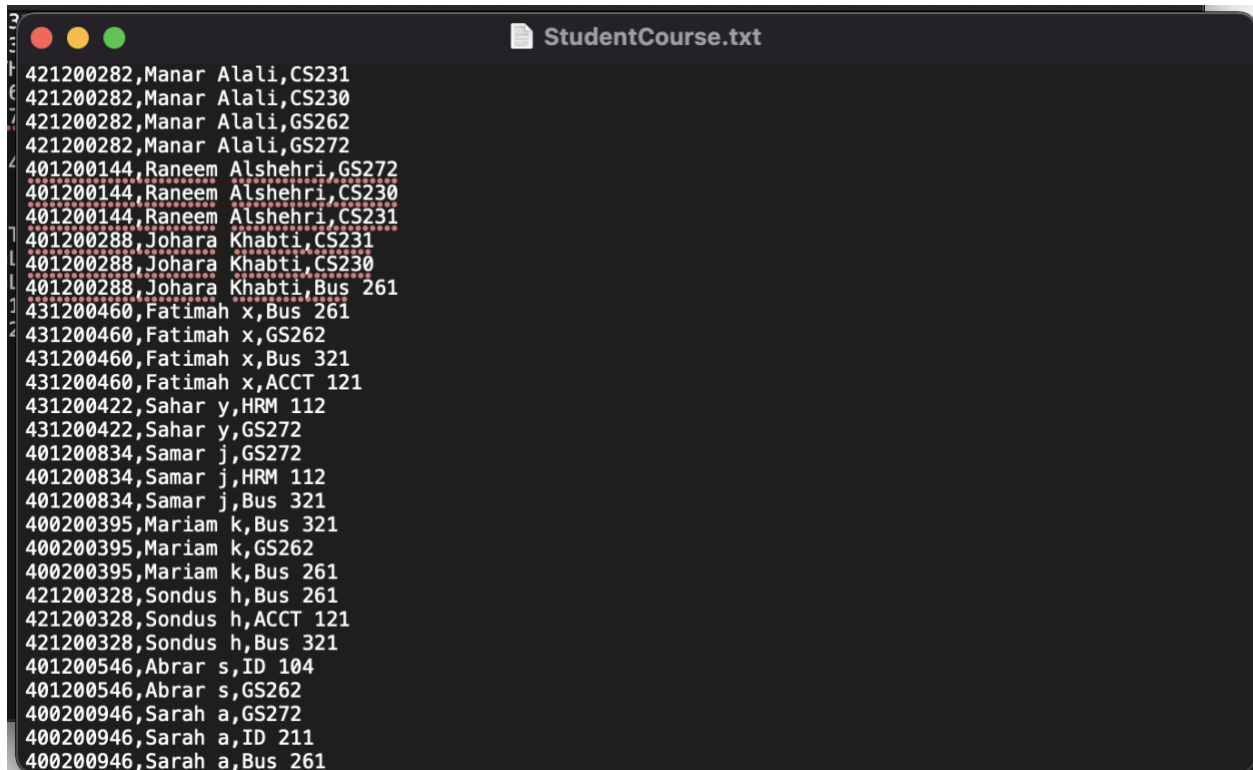
Name	Date of Birth	ID Number	Location	Major
Manar, Alali	10-10-2002	421200282	Jubail, 966	CS
Raneem, Alshehri	10-5-2001	401200144	Jubail, 966	CS
Johara, Khabti	5-10-2001	401200288	Jubail, 966	CS
Fatimah, x	1-4-2000	431200460	Jubail, 966	Business
Sahar, y	8-3-2003	431200422	Jubail, 966	Business
Samar, j	9-6-1999	401200834	Jubail, 966	HR
Mariam, k	4-1-1998	400200395	Jubail, 966	HR
Sondus, h	7-1-2002	421200328	Jubail, 966	Accounting
Abrar, s	3-10-2001	401200546	Jubail, 966	Interior Design
Sarah, a	8-6-2000	400200946	Jubail, 966	Interior Design
Rawan, s	1-2-2001	402200701	Jubail, 966	CS
Shatha, r	3-7-2003	431200567	Jubail, 966	English
Maysa, q	20-2-2003	431200423	Jubail, 966	English
Shahad, o	14-12-2002	421200641	Jubail, 966	English
Camilla, a	19-10-2001	402200399	Jubail, 966	CS
Shoug, r	26-8-2001	400200721	Jubail, 966	Interior Design
Joud, m	23-11-2002	422200587	Jubail, 966	MIS
Lujain, v	16-12-2002	421200337	Jubail, 966	MIS

10. "Course.txt", is another file that stores information that is valuable for the program. It is first written to and holds the data for all the courses. Then, it reads from the file and displays the data.



```
CS231,Programming II,4,6
CS230,Digital Logic Design,4,6
MATH211,Calculus III,3,3
GS262,Arabic Proffesional Writing,2,2
GS272,Proffesional Ethics,2,2
Bus 261,Business Communication,3,3
CS 415,Computer Communication,3,3
Bus 321,Business Research Methods,2,2
HRM 112,Human Resources Management,4,4
ACCT 121,Introduction to Financial Accounting,3,4
ENGL 241,Introduction to Literature,3,3
ENGL 223,JJournalistic Writing,3,3
ID 104,Foundation of Drawing,4,6
ID 211,Interior Design Studio II,4,6
```

11. The “StudentCourse.txt” file, combines both the above-mentioned files. It creates a file that is useful for the process of adding a course for a specific student.



```
StudentCourse.txt
421200282,Manar Alali,CS231
421200282,Manar Alali,CS230
421200282,Manar Alali,GS262
421200282,Manar Alali,GS272
401200144,Raneem Alshehri,GS272
401200144,Raneem Alshehri,CS230
401200144,Raneem Alshehri,CS231
401200288,Johara Khabti,CS231
401200288,Johara Khabti,CS230
401200288,Johara Khabti,Bus 261
431200460,Fatimah x,Bus 261
431200460,Fatimah x,GS262
431200460,Fatimah x,Bus 321
431200460,Fatimah x,ACCT 121
431200422,Sahar y,HRM 112
431200422,Sahar y,GS272
401200834,Samar j,GS272
401200834,Samar j,HRM 112
401200834,Samar j,Bus 321
400200395,Mariam k,Bus 321
400200395,Mariam k,GS262
400200395,Mariam k,Bus 261
421200328,Sondus h,Bus 261
421200328,Sondus h,ACCT 121
421200328,Sondus h,Bus 321
401200546,Abrar s,ID 104
401200546,Abrar s,GS262
400200946,Sarah a,GS272
400200946,Sarah a,ID 211
400200946,Sarah a,Bus 261
```

## 12.Text of the code:

- **//Making the code visible:**

- **package** java2project;
- 
- **public class** java2project {
- 
- **public static void** main(String[] args) {
- 
- **// instantiating the window**
- **adminLoginWindow** admin\_Login\_Window= **new**
- adminLoginWindow();
- **//** **mainWindow** main\_Window=new mainWindow();
- 
- **admin\_Login\_Window.setVisible(true);**
- **//** **main\_Window.setVisible(true);**
- 
- **}**
- 
- **}**

--//-- --//-- --//-- --//-- --//-- --//-- --//-- --//-- --//--

- **//Main Window:**

- **package** java2project;
- 
- **import** java.awt.\*;
- **import** java.awt.event.ActionEvent;
- **import** java.awt.event.ActionListener;
- 
- 
- **import** javax.swing.\*;
- 
- 
- **public class** mainWindow **extends** JFrame **implements**  
ActionListener {
- 
- **public** mainWindow() {
- 
- **//window 2 that has 2 options for students and courses**
- 
- **this.setTitle("Main Window");**
- **this.setSize(400,350);**
- 
- **this.setDefaultCloseOperation(*HIDE\_ON\_CLOSE*);**

- `this.setLayout(new FlowLayout());`
- 
- 
- `// menu`
- `JMenu Manage= new JMenu("MANAGE");`
- `this.add(Manage);`
- 
- `//menu item`
- `JMenuItem Manage_Students_Window =`  
`new JMenuItem("Students");`
- 
- `Manage_Students_Window.addActionListener(this);`
- `Manage.add(Manage_Students_Window);`
- 
- `JMenuItem Manage_Courses_Window=`  
`new JMenuItem("Courses");`
- 
- `Manage_Courses_Window.addActionListener(this);`
- `Manage.add(Manage_Courses_Window);`
- 
- 
- `//menu bar`
- `JMenuBar menuBar= new JMenuBar();`
- 
- `this.setJMenuBar(menuBar);`

- `this.add(menuBar);`
- `menuBar.add(Manage);`
- 
- `//icon`
- 
- `ImageIcon JUC_icon= new`
- `ImageIcon("Unknown.png");`
- `JLabel JUC_label=new JLabel(JUC_icon);`
- `this.add(JUC_label);`
- 
- 
- 
- `//message`
- `JLabel JUC_text=new JLabel("welcome to`
- `JUC online services");`
- `this.add(JUC_text);`
- 
- 
- 
- `//font`
- `JUC_text.setFont (new Font`
- `("TimesRoman", Font.BOLD | Font.ITALIC, 17));`
- 
- `this.setVisible(true);`
-



- }
- 
- @Override
- **public void** actionPerformed(ActionEvent e) {
- // TODO Auto-generated method stub
- String actioncommand= e.getActionCommand();
- System.*out*.println(actioncommand);
- 
- **if**(actioncommand.equals("Students")){
- ManageStudent Manage\_Student= **new**
- ManageStudent();
- Manage\_Student.setVisible(**true**);
- }
- **if**(actioncommand.equals("Courses")){
- ManageCourse Manage\_Course= **new**
- ManageCourse();
- Manage\_Course.setVisible(**true**);
- 
- }
- 
- }
- 
- }
- 
- }

--//-- --//-- --//-- --//-- --//-- --//-- --//-- --//-- --//--

## • Manage Students:

- **package** java2project;
- 
- **import** java.awt.\*;
- **import** java.awt.event.ActionEvent;
- **import** java.awt.event.ActionListener;
- **import** javax.swing.\*;
- 
- 
- 
- **public class** ManageStudent **extends** JFrame **implements** ActionListener{
- 
- **public** ManageStudent(){
- 
- 
- 
- **this**.setTitle("Manage Student Window");
- **this**.setSize(400,350);
- **this**.setDefaultCloseOperation(*HIDE\_ON\_CLOSE*);
- **this**.setLayout(**new** BorderLayout() );
-

- `// menu`
- `JMenu Manage= new JMenu("MANAGE STUDENTS");`
- `this.add(Manage);`
- 
- `//menu item`
- `JMenuItem Add_Students = new JMenuItem("Add New Students");`
- `Add_Students.addActionListener(this);`
- `JMenuItem Display_Students= new JMenuItem("Display`  
`Students");`
- `Display_Students.addActionListener(this);`
- `JMenuItem Add_Course= new JMenuItem("Add Course");`
- `Add_Course.addActionListener(this);`
- 
- `Manage.add(Add_Students);`
- `Manage.add(Display_Students);`
- `Manage.add(Add_Course);`
- 
- 
- `//menu bar`
- `JMenuBar menuBar= new JMenuBar();`
- 
- `this.setJMenuBar(menuBar);`
- `// this.add(menuBar);`
- `menuBar.add(Manage);`
- 
- 
- `//icon`

- 
- Imagelcon JUC\_icon= **new** Imagelcon("juc.png");
- JLabel JUC\_label=**new** JLabel(JUC\_icon);
- //this.add(JUC\_label);
- 
- 
- 
- 
- //message
- JLabel JUC\_text=**new** JLabel("welcome to JUC online services");
- //this.add(JUC\_text);
- 
- 
- 
- //font
- JUC\_text.setFont (**new** Font ("TimesRoman", Font.**BOLD** |
- Font.**ITALIC**, 17));
- 
- 
- 
- //panel
- JPanel Panel1 = **new** JPanel(**new** FlowLayout());
- Panel1.add(menuBar);
- JPanel Panel2 = **new** JPanel(**new** FlowLayout());
- Panel2.add(JUC\_text);
- Panel2.add(JUC\_label);
- 
- //adding to the frame

- `this.add(Panel1,BorderLayout.NORTH);`
- `this.add(Panel2,BorderLayout.CENTER);`
- 
- 
- `}`
- 
- `@Override`
- `public void actionPerformed(ActionEvent e) {`
- `// TODO Auto-generated method stub`
- 
- 
- 
- `String actioncommand= e.getActionCommand();`
- 
- 
- `if(actioncommand.equals("Add New Students")){`
- `System.out.println("Manage");`
- `AddStudents Add_Students= new AddStudents();`
- `Add_Students.setVisible(true);`
- `}`
- `if(actioncommand.equals("Display Students")){`
- `DisplayStudents Display_Students= new DisplayStudents();`
- `Display_Students.setVisible(true);`
- `}`
- 
- 
- `if(actioncommand.equals("Add Course")){`
- `AddCourse Add_Course= new AddCourse();`
- `Add_Course.setVisible(true);`

- }
- }
- 
- }

--//-- --//-- --//-- --//-- --//-- --//-- --//-- --//-- --//--

## • Manage Courses:

- **package** java2project;
- 
- **import** java.awt.\*;
- **import** java.awt.event.ActionEvent;
- **import** java.awt.event.ActionListener;
- **import** javax.swing.\*;
- 
- 
- 
- **public class** ManageCourse **extends** JFrame **implements** ActionListener{
- 
- 
- **public** ManageCourse() {
- 
- **this**.setTitle("Manage Course Window");
- **this**.setSize(400,350);

- `this.setDefaultCloseOperation(HIDE_ON_CLOSE);`
- `this.setLayout(new BorderLayout());`
- 
- `// menu`
- `JMenu Manage= new JMenu("MANAGE COURSES");`
- 
- `this.add(Manage);`
- 
- `//menu item`
- `JMenuItem Add_Course = new JMenuItem("Add New Course");`
- `Add_Course.addActionListener(this);`
- `JMenuItem Display_Courses= new JMenuItem("Display Courses");`
- `Display_Courses.addActionListener(this);`
- 
- 
- `Manage.add(Add_Course);`
- `Manage.add(Display_Courses);`
- 
- 
- 
- `//menu bar`
- `JMenuBar menuBar= new JMenuBar();`
- 
- `this.setJMenuBar(menuBar);`
- `// this.add(menuBar);`
- `menuBar.add(Manage);`
-

- 
- `//icon`
- 
- `ImageIcon JUC_icon= new ImageIcon("juc.png");`
- `JLabel JUC_label=new JLabel(JUC_icon);`
- `//this.add(JUC_label);`
- 
- 
- 
- `//message`
- `JLabel JUC_text=new JLabel("welcome to JUC online services");`
- `//this.add(JUC_text);`
- 
- 
- 
- `//font`
- `JUC_text.setFont (new Font ("TimesRoman", Font.BOLD | Font.ITALIC,`
- 17));
- 
- 
- 
- `//panel`
- `JPanel Panel1 = new JPanel(new FlowLayout());`
- `Panel1.add(menuBar);`
- `JPanel Panel2 = new JPanel(new FlowLayout());`
- `Panel2.add(JUC_text);`
- `Panel2.add(JUC_label);`



- 
- //adding to the frame
- **this**.add(Panel1, BorderLayout.*NORTH*);
- **this**.add(Panel2, BorderLayout.*CENTER*);
- 
- 
- }
- 
- @Override
- **public void** actionPerformed(ActionEvent e) {
- 
- String actioncommand= e.getActionCommand();
- //System.out.println(actioncommand);
- 
- **if**(actioncommand.equals("Add New Course")){
- AddNewCourse Add\_New\_Course= **new** AddNewCourse();
- Add\_New\_Course.setVisible(**true**);
- }
- **if**(actioncommand.equals("Display Courses")){
- DisplayCourses Display\_Courses= **new** DisplayCourses();
- Display\_Courses.setVisible(**true**);
- 
- }
- }
- }
- }
- 
- --//-- --//-- --//-- --//-- --//-- --//-- --//-- --//-- --//--

- Add Student:

```
package java2project;
```

```
import java.awt.*;
```

```
import java.awt.event.ActionEvent;
```

```
import java.awt.event.ActionListener;
```

```
import java.io.*;
```

```
import javax.swing.*;
```

```
public class AddStudents extends JFrame implements ActionListener{
```

```
    JLabel info;
```

```
    JTextField text1, text2 ,dayText , monthText, yearText, text4 ,text5,text6,text7;
```

```
public AddStudents(){
```

```
    super("Add Students");
```

```
    setSize(400,500);
```

```
    this.setDefaultCloseOperation(JFrame.HIDE_ON_CLOSE);
```

```
    setLocationRelativeTo(null);
```

```
    setLayout(null);
```

```
JLabel lbl= new JLabel("Add New Student");  
Font Title_Font = new Font("Cambria", Font.BOLD, 28);  
lbl.setFont(Title_Font);  
lbl.setBounds(70,20,250,50);  
add(lbl);
```

```
Font Label_Font = new Font("Cambria", Font.BOLD, 14);
```

```
info= new JLabel("");  
info.setForeground(Color.RED);  
info.setFont(Label_Font);  
info.setBounds(50,80,320,50);  
add(info);
```

```
//First name * , Last name*, Birthdate*, ID*,address, major, and Phone
```

```
JLabel l1= new JLabel("First Name*",SwingConstants.RIGHT);  
l1.setFont(Label_Font);  
l1.setBounds(70,120,90,50);  
text1 = new JTextField(10);  
text1.setBounds(160,130,150,30);
```

```
JLabel l2= new JLabel("Last Name*",SwingConstants.RIGHT);  
l2.setFont(Label_Font);
```

```
l2.setBounds(70,160,90,50);  
text2 = new JTextField(10);  
text2.setBounds(160,170,150,30);
```

```
JLabel l3= new JLabel("Birthdate*",SwingConstants.RIGHT);  
l3.setFont(Label_Font);  
l3.setBounds(70,200,80,50);  
dayText = new JTextField(4);  
dayText.setBounds(160,210,35,30);  
monthText = new JTextField(4);  
monthText.setBounds(205,210,35,30);  
yearText = new JTextField(4);  
yearText.setBounds(255,210,55,30);
```

```
JLabel l4= new JLabel("ID*",SwingConstants.RIGHT);  
l4.setFont(Label_Font);  
l4.setBounds(70,240,80,50);  
text4 = new JTextField(10);  
text4.setBounds(160,250,150,30);
```

```
JLabel l5= new JLabel("Address",SwingConstants.RIGHT);  
l5.setFont(Label_Font);  
l5.setBounds(70,280,80,50);  
text5 = new JTextField(10);  
text5.setBounds(160,290,150,30);
```

```
JLabel l6= new JLabel("Phone",SwingConstants.RIGHT);  
l6.setFont(Label_Font);  
l6.setBounds(70,320,80,50);  
text6 = new JTextField(10);  
text6.setBounds(160,330,150,30);
```

```
JLabel l7= new JLabel("Major*",SwingConstants.RIGHT);  
l7.setFont(Label_Font);  
l7.setBounds(70,360,80,50);  
text7 = new JTextField(10);  
text7.setBounds(160,370,150,30);
```

```
add(l1);  
add(text1);  
add(l2);  
add(text2);  
add(l3);  
add(dayText);  
add(monthText);  
add(yearText);  
add(l4);
```

```
add(text4);
add(l5);
add(text5);
add(l6);
add(text6);
add(l7);
add(text7);
```

```
JButton add_btn = new JButton("Add");
add_btn.addActionListener(this);
add_btn.setBounds(230,410,80,30);
```

```
add(add_btn);
```

```
JLabel msg2 = new JLabel("field with an * must be filled");
msg2.setBounds(100, 400, 250, 100);
add(msg2);
```

```
}
```

```
public void actionPerformed(ActionEvent e){
    // read all data from gui
    String fname = text1.getText().trim();
    String lname = text2.getText().trim();
    String day = dayText.getText().trim();
```

```
String month = monthText.getText().trim();
```

```
String year = yearText.getText().trim();
```

```
String id = text4.getText().trim();
```

```
String address = text5.getText().trim();
```

```
String phone = text6.getText().trim();
```

```
String maj = text7.getText().trim();
```

```
String birth= "";
```

```
if (fname.equals("") || lname.equals("") || id.equals("") || day.equals("") ||  
month.equals("") || year.equals("") || maj.equals("")){  
    info.setText("Warning! Information is incomplete, data is required to add  
student");  
}  
else{
```

```
    int dayNo = Integer.parseInt(day);
```

```
    int monthNo =Integer.parseInt( month);
```

```
    int yearNo = Integer.parseInt(year);
```

```
    boolean d = true , m = true , y=true;
```

```
    if(dayNo<1||dayNo>31)
```

```
        d = false;
```

```
    if (monthNo<1 || monthNo>12)
```

```
        m = false;
    if(yearNo < 1970 || yearNo >2005)
        y = false;

    if(y && m && d){
        birth = dayNo+"-"+monthNo+"-"+yearNo;

        PrintWriter outPutStream = null;
        try
        {
            outPutStream = new PrintWriter(new
FileOutputStream("Student.txt",true));

            outPutStream.println(fname+","+lname+","+birth+","+id+","+address+","+
phone+","+maj);

            outPutStream.close();
            info.setText("Student added successfully..");
        }
        catch(FileNotFoundException ex)
        {
            System.out.println("Error opening the file student.txt "+
ex.getMessage());
            info.setText("File Student does not exist..");
        }
    }
```



```
}
```

```
else
```

```
info.setText("Not Correct date..");
```

```
}
```

```
}
```

```
}
```

```
--//-- --//-- --//-- --//-- --//-- --//-- --//-- --//-- --//--
```

- Add New Course:

```
package java2project;
```

```
import java.awt.*;
```

```
import java.awt.event.ActionEvent;
```

```
import java.awt.event.ActionListener;
```

```
import java.io.*;
```

```
import javax.swing.*;
```

```
public class AddNewCourse extends JFrame implements ActionListener{
```

```
    JLabel message;
```

```
    JTextField text1, text2, text3 , text4;
```

```
    public AddNewCourse(){
```

```
        super("Add New Course");
```

```
        setSize(400,500);
```

```
        this.setDefaultCloseOperation(JFrame.HIDE_ON_CLOSE);
```

```
        setLocationRelativeTo(null);
```

```
        setLayout(null);
```

```
        JLabel lbl= new JLabel("Add New Course");
```

```
        Font Title_Font = new Font("Cambria", Font.BOLD, 28);
```

```
        lbl.setFont(Title_Font);
```

```
        lbl.setBounds(70,20,300,50);
```

```
        add(lbl);
```

```
        Font Label_Font = new Font("Cambria", Font.BOLD, 14);
```

```
        message= new JLabel("");
```

```
        message.setForeground(Color.RED);
```

```
        message.setFont(Label_Font);
```

```
        message.setBounds(50,80,320,50);
```

```
        add(message);
```

```
JLabel l1= new JLabel("Course Code*",SwingConstants.RIGHT);  
l1.setFont(Label_Font);  
l1.setBounds(30,120,120,50);  
text1 = new JTextField(10);  
text1.setBounds(160,130,150,30);
```

```
JLabel l2= new JLabel("Course Name*",SwingConstants.RIGHT);  
l2.setFont(Label_Font);  
l2.setBounds(30,160,120,50);  
text2 = new JTextField(10);  
text2.setBounds(160,170,150,30);
```

```
JLabel l3= new JLabel("Credit Hours*",SwingConstants.RIGHT);  
l3.setFont(Label_Font);  
l3.setBounds(30,200,120,50);  
text3 = new JTextField(10);  
text3.setBounds(160,210,150,30);
```

```
JLabel l4= new JLabel("Contact Hours*",SwingConstants.RIGHT);  
l4.setFont(Label_Font);  
l4.setBounds(30,240,120,50);  
text4 = new JTextField(10);  
text4.setBounds(160,250,150,30);
```

```
add(l1);
add(text1);
add(l2);
add(text2);
add(l3);
add(text3);
add(l4);
add(text4);
```

```
JButton addButton = new JButton("Add");
addButton.setBounds(230,290,80,30);
addButton.addActionListener(this);
add(addButton);
```

```
JLabel msg2 = new JLabel("field with an * must be filled");
msg2.setBounds(100, 350, 250, 100);
add(msg2);
```

```
}
```

```
public void actionPerformed(ActionEvent e){
```

```
    // read all data from gui
```

```
    String code = text1.getText().trim();
```

```
    String name = text2.getText().trim();
```

```
    String creditHour = text3.getText().trim();
```

```

String contactHour = text4.getText().trim();

PrintWriter outPutStream = null;
try
{
    outPutStream = new PrintWriter(new
FileOutputStream("Course.txt",true));

    outPutStream.println(code+","+name+","+creditHour+","+contactHour);
    outPutStream.close();

    message.setText("Course added successfully!");
}
catch(FileNotFoundException ex)
{
    System.out.println("Error opening the file Course.txt "+ ex.getMessage());
    message.setText("File Course does not exist..");
}

}

}

```

--//-- --//-- --//-- --//-- --//-- --//-- --//-- --//-- --//--

- Add Course:

- `package` java2project;
- 
- `import` java.awt.\*;
- `import` java.awt.event.ActionEvent;
- `import` java.awt.event.ActionListener;
- `import` java.io.\*;
- `import` javax.swing.\*;
- 
- 
- `public class` AddCourse `extends` JFrame `implements` ActionListener{
- 
- JLabel `message`;
- JTextField `text1`,`text2`;
- 
- `public` AddCourse(){
- `super`("Add Course");
- setSize(400,400);
- `this`.setDefaultCloseOperation(JFrame.*HIDE\_ON\_CLOSE*);
- setLocationRelativeTo(`null`);
- setLayout(`null`);
- 
- 
- JLabel lbl= `new` JLabel("Add Course to Student");
- Font Title\_Font = `new` Font("Cambria", Font.*BOLD*, 28);
- lbl.setFont(Title\_Font);
- lbl.setBounds(50,20,350,50);

- add(lbl);
- 
- Font Label\_Font = new Font("Cambria", Font.**BOLD**, 14);
- 
- message= new JLabel("");
- message.setForeground(Color.**RED**);
- message.setFont(Label\_Font);
- message.setBounds(60,60,320,50);
- add(message);
- 
- JLabel l1= new JLabel("Student ID\*",SwingConstants.**RIGHT**);
- l1.setFont(Label\_Font);
- l1.setBounds(30,100,120,50);
- text1 = new JTextField(10);
- text1.setBounds(160,110,150,30);
- 
- JLabel l2= new JLabel("Course Code\*",SwingConstants.**RIGHT**);
- l2.setFont(Label\_Font);
- l2.setBounds(30,140,120,50);
- text2 = new JTextField(10);
- text2.setBounds(160,150,150,30);
- 
- add(l1);
- add(text1);
- add(l2);
- add(text2);
-

- JButton add\_btn = new JButton("Add Course");
- add\_btn.setBounds(210,190,100,30);
- add\_btn.addActionListener(this);
- add(add\_btn);
- }
- 
- public void actionPerformed(ActionEvent e){
- 
- String std\_name="";
- String std\_id = text1.getText().trim();
- String code = text2.getText().trim();
- boolean f1 = false , f2 = false;
- 
- // read from student and course files to check if already added.
- Scanner std\_Stream = null , course\_Stream = null;
- 
- try{
- std\_Stream = new Scanner(new FileInputStream("student.txt"));
- course\_Stream = new Scanner(new
- FileInputStream("Course.txt"));
- 
- //search for student is valid or not
- while(std\_Stream.hasNextLine()){
- String line = std\_Stream.nextLine();
- String[] tokens = line.split(",");
- if (tokens[3].equals(std\_id)){
- f1 = true;



- std\_name = tokens[0]+" "+tokens[1];
- break;
- } // end if
- } // end while loop
- 
- 
- //search for course is vaild or not
- while(course\_Stream.hasNextLine()){
- String line = course\_Stream.nextLine();
- String[] tokens = line.split(",");
- if (tokens[0].equals(code)){
- f2 = true;
- break;
- } // end if
- } // end while
- 
- } // end try
- catch (FileNotFoundException ex){
- System.out.println("Files not found");
- message.setText("Files are not exist.. check it");
- }
- 
- 
- 
- if (f2 && f1){
- PrintWriter outputStream = null;
- try

- {
- outStream = new PrintWriter(new
- FileOutputStream("StudentCourse.txt",true));
- outStream.println(std\_id+","+std\_name+","+code);
- outStream.close();
- message.setText("Course addedd successfully to a Student");
- }
- catch(FileNotFoundException ex)
- {
- System.out.println("Error opening the file studentCourse.txt "+
- ex.getMessage());
- message.setText("File studentCourse does not exist..");
- }
- 
- } // end if
- else
- {
- message.setText("Check your student Id and Course Code");
- } //end else
- 
- 
- 
- 
- } // end action perform
- 
- }

--//-- --//-- --//-- --//-- --//-- --//-- --//-- --//-- --//--

## ● Display Student:

- **package** java2project;
- 
- **import** java.awt.\*;
- **import** java.awt.event.ActionEvent;
- **import** java.awt.event.ActionListener;
- **import** java.io.\*;
- **import** java.util.Scanner;
- **import** javax.swing.\*;
- 
- **public class** DisplayStudents **extends** JFrame **implements** ActionListener{
- 
- JTextField **text1**;
- JTextArea **area**;
- **public** DisplayStudents(){
- **super**("Display Students");
- setSize(400,400);
- **this**.setDefaultCloseOperation(JFrame.**HIDE\_ON\_CLOSE**);
- setLocationRelativeTo(**null**);
- setLayout(**null**);
-

- `//title`
- `JLabel lbl= new JLabel("Display Students");`
- `Font Title_Font = new Font("Cambria", Font.BOLD, 28);`
- `lbl.setFont(Title_Font);`
- `lbl.setBounds(70,20,250,50);`
- `add(lbl);`
- 
- `Font Label_Font = new Font("Cambria", Font.BOLD, 14);`
- 
- `//enter major name`
- `JLabel l1= new JLabel("Course Code:",SwingConstants.RIGHT);`
- `l1.setFont(Label_Font);`
- `l1.setBounds(30,70,120,50);`
- `text1 = new JTextField(10);`
- `text1.setBounds(160,80,150,30);`
- `text1.addActionListener(this);`
- `add(l1);`
- `add(text1);`
- 
- `area = new JTextArea(15,30);`
- `JScrollPane scrolledText = new JScrollPane(area);`
- `scrolledText.setBounds(50,140,300,150);`
- 
- `scrolledText.setVerticalScrollBarPolicy(JScrollPane.VERTICAL_SCROLLBAR_ALWAYS);`
- `scrolledText.setHorizontalScrollBarPolicy(JScrollPane.HORIZONTAL_SCROLLBAR_ALWAYS);`

- add(scrolledText);
- 
- 
- JButton back\_btn = new JButton("Back");
- back\_btn.setBounds(160,300,80,30);
- 
- add(back\_btn);
- }
- 
- **public void** actionPerformed(ActionEvent e){
- String code = text1.getText();
- String names = "";
- 
- 
- Scanner inputStream = null;
- // read StudentCourse file
- **try**{
- inputStream = new Scanner(new
- FileInputStream("StudentCourse.txt"));
- **while**(inputStream.hasNextLine()){
- String line = inputStream.nextLine();
- String[] tokens = line.split(",");
- **if** (tokens[2].equals(code)){
- names += "\t"+tokens[1]+"\\n";
- } // end if
- 
- } //end while

- 
- `area.setText(names);`
- 
- `}`
- `catch (FileNotFoundException ex){`
- `System.out.println("File studentCourse.txt was not found");`
- `}`
- 
- `}`
- 
- `}`

--//-- --//-- --//-- --//-- --//-- --//-- --//-- --//-- --//--

## • Display Courses:

- `package java2project;`
- 
- `import java.awt.*;`
- `import java.io.*;`
- `import java.util.Scanner;`
- `import javax.swing.*;`
- 
- 
- `public class DisplayCourses extends JFrame{`
-

- JTextArea `area` ;
- `public` DisplayCourses(){
- `super`("Display Courses");
- setSize(400,400);
- setLocationRelativeTo(`null`);
- `this`.setDefaultCloseOperation(JFrame.*HIDE\_ON\_CLOSE*);
- 
- setLayout(`null`);
- 
- //title
- JLabel lbl= `new` JLabel("Display All Courses");
- Font Title\_Font = `new` Font("Cambria", Font.*BOLD*, 24);
- lbl.setFont(Title\_Font);
- lbl.setBounds(80,20,250,50);
- add(lbl);
- 
- Font Label\_Font = `new` Font("Cambria", Font.*BOLD*, 14);
- 
- `area` = `new` JTextArea(15,150);
- JScrollPane scrolledText = `new` JScrollPane(`area`);
- scrolledText.setBounds(70,80,250,200);
- 
- scrolledText.setVerticalScrollBarPolicy(JScrollPane.*VERTICAL\_SCROLLBAR\_ALWAYS*);
- scrolledText.setHorizontalScrollBarPolicy(JScrollPane.*HORIZONTAL\_SCROLLBAR\_ALWAYS*);
- add(scrolledText);

- 
- 
- JButton back\_btn = new JButton("Back");
- back\_btn.setBounds(150,300,80,30);
- 
- add(back\_btn);
- 
- // read all course from Course File
- Scanner inputStream = null;
- // read admin from a file
- try{
- inputStream = new Scanner(new FileInputStream("Course.txt"));
- 
- String course\_data="";
- //read all file
- while(inputStream.hasNextLine()){
- String line = inputStream.nextLine();
- String[] tokens = line.split(",");
- course\_data += tokens[0]+"\\t"+tokens[1]+"\\n";
- } // end while loop
- 
- area.setText(course\_data);
- 
- }
- catch (FileNotFoundException ex){
- System.out.println("File Course.txt was not found");
- }



- 
- 
- }
- 
- }

```
--//-- --//-- --//-- --//-- --//-- --//-- --//-- --//-- --//--
```

## • Admin Login Window:

```
package java2project;
```

```
import java.awt.*;
```

```
import java.awt.event.ActionEvent;
```

```
import java.awt.event.ActionListener;
```

```
import java.util.Arrays;
```

```
import javax.swing.*;
```

```
public class adminLoginWindow extends JFrame implements ActionListener {
```

```
    private JTextField userNameText;
```

```
    private JPasswordField passWordText;
```

```
    private JLabel message;
```

```
    public adminLoginWindow () {
```

```
//window 1
```

```
this.setTitle("Admin Login Window");  
this.setSize(400,200);  
this.setDefaultCloseOperation(EXIT_ON_CLOSE);  
this.setLayout(new BorderLayout());
```

```
// labels
```

```
JLabel login = new JLabel("Login Page");  
JLabel userNameLbl = new JLabel("User Name:");  
JLabel passLbl = new JLabel("Paswword:");  
message = new JLabel("Message:");
```

```
// initializing the textfields
```

```
userNameText = new JTextField();  
passWordText = new JPasswordField();
```

```
// buttons
```

```
JButton submit = new JButton("Submit");  
submit.addActionListener(this);  
JButton clear = new JButton("Clear");  
clear.addActionListener(this);
```

```
//font
```

```
login.setFont (new Font ("TimesRoman", Font.BOLD | Font.ITALIC, 17));
```

```
JPanel northPanel = new JPanel(new FlowLayout());  
northPanel.add(login);
```

```
JPanel centerPanel = new JPanel(new GridLayout(3,1));  
centerPanel.add(userNameLbl);  
centerPanel.add(userNameText);  
centerPanel.add(passLbl);  
centerPanel.add(passWordText);  
centerPanel.add(submit);  
centerPanel.add(clear);
```

```
//JPanel southPanel = new JPanel(new GridLayout(1,1));
```

```
//adding to the frame
```

```
add(northPanel,BorderLayout.NORTH);  
add(centerPanel,BorderLayout.CENTER);  
add(message,BorderLayout.SOUTH) ;  
//add(southPanel,BorderLayout.SOUTH);
```

```
}
```

```

@Override
public void actionPerformed(ActionEvent e) {
    // TODO Auto-generated method stub

    String command = e.getActionCommand();

    char [] pass = passWordText.getPassword();
    char [] myPass = {'4','2','1'};

    if (command.equals("Submit")) {
        // System.out.println("submit");
        if (userNameText.getText().equals("manar") &&
Arrays.equals(pass,myPass)) {
            //System.out.println(userNameText.equals("manar") &&
Arrays.equals(pass,myPass));
            message.setText("Message : You are logged in Successfully");

            mainWindow main_Window= new mainWindow();
            main_Window.setVisible(true);
        }
        else {
            //System.out.println("Sorry!! Check your username and password");
            message.setText("Message : Sorry!! Check your username and
password");
        }
    }
}

```

```
}
```

```
else if (command.equals("Clear")) {
```

```
    userNameText.setText(null);
```

```
    passWordText.setText(null);
```

```
    message.setText(null);
```

```
}
```

```
}
```

```
}
```

```
--//-- --//-- --//-- --//-- --//-- --//-- --//-- --//-- --//--
```