



 slington college
(इस्लिङ्टन कलेज)

Module Code & Module Title
CC4002NA Information Systems

Assessment Weightage & Type
20% Individual Coursework

Year and Semester
2018-19 Autumn

Student Name: Animesh Gautam

London Met ID:

College ID: np01cp4a180083

Assignment Due Date: 19th April 2019

Assignment Submission Date: 19th April 2019

I confirm that I understand my coursework needs to be submitted online via Google Classroom under the relevant module page before the deadline in order for my assignment to be accepted and marked. I am fully aware that late submissions will be treated as non-submission and a marks of zero will be awarded.

Contents

| | |
|---------------------------------------|----|
| INTRODUCTION | 1 |
| Class | 2 |
| Pseudocode | 3 |
| Description | 7 |
| Class RigoTechnology | 7 |
| main | 7 |
| GUI | 7 |
| windowClosing | 7 |
| actionPerformed | 7 |
| Test | 8 |
| • Test 1 | 8 |
| • Test 2 | 8 |
| • Test 3 | 12 |
| Error detection and correction | 13 |
| • Error 1 | 13 |
| • Error 2 | 14 |
| • Error 3 | 16 |
| CONCLUSION | 18 |
| Appendix | 19 |
| Class RigoTechnology | 19 |

| | |
|---|----|
| Figure 1-Class Diagram | 2 |
| Figure 2-Testing using cmd..... | 8 |
| Figure 3-Adding platform for junior developer | 8 |
| Figure 4-Dialogbox after adding | 9 |
| Figure 5-Adding platform for senior developer | 9 |
| Figure 6-Dialogbox after adding | 9 |
| Figure 7-Appointing senior developer..... | 10 |
| Figure 8-Dialogbox after appointing | 10 |
| Figure 9-Appointing junior developer..... | 10 |
| Figure 10-Dialogbox after adding | 11 |
| Figure 11-Termination developer contract..... | 11 |
| Figure 12-Dialogbox after adding | 11 |
| Figure 13-Error due to wrong platform no | 12 |
| Figure 14-Error due to wrong type | 12 |
| Figure 15-Error due to wrong input of platform no..... | 12 |
| Figure 16-Syntax Error | 13 |
| Figure 17-Error Correction | 14 |
| Figure 18-Inputing Value | 14 |
| Figure 19-RuntimeError..... | 15 |
| Figure 20-Error in code | 15 |
| Figure 21-Error Correction | 15 |
| Figure 22-Working of program after error correction | 16 |
| Figure 23-Error Displayed | 16 |
| Figure 24-Logical Error..... | 17 |
| Figure 25_Error Correction..... | 17 |
| Figure 26-Successful display after error..... | 17 |

INTRODUCTION

This is our second coursework of 2nd semester of programming which was given to us in the 20th week and is to be submitted till 24th week. In this coursework we were given to do coding and make a GUI for inputting values of constructor of senior developer and junior developer. This is the extended coursework of previous coursework. This coursework was given to us so that we can make a GUI take the input from user, get it and do the specific function by calling the function of Senior and Junior Developer. This was to be done in a software named BlueJ. After the completion of coding the program was to be tested and shown in documentation.

Class Diagram

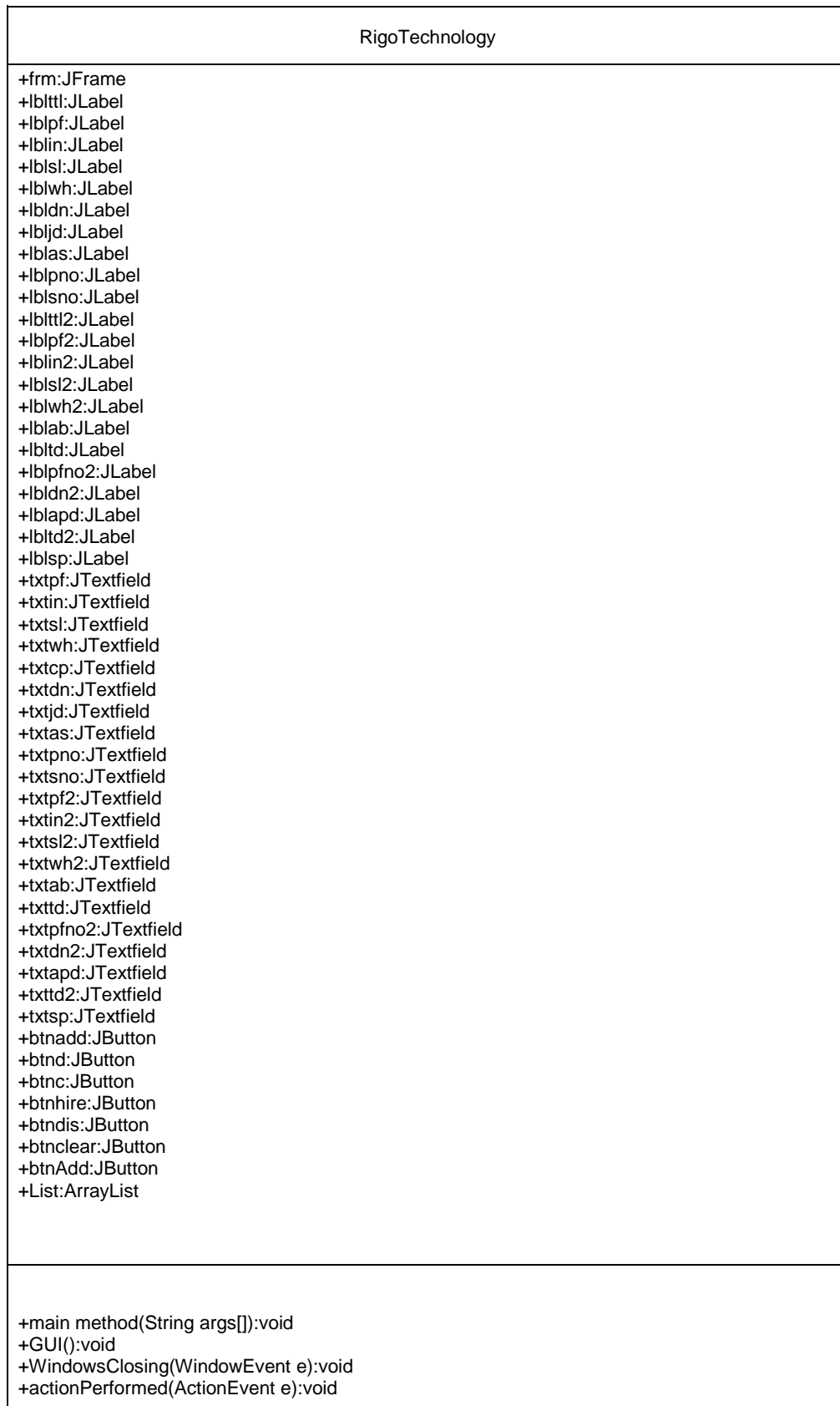


Figure 1-Class Diagram

Pseudocode

Class RigoTechnology

DO

DECLARE JFrame

DECLARE all JLabels

DECLARE all JTextFields

DECLARE all JButtons

DECLARE arraylist

DECLARE void main(String args[])

DO

CALL GUI()

END DO

DECLARE GUI()

DO

SET framename

SET size

SET visible

SET resizable

SET layout

Create object for all JLabels

SET bounds for all JLabels

ADD all JLabels in frame

CREATE object for all JTextFields

SET bounds for all JTextFields

ADD all JTextFields in frame

CREATE object for all JButtons

SET bounds for all JButtons

ADD all JButtons in frame

ADD ActionListener in all JButtons

END DO

DECLARE actionPerformed(ActionEvent e)

DO

IF Button IS ADD of SeniorDeveloper

String Platform= txtpf.getText();

String InterviewerName=txtin.getText();

String Salary=txtsl.getText();

String Workinghour=txtwh.getText();

String Contractperiod=txtcp.getText();

IF(Any TextField IS EMPTY)

DISPLAY Dialogbox ("Empty Text Field.Please fill all the fields")

ELSE

```

        int WorkingHour=Integer.parseInt(Workinghour);
        int Salaryint=Integer.parseInt(Salary);
        int ContractPeriod=Integer.parseInt(Contractperiod);
        IF Workinghour IS GREATER THAN OR EQUAL
        TO 8 AND LESS THAN OR EQUAL TO 12
            SeniorDeveloper detailSen= new SeniorDeveloper
            (Platform,InterviewerName,WorkingHour,Salary
            int,ContractPeriod);
            ADD detailSen in list
            DISPLAY Dialogbox ("Successfully Added")
        ELSE
            DISPLAY Dialogbox ("Limit Reached of Working
            Hours. Limit 8-12");

    IF Button IS APPOINT of SeniorDeveloper
        String DeveloperName=txtDn.getText();
        String Joiningdate=txtjd.getText();
        String Advancesalary=txtas.getText();
        String Roomno=txtsno.getText();
        String Platformno=txtpno.getText();
        IF(Any TextField IS EMPTY)
            DISPLAY Dialogbox ("Empty Text Field.Please fill
            all the fields")
        ELSE
            int AdvanceSalary=Integer.parseInt(Advancesalary)
            int PlatformNo=((Integer.parseInt(Platformno))-1);
            IF PlatformNo IS GREATER THAN OR EQUAL TO 0 AND
            LESS THAN OR EQUAL TO size of arraylist
                IF PlatformNo IS INSTANCEOF SeniorDeveloper
                    SeniorDeveloper appSen=(SeniorDeveloper)list.
                    get(PlatformNo);
                    CALL hire of SeniorDeveloper(DeveloperName,
                    Joiningdate,AdvanceSalary,Roomno);
                    DISPLAY Dialogbox ("Senior Developer has been
                    appointed")
                ELSE
                    DISPLAY Dialogbox ("Not a valid Platform. Please
                    enter a valid Platform No.")
            ELSE
                DISPLAY Dialogbox ("Not a valid Platform. Please
                enter a valid Platform No.")

    IF Button IS TERMINATE of SeniorDeveloper
        String Platformno=txtpno.getText();
        IF Platformno IS EMPTY

```



```

        DISPLAY Dialogbox ("Empty Text Field.Please fill
        all the fields")
    ELSE
        int PlatformNo=((Integer.parseInt(Platformno))-1);
        IF PlatformNo from list IS INSTANCEOF SeniorDeveloper

            SeniorDeveloper termDev=(SeniorDeveloper)list.
            get(PlatformNo);
            IF termDev.getappointed()==false
                DISPLAY Dialogbox("Developer not appointed
                yet")
            ELSE
                termDev.termination();
                DISPLAY Dialogbox("Developer Terminated")
        ELSE
            DISPLAY Dialogbox("Not a valid Platform .Please
            enter a valid Platform No.")

    IF Button IS ADD of JuniorDeveloper
        String JPlatform=txtpf2.getText();
        String JInterviewerName=txtin2.getText();
        String JWorkinghour=txtwh2.getText();
        String Jsalary=txtsl2.getText();
        String JAppoint=txtab.getText();
        String JTermination=txttd.getText();
        IF(Any TextField IS EMPTY)
            DISPLAY Dialogbox ("Empty Text Field.Please fill
            all the fields")
        ELSE
            int JWorkingHour=Integer.parseInt(JWorkinghour);
            int JSalary=Integer.parseInt(Jsalary);
            JuniorDeveloper detailJun=new JuniorDeveloper
            (JPlatform,JInterviewerName,JWorkingHour,JSalary,
            JAppoint,JPromotion);
            ADD detailJun in list
            DISPLAY Dialogbox ("Successfully Added")

    IF Button IS HIRE of JuniorDeveloper
        String JPlatformno=txtpfno2.getText();
        String JDeveloperName=txttn2.getText();
        String JAppointedDate=txtapd.getText();
        String JTerminationDate=txttd2.getText();
        String JSpecialization=txtsp.getText();
        IF(Any TextField IS EMPTY)
            DISPLAY Dialogbox ("Empty Text Field.Please fill
            all the fields")

```

```

ELSE
    int JPlatformNo=((Integer.parseInt(JPlatformno))-1);
    IF PlatfromNo IS GREATER THAN OR EQUAL TO 0 AND
    LESS THAN OR EQUAL TO size of arraylist
        JuniorDeveloper hireJun=(JuniorDeveloper)list.
        get(JPlatformNo);
        hireJun.appoint(JDeveloperName, JAppointedDate,
        JTerminationDate, JSpecialization);
        DISPLAY Dialogbox ("Junior Developer has been
        appointed")
    ELSE
        DISPLAY Dialogbox("Not a valid Platform No.
        Please enter a valid Platform No.")

IF Button IS DISPLAY
    IF list IS EMPTY
        DISPLAY Dialogbox("Please enter the data first.")
    ELSE
        FOR Developer dev:list
            IF dev IS INSTANCEOF SeniorDeveloper
                SeniorDeveloper Sdis=(SeniorDeveloper)
                dev;
                Sdis.display();
            IF dev IS INSTANCEOF JuniorDeveloper
                JuniorDeveloper Jdis=(JuniorDeveloper)dev;
                Jdis.display();

IF Button IS CLEAR
    SET empty text for all JTextField

```

Description

Class RigoTechnology

main

This is the main method of the program from where execution starts. This method simply calls GUI method of the same class.

GUI

In this method the creation of gui is done. Frame is created using JFrame. JLabel, JTextField, JButton are used for labels, textfields, and buttons. Size for labels, textfields and buttons are given. ActionListener is also added. GUI is created through this method.

windowClosing

This method simply creates a dialog box asking whether to exit or not and according the function is carried out.

actionPerformed

In this method all the work of buttons are done. When user click any button their specific functions are carried out and shows dialog box after the completion of the process or when an error occurs.

Test

- Test 1

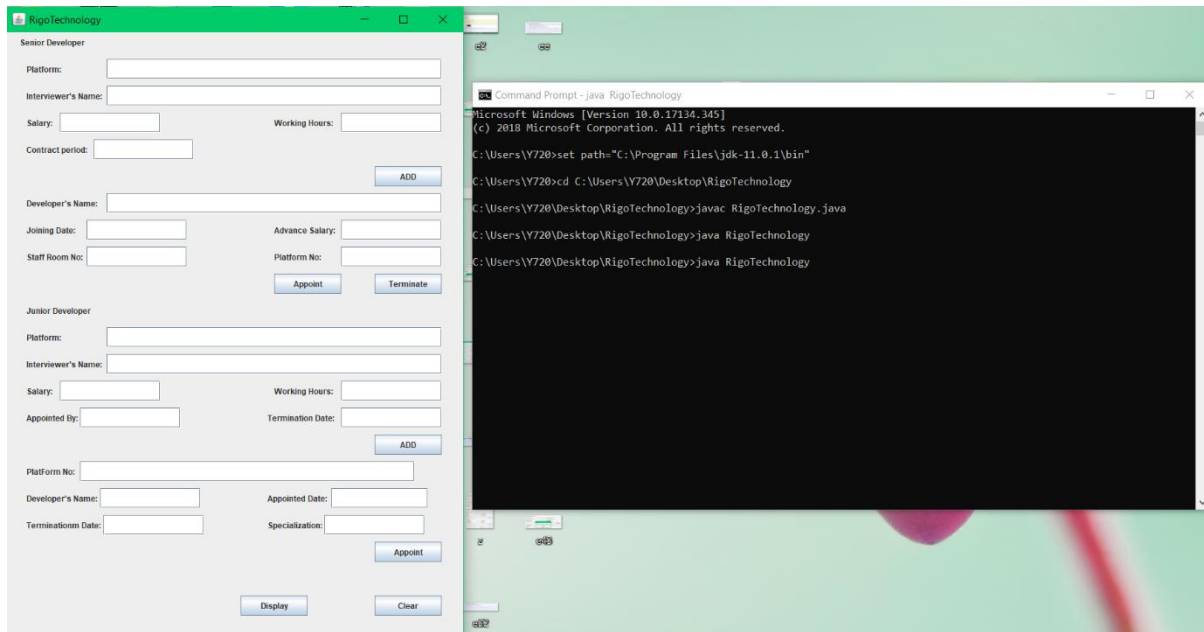


Figure 2-Testing using cmd

- Test 2

Adding platform for junior developer list

This is a screenshot of the 'Junior Developer' form within the RigoTechnology application. The form has a light gray background and a white title bar. It contains several input fields with labels: 'Platform:' with the value 'java', 'Interviewer's Name:' with the value 'Bhim', 'Salary:' with the value '1000', 'Working Hours:' with the value '10', 'Appointed By:' with the value 'Sukrit', and 'Termination Date:' with the value 'jan 2022'. There is an 'ADD' button at the bottom right of the form.

Figure 3-Adding platform for junior developer

Junior Developer

Platform:

Interviewer's Name:

Salary: Working Hours:

Appointed By: Termination Date:

Message

Successfully Added

Figure 4-Dialogbox after adding

Adding platform for senior developer list

Senior Developer

Platform:

Interviewer's Name:

Salary: Working Hours:

Contract period:

Figure 5-Adding platform for senior developer

Senior Developer

Platform:

Interviewer's Name:

Salary: Working Hours:

Contract period:

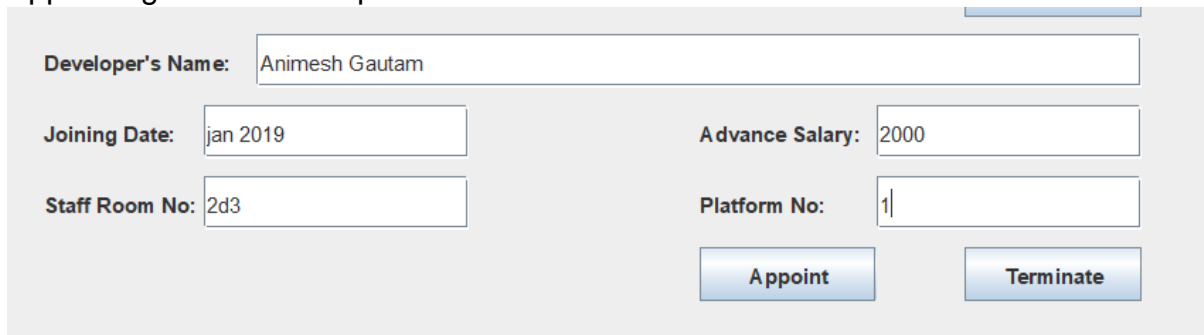
Developer's Name:

Message

Successfully Added

Figure 6-Dialogbox after adding

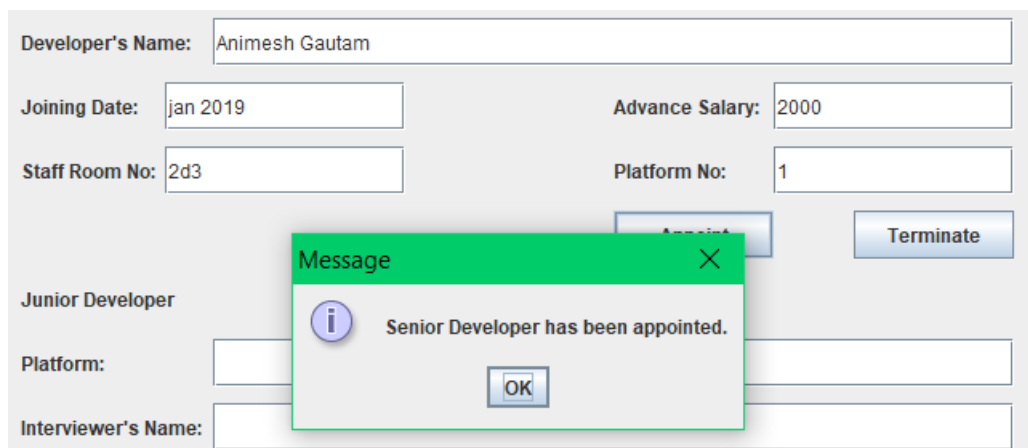
Appointing senior developer



A screenshot of a web form for appointing a senior developer. The form has a light gray background. It contains several input fields: 'Developer's Name' with the value 'Animesh Gautam', 'Joining Date' with 'jan 2019', 'Staff Room No' with '2d3', 'Advance Salary' with '2000', and 'Platform No' with '1'. At the bottom right, there are two blue buttons: 'Appoint' and 'Terminate'.

| | | | |
|-------------------|----------------|-----------------|-----------|
| Developer's Name: | Animesh Gautam | | |
| Joining Date: | jan 2019 | Advance Salary: | 2000 |
| Staff Room No: | 2d3 | Platform No: | 1 |
| | | Appoint | Terminate |

Figure 7-Appointing senior developer



A screenshot of the same web form as in Figure 7, but with a confirmation dialog box overlaid in the center. The dialog box has a green header bar with the title 'Message' and a close button (X). It contains an information icon (i) and the text 'Senior Developer has been appointed.' Below the text is an 'OK' button. The background form is partially visible, showing the same input fields and buttons as before.

| | | | |
|-------------------|----------------|-----------------|-----------|
| Developer's Name: | Animesh Gautam | | |
| Joining Date: | jan 2019 | Advance Salary: | 2000 |
| Staff Room No: | 2d3 | Platform No: | 1 |
| | | Appoint | Terminate |

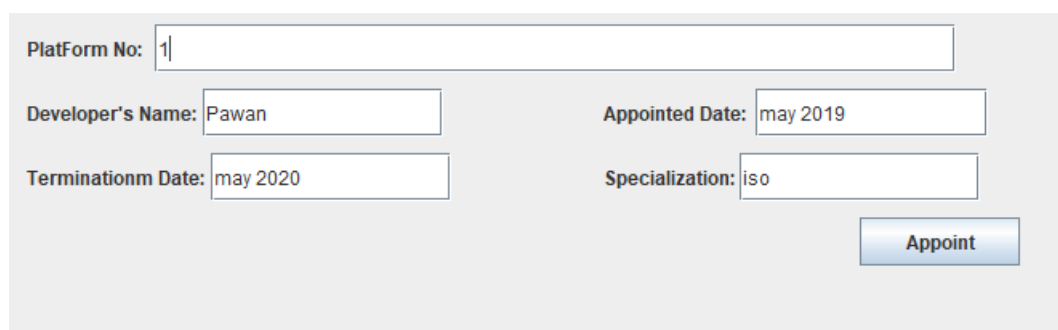
Message

Senior Developer has been appointed.

OK

Figure 8-Dialogbox after appointing

Appoint junior developer



A screenshot of a web form for appointing a junior developer. The form has a light gray background. It contains several input fields: 'Platform No' with '1', 'Developer's Name' with 'Pawan', 'Appointed Date' with 'may 2019', 'Termination Date' with 'may 2020', and 'Specialization' with 'iso'. At the bottom right, there is a blue button labeled 'Appoint'.

| | | | |
|-------------------|----------|-----------------|----------|
| Platform No: | 1 | | |
| Developer's Name: | Pawan | Appointed Date: | may 2019 |
| Termination Date: | may 2020 | Specialization: | iso |
| | | Appoint | |

Figure 9-Appointing junior developer

Interviewer's Name: Durba

Salary: 20000

Appointed By: Bibek

Platform No: 1

Developer's Name: Pawan

Appointed Date: may 2019

Termination Date: may 2020

Specialization: iso

ADD

Appoint

Message X

Junior Developer has been appointed.

OK

Figure 10-Dialogbox after adding

Terminating developer contract

Developer's Name: Animesh Gautam

Joining Date: jan 2019

Staff Room No: 2d3

Advance Salary: 2000

Platform No: 2

Appoint

Terminate

Figure 11-Termination developer contract

Developer's Name: Animesh Gau

Joining Date: jan 2019

Staff Room No: 2d3

Platform No: 2

Appoint

Terminate

ADD

Message X

Developer Terminated

OK

Figure 12-Dialogbox after adding

- Test 3

Developer's Name: Animesh Gautam

Joining Date: jan 2019

Advance Salary: 1000

Staff Room No: 2d3

Platform No: 2

Buttons: Assign, Terminate

Junior Developer

Platform:

Interviewer's Name:

Message

Not a valid Platform .Please enter a valid Platform No.

OK

Figure 13-Error due to wrong platform no

Developer's Name: Animesh Gautam

Joining Date: jan 2019

Advance Salary: 1000

Staff Room No: 2d3

Platform No: ac

Buttons: Assign, Terminate

Junior De

Platform:

Interviewer's name:

Message

Please Enter valid input. Keep Integer values in Advance Salary and Platform No.

OK

Figure 14-Error due to wrong type

Developer's Name: Animesh Gautam

Joining Date: jan 2019

Advance Salary: 1000

Staff Room No: 2d3

Platform No: 1

Buttons: Assign, Terminate

Junior Devel

Platform:

Interviewer's name: joberk

Message

Not a valid Platam of SeniorDeveloper.Please enter a valid Platform No.

OK

Figure 15-Error due to wrong input of platform no

Error detection and correction

- Error 1

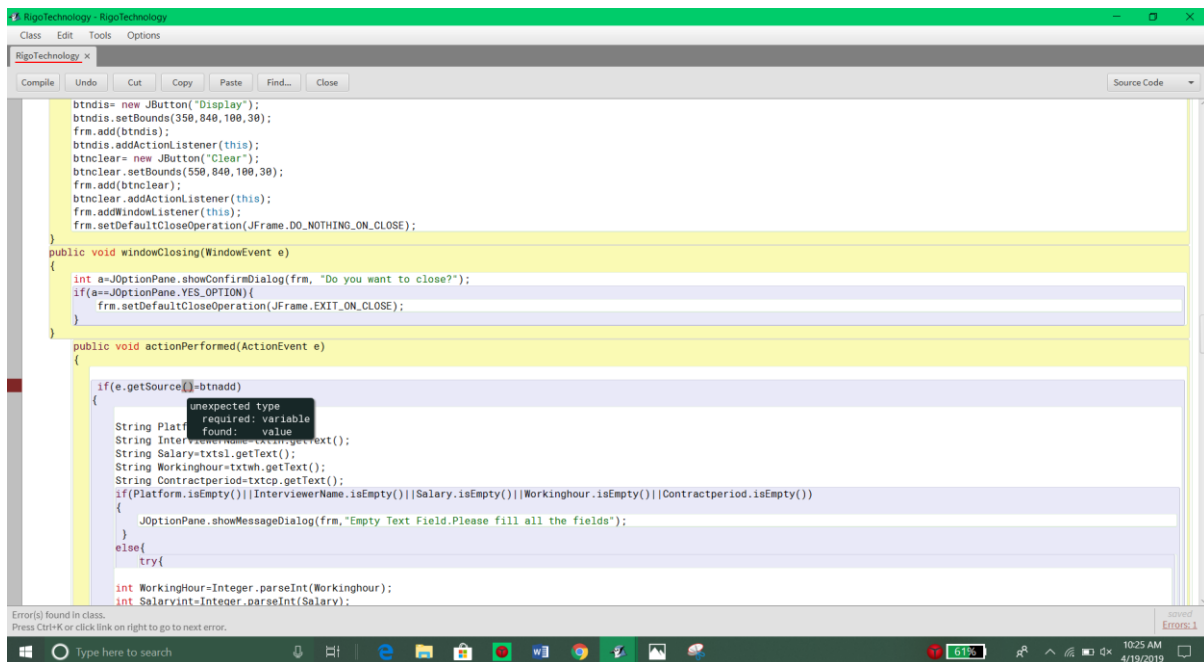


Figure 16-Syntax Error

The above error in the figure is a syntax error. It says unexpected type. This kind of error occurs when there are mistakes in the syntax, missing of any symbols, or the grammar and spelling of the language. This error was encountered due to the missing symbol i.e. =. This error was rectified by keeping the missing symbol =. The corrected code is given below.

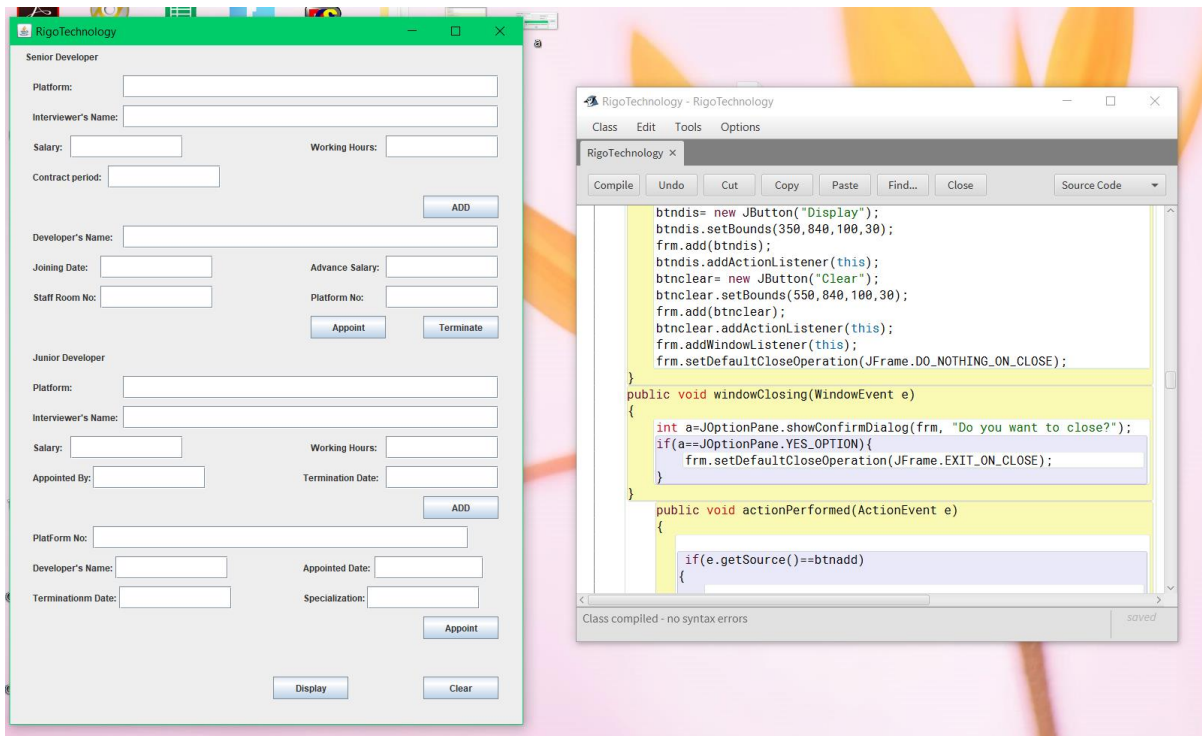


Figure 17-Error Correction

- Error 2

Platform No:

Developer's Name: Appointed Date:

Termination Date: Specialization:

Figure 18-Inputing Value

```
Exception in thread "AWT-EventQueue-0" java.lang.ClassCastException: SeniorDeveloper cannot be cast to JuniorDeveloper
at RigoTechnology.actionPerformed(RigoTechnology.java:371)
at javax.swing.AbstractButton.fireActionPerformed(AbstractButton.java:2022)
at javax.swing.AbstractButton$Handler.actionPerformed(AbstractButton.java:2348)
at javax.swing.DefaultButtonModel.fireActionPerformed(DefaultButtonModel.java:402)
at javax.swing.DefaultButtonModel.setPressed(DefaultButtonModel.java:259)
at javax.swing.plaf.basic.BasicButtonListener.mouseReleased(BasicButtonListener.java:252)
at java.awt.Component.processMouseEvent(Component.java:6539)
at javax.swing.JComponent.processMouseEvent(JComponent.java:3324)
at java.awt.Component.processEvent(Component.java:6304)
at java.awt.Container.processEvent(Container.java:2239)
at java.awt.Component.dispatchEventImpl(Component.java:4889)
at java.awt.Container.dispatchEventImpl(Container.java:2297)
at java.awt.Component.dispatchEvent(Component.java:4711)
at java.awt.LightweightDispatcher.retargetMouseEvent(Container.java:4904)
at java.awt.LightweightDispatcher.processMouseEvent(Container.java:4535)
at java.awt.LightweightDispatcher.dispatchEvent(Container.java:4476)
```

Figure 19-RuntimeError

```
int JPlatformNo=((Integer.parseInt(JPlatformNo))-1);
if(JPlatformNo>=0 && JPlatformNo<list.size())
{
    JuniorDeveloper hireJun=(JuniorDeveloper)list.get(JPlatformNo);
    hireJun.appoint(JDeveloperName, JAppointedDate, JTerminationDate, JSpecialization);
    JOptionPane.showMessageDialog(frm,"Junior Developer has been appointed.");
}

else{
    JOptionPane.showMessageDialog(frm,"Not a valid Platform No.Please enter a valid Platform No.");
}

}
catch(NumberFormatException a)
{
```

Figure 20-Error in code

This error is called runtime error. As seen in above figure even after keeping the value of Platform No This kind of error occurs when the program is asked to do something that is unreliable and not possible. The corrected code for this error is show below.

```
else{
    try{
        int JPlatformNo=((Integer.parseInt(JPlatformNo))-1);
        if(JPlatformNo>=0 && JPlatformNo<list.size())
        {
            if(list.get(JPlatformNo) instanceof JuniorDeveloper)
            {
                JuniorDeveloper hireJun=(JuniorDeveloper)list.get(JPlatformNo);
                hireJun.appoint(JDeveloperName, JAppointedDate, JTerminationDate, JSpecialization);
                JOptionPane.showMessageDialog(frm,"Junior Developer has been appointed.");
            }
            else{
                JOptionPane.showMessageDialog(frm,"Not a valid Platform No of Junior Developer.Please enter a valid Platform No.");
            }
        }
    }
    else{
        JOptionPane.showMessageDialog(frm,"Not a valid Platform No.Please enter a valid Platform No.");
    }
}
```

Figure 21-Error Correction

Junior Developer

Platform:

Interviewer's Name:

Salary:

Appointed By:

Platform No:

Developer's Name:

Appointed Date:

Termination Date:

Specialization:

Figure 22-Working of program after error correction

- Error 3

Developer's Name:

Joining Date:

Advance Salary:

Staff Room No:

Platform No:

Junior Developer

Platform:

Figure 23-Error Displayed

```

if(PlatformNo>0 && PlatformNo<list.size())
{
    if(list.get(PlatformNo) instanceof SeniorDeveloper)
    {
        SeniorDeveloper appSen=(SeniorDeveloper)list.get(PlatformNo);
        appSen.hire(DeveloperName,Joiningdate,AdvanceSalary,Roomno);
        JOptionPane.showMessageDialog(frm,"Senior Developer has been appointed.");
    }
    else{
        JOptionPane.showMessageDialog(frm,"Not a valid Platform of SeniorDeveloper.Please enter a valid Platform No.");
    }
}
else{
    JOptionPane.showMessageDialog(frm,"Not a valid Platform .Please enter a valid Platform No.");
}
}

```

Figure 24-Logical Error

This type of error is called logical error. It occurs when there is flaw in the flow of program or the concept. The program does not crash rather . As seen in the figure above due to the fault in the program the platform inputted shows error due to wrong logic. Below are the figures of the corrected ones.

```

if(PlatformNo>=0 && PlatformNo<list.size())
{
    if(list.get(PlatformNo) instanceof SeniorDeveloper)
    {
        SeniorDeveloper appSen=(SeniorDeveloper)list.get(PlatformNo);
        appSen.hire(DeveloperName,Joiningdate,AdvanceSalary,Roomno);
        JOptionPane.showMessageDialog(frm,"Senior Developer has been appointed.");
    }
    else{
        JOptionPane.showMessageDialog(frm,"Not a valid Platform of SeniorDeveloper.Please enter a valid Platform No.");
    }
}
else{
    JOptionPane.showMessageDialog(frm,"Not a valid Platform .Please enter a valid Platform No.");
}
}

```

Figure 25_Error Correction

The screenshot shows a Java Swing application window with the following fields and buttons:

- Developer's Name: Animesh
- Joining Date: jan 2019
- Advance Salary: 1000
- Staff Room No: 2d3
- Platform No: 1
- Buttons: Terminate, OK

A green message dialog box is displayed in the center of the window with the text "Senior Developer has been appointed." and an "OK" button.

Figure 26-Successful display after error

CONCLUSION

The main aim of this coursework was to create a class to build a GUI and relate it with Senior and Junior Developer. After completion of this project, I learned all basic concepts of creating GUI ,calling functions and carrying out its own specific function. We learnt how to do type conversion and create label, buttons, textfield, etc. We also wrote pseudocode, drawn class diagram so that it can be easier to understand for others too. We gained a lot of knowledge on programing, causes of error and different ways to debug them. With the help of our module teacher we were able to do this project with minor difficulties.

Appendix

Class RigoTechnology

```
public class RigoTechnology extends WindowAdapter implements ActionListener
{
    JFrame frm;
    JLabel
lblttl,lblpf,lblin,lblsl,lblwh,lbldn,bljd,lblas,blpno,blsno,blttl2,blpf2,blin2,blsl2,blwh2,l
blab,bltd,blpfno2,lbldn2,blapd,bltd2,blsp,blcp;
    JTextField
txtpf,txtin,txtsl,txtwh,txtcp,txtdn,txtjd,txtas,txtpno,txtsno,txtpf2,txtin2,txtsl2,txtwh2,txtab
,txtd,txtpfno2,txtdn2,txtdp,txtd2,txtsp;
    JButton btnadd,btnnd,btnc,btnhire,btndis,btnclear,btnAdd;
    ArrayList<Developer> list= new ArrayList<Developer>();

    public static void main(String args[])
    {
        new RigoTechnology().GUI();
    }
    public void GUI(){
        frm= new JFrame("RigoTechnology");
        frm.setSize(700,950);
        frm.setVisible(true);
        frm.setResizable(true);
        frm.setLayout(null);
        JPanel pane= new JPanel();

        lblttl=new JLabel();
        lblttl.setBounds(20,0,200,30);
        frm.add(lblttl);
        lblttl.setText("Senior Developer");
        lblpf= new JLabel("Platform:");
        lblpf.setBounds(30,40,60,30);
        frm.add(lblpf);
        txtpf= new JTextField();
        txtpf.setBounds(150,40,500,30);
        frm.add(txtpf);
        lblin= new JLabel("Interviewer's Name:");
        lblin.setBounds(30,80,150,30);
        frm.add(lblin);
        txtin= new JTextField();
        txtin.setBounds(150,80,500,30);
        frm.add(txtin);
```

```

lblsl= new JLabel("Salary:");
lblsl.setBounds(30,120,60,30);
frm.add(lblsl);
txtsl= new JTextField();
txtsl.setBounds(80,120,150,30);
frm.add(txtsl);
lblwh= new JLabel("Working Hours:");
lblwh.setBounds(400,120,100,30);
frm.add(lblwh);

txtwh= new JTextField();
txtwh.setBounds(500,120,150,30);
frm.add(txtwh);
lblcp= new JLabel("Contract period:");
lblcp.setBounds(30,160,100,30);
frm.add(lblcp);
txtcp=new JTextField();
txtcp.setBounds(130,160,150,30);
frm.add(txtcp);
btnadd= new JButton("ADD");
btnadd.setBounds(550,200,100,30);
frm.add(btnadd);
btnadd.addActionListener(this);
lbldn= new JLabel("Developer's Name:");
lbldn.setBounds(30,240,150,30);
frm.add(lbldn);
txtdn= new JTextField();
txtdn.setBounds(150,240,500,30);
frm.add(txtdn);
lbldj= new JLabel("Joining Date:");
lbldj.setBounds(30,280,100,30);
frm.add(lbldj);

txtjd= new JTextField();
txtjd.setBounds(120,280,150,30);
frm.add(txtjd);
lblas= new JLabel("Advance Salary:");
lblas.setBounds(400,280,100,30);
frm.add(lblas);
txtas= new JTextField();
txtas.setBounds(500,280,150,30);
frm.add(txtas);
lbldpno = new JLabel ("Platform No:");
lbldpno.setBounds(400,320,100,30);
frm.add(lbldpno);
txtpno= new JTextField();

```



```

txtpno.setBounds(500,320,150,30);
frm.add(txtpno);
lblsno= new JLabel("Staff Room No:");
lblsno.setBounds(30,320,100,30);
frm.add(lblsno);
txtsno= new JTextField();
txtsno.setBounds(120,320,150,30);
frm.add(txtsno);
btnd= new JButton("Appoint");
btnd.setBounds(400,360,100,30);
frm.add(btnd);
btnd.addActionListener(this);
btnc= new JButton("Terminate");
btnc.setBounds(550,360,100,30);
frm.add(btnc);
btnc.addActionListener(this);

```

```

lblttl2= new JLabel("Junior Developer");
lblttl2.setBounds(30,400,200,30);
frm.add(lblttl2);
lblpf2= new JLabel("Platform:");
lblpf2.setBounds(30,440,60,30);
frm.add(lblpf2);
txtpf2= new JTextField();
txtpf2.setBounds(150,440,500,30);
frm.add(txtpf2);
lblin2= new JLabel("Interviewer's Name:");
lblin2.setBounds(30,480,150,30);
frm.add(lblin2);
txtin2= new JTextField();
txtin2.setBounds(150,480,500,30);
frm.add(txtin2);
lblsl2= new JLabel("Salary:");
lblsl2.setBounds(30,520,60,30);
frm.add(lblsl2);
txtsl2= new JTextField();
txtsl2.setBounds(80,520,150,30);
frm.add(txtsl2);
lblwh2= new JLabel("Working Hours:");
lblwh2.setBounds(400,520,100,30);
frm.add(lblwh2);
txtwh2= new JTextField();
txtwh2.setBounds(500,520,150,30);
frm.add(txtwh2);

```

```

lblab= new JLabel("Appointed By:");
lblab.setBounds(30,560,100,30);
frm.add(lblab);
txtab= new JTextField();
txtab.setBounds(110,560,150,30);
frm.add(txtab);
lbltd= new JLabel("Termination Date:");
lbltd.setBounds(390,560,100,30);
frm.add(lbltd);
txttd= new JTextField();
txttd.setBounds(500,560,150,30);
frm.add(txttd);
btnAdd= new JButton("ADD");
btnAdd.setBounds(550,600,100,30);
frm.add(btnAdd);
btnAdd.addActionListener(this);
lblpfno2= new JLabel("PlatForm No:");
lblpfno2.setBounds(30,640,100,30);
frm.add(lblpfno2);
txtpfno2=new JTextField();
txtpfno2.setBounds(110,640,500,30);
frm.add(txtpfno2);
lbldn2= new JLabel("Developer's Name:");
lbldn2.setBounds(30,680,150,30);
frm.add(lbldn2);
txtdn2=new JTextField();
txtdn2.setBounds(140,680,150,30);
frm.add(txtdn2);
lblapd= new JLabel("Appointed Date:");
lblapd.setBounds(390,680,100,30);
frm.add(lblapd);
txtapd=new JTextField();
txtapd.setBounds(485,680,145,30);
frm.add(txtapd);
lbltd2= new JLabel("Terminationm Date:");
lbltd2.setBounds(30,720,150,30);
frm.add(lbltd2);
txttd2=new JTextField();
txttd2.setBounds(145,720,150,30);
frm.add(txttd2);
lblsp= new JLabel("Specialization:");
lblsp.setBounds(390,720,100,30);
frm.add(lblsp);
txtsp=new JTextField();
txtsp.setBounds(475,720,150,30);
frm.add(txtsp);

```

```

        btnhire= new JButton("Appoint");
        btnhire.setBounds(550,760,100,30);
        frm.add(btnhire);
        btnhire.addActionListener(this);
        btndis= new JButton("Display");
        btndis.setBounds(350,840,100,30);
        frm.add(btndis);
        btndis.addActionListener(this);
        btnclear= new JButton("Clear");
        btnclear.setBounds(550,840,100,30);
        frm.add(btnclear);
        btnclear.addActionListener(this);
        frm.addWindowListener(this);
        frm.setDefaultCloseOperation(JFrame.DO_NOTHING_ON_CLOSE);
    }
    public void windowClosing(WindowEvent e)
    {
        int a=JOptionPane.showConfirmDialog(frm, "Do you want to close?");
        if(a==JOptionPane.YES_OPTION){
            frm.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        }
    }
    public void actionPerformed(ActionEvent e)
    {
        if(e.getSource()==btnadd)
        {
            String Platform= txtpf.getText();
            String InterviewerName=txtin.getText();
            String Salary=txtsl.getText();
            String Workinghour=txtwh.getText();
            String Contractperiod=txtcp.getText();

            if(Platform.isEmpty()||InterviewerName.isEmpty()||Salary.isEmpty()||Workinghour.isE
            mpty()||Contractperiod.isEmpty())
            {
                JOptionPane.showMessageDialog(frm,"Empty Text Field.Please fill all
                the fields");
            }
            else{
                try{

                    int WorkingHour=Integer.parseInt(Workinghour);
                    int Salaryint=Integer.parseInt(Salary);
                    int ContractPeriod=Integer.parseInt(Contractperiod);

```

```

        if(WorkingHour>=8 && WorkingHour<=12)
        {
            SeniorDeveloper detailSen= new
SeniorDeveloper(Platform,InterviewerName,WorkingHour,Salaryint,ContractPeriod);
            list.add(detailSen);
            JOptionPane.showMessageDialog(frm,"Successfully Added");
        }
        else{
            JOptionPane.showMessageDialog(frm,"Limit Reached of Working Hours.
Limit 8-12");
        }
    }
    catch(NumberFormatException err)
    {
        JOptionPane.showMessageDialog(frm,"Please Enter valid input. Keep
Integer values in Working Hours, Salary and Contact period.");
    }
}

if(e.getSource()==btnd)
{
    String DeveloperName=txtDN.getText();
    String Joiningdate=txtjd.getText();
    String Advancesalary=txtas.getText();
    String Roomno=txtsno.getText();
    String Platformno=txtpno.getText();

    if(DeveloperName.isEmpty()||Joiningdate.isEmpty()||Advancesalary.isEmpty()||Room
no.isEmpty()||Platformno.isEmpty())
    {
        JOptionPane.showMessageDialog(frm,"Empty Text Field.Please fill all
the fields");
    }
    else{
        try{

            int AdvanceSalary=Integer.parseInt(Advancesalary);
            int PlatformNo=((Integer.parseInt(Platformno))-1);

            if(PlatformNo>=0 && PlatformNo<list.size())
            {

                if(list.get(PlatformNo) instanceof SeniorDeveloper)
                {

```

```

        SeniorDeveloper appSen=(SeniorDeveloper)list.get(PlatformNo);
        appSen.hire(DeveloperName,Joiningdate,AdvanceSalary,Roomno);
        JOptionPane.showMessageDialog(frm,"Senior Developer has been
appointed.");

    }
    else{
        JOptionPane.showMessageDialog(frm,"Not a valid Platform  of
SeniorDeveloper.Please enter a valid Platform No.");
    }
}
else{
    JOptionPane.showMessageDialog(frm,"Not a valid Platform .Please
enter a valid Platform No.");
}
}
catch(NumberFormatException a)
{
    JOptionPane.showMessageDialog(frm,"Please Enter valid input. Keep
Integer values in Advance Salary and Platform No.");
}
}
}

if(e.getSource()==btnc)
{

    String Platformno=txtpno.getText();
    if(Platformno.isEmpty()){
        JOptionPane.showMessageDialog(frm,"Empty Text Field.Please fill
all the fields");
    }
    else{
        try{
            int PlatformNo=((Integer.parseInt(Platformno))-1);
            if(list.get(PlatformNo) instanceof SeniorDeveloper)
            {
                SeniorDeveloper
termDev=(SeniorDeveloper)list.get(PlatformNo);
                if(termDev.getappointed()==false)
                {
                    JOptionPane.showMessageDialog(frm,"Developer not
appointed yet");
                }
            }
        }
    }
}

```

```

        else{
            termDev.termination();
            JOptionPane.showMessageDialog(frm,"Developer
Terminated");
        }
    }
    else{
        JOptionPane.showMessageDialog(frm,"Not a valid Platform
No.Please enter a valid Platform No.");
    }
}
catch(Exception b)
{
    JOptionPane.showMessageDialog(frm,"Please Enter valid input.
Keep Integer values in Platform No.");
}
}

}
if(e.getSource()==btnAdd)
{
    String JPlatform=txtpf2.getText();
    String JInterviewerName=txtin2.getText();
    String JWorkinghour=txtwh2.getText();
    String Jsalary=txtsl2.getText();
    String JAppoint=txtab.getText();
    String JTermination=txttd.getText();

    if(JPlatform.isEmpty()||JInterviewerName.isEmpty()||JWorkinghour.isEmpty()||Jsalary
.isEmpty()||JAppoint.isEmpty()||JTermination.isEmpty())
    {
        JOptionPane.showMessageDialog(frm,"Empty Text Field.Please fill all
the fields");
    }
    else{
        try{
            int JWorkingHour=Integer.parseInt(JWorkinghour);
            int JSalary=Integer.parseInt(Jsalary);
            JuniorDeveloper detailJun=new
JuniorDeveloper(JPlatform,JInterviewerName,JWorkingHour,JSalary,JAppoint,JTer
mination);
            list.add(detailJun);
            JOptionPane.showMessageDialog(frm,"Successfully Added");

```

```

        }
        catch(Exception a)
        {
            JOptionPane.showMessageDialog(frm,"Please Enter valid input.
Keep Integer values in Working Hours and Salary .");
        }
    }
}
if(e.getSource()==btnhire)
{
    String JPlatformno=txtpfno2.getText();
    String JDeveloperName=txttn2.getText();
    String JAppointedDate=txtapd.getText();
    String JTerminationDate=txttd2.getText();
    String JSpecialization=txtsp.getText();

    if(JPlatformno.isEmpty()||JDeveloperName.isEmpty()||JAppointedDate.isEmpty()||JTerminationDate.isEmpty()||JSpecialization.isEmpty())
    {
        JOptionPane.showMessageDialog(frm,"Empty Text Field.Please fill all
the fields");
    }
    else{
        try{
            int JPlatformNo=((Integer.parseInt(JPlatformno))-1);
            if(JPlatformNo>=0 && JPlatformNo<list.size())
            {
                if(list.get(JPlatformNo) instanceof JuniorDeveloper)
                {
                    JuniorDeveloper hireJun=(JuniorDeveloper)list.get(JPlatformNo);
                    hireJun.appoint(JDeveloperName, JAppointedDate,
JTerminationDate, JSpecialization);
                    JOptionPane.showMessageDialog(frm,"Junior Developer has been
appointed.");
                }
                else{
                    JOptionPane.showMessageDialog(frm,"Not a valid Platform No
of Junior Developer.Please enter a valid Platform No.");
                }
            }
        }

        else{
            JOptionPane.showMessageDialog(frm,"Not a valid Platform
No.Please enter a valid Platform No.");
        }
    }
}

```

```

        }

    }
    catch(NumberFormatException a)
    {
        JOptionPane.showMessageDialog(frm,"Please Enter valid input.
Keep Integer values in Platform No.");
    }
}
}
if(e.getSource()==btndis)
{

    if(list.isEmpty())
    {
        JOptionPane.showMessageDialog(frm,"Please enter the data first.");
    }
    else
    {
        for(Developer dev:list)
        {
            if(dev instanceof SeniorDeveloper)
            {

                SeniorDeveloper Sdis=(SeniorDeveloper)dev;
                Sdis.display();
            }
            if(dev instanceof JuniorDeveloper)
            {
                JuniorDeveloper Jdis=(JuniorDeveloper)dev;
                Jdis.display();
            }
        }
    }
}

if(e.getSource()==btnclear)
{
    txtpf.setText("");
    txtin.setText("");
    txtsl.setText("");
    txtwh.setText("");
    txtcp.setText("");
    txtdn.setText("");
    txtjd.setText("");
    txtas.setText("");
}

```



```
        txtpno.setText("");
        txtsno.setText("");
        txtpf2.setText("");
        txtin2.setText("");
        txtsl2.setText("");
        txtwh2.setText("");
        txtab.setText("");
        txttd.setText("");
        txtpfno2.setText("");
        txtdn2.setText("");
        txtapd.setText("");
        txttd2.setText("");
        txtsp.setText("");
    }
}

}
```