Experiment 16
Calculator using Java Swing

Date of submission: 18-11-2020

Aim: Write a Java program that works as a simple calculator. Arrange Buttons for digits and the + - * / operations properly. Add a text field to display the result. Handle any possible exceptions like divide by zero. Use Java Swing.

Concepts Used: Java Swing and event handling

Algorithm:

Class Calulator

```
Create JFrame with title Calculator and setDefaultCloseOperation to EXIT_ON_CLOSE
Set the layout manger for the JFrame to GridBagLayout
Create the JLabel to display the result and to show the numbers entered
Add the Jlabel to `display` the number entered and the result
Add the Jbuttons with the symbols 1,2,3,4,5,6,7,8,9,0 for the numbers and +,-,*,/ for the operations
//Event handling
For every numberbutton:
    if the button is pressed, append the character to the Jlabel
Done

If +,-,*,/ button pressed
    case + : add = true,sub=false,mul=false,div=false
        break
    case - : add = false,sub=true,mul=false,div=false
        break
    case * : add = true,sub=false,mul=true,div=false
        break
    case / : add = false,sub=false,mul=false,div=true
        break
    endcase
a = Float.valueOf(display.getText())
endFor

If '=' button is pressed
    flag =false
    if(add or sub or mul or div of mod)
        b = Float.valueOf(display.getText())
        case add: res=a+b
            break
        case sub: res = a-b
            break
        case mul: res=a*b
            break
        case div: if b==0
                flag=true
            else
                res = a/b
            endif
        endcase
        if flag then
            display.setText(“Divide by 0 error”)
        else
            display.setText(res)
        endif
    endif
```

Result: The program is successfully compiled and the required output is obtained

Program code:

```
/****************************************
 * Calculator implementing the funtions +,-,*,/
 * Done By: Rohit Karunakaran
 * ************************************/
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;


public class Calculator
{
   JFrame jfrm;
   JLabel disp;
   GridBagConstraints c ;
   ActionListener numberButtonPressed;
   ActionListener mathButtonPressed;
   boolean add;
   boolean sub;
   boolean mul;
   boolean div;
   boolean done;

   double calc;

   public Calculator()
   {
      jfrm = new JFrame("Calculator");
```

```
jfrm.setLayout(new GridBagLayout());
c = new GridBagConstraints();
calc = 0;
add=false;
sub=false;
div=false;
mul=false;
done = true;

jfrm.setSize(270,330);
jfrm.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

disp = new JLabel(String.valueOf(calc));

numberButtonPressed = new ActionListener()
{
    public void actionPerformed(ActionEvent ae)
    {
        JButton callerButton = (JButton)ae.getSource();
        String val = callerButton.getText();
        String displayText = disp.getText();

        if(displayText.equals("ERROR"))
        {
            disp.setText(val);
        }
        else
        {
            if(done ==true){
                    disp.setText(String.valueOf(Double.parseDouble(val)));
            }
            else{
                    double newVal = Double.parseDouble(displayText)*10+Double.parseDouble(val);
                    disp.setText(String.valueOf(newVal));
                }
        }
        done = false;
    }
};

mathButtonPressed = new ActionListener()
{
    public void actionPerformed(ActionEvent ae)
    {
        //done = true;
        JButton b = (JButton) ae.getSource();
        double result = 0.0;
        String val = disp.getText();
        if(!done)
```

```
{
    if(val.equals("ERROR"))
    {
        disp.setText("0.0");
        calc = 0.0;
    }
    else
    {
        if(add||sub||div||mul)
        {
            double operand= Double.parseDouble(val);
            if(add)
            {
                result = calc+operand;
                add = false;
            }
            else if(sub)
            {
                result = calc-operand;
                sub = false;
            }
            else if(mul)
            {
                result = calc*operand;
                mul = false;
            }
            else if(div)
            {
                if(operand!=0.0)
                    result = calc/operand;
                else{
                    disp.setText("ERROR");
                    done = true;
                    return;
                }
                div = false;
            }
            calc = result;
            disp.setText(String.valueOf(calc));

        }
        else
        {
            calc = Double.parseDouble(val);
            char op = b.getText().charAt(0);
            switch(op)
            {
                case '+':add=true;break;
                case '*':mul=true;break;
```

```
                                case '/':div=true;break;
                                case '-':sub=true;break;
                            }
                        }

                    }
                    done = true;
                }
            }
        };

        c.anchor = GridBagConstraints.FIRST_LINE_START;
        c.fill= GridBagConstraints.VERTICAL;
        c.weightx=0.5;
        c.gridx=0; c.gridy=0;
        c.gridwidth = 3;
        c.ipady = 20;

        jfrm.add(disp,c);
        addButtons();
        jfrm.setVisible(true);
    }

    private void addButtons()
    {
        c.ipady = 10;
        c.gridwidth = 1;
        c.fill= GridBagConstraints.HORIZONTAL;
        c.anchor = GridBagConstraints.LINE_START;
        JButton numbers[] = new JButton[10];
        for(int i = 0;i<10;i++)
        {
            numbers[i] = new JButton(String.valueOf(i));
            numbers[i].addActionListener(numberButtonPressed);
            if(i!=0)
                c.gridx = (i+2)%3;
            else
                c.gridx = 1;
            c.gridy = i%3==0?4-(i/3-1):4-(i/3);
            jfrm.add(numbers[i],c);
        }
        JButton addButton= new JButton("+");
        addButton.addActionListener(mathButtonPressed);

        JButton subButton= new JButton("-");
        subButton.addActionListener(mathButtonPressed);

        JButton mulButton= new JButton("x");
        mulButton.addActionListener(mathButtonPressed);
```

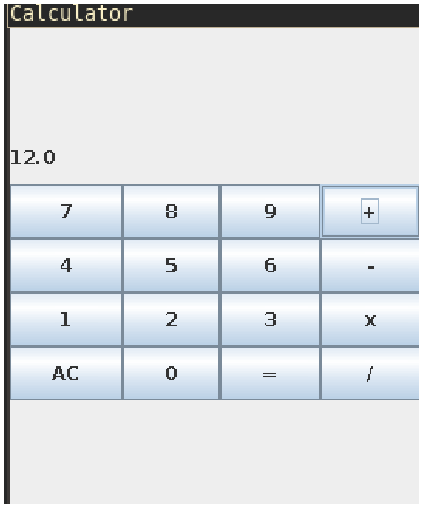
```
JButton divButton= new JButton("/");
divButton.addActionListener(mathButtonPressed);

c.gridx = 3;
c.gridy = 2;
jfrm.add(addButton,c);
c.gridy = 3;
jfrm.add(subButton,c);
c.gridy =4;
jfrm.add(mulButton,c);
c.gridy = 5;
jfrm.add(divButton,c);

JButton equalButton = new JButton("=");
equalButton.addActionListener(new ActionListener()
        {
            public void actionPerformed(ActionEvent ae)
            {
                double result = 0.0;
                String val = disp.getText();
                if(!done)
                {
                    if(val.equals("ERROR"))
                    {
                        disp.setText("0.0");
                        calc = 0.0;
                    }
                    else
                    {
                        if(add||sub||div||mul)
                        {
                            double operand= Double.parseDouble(val);
                            if(add)
                            {
                                result = calc+operand;
                                add = false;
                            }
                            else if(sub)
                            {
                                result = calc-operand;
                                sub = false;
                            }
                            else if(mul)
                            {
                                result = calc*operand;
                                mul = false;
                            }
                            else if(div)
```

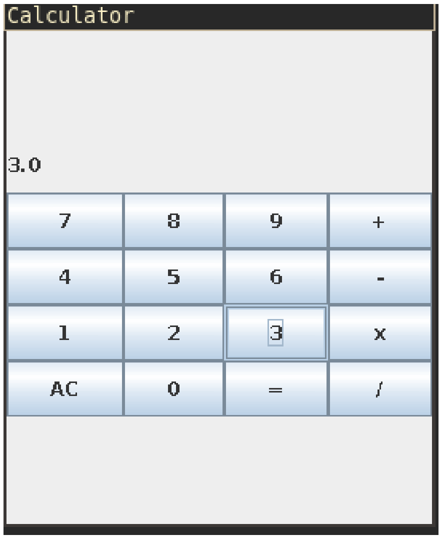
```
                                    {
                                        if(operand!=0.0)
                                            result = calc/operand;
                                        else{
                                            disp.setText("ERROR");
                                            done = true;
                                            return;
                                            }
                                        div = false;
                                    }
                                    calc = result;
                                    disp.setText(String.valueOf(calc));
                                }
                                else
                                {
                                    calc = Double.parseDouble(val);
                                }

                            }
                            done = true;
                        }
                    }
                });

        c.gridy=5;
        c.gridx =2;
        jfrm.add(equalButton,c);

        c.gridx = 0;
        JButton clearAll = new JButton("AC");
        clearAll.addActionListener(new ActionListener()
                {
                    public void actionPerformed(ActionEvent ae)
                    {
                        add = mul = div = sub = false;
                        done = true;
                        disp.setText("0.0");
                    }
                }
                );
        jfrm.add(clearAll,c);

    }


    public static void main(String[] args)
    {
        SwingUtilities.invokeLater(new Runnable()
                {
```

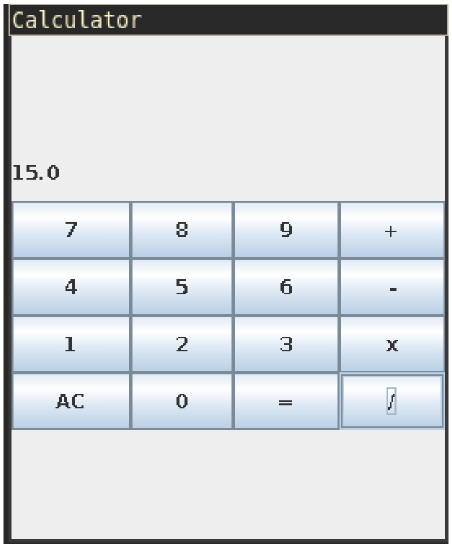
```
               public void run()
               {
                  new Calculator();
               }
            });
   }
}
```

Sample output

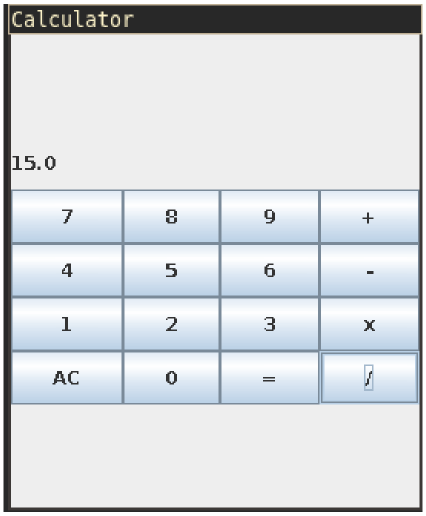
12+3 =15

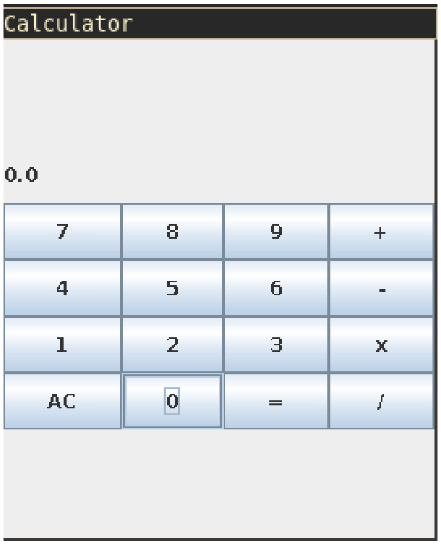

12.0

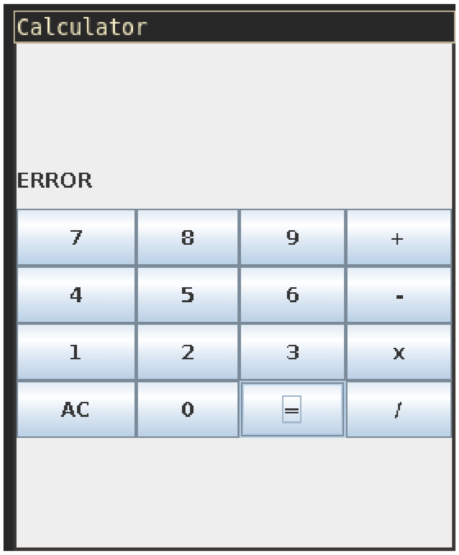

3.0


15.0

15 / 0 = ERROR


15.0


0.0


ERROR

Experiment 17
Traffic Lights using Java Swing

Date of submission: 18-11-2020

Aim: Write a Java program that simulates a traffic light. The program lets the user select one of three lights: red, yellow, or green. When a radio button is selected, the light is turned on, and only one light can be on at a time. No light is on when the program starts.

Concepts Used: Java Swing, Event handling

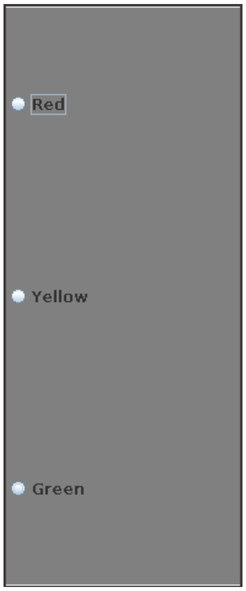
Algorithm:

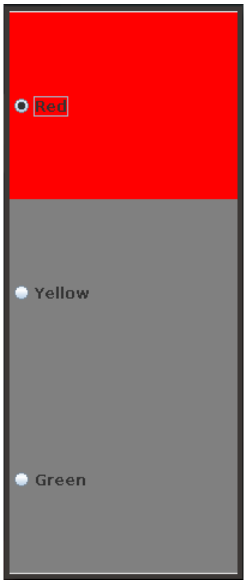
```
1.  Create 3 JradioButtons buttons[] with the values “Red” “Yellow” “Green” for the three buttons
2.  buttons[0].setBackground(Color..GRAY)
3.  buttons[2].setBackground(Color..GRAY)
4.  buttons[1].setBackground(Color..GRAY)
5.
6.  add the buttons to the Jframe
7.  Create a ButtonGroup bg
8.  Add the buttons to bg
9.
10. //Event handling
11. button[0] is pressed
12.         set button[1] and button[2] to Grey
13.         set button[0] tored
14. button[1] is pressed
15.         set button[1] to yellow
16.         set button[0] and button[2] to Grey
17. button[2] is pressed
18.         set button[2] to green
19.         set button[0] and button[1] to Grey
```

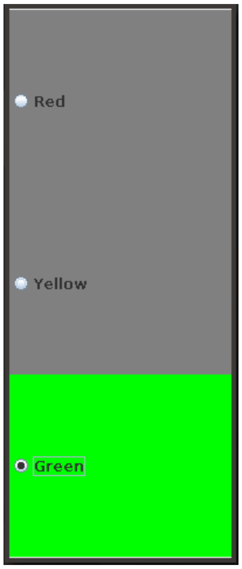
Program Code

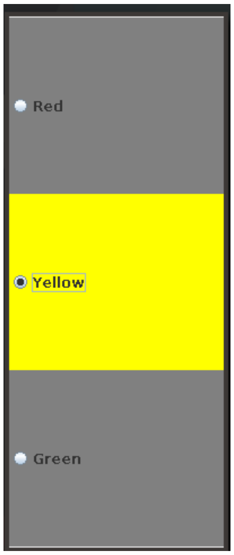
```
import java.awt.*;
import javax.swing.*;
import java.awt.event.*;

class TrafficLights{
    TrafficLights(){
        JFrame jfrm = new JFrame();
        jfrm.setLayout(new GridLayout(3,1));
        jfrm.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        jfrm.setSize(200,500);
        JRadioButton signalButtons[] = new JRadioButton[3];
        ButtonGroup bg = new ButtonGroup();

        signalButtons[0]=new JRadioButton("Red");
        signalButtons[0].setBackground(Color.GRAY);;

        signalButtons[1]=new JRadioButton("Yellow");
        signalButtons[1].setBackground(Color.GRAY);;

        signalButtons[2]=new JRadioButton("Green");
        signalButtons[2].setBackground(Color.GRAY);;

        bg.add(signalButtons[0]);
        bg.add(signalButtons[1]);
        bg.add(signalButtons[2]);

        jfrm.add(signalButtons[0]);
        jfrm.add(signalButtons[1]);
        jfrm.add(signalButtons[2]);

        signalButtons[0].addActionListener(new ActionListener(){
            public void actionPerformed(ActionEvent e){
                signalButtons[0].setBackground(Color.RED);
                signalButtons[1].setBackground(Color.GRAY);;
                signalButtons[2].setBackground(Color.GRAY);;
            }
        });

        signalButtons[1].addActionListener(new ActionListener(){
            public void actionPerformed(ActionEvent e){
                signalButtons[0].setBackground(Color.GRAY);
                signalButtons[1].setBackground(Color.YELLOW);;
                signalButtons[2].setBackground(Color.GRAY);;
            }
        });

        signalButtons[2].addActionListener(new ActionListener(){
            public void actionPerformed(ActionEvent e){
                signalButtons[0].setBackground(Color.GRAY);
                signalButtons[1].setBackground(Color.GRAY);;
                signalButtons[2].setBackground(Color.GREEN);;
            }
        });
        jfrm.setVisible(true);

    }
    public static void main(String[] args){
        new TrafficLights();
    }

}
```

Sample output


○ Red
○ Yellow
○ Green


● Red
○ Yellow
○ Green


○ Red
○ Yellow
● Green


○ Red
● Yellow
○ Green