11 Nov 2020 Rinoy Kuriyakose R3 56

**Experiment: 16**

**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

**Concept Used:**

Java Swing –Graphical User Interface and Event handling.

**Algorithm:**

Algorithm Calculator

1. Import java.awt, java.awt.event, java.swing and java.util classes

2. Initialise JFrame frame with title “ Calculator”

3. Initialise JPanel panel1, panel2

4. Set layout managers GridLayout(4,5), BorderLayout() for panel2 and frame respectively

5. frame.setSize(350,350)

6. frame.setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE)

//Components

7. JTextField input = new JTextField()

8. JButton b1 = new JButton("1")

9. JButton b2 = new JButton("2")

10. JButton b3 = new JButton("3")

11. JButton b4 = new JButton("4")

12. JButton b5 = new JButton("5")

13. JButton b6 = new JButton("6")

14. JButton b7 = newJButton("7")

15. JButton b8 = new JButton("8")

16. JButton b0 = new JButton("0"

17. JButton b9 = new JButton("9")

18. JButton b17 = new JButton("AC")

19. JButton b19 = new JButton(".")

20. JButton b10 = new JButton("+")

21. JButton b11 = new JButton("-")

22. JButton b13 = new JButton("\*")

23. JButton b12 = new JButton("/")

24. JButton b14 = new JButton("%")

25. JButton b16 = new JButton("=")

26. . JButton b15 = new JButton("^")

27. Add buttons to panel2

28. input.setHorizontalAlignment(JTextField.CENTER)

29. panel1.add(input)

30. frame.add(panel1,BorderLayout.NORTH)

31. frame.add(panel1,BorderLayout.CENTER)

32. frame.setVisible(true)

//Event Handling

//For the input buttons, the event handler is,

33. input.setText(result.getText() + "0") //Button 0

// Same for all other from 1-9

//For AC button, the event handler is,

34. input.setText("")

//For DEL button, the event handler is,

35. input.setText(input.setText(input.getText().substring(0, input.getText().length() - 1)

//For Equals button, the event handler is,

36. inp = input.getText()

37. i = 0

38. result = 0

39. op = exp.charAt(0)

40. try

41. while(z != '+' && z != '-' && z != '\*' && z != '/' && z != '%')

42. op = exp.charAt(i)

43. i++

44. endwhile

45. x = Float.parseFloat(exp.substring(0,i-1))

46. y = Float.parseFloat(exp.substring(i,inp.length()))

47. switch(op)

48. case + : result = x+y

49. break

50. case - : result = x-y

51. break

52. case \* : result = x\*y

53. break

54. case / : if(y == 0)

55. break

56. result = x/y

57. break

58. case %:result = x%y

59. break

60. case ^:result = Math.pow(x,y)

61. break

62. endcase

63. endswitch

64. if(y == 0)

65. result.setText("Not defined!")

66. else

67. result.setText(res+"")

68. endif

69. catch

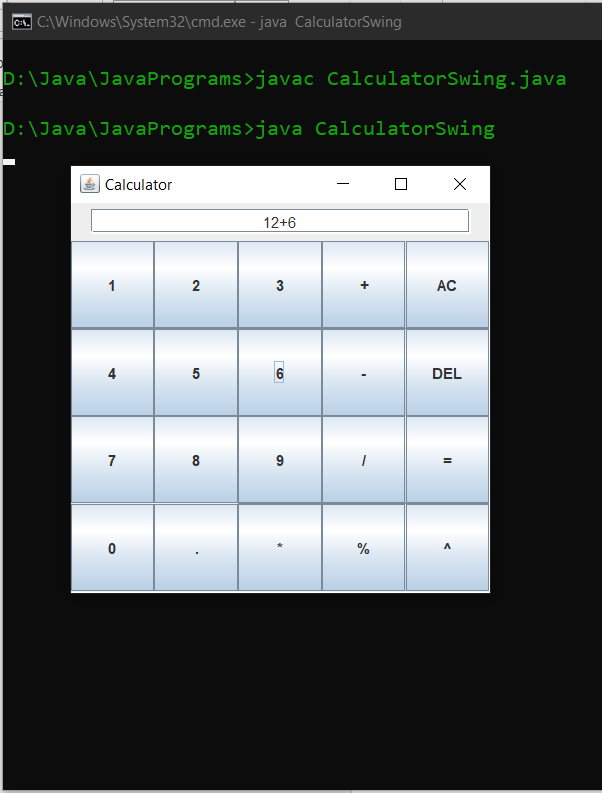
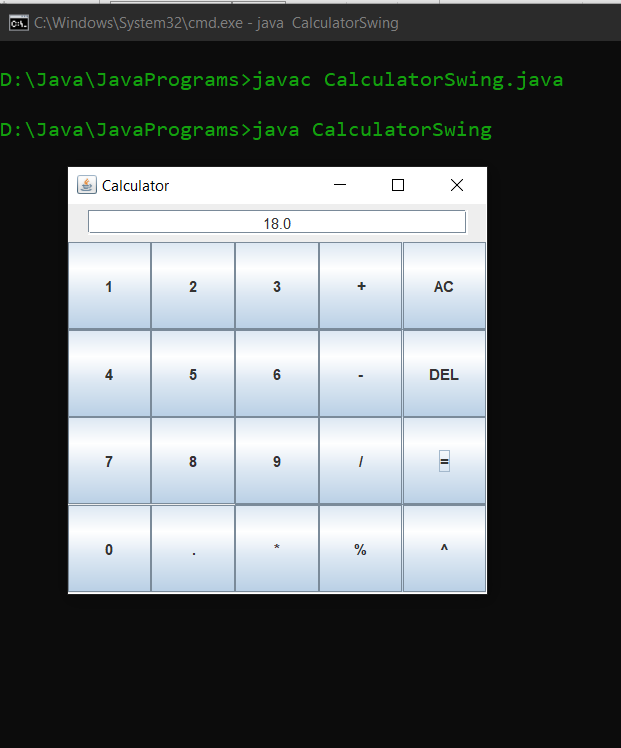
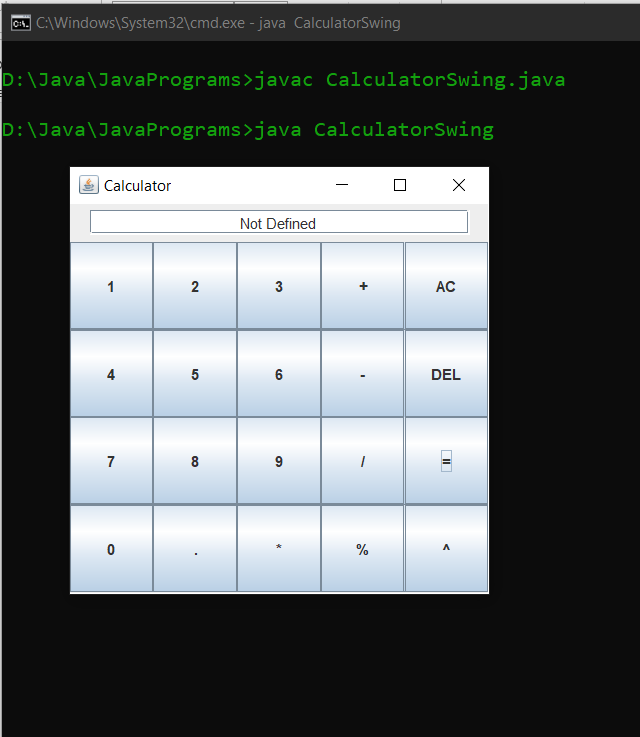
70. result.setText("Invalid Input")

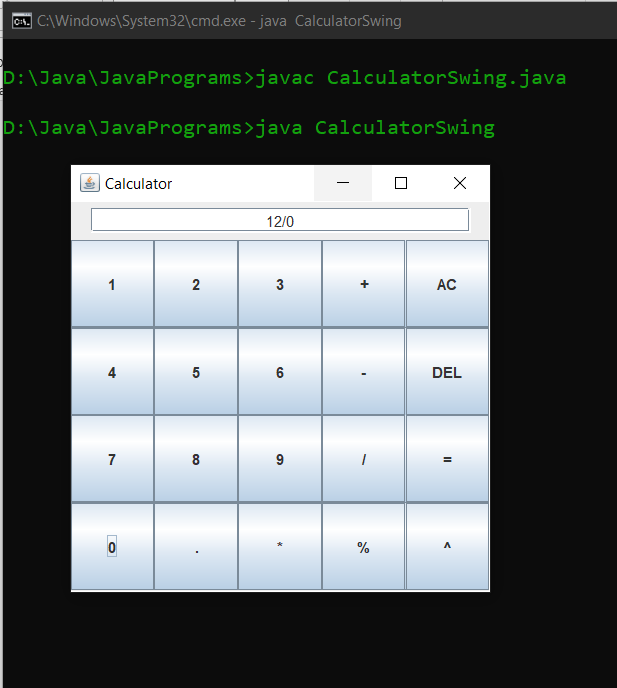
71. endtry

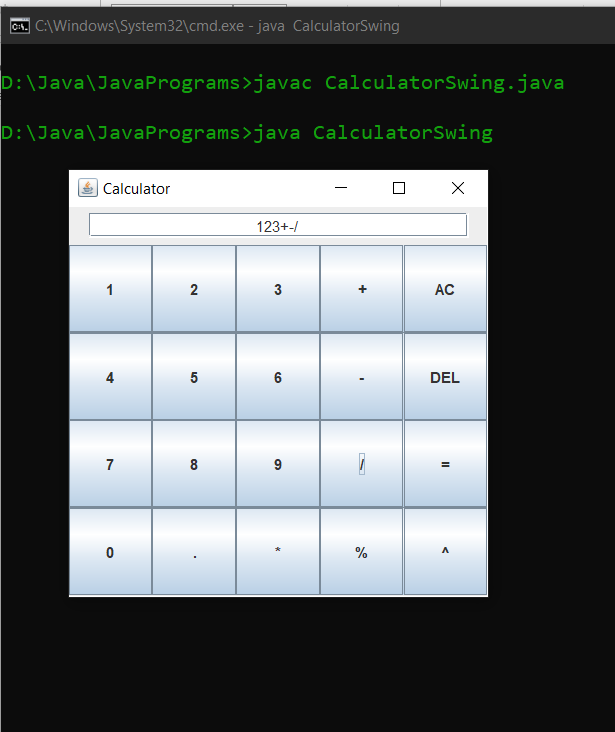
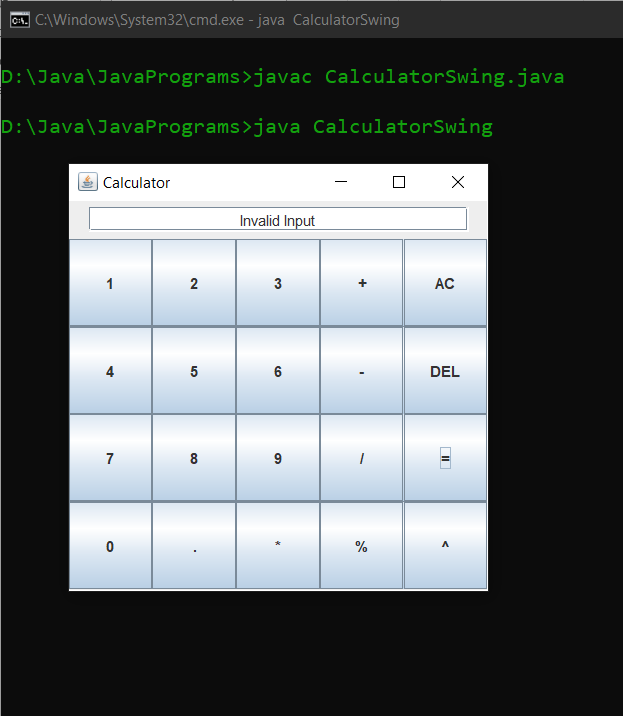
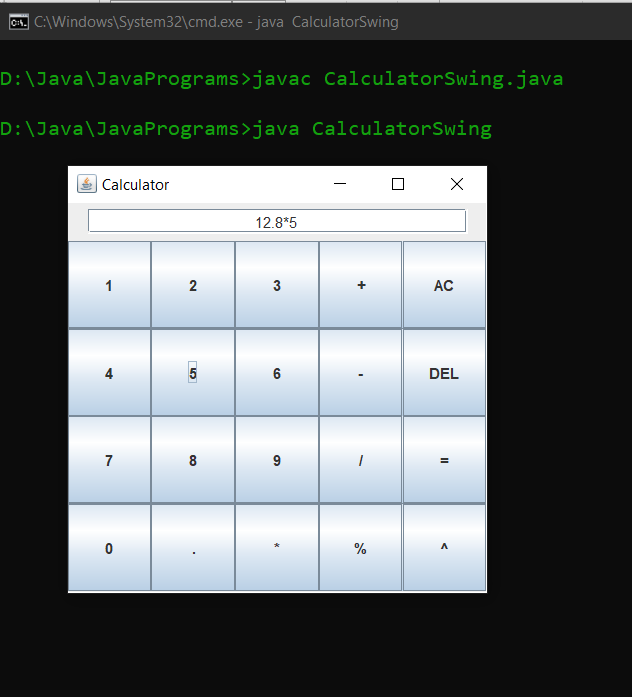
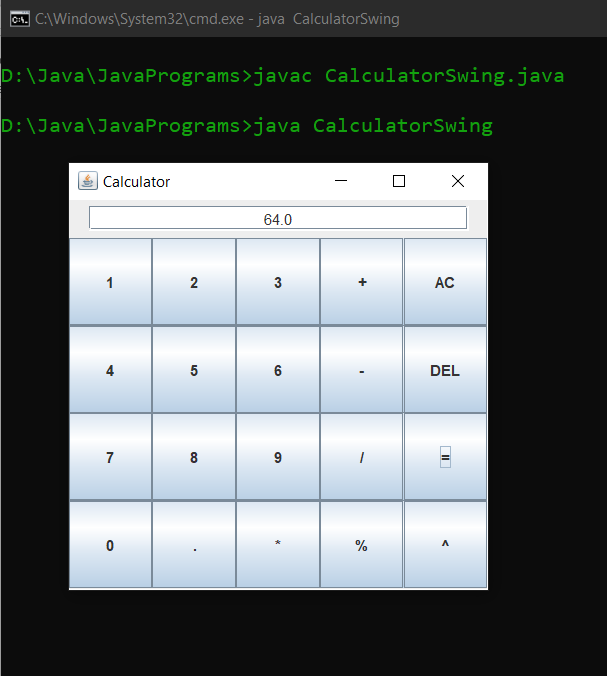
**Program:**

import java.awt.\*;  
import java.awt.event.\*;  
import javax.swing.\*;  
public class CalculatorSwing {  
 CalculatorSwing(){  
 JFrame frame=new JFrame("Calculator");  
 JPanel panel1=new JPanel();  
 JPanel panel2=new JPanel();  
  
 frame.setLayout(new BorderLayout());  
 panel2.setLayout(new GridLayout(4,5));  
 frame.setSize(350,350);  
  
 frame.setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);  
 JButton b0=new JButton("0");  
 JButton b1=new JButton("1");  
 JButton b2=new JButton("2");  
 JButton b3=new JButton("3");  
 JButton b4=new JButton("4");  
 JButton b5=new JButton("5");  
 JButton b6=new JButton("6");  
 JButton b7=new JButton("7");  
 JButton b8=new JButton("8");  
 JButton b9=new JButton("9");  
 JButton b10=new JButton("+");  
 JButton b11=new JButton("-");  
 JButton b12=new JButton("/");  
 JButton b13=new JButton("\*");  
 JButton b14=new JButton("%");  
 JButton b15=new JButton("^");  
 JButton b16=new JButton("=");  
 JButton b17=new JButton("AC");  
 JButton b18=new JButton("DEL");  
 JButton b19=new JButton(".");  
  
 JTextField input = new JTextField(30);  
  
 input.setHorizontalAlignment(JTextField.CENTER);  
 panel1.add(input);  
 panel2.add(b1);  
 panel2.add(b2);  
 panel2.add(b3);  
 panel2.add(b10);  
 panel2.add(b17);  
  
 panel2.add(b4);  
 panel2.add(b5);  
 panel2.add(b6);  
 panel2.add(b11);  
 panel2.add(b18);  
  
 panel2.add(b7);  
 panel2.add(b8);  
 panel2.add(b9);  
 panel2.add(b12);  
 panel2.add(b16);  
  
 panel2.add(b0);  
 panel2.add(b19);  
 panel2.add(b13);  
 panel2.add(b14);  
 panel2.add(b15);  
 frame.add(panel1,BorderLayout.NORTH);  
 frame.add(panel2,BorderLayout.CENTER);  
  
 frame.setVisible(true);  
 b0.addActionListener(new ActionListener(){  
 public void actionPerformed(ActionEvent ae){  
 if(input.getText().equals("Invalid Input")||input.getText().equals("Not Defined")){  
 input.setText("");  
 }  
 input.setText(input.getText()+"0");  
 }});  
 b1.addActionListener(new ActionListener(){  
 public void actionPerformed(ActionEvent ae){  
 if(input.getText().equals("Invalid Input")||input.getText().equals("Not Defined")){  
 input.setText("");  
 }  
 input.setText(input.getText()+"1");  
 }});  
 b2.addActionListener(new ActionListener(){  
 public void actionPerformed(ActionEvent ae){  
 if(input.getText().equals("Invalid Input")||input.getText().equals("Not Defined")){  
 input.setText("");  
 }  
 input.setText(input.getText()+"2");  
 }});  
 b3.addActionListener(new ActionListener(){  
 public void actionPerformed(ActionEvent ae){  
 if(input.getText().equals("Invalid Input")||input.getText().equals("Not Defined")){  
 input.setText("");  
 }  
 input.setText(input.getText()+"3");  
 }});  
 b4.addActionListener(new ActionListener(){  
 public void actionPerformed(ActionEvent ae){  
 if(input.getText().equals("Invalid Input")||input.getText().equals("Not Defined")){  
 input.setText("");  
 }  
 input.setText(input.getText()+"4");  
 }});  
 b5.addActionListener(new ActionListener(){  
 public void actionPerformed(ActionEvent ae){  
 if(input.getText().equals("Invalid Input")||input.getText().equals("Not Defined")){  
 input.setText("");  
 }  
 input.setText(input.getText()+"5");  
 }});  
 b6.addActionListener(new ActionListener(){  
 public void actionPerformed(ActionEvent ae){  
 if(input.getText().equals("Invalid Input")||input.getText().equals("Not Defined")){  
 input.setText("");  
 }  
 input.setText(input.getText()+"6");  
 }});  
 b7.addActionListener(new ActionListener(){  
 public void actionPerformed(ActionEvent ae){  
 if(input.getText().equals("Invalid Input")||input.getText().equals("Not Defined")){  
 input.setText("");  
 }  
 input.setText(input.getText()+"7");  
 }});  
 b8.addActionListener(new ActionListener(){  
 public void actionPerformed(ActionEvent ae){  
 if(input.getText().equals("Invalid Input")||input.getText().equals("Not Defined")){  
 input.setText("");  
 }  
 input.setText(input.getText()+"8");  
 }});  
 b9.addActionListener(new ActionListener(){  
 public void actionPerformed(ActionEvent ae){  
 if(input.getText().equals("Invalid Input")||input.getText().equals("Not Defined")){  
 input.setText("");  
 }  
 input.setText(input.getText()+"9");  
 }});  
 b10.addActionListener(new ActionListener(){  
 public void actionPerformed(ActionEvent ae){  
 if(input.getText().equals("Invalid Input")||input.getText().equals("Not Defined")){  
 input.setText("");  
 }  
 input.setText(input.getText()+"+");  
 }});  
 b11.addActionListener(new ActionListener(){  
 public void actionPerformed(ActionEvent ae){  
 if(input.getText().equals("Invalid Input")||input.getText().equals("Not Defined")){  
 input.setText("");  
 }  
 input.setText(input.getText()+"-");  
 }});  
 b12.addActionListener(new ActionListener(){  
 public void actionPerformed(ActionEvent ae){  
 if(input.getText().equals("Invalid Input")||input.getText().equals("Not Defined")){  
 input.setText("");  
 }  
 input.setText(input.getText()+"/");  
 }});  
 b13.addActionListener(new ActionListener(){  
 public void actionPerformed(ActionEvent ae){  
 if(input.getText().equals("Invalid Input")||input.getText().equals("Not Defined")){  
 input.setText("");  
 }  
 input.setText(input.getText()+"\*");  
 }});  
 b14.addActionListener(new ActionListener(){  
 public void actionPerformed(ActionEvent ae){  
 if(input.getText().equals("Invalid Input")||input.getText().equals("Not Defined")){  
 input.setText("");  
 }  
 input.setText(input.getText()+"%");  
 }});  
 b15.addActionListener(new ActionListener(){  
 public void actionPerformed(ActionEvent ae){  
 if(input.getText().equals("Invalid Input")||input.getText().equals("Not Defined")){  
 input.setText("");  
 }  
 input.setText(input.getText()+"^");  
 }});  
 b16.addActionListener(new ActionListener(){  
 public void actionPerformed(ActionEvent ae){  
 String inp = input.getText();  
 int i = 0;  
 float a, b, result = 0;  
 char op = inp.charAt(0);  
 try  
 {  
 while(op != '+' && op != '-' && op!= '\*' && op != '/' && op != '%' && op != '^')  
 {  
 op = inp.charAt(i);  
 i++;  
 }  
 a = Float.parseFloat(inp.substring(0,i-1));  
 b = Float.parseFloat(inp.substring(i,inp.length()));  
 switch(op)  
 {  
 case '+': result = a+b;  
 break;  
 case '-': result = a-b;  
 break;  
 case '/': try{  
 result = a/b;  
 }catch(Exception e1){}  
 break;  
 case '\*': result = a\*b;  
 break;  
  
 case '%': result = a%b;  
 break;  
 case '^': result = (float) Math.pow(a,b);  
 break;  
 }  
 if(b!=0){  
 input.setText(result+"");  
 }else{  
 input.setText("Not Defined");  
 }  
 }catch(Exception e2){  
 input.setText("Invalid Input");  
 }  
 }});  
 b17.addActionListener(new ActionListener(){  
 public void actionPerformed(ActionEvent ae){  
 input.setText("");  
 }});  
 b18.addActionListener(new ActionListener(){  
 public void actionPerformed(ActionEvent ae){  
 if(input.getText().equals("Invalid Input")||input.getText().equals("Not Defined")){  
 input.setText("");  
 }else {  
 try {  
 input.setText(input.getText().substring(0, input.getText().length() - 1));  
 } catch (Exception e3) {  
 }  
 }  
 }});  
 b19.addActionListener(new ActionListener(){  
 public void actionPerformed(ActionEvent ae){  
 if(input.getText().equals("Invalid Input")||input.getText().equals("Not Defined")){  
 input.setText("");  
 }  
 input.setText(input.getText()+".");  
 }});  
 }  
  
 public static void main(String[] args) {  
 new CalculatorSwing();  
 }  
}

**Output:**







**Result:**

A simple calculator with GUI is created using Java Swing.

11 Nov 2020 Rinoy Kuriyakose R3 56

**Experiment: 17**

**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.

**Concept Used:**

Java Swing – GUI and Event Handling.

**Algorithm:**

1.Import java.awt, java.awt.event and java.swing classes

2. JFrame frame = new JFrame("Traffic Light")

3. JRadioButton red = new JradioButton

4. JRadioButton yellow = new JradioButton

5. JRadioButton green = new JradioButton

6. ButtonGroup traffic = new ButtonGroup()

7. red.setBackground(Color.WHITE)

8. yellow.setBackground(Color.WHITE)

9. green.setBackground(Color.WHITE)

10. traffic.add(red)

11. traffic.add(yellow)

12.traffic.add(green)

13. frame.add(red)

14. frame.add(yellow)

15.frame.add(green)

16. frame.setSize(300,100)

17. frame.setLayout(new GridLayout(1,3))

18. frame.setVisible(true)

19. frame.setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE)

//Event handler code for red button

20. red.setBackground(Color.RED)

21. yellow.setBackground(Color.WHITE)

22. green.setBackground(Color.WHITE)

23. red.setText("STOP")

24. yellow.setText("")

25. green.setText("")

//Event handler code for yellow button

26. red.setBackground(Color.WHITE)

27. yellow.setBackground(Color.YELLOW)

28. green.setBackground(Color.WHITE)

29. red.setText("")

30. yellow.setText("WAIT")

31. green.setText("")

//Event handler code for RED button

32. red.setBackground(Color.WHITE)

33. yellow.setBackground(Color.WHITE)

34. green.setBackground(Color.GREEN)

35. red.setText("")

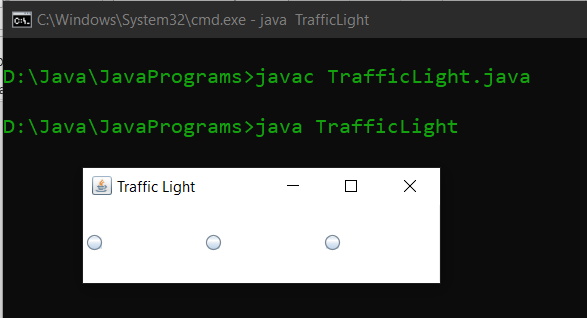
36. yellow.setText("")

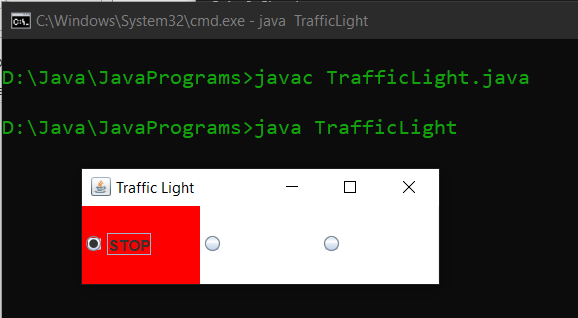
37. green.setText("GO")

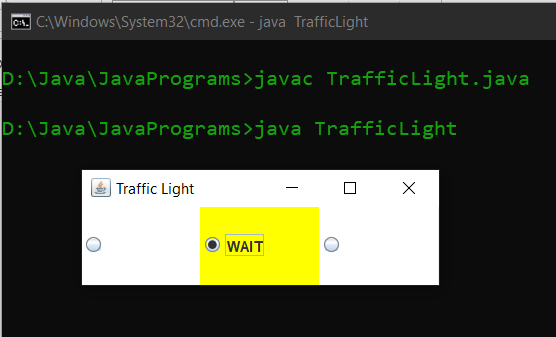
**Program:**

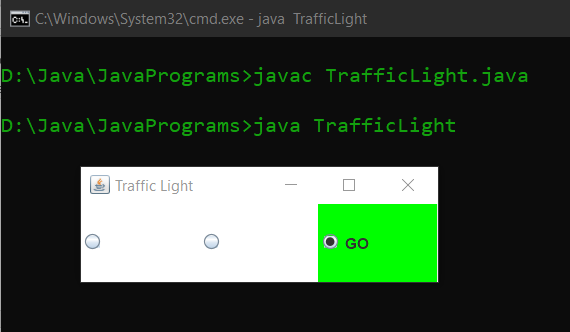
import java.awt.\*;  
import java.awt.event.\*;  
import javax.swing.\*;  
public class TrafficLight  
{  
 TrafficLight() {  
 JFrame frame = new JFrame("Traffic Light");  
 JRadioButton red = new JRadioButton();  
 JRadioButton yellow = new JRadioButton();  
 JRadioButton green = new JRadioButton();  
 ButtonGroup traffic = new ButtonGroup();  
 traffic.add(red);  
 traffic.add(yellow);  
 traffic.add(green);  
 red.setBackground(Color.WHITE);  
 yellow.setBackground(Color.WHITE);  
 green.setBackground(Color.WHITE);  
 frame.add(red);  
 frame.add(yellow);  
 frame.add(green);  
 frame.setSize(300,100);  
 frame.setLayout(new GridLayout(1,3));  
 frame.setVisible(true);  
 frame.setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);  
 red.addActionListener(new ActionListener(){  
 public void actionPerformed(ActionEvent ae){  
 red.setBackground(Color.RED);  
 yellow.setBackground(Color.WHITE);  
 green.setBackground(Color.WHITE);  
 red.setText("STOP");  
 yellow.setText("");  
 green.setText("");  
 }  
 });  
 yellow.addActionListener(new ActionListener() {  
 public void actionPerformed(ActionEvent ae){  
 red.setBackground(Color.WHITE);  
 yellow.setBackground(Color.YELLOW);  
 green.setBackground(Color.WHITE);  
 red.setText("");  
 yellow.setText("WAIT");  
 green.setText("");  
 }  
 });  
 green.addActionListener(new ActionListener() {  
 public void actionPerformed(ActionEvent ae){  
 red.setBackground(Color.WHITE);  
 yellow.setBackground(Color.WHITE);  
 green.setBackground(Color.GREEN);  
 red.setText("");  
 yellow.setText("");  
 green.setText("GO");  
 }  
 });  
 }  
 public static void main(String[] args)  
 {  
 new TrafficLight();  
 }  
}

**Output:**









**Result:**

A traffic light is stimulated using Java Swing.