

```

1
2 import java.awt.*;
3 import java.awt.event.*;
4 public class TextCopy extends Frame implements ActionListener
5 {
6     TextField text1,text2;
7     Label t1,t2;
8     Button paste;
9     public TextCopy(){
10         setLayout(new FlowLayout());
11         paste=new Button("Paste");
12         Label t1=new Label("Text1: ",Label.RIGHT);
13         Label t2=new Label("Text2: ",Label.RIGHT);
14         text1=new TextField(12);
15         text2=new TextField(12);
16         add(t1);
17         add(text1);
18         add(paste);
19         add(t2);
20         add(text2);
21
22         paste.addActionListener(this);
23         addWindowListener(new WindowAdapter1());
24     }
25     public void actionPerformed(ActionEvent ae)
26     {
27         if(ae.getSource()==paste)
28         {
29             String tf=text1.getText();
30             text2.setText(tf);
31         }
32     }
33
34 Run | Debug
35 public static void main(String args[]) {
36     TextCopy tc = new TextCopy();
37     tc.setSize(new Dimension(400, 400));
38     tc.setTitle("TEXT COPY PASTE");
39     tc.setVisible(true);
40 }

```

TextCopy.java × TextFieldFour.java

TextCopy.java > TextCopy > actionPerformed(ActionEvent)

```
27 if(ae.getSource()==paste)
28 {
29     String tf=text1.getText();
30     text2.setText(tf);
31 }
32 }
```

Run | Debug

```
33 public static void main(String args[]) {
34     TextCopy tc = new TextCopy();
35     tc.setSize(new Dimension(400, 400));
36     tc.setTitle("TEXT COPY PASTE");
37     tc.setVisible(true);
38 }
39
40 class WindowAdapter1 extends WindowAdapter {
41     public void windowClosing(WindowEvent we) {
42         System.exit(0);
43     }
44 }
45 }
```

OUTPUT DEBUG CONSOLE TERMINAL

1: cmd

C:\Users\akki\Desktop\java files>javac TextCopy.java

C:\Users\akki\Desktop\java files>java TextCopy

C:\Users\akki\Desktop\java files>

Text1:   Text2:

```

1
2 import java.awt.*;
3 import java.awt.event.*;
4 public class TextFieldFour extends Frame implements ActionListener{
5     TextField tf1,tf2,tf3,tf4;
6     Label t1,t2,t3,t4;
7     Button b;
8     public TextFieldFour(){
9         setLayout(new FlowLayout());
10        Label t1=new Label("NUM1: ",Label.RIGHT);
11        Label t2=new Label("NUM2: ",Label.RIGHT);
12        Label t3=new Label("Arithmetic operator: ",Label.RIGHT);
13        Label t4=new Label("Result:",Label.RIGHT);
14        b=new Button("Result");
15        tf1=new TextField(12);
16        tf2=new TextField(12);
17        tf3=new TextField(12);
18        tf4=new TextField(12);
19        add(t1);
20        add(tf1);
21        add(t2);
22        add(tf2);
23        add(t3);
24        add(tf3);
25        add(b);
26        add(t4);
27        add(tf4);
28        b.addActionListener(this);
29        addWindowListener(new WindowAdapter1());
30    }
31    public void actionPerformed(ActionEvent ae) {
32        if (ae.getSource() == b) {
33
34            int n1 = Integer.parseInt(tf1.getText());
35            int n2 = Integer.parseInt(tf2.getText());
36            int res = 0;
37            String op = tf3.getText();
38            switch(op){
39                case "+": res = n1+n2;
40                break;

```

```

38         switch(op){
39             case "+": res = n1+n2;
40             break;
41
42             case "-": res = n1-n2;
43             break;
44
45             case "*": res = n1 * n2;
46             break;
47
48             case "/": res = n1 / n2;
49         }
50         tf4.setText(String.valueOf(res));
51     }
52 }
53 }

```

Run | Debug

```

54 public static void main(String args[]){
55     TextFieldFour obj=new TextFieldFour();
56     obj.setSize(new Dimension(400, 400));
57     obj.setTitle("Arithmetic Operation");
58     obj.setVisible(true);
59 }
60 class WindowAdapter1 extends WindowAdapter {
61     public void windowClosing(WindowEvent we) {
62         System.exit(0);
63     }
64 }
65 }
66

```

OUTPUT    DEBUG CONSOLE    TERMINAL

1: java

C:\Users\akki\Desktop\java files>javac TextFieldFour.java

C:\Users\akki\Desktop\java files>java TextFieldFour

NUM1:  NUM2:  Arithmetic operator:   Result: