```
 1  // joi/10/joi/JOIPanel.java
 2  //
 3  //
 4  // Copyright 2003 Bill Campbell and Ethan Bolker
 5
 6  import java.applet.*;
 7  import java.awt.*;
 8  import java.awt.event.*;
 9
10  /**
11   * A JOIPanel displays a button and a message.
12   * Pushing the button changes the message.
13   *
14   * This panel can be displayed either from an applet
15   * in a browser or by the JVM as an application.
16   *
17   * @version 10
18   */
19
20  public class JOIPanel extends Applet
21  {
22      private static final String MESSAGE1 = "Java Outside In";
23      private static final String MESSAGE2 = "Java Inside Out";
24      private String currentMessage = MESSAGE1; // currently displayed
25
26      private Font font;                // for printing the message
27      private Button button;            // for changing messages
28
29      /**
30       * Equip this Panel with a Button
31       * and an associated ButtonListener, and
32       * set the font for the message.
33       */
34
35      public void init()
36      {
37          // what this Panel looks like
38          button = new Button( "Press Me" );
39          this.add( button );
40          font = new Font("Garamond", Font.BOLD, 48);
41
42          // how this Panel behaves
43          button.addActionListener( new JOIButtonListener( this ) );
44      }
45
46      /**
47       * Method that responds when the ButtonListener sends a
48       * changeMessage message.
49       */
50
51      public void changeMessage()
52      {
53          currentMessage =
54              currentMessage.equals(MESSAGE1) ? MESSAGE2 : MESSAGE1;
55          this.repaint();
56      }
```



```
57
58      /**
59       * Draw the current message on this Panel.
60       *
61       * (The button is already there.)
62       *
63       * @param g an object encapsulating the graphics (e.g. pen)
64       *          properties.
65       */
66
67      public void paint(Graphics g)
68      {
69          g.setColor(Color.black);
70          g.setFont(font);
71          g.drawString(currentMessage, 40, 75);
72      }
73
74      /**
75       * Ask the JVM to display this Panel.
76       */
77
78      public static void main( String[] args )
79      {
80          Terminal t     = new Terminal();
81          Frame frame    = new Frame();
82          JOIPanel panel = new JOIPanel();
83          panel.init();
84          frame.add(panel);
85          frame.setSize(400,120);
86          frame.show();
87          t.readLine("Type return to close the window ... ");
88          System.exit(0);
89      }
90  }
```