

```
24
25 // Fuege die Komponenten dem Frame hinzu
26 c.add(b1);
27 c.add(b2);
28 c.add(b3);
29 c.add(cb);
30
31 // Erzeuge den Listener und registriere ihn
32 LafListener ll = new LafListener();
33 b1.addActionListener(ll);
34 b2.addActionListener(ll);
35 b3.addActionListener(ll);
36 cb.addItemListener(ll);
37 }
38 // Innere Listener-Klasse
39 public class LafListener implements ItemListener, ActionListener {
40     // Fuer die Look-and-feel-Auswahl
41     String[] laf =
42         {"javax.swing.plaf.metal.MetalLookAndFeel",
43          "com.sun.java.swing.plaf.motif.MotifLookAndFeel",
44          "com.sun.java.swing.plaf.windows.WindowsLookAndFeel"};
45
46     // Fuer das ItemListener-Interface
47     public void itemStateChanged(ItemEvent e) {
48         try {
49             int i = cb.getSelectedIndex();
50             UIManager.setLookAndFeel(laf[i]);
51         }
52         catch (Exception ex) {
53             System.err.println(ex);
54         }
55         SwingUtilities.updateComponentTreeUI(f);
56     }
57     // Fuer das ActionListener-Interface
58     public void actionPerformed(ActionEvent e) {
59         try {
60             int i;
61             if (e.getSource() == b1)
62                 i = 0;
63             else if (e.getSource() == b2)
64                 i = 1;
65             else
66                 i = 2;
67             UIManager.setLookAndFeel(laf[i]);
68             cb.setSelectedIndex(i);
69         }
70         catch (Exception ex) {
71             System.err.println(ex);
72         }
73         SwingUtilities.updateComponentTreeUI(f);
74     }
75 }
76 public static void main(String[] args) {
77     LookAndFeel fenster = new LookAndFeel();
78     fenster.setTitle("Look and feel einstellen");
```