

Import Statements

```
1 import java.awt.*;
2 import java.awt.event.*;
3 import javax.swing.*;
4 import java.applet.AudioClip;
5 import java.net.URL;
```

Components are *Instance Data Items*

```
6 public class JukeBoxControls2 extends JPanel
7 {
8     private JComboBox musicCombo;
9     private JButton stopButton, playButton;
10    private AudioClip[] music;
11    private AudioClip current;
12
13    //-----
14    // Sets up the GUI for the juke box.
15    //-----
16    public JukeBoxControls2()
17    {
18        URL url1, url2, url3, url4, url5, url6;
19        url1 = url2 = url3 = url4 = url5 = url6 = null;
20
21        // Obtain and store the audio clips to play
22        try
23        {
24            url1 = new URL ("file", "localhost", "westernBeat.wav");
25            url2 = new URL ("file", "localhost", "classical.wav");
26            url3 = new URL ("file", "localhost", "jeopardy.au");
27            url4 = new URL ("file", "localhost", "newAgeRythm.wav");
28            url5 = new URL ("file", "localhost", "eightiesJam.wav");
29            url6 = new URL ("file", "localhost", "hitchcock.wav");
30        }
31        catch (Exception exception) {}
32
33        music = new AudioClip[7];
34        music[0] = null; // Corresponds to "Make a Selection..."
35        music[1] = JApplet.newAudioClip (url1);
36        music[2] = JApplet.newAudioClip (url2);
37        music[3] = JApplet.newAudioClip (url3);
38        music[4] = JApplet.newAudioClip (url4);
39        music[5] = JApplet.newAudioClip (url5);
40        music[6] = JApplet.newAudioClip (url6);
41
42        // Create the list of strings for the combo box options
43        String[] musicNames = {"Make A Selection...", "Western Beat",
44                               "Classical Melody", "Jeopardy Theme", "New Age Rythm",
45                               "Eighties Jam", "Alfred Hitchcock's Theme"};
46
47        musicCombo = new JComboBox (musicNames);
48        musicCombo.setAlignmentX (Component.CENTER_ALIGNMENT);
49
50        // Set up the buttons
51        playButton = new JButton ("Play", new ImageIcon ("play.gif"));
52        playButton.setBackground (Color.white);
53        playButton.setMnemonic ('p');
54        stopButton = new JButton ("Stop", new ImageIcon ("stop.gif"));
55        stopButton.setBackground (Color.white);
56        stopButton.setMnemonic ('s');
57
58        JPanel buttons = new JPanel();
59        buttons.setLayout (new BoxLayout (buttons, BoxLayout.X_AXIS));
60        buttons.add (playButton);
61        buttons.add (Box.createRigidArea (new Dimension(5,0)));
62        buttons.add (stopButton);
63        buttons.setBackground (Color.cyan);
64
65        // Set up this panel
66        setPreferredSize (new Dimension (300, 100));
67        setBackground (Color.cyan);
68        setLayout (new BoxLayout (this, BoxLayout.Y_AXIS));
69        add (Box.createRigidArea (new Dimension(0,5)));
70        add (musicCombo);
71        add (Box.createRigidArea (new Dimension(0,5)));
72        add (buttons);
73        add (Box.createRigidArea (new Dimension(0,5)));
74
75        musicCombo.addActionListener (new ComboListener());
76        stopButton.addActionListener (new ButtonListener());
77        playButton.addActionListener (new ButtonListener());
78
79        current = null;
80    }
81
82    //*****
83    // Represents the action listener for the combo box.
84    //*****
85    private class ComboListener implements ActionListener
86    {
87        //-----
88        // Stops playing the current selection (if any) and resets
89        // the current selection to the one chosen.
90        //-----
91        public void actionPerformed (ActionEvent event)
92        {
93            if (current != null)
94                current.stop();
95
96            current = music[musicCombo.getSelectedIndex()];
97        }
98    }
99
100    //*****
101    // Represents the action listener for both control buttons.
102    //*****
103    private class ButtonListener implements ActionListener
104    {
105        //-----
106        // Stops the current selection (if any) in either case. If
107        // the play button was pressed, start playing it again.
108        //-----
109        public void actionPerformed (ActionEvent event)
110        {
111            if (current != null)
112                current.stop();
113
114            if (event.getSource() == playButton)
115                if (current != null)
116                    current.play();
117        }
118    }
119 }
120
```

Constructor

Components are instantiated
and added to the panel

(Ignore the Layout Manager for now)

Listeners are instantiated and added to
(associated with) components. Notice the
use of a listener named *ComboListener* and
ButtonListener

ComboListener implements *ActionListener*

ButtonListener implements *ActionListener*

This is an example of two listeners in one panel.

Each listener appears after the constructor but within the class.
Because *ComboListener* and *ButtonListener* are both *ActionListeners*,
they could be combined.
However, that is only possible because they are the same type of listener.