```
Loading vc-cvs...
package caltool.options_ui;

import caltool.options.*;
import javax.swing.*;
import java.awt.*;
import java.awt.event.*;
import mvp.*;

/****
 *
 * Class OptionsMenu is the pulldown menu view of the <a href = "Options.html"
 * Options </a> model class.  The OptionsMenu widget is a Java JMenu.
 * Anonymous instances of JMenuItem are defined for each item in the menu.
 *
 */
public class OptionsMenu extends mvp.View {

    /**
     * Construct this with the given Options model.
     */
    public OptionsMenu(Screen screen, Options options, OptionsUI optionsUI) {
        super(screen, options);
        this.optionsUI = optionsUI;
    }

    /**
     * Compose this by inserting each of its three menu items into the pulldown
     * menu.  The items are Times and Dates, Scheduling, Viewing, and
     * Administrative.
     */
    public Component compose() {

        /*
         * Make the widget of this the JMenu.
         */
        widget = new JMenu("Options");

        /*
         * Create a constant reference to the Options model for use in the
         * menu item action listeners.  This is necessary due to the subtle
         * rules of anonymous inner classes in Java.
         */
        final Options options = (Options) model;

        /*
         * Create the menu items using the following pattern:
         *                                                                  <pre>
         *     JMenu.add(new JMenuItem("<em>Item name</em>").addActionListener(
         *         new ActionListener() {
         *             public void actionPerformed(ActionEvent e) {
         *                 <em>Model.method()</em>
         *                                                                  </pre>
         */


        /*
         * Add the 'Times and Dates' menu item.
         */
        ((JMenu) widget).add(new JMenuItem("Times and Dates ...")).addActionListener(
            new ActionListener() {
                public void actionPerformed(ActionEvent e) {
                    optionsUI.show();
                    optionsUI.selectTab("Times and Dates");
//                  System.out.println("In Options.timesAndDates.");
                }
            }
        );

        /*
         * add the 'Fonts ...' menu item.););
         */
        ((JMenu) widget).add(new JMenuItem("Fonts ...")).addActionListener(
            new ActionListener() {
                public void actionPerformed(ActionEvent e) {
                    optionsUI.show();
                    optionsUI.selectTab("Fonts");
//                  System.out.println("In Options.viewing.");
                }
            }
        );

        /*
         * Add the 'Scheduling ...' menu item.);
         */
        ((JMenu) widget).add(new JMenuItem("Scheduling ...")).addActionListener(
            new ActionListener() {
                public void actionPerformed(ActionEvent e) {
                    optionsUI.show();
                    optionsUI.selectTab("Scheduling");
//                  System.out.println("In Options.scheduling.");
                }
            }
        );

        /*
         * add the 'Viewing ...' menu item.););
         */
        ((JMenu) widget).add(new JMenuItem("Viewing ...")).addActionListener(
            new ActionListener() {
                public void actionPerformed(ActionEvent e) {
                    optionsUI.show();
                    optionsUI.selectTab("Viewing");
//                  System.out.println("In Options.viewing.");
                }
            }
        );

        /*
         * add the 'Administrative ...' menu item.););
         */
        ((JMenu) widget).add(new JMenuItem("Administrative ...")).addActionListener(
```

```
112             new ActionListener() {
113                 public void actionPerformed(ActionEvent e) {
114                     optionsUI.show();
115                     optionsUI.selectTab("Administrative");
116 //                  System.out.println("In Options.administrative.");
117                 }
118             }
119         );
120 
121         /*
122          * Add a separator.
123          */
124         ((JMenu) widget).add(new JSeparator());
125 
126         /*
127          * add the 'Global ...' menu item.););
128          */
129         ((JMenu) widget).add(new JMenuItem("Global ...")).addActionListener(
130             new ActionListener() {
131                 public void actionPerformed(ActionEvent e) {
132                     System.out.println("In OptionsMenu 'Global' listener.");
133                 }
134             }
135         );
136 
137         /*
138          * Add a separator.
139          */
140         ((JMenu) widget).add(new JSeparator());
141 
142         /*
143          * add the 'Restore Defaults' menu item.););
144          */
145         ((JMenu) widget).add(new JMenuItem("Restore Defaults")).addActionListener(
146             new ActionListener() {
147                 public void actionPerformed(ActionEvent e) {
148                     System.out.println("In OptionsMenu 'Restore Defaults' listener.");
149                 }
150             }
151         );
152 
153         return widget;
154     }
155 
156     /** Parent view */
157     protected OptionsUI optionsUI;
158 
159 }
```