

```

Loading vc-cvs...
1 package caltool.edit_ui;
2
3 import caltool.edit.*;
4 import javax.swing.*;
5 import java.awt.*;
6 import java.awt.event.*;
7 import.mvp.*;
8
9 /**
10 *
11 * Class EditMenu is the pulldown menu view of the <a href = "Edit.html" Edit
12 * </a> model class. The EditMenu widget is a Java JMenu. Anonymous instances
13 * of JMenuItem are defined for each item in the menu.
14 *
15 */
16 public class EditMenu extends View {
17
18     /**
19     * Construct this with the given name as the pulldown label. The given
20     * Edit is the companion model.
21     */
22     public EditMenu(Screen screen, Edit edit, EditUI editUI) {
23         super(screen, edit);
24     }
25
26     /**
27     * Compose this by inserting each of its ten menu items into the pulldown
28     * menu. The items are Undo, Redo, Cut, copy, Past, Delete, Select All,
29     * and Find. JSeparators are placed after the 'Redo' and 'Select All'
30     * items.
31     */
32     public Component compose() {
33
34         /**
35         * Make the widget of this the JMenu.
36         */
37         widget = new JMenu("Edit");
38
39         /**
40         * Create a constant reference to the File model for use in the menu
41         * item action listeners. This is necessary due to the subtle rules of
42         * anonymous inner classes in Java.
43         */
44         final Edit edit = (Edit) model;
45
46         /**
47         * Create the menu items using the following pattern:
48         *
49         *      JMenu.add(new JMenuItem("<em>Item name</em>").addActionListener(
50         *          new ActionListener() {
51         *              public void actionPerformed(ActionEvent e) {
52         *                  <em>Model.method()</em>
53         *              }
54         *          }
55         *      );
56
57         /**
58         * Add the 'Undo' menu item.
59         */
60         ((JMenu) widget).add(new JMenuItem("Undo")).addActionListener(
61             new ActionListener() {
62                 public void actionPerformed(ActionEvent ev) {
63                     edit.undo();
64                 }
65             }
66         );
67
68         /**
69         * Add the 'Redo' menu item.
70         */
71         ((JMenu) widget).add(new JMenuItem("Redo")).addActionListener(
72             new ActionListener() {
73                 public void actionPerformed(ActionEvent ev) {
74                     edit.redo();
75                 }
76             }
77         );
78
79         /**
80         * Add a separator.
81         */
82         ((JMenu) widget).add(new JSeparator());
83
84         /**
85         * Add the 'Cut' menu item.
86         */
87         ((JMenu) widget).add(new JMenuItem("Cut")).addActionListener(
88             new ActionListener() {
89                 public void actionPerformed(ActionEvent ev) {
90                     edit.cut();
91                 }
92             }
93         );
94
95         /**
96         * Add the 'Copy' menu item.
97         */
98         ((JMenu) widget).add(new JMenuItem("Copy")).addActionListener(
99             new ActionListener() {
100                 public void actionPerformed(ActionEvent ev) {
101                     edit.copy();
102                 }
103             }
104         );
105
106         /**
107         * Add the 'Paste' menu item.
108         */
109         ((JMenu) widget).add(new JMenuItem("Paste")).addActionListener(
110

```

```
112         new ActionListener() {
113             public void actionPerformed(ActionEvent ev) {
114                 edit.paste();
115             }
116         }
117     );
118
119     /*
120     * Add the 'Delete' menu item.
121     */
122     ((JMenu) widget).add(new JMenuItem("Delete")).addActionListener(
123         new ActionListener() {
124             public void actionPerformed(ActionEvent ev) {
125                 edit.delete();
126             }
127         }
128     );
129
130     /*
131     * Add the 'Select All' menu item.
132     */
133     ((JMenu) widget).add(new JMenuItem("Select All")).addActionListener(
134         new ActionListener() {
135             public void actionPerformed(ActionEvent ev) {
136                 edit.selectAll();
137             }
138         }
139     );
140
141     /*
142     * Add a separator.
143     */
144     ((JMenu) widget).add(new JSeparator());
145
146     /*
147     * Add the 'Find ...' menu item.
148     */
149     ((JMenu) widget).add(new JMenuItem("Find ...")).addActionListener(
150         new ActionListener() {
151             public void actionPerformed(ActionEvent ev) {
152                 JOptionPane.showInputDialog("Search for: ");
153                 ((Edit) model).find(null);
154             }
155         }
156     );
157
158     return widget;
159 }
160
161 }
```