

```

Loading vc-cvs...
1  package caltool.options_ui;
2
3  import caltool.caltool_ui.BoxPanel;
4  import caltool.options.*;
5  import mvp.*;
6  import javax.swing.*;
7  import java.awt.*;
8  import java.awt.event.*;
9
10 public class ViewingListsAppointmentsOptionsPanel extends View {
11
12     public ViewingListsAppointmentsOptionsPanel(
13         Screen screen, Options options) {
14         super(screen, options);
15         maxTextFieldHeight = 1.9;
16     }
17
18     public Component compose() {
19
20         BoxPanel outerVbox = BoxPanel.createVerticalBox();
21         BoxPanel outerHbox = BoxPanel.createHorizontalBox();
22         BoxPanel hbox1 = BoxPanel.createHorizontalBox();
23         BoxPanel hbox2 = BoxPanel.createHorizontalBox();
24         BoxPanel hbox3 = BoxPanel.createHorizontalBox();
25         BoxPanel hbox4 = BoxPanel.createHorizontalBox();
26         BoxPanel hbox5 = BoxPanel.createHorizontalBox();
27         BoxPanel hbox6 = BoxPanel.createHorizontalBox();
28         BoxPanel vbox1 = BoxPanel.createVerticalBox();
29         BoxPanel vbox2 = BoxPanel.createVerticalBox();
30         BoxPanel vbox3 = BoxPanel.createVerticalBox();
31         BoxPanel vbox4 = BoxPanel.createVerticalBox();
32         BoxPanel vbox5 = BoxPanel.createVerticalBox();
33         BoxPanel vbox6 = BoxPanel.createVerticalBox();
34         JLabel label;
35
36         vbox1.add(label = new JLabel("Show these fields:"));
37         label.setAlignmentX(Component.LEFT_ALIGNMENT);
38         vbox2.add(titleCheckBox = new JCheckBox("Title"));
39         titleCheckBox.setSelected(true);
40         vbox2.add(dateCheckBox = new JCheckBox("Date"));
41         dateCheckBox.setSelected(true);
42         vbox2.add(timeCheckBox = new JCheckBox("Time"));
43         timeCheckBox.setSelected(true);
44         vbox2.add(durationCheckBox = new JCheckBox("Duration"));
45         durationCheckBox.setSelected(true);
46         vbox2.add(recursCheckBox = new JCheckBox("Recurs"));
47         recursCheckBox.setSelected(true);
48         vbox2.setAlignmentY(Component.TOP_ALIGNMENT);
49
50         vbox3.add(categoryCheckBox = new JCheckBox("Category"));
51         categoryCheckBox.setSelected(true);
52         vbox3.add(locationCheckBox = new JCheckBox("Location"));
53         locationCheckBox.setSelected(true);
54         vbox3.add(securityCheckBox = new JCheckBox("Security"));
55         securityCheckBox.setSelected(true);
56
57         vbox3.add(priorityCheckBox = new JCheckBox("Priority"));
58         priorityCheckBox.setSelected(true);
59         vbox3.setAlignmentY(Component.TOP_ALIGNMENT);
60
61         hbox1.add(vbox2);
62         hbox1.add(BoxPanel.createHorizontalStrut(10));
63         hbox1.add(vbox3);
64         hbox1.setAlignmentX(Component.LEFT_ALIGNMENT);
65         vbox1.add(hbox1);
66         vbox1.setAlignmentY(Component.TOP_ALIGNMENT);
67
68         hbox2.add(new JLabel("Primary sort key: "));
69         String[] selections = {"Title", "Date", "Time", "Duration", "Recurs",
70             "Category", "Location", "Security", "Priority"};
71         hbox2.add(sortKeyCombo = new JComboBox(selections));
72         sortKeyCombo.setSelectedItem("Date");
73         sortKeyCombo.setMaximumSize(
74             new Dimension(
75                 (int)("Category".length() * sortKeyCombo.getFont().getSize()),
76                 (int)(maxTextFieldHeight * sortKeyCombo.getFont().getSize())));
77         hbox2.setAlignmentX(Component.LEFT_ALIGNMENT);
78
79         hbox3.add(new JLabel("Maximum number of items: "));
80         hbox3.add(maxItemsTextField = new JTextField());
81         maxItemsTextField.setMaximumSize(
82             new Dimension(
83                 (int)(4 * maxItemsTextField.getFont().getSize()),
84                 (int)(maxTextFieldHeight *
85                     maxItemsTextField.getFont().getSize())));
86         hbox3.setAlignmentX(Component.LEFT_ALIGNMENT);
87
88         hbox4.add(new JLabel("Initial display height: "));
89         hbox4.add(displayHeightTextField = new JTextField());
90         displayHeightTextField.setMaximumSize(
91             new Dimension(
92                 (int)(4 * maxItemsTextField.getFont().getSize()),
93                 (int)(maxTextFieldHeight *
94                     maxItemsTextField.getFont().getSize())));
95         hbox4.add(new JLabel(" items"));
96         hbox4.setAlignmentX(Component.LEFT_ALIGNMENT);
97
98         hbox5.add(label = new JLabel("Start Date: "));
99         label.setAlignmentY(Component.TOP_ALIGNMENT);
100        hbox5.add(startDateTextField = new JTextField(" 1"));
101        startDateTextField.setMaximumSize(
102            new Dimension(
103                (int)(3 * startDateTextField.getFont().getSize()),
104                (int)(maxTextFieldHeight *
105                    startDateTextField.getFont().getSize())));
106        /*
107        * It seems that Java ignores the following minimum setting, but we do
108        * it in case things ever get better.
109        */
110        startDateTextField.setMinimumSize(
111            new Dimension(
112                (int)(3 * startDateTextField.getFont().getSize()),

```

```

112         (int)(maxTextFieldHeight *
113             startDateTextField.getFont().getSize()));
114     String[] selections2 = {"any", "today", "days", "weeks", "months",
115         "years", "specific date"};
116     startDateTextField.setAlignmentY(Component.TOP_ALIGNMENT);
117     hbox5.add(BoxPanel.createHorizontalStrut(4));
118     vbox5.add(startDateCombo = new JComboBox(selections2));
119     startDateCombo.setSelectedItem("weeks");
120     startDateCombo.setMaximumSize(
121         new Dimension(
122             "specific date".length() * startDateCombo.getFont().getSize(),
123             (int)(maxTextFieldHeight *
124                 startDateCombo.getFont().getSize()));
125     startDateCombo.setAlignmentX(Component.LEFT_ALIGNMENT);
126     String[] selections3 = {"ago", "from now"};
127     vbox5.add(startDateDirectionCombo = new JComboBox(selections3));
128     startDateDirectionCombo.setMaximumSize(
129         new Dimension(
130             "from now".length() *
131                 startDateDirectionCombo.getFont().getSize(),
132             (int)((maxTextFieldHeight + 0.5) * // Unknown why 0.5 fudge
133                 // factor is needed
134                 startDateDirectionCombo.getFont().getSize()));
135     startDateDirectionCombo.setAlignmentX(Component.LEFT_ALIGNMENT);
136     vbox5.setAlignmentY(Component.TOP_ALIGNMENT);
137     hbox5.add(vbox5);
138     hbox5.setAlignmentX(Component.LEFT_ALIGNMENT);
139
140     /*
141     * The following doesn't work right yet.
142     */
143     startDateTextField.addActionListener(
144         new ActionListener() {
145             public void actionPerformed(ActionEvent e) {
146                 if (Integer.parseInt(startDateTextField.getText()) == 1) {
147                     String item = (String)startDateCombo.getSelectedItem();
148                     if (item.equals("days")) {
149                         startDateCombo.setEditable(true);
150                         startDateCombo.setSelectedItem("day");
151                         startDateCombo.setEditable(false);
152                     }
153                     if (item.equals("weeks")) {
154                         startDateCombo.setEditable(true);
155                         startDateCombo.setSelectedItem("week");
156                         startDateCombo.setEditable(false);
157                     }
158                     if (item.equals("months")) {
159                         startDateCombo.setEditable(true);
160                         startDateCombo.setSelectedItem("month");
161                         startDateCombo.setEditable(false);
162                     }
163                     if (item.equals("years")) {
164                         startDateCombo.setEditable(true);
165                         startDateCombo.setSelectedItem("year");
166                         startDateCombo.setEditable(false);
167                     }
168                 }
169             }
170         });
171     /*
172     */
173     startDateCombo.addItemListener(
174         new ItemListener() {
175             public void itemStateChanged(ItemEvent e) {
176                 String item = (String) e.getItem();
177                 System.out.println(item);
178                 /*
179                 * The logic here may not be optimal, but other versions
180                 * have been tried without success. The trick is to avoid
181                 * having to check for equality with specific date strings.
182                 */
183                 if (item.equals("days") || item.equals("months") ||
184                     item.equals("weeks") || item.equals("years")) {
185                     startDateTextField.setEnabled(true);
186                     startDateTextField.setText("");
187                     startDateDirectionCombo.setEnabled(true);
188                     startDateCombo.setEditable(false);
189                 }
190                 else {
191                     startDateTextField.setEnabled(false);
192                     startDateDirectionCombo.setEnabled(false);
193                     startDateTextField.setText("");
194                 }
195                 if (item.equals("any") || item.equals("today")) {
196                     startDateCombo.setEditable(false);
197                 }
198                 if (item.equals("specific date") && !specificSelected) {
199                     startDateCombo.setEditable(true);
200                     startDateCombo.getEditor().selectAll();
201                     specificSelected = true;
202                     /*
203                     * BTW, the following code leads to infinite recursion.
204                     */
205                     startDateCombo.setSelectedItem("---enter date---");
206                 }
207                 else {
208                     specificSelected = false;
209                 }
210             }
211         });
212     protected boolean specificSelected = false;
213
214     };
215
216     hbox6.add(label = new JLabel("End Date: "));
217     label.setAlignmentY(Component.TOP_ALIGNMENT);
218     hbox6.add(endDateTextField = new JTextField(" 1"));
219     endDateTextField.setMaximumSize(
220         new Dimension(
221             (int)(3 * endDateTextField.getFont().getSize()),
222             (int)(maxTextFieldHeight *
223                 endDateTextField.getFont().getSize()));

```

```

224     endDateTextField.setAlignmentY(Component.TOP_ALIGNMENT);
225     hbox6.add(BoxPanel.createHorizontalStrut(4));
226     vbox6.add(endDateCombo = new JComboBox(selections2));
227     endDateCombo.setSelectedItem("weeks");
228     endDateCombo.setMaximumSize(
229         new Dimension(
230             "specific date".length() * endDateCombo.getFont().getSize(),
231             (int)(maxTextFieldHeight *
232                 endDateCombo.getFont().getSize()));
233     endDateCombo.setAlignmentX(Component.LEFT_ALIGNMENT);
234     vbox6.add(endDateDirectionCombo = new JComboBox(selections3));
235
236     endDateDirectionCombo.setMaximumSize(
237         new Dimension(
238             "from now".length() *
239                 endDateDirectionCombo.getFont().getSize(),
240             (int)((maxTextFieldHeight + 0.5) * // Unknown why 0.75 fudge
241                 // factor is needed
242                 endDateDirectionCombo.getFont().getSize()));
243     endDateDirectionCombo.setAlignmentX(Component.LEFT_ALIGNMENT);
244     vbox6.add(BoxPanel.createVerticalStrut(10));
245     vbox6.setAlignmentY(Component.TOP_ALIGNMENT);
246     hbox6.add(vbox6);
247     hbox6.setAlignmentX(Component.LEFT_ALIGNMENT);
248
249     vbox4.add(hbox2);
250     vbox4.add(BoxPanel.createVerticalStrut(10));
251     vbox4.add(hbox3);
252     vbox4.add(BoxPanel.createVerticalStrut(10));
253     vbox4.add(hbox4);
254     vbox4.add(BoxPanel.createVerticalStrut(10));
255     vbox4.add(hbox5);
256     vbox4.add(BoxPanel.createVerticalStrut(10));
257     vbox4.add(hbox6);
258     vbox4.add(BoxPanel.createVerticalStrut(10));
259     vbox4.setAlignmentY(Component.TOP_ALIGNMENT);
260
261     outerHbox.add(BoxPanel.createHorizontalStrut(30));
262     outerHbox.add(BoxPanel.createHorizontalGlue());
263     outerHbox.add(vbox1);
264     outerHbox.add(BoxPanel.createHorizontalStrut(20));
265     outerHbox.add(BoxPanel.createHorizontalGlue());
266     outerHbox.add(vbox4);
267     outerHbox.add(BoxPanel.createHorizontalStrut(30));
268     outerHbox.add(BoxPanel.createHorizontalGlue());
269
270     outerVbox.add(BoxPanel.createVerticalStrut(30));
271     outerHbox.add(BoxPanel.createVerticalGlue());
272     outerVbox.add(outerHbox);
273     outerVbox.add(BoxPanel.createVerticalStrut(30));
274     outerHbox.add(BoxPanel.createVerticalGlue());
275
276     return (widget = outerVbox);
277 }
278
279

```

```

280     JCheckBox titleCheckBox;
281     JCheckBox dateCheckBox;
282     JCheckBox timeCheckBox;
283     JCheckBox durationCheckBox;
284     JCheckBox recursCheckBox;
285     JCheckBox categoryCheckBox;
286     JCheckBox locationCheckBox;
287     JCheckBox securityCheckBox;
288     JCheckBox priorityCheckBox;
289     JComboBox sortKeyCombo;
290     JTextField maxItemsTextField;
291     JTextField displayHeightTextField;
292     JTextField startDateTextField;
293     JComboBox startDateCombo;
294     JComboBox startDateDirectionCombo;
295     JTextField endDateTextField;
296     JComboBox endDateCombo;
297     JComboBox endDateDirectionCombo;
298
299     /** The max height of a text field */
300     protected final double maxTextFieldHeight;
301
302
303 }

```