```
Loading vc-cvs...
```

```
package caltool.options_ui;

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

public class ViewingListsAppointmentsOptionsPanel extends View {

    public ViewingListsAppointmentsOptionsPanel(
            Screen screen, Options options) {
        super(screen, options);
        maxTextFieldHeight = 1.9;
    }

    public Component compose() {

        BoxPanel outerVbox = BoxPanel.createVerticalBox();
        BoxPanel outerHbox = BoxPanel.createHorizontalBox();
        BoxPanel hbox1 = BoxPanel.createHorizontalBox();
        BoxPanel hbox2 = BoxPanel.createHorizontalBox();
        BoxPanel hbox3 = BoxPanel.createHorizontalBox();
        BoxPanel hbox4 = BoxPanel.createHorizontalBox();
        BoxPanel hbox5 = BoxPanel.createHorizontalBox();
        BoxPanel hbox6 = BoxPanel.createHorizontalBox();
        BoxPanel vbox1 = BoxPanel.createVerticalBox();
        BoxPanel vbox2 = BoxPanel.createVerticalBox();
        BoxPanel vbox3 = BoxPanel.createVerticalBox();
        BoxPanel vbox4 = BoxPanel.createVerticalBox();
        BoxPanel vbox5 = BoxPanel.createVerticalBox();
        BoxPanel vbox6 = BoxPanel.createVerticalBox();
        JLabel label;

        vbox1.add(label = new JLabel("Show these fields:"));
        label.setAlignmentX(Component.LEFT_ALIGNMENT);
        vbox2.add(titleCheckBox = new JCheckBox("Title"));
        titleCheckBox.setSelected(true);
        vbox2.add(dateCheckBox = new JCheckBox("Date"));
        dateCheckBox.setSelected(true);
        vbox2.add(timeCheckBox = new JCheckBox("Time"));
        timeCheckBox.setSelected(true);
        vbox2.add(durationCheckBox = new JCheckBox("Duration"));
        durationCheckBox.setSelected(true);
        vbox2.add(recursCheckBox = new JCheckBox("Recurs"));
        recursCheckBox.setSelected(true);
        vbox2.setAlignmentY(Component.TOP_ALIGNMENT);

        vbox3.add(categoryCheckBox = new JCheckBox("Category"));
        categoryCheckBox.setSelected(true);
        vbox3.add(locationCheckBox = new JCheckBox("Location"));
        locationCheckBox.setSelected(true);
        vbox3.add(securityCheckBox = new JCheckBox("Security"));
        securityCheckBox.setSelected(true);
        vbox3.add(priorityCheckBox = new JCheckBox("Prioity"));
        priorityCheckBox.setSelected(true);
        vbox3.setAlignmentY(Component.TOP_ALIGNMENT);

        hbox1.add(vbox2);
        hbox1.add(BoxPanel.createHorizontalStrut(10));
        hbox1.add(vbox3);
        hbox1.setAlignmentX(Component.LEFT_ALIGNMENT);
        vbox1.add(hbox1);
        vbox1.setAlignmentY(Component.TOP_ALIGNMENT);

        hbox2.add(new JLabel("Primary sort key: "));
        String[] selections = {"Title", "Date", "Time", "Duration", "Recurs",
            "Category", "Location", "Security", "Priority"};
        hbox2.add(sortKeyCombo = new JComboBox(selections));
        sortKeyCombo.setSelectedItem("Date");
        sortKeyCombo.setMaximumSize(
            new Dimension(
                (int)("Category".length() * sortKeyCombo.getFont().getSize()),
                (int)(maxTextFieldHeight * sortKeyCombo.getFont().getSize())));
        hbox2.setAlignmentX(Component.LEFT_ALIGNMENT);

        hbox3.add(new JLabel("Maximum number of items: "));
        hbox3.add(maxItemsTextField = new JTextField());
        maxItemsTextField.setMaximumSize(
            new Dimension(
                (int)(4 * maxItemsTextField.getFont().getSize()),
                (int)(maxTextFieldHeight *
                    maxItemsTextField.getFont().getSize())));
        hbox3.setAlignmentX(Component.LEFT_ALIGNMENT);

        hbox4.add(new JLabel("Initial display height: "));
        hbox4.add(displayHeightTextField = new JTextField());
        displayHeightTextField.setMaximumSize(
            new Dimension(
                (int)(4 * maxItemsTextField.getFont().getSize()),
                (int)(maxTextFieldHeight *
                    maxItemsTextField.getFont().getSize())));
        hbox4.add(new JLabel(" items"));
        hbox4.setAlignmentX(Component.LEFT_ALIGNMENT);

        hbox5.add(label = new JLabel("Start Date: "));
        label.setAlignmentY(Component.TOP_ALIGNMENT);
        hbox5.add(startDateTextField = new JTextField(" 1"));
        startDateTextField.setMaximumSize(
            new Dimension(
                (int)(3 * startDateTextField.getFont().getSize()),
                (int)(maxTextFieldHeight *
                    startDateTextField.getFont().getSize())));
        /*
         * It seems that Java ignores the following miniumum setting, but we do
         * it in case things ever get better.
         */
        startDateTextField.setMinimumSize(
            new Dimension(
                (int)(3 * startDateTextField.getFont().getSize()),
```

```
                (int)(maxTextFieldHeight *
                    startDateTextField.getFont().getSize())));
        String[] selections2 = {"any", "today", "days", "weeks", "months",
            "years", "specific date"};
        startDateTextField.setAlignmentY(Component.TOP_ALIGNMENT);
        hbox5.add(BoxPanel.createHorizontalStrut(4));
        vbox5.add(startDateCombo = new JComboBox(selections2));
        startDateCombo.setSelectedItem("weeks");
        startDateCombo.setMaximumSize(
            new Dimension(
                "specific date".length() * startDateCombo.getFont().getSize(),
                (int)(maxTextFieldHeight *
                    startDateCombo.getFont().getSize())));
        startDateCombo.setAlignmentX(Component.LEFT_ALIGNMENT);
        String[] selections3 = {"ago", "from now"};
        vbox5.add(startDateDirectionCombo = new JComboBox(selections3));
        startDateDirectionCombo.setMaximumSize(
            new Dimension(
                "from now".length() *
                    startDateDirectionCombo.getFont().getSize(),
                (int)((maxTextFieldHeight + 0.5) *   // Unknown why 0.5 fudge
                                                     // factor is needed
                    startDateDirectionCombo.getFont().getSize())));
        startDateDirectionCombo.setAlignmentX(Component.LEFT_ALIGNMENT);
        vbox5.setAlignmentY(Component.TOP_ALIGNMENT);
        hbox5.add(vbox5);
        hbox5.setAlignmentX(Component.LEFT_ALIGNMENT);

        /*
         * The following doesn't work right yet.
         *
        startDateTextField.addActionListener(
            new ActionListener() {
                public void actionPerformed(ActionEvent e) {
                    if (Integer.parseInt(startDateTextField.getText()) == 1) {
                        String item = (String)startDateCombo.getSelectedItem();
                        if (item.equals("days")) {
                            startDateCombo.setEditable(true);
                            startDateCombo.setSelectedItem("day");
                            startDateCombo.setEditable(false);
                        }
                        if (item.equals("weeks")) {
                            startDateCombo.setEditable(true);
                            startDateCombo.setSelectedItem("week");
                            startDateCombo.setEditable(false);
                        }
                        if (item.equals("months")) {
                            startDateCombo.setEditable(true);
                            startDateCombo.setSelectedItem("month");
                            startDateCombo.setEditable(false);
                        }
                        if (item.equals("years")) {
                            startDateCombo.setEditable(true);
                            startDateCombo.setSelectedItem("year");
                            startDateCombo.setEditable(false);
                        }
                    }
                }
            }
        );
         *
         */

        startDateCombo.addItemListener(
            new ItemListener() {
                public void itemStateChanged(ItemEvent e) {
                    String item = (String) e.getItem();
                    System.out.println(item);
                    /*
                     * The logic here may not be optimal, but other versions
                     * have been tried without success.  The trick is to avoid
                     * having to check for equality with specific date strings.
                     */
                    if (item.equals("days") || item.equals("months") ||
                            item.equals("weeks") || item.equals("years")) {
                        startDateTextField.setEnabled(true);
                        startDateTextField.setText("");
                        startDateDirectionCombo.setEnabled(true);
                        startDateCombo.setEditable(false);
                    }
                    else {
                        startDateTextField.setEnabled(false);
                        startDateDirectionCombo.setEnabled(false);
                        startDateTextField.setText("");
                    }
                    if (item.equals("any") || item.equals("today")) {
                        startDateCombo.setEditable(false);
                    }
                    if (item.equals("specific date") && !specificSelected) {
                        startDateCombo.setEditable(true);
                        startDateCombo.getEditor().selectAll();
                        specificSelected = true;
                        /*
                         * BTW, the following code leads to infinite recursion.
                        startDateCombo.setSelectedItem("--enter date--");
                         */
                    } else {
                        specificSelected = false;
                    }
                }
                protected boolean specificSelected = false;
            }
        );

        hbox6.add(label = new JLabel("End Date: "));
        label.setAlignmentY(Component.TOP_ALIGNMENT);
        hbox6.add(endDateTextField = new JTextField(" 1"));
        endDateTextField.setMaximumSize(
            new Dimension(
                (int)(3 * endDateTextField.getFont().getSize()),
                (int)(maxTextFieldHeight *
                    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 }
```