Recitation No. 11: SWT GUI Package

Model-View Separation

- Separate between the application logic (model) part and the GUI (view) part.
- Ensures that view changes have no effect on the basic model.
- Enables us to maintain one model for several different views.

Example: Address Book

The Model

```java
ADDRESSBOOK CLASS
- add(Contact c)
- get(String name)
- delete(String name)
- modify(Contact c)
- search(String prefix)
- getContacts()
- getCount()
- save(String filename)
- load(String filename)

CONTACT CLASS
- has
- name
- email
- telephone
- address
```

The View

The class diagram:

```
ADDRESSBOOK VIEWER CLASS
- add, delete, modify, search
```

The implementation:

- based on the SWT GUI library
**SWT**

- Online Documentation:
  - JavaDoc
  - Snippets

**Widgets**

- `<abstract class>` Widget
- `<class>` Control
  - `<abstract class>` ScrollBar
  - `<class>` Menu
  - `<class>` Table
  - `<class>` Sash
  - `<class>` Composite
  - `<class>` Label
  - `<class>` Button
  - `<class>` Composite
  - `<class>` Table

**Layouts**

- A Layout controls the position and size of Control widgets in a Composite.

**GridLayout**

- Lays out the Control widgets in a grid.

**GridLayout Configuration fields:**

<table>
<thead>
<tr>
<th>Field</th>
<th>Default</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>horizontalSpacing</td>
<td>5</td>
<td>Horizontal/vertical space between the grid cells</td>
</tr>
<tr>
<td>verticalSpacing</td>
<td>5</td>
<td>The size of the horizontal/vertical margins of the layout</td>
</tr>
<tr>
<td>marginHeight</td>
<td>5</td>
<td>Number of columns</td>
</tr>
<tr>
<td>marginWidth</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>numColumns</td>
<td>false</td>
<td>If true, all columns will have the same size</td>
</tr>
</tbody>
</table>

Each column is as wide as Wide Button 2.
GridLayout (cont.)

GridData:
- Use GridData objects to configure the Control widgets in a GridLayout.
- Use the setLayoutData() to set a GridData object into a Control, e.g. label.setLayoutData(new GridData(...));
- Do not reuse GridData objects

GridLayout (cont.)

GridData Configuration Fields:

<table>
<thead>
<tr>
<th>Field</th>
<th>Default</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>gridWidth</td>
<td>false</td>
<td>If true, the width/length of the widget will be as large as possible to fill the remaining space.</td>
</tr>
<tr>
<td>gridHeight</td>
<td>SWT.DEFAULT</td>
<td>A minimum width/height for the widget.</td>
</tr>
<tr>
<td>horizontalSpace</td>
<td>1</td>
<td>the number of column/row cells that the widget will take up.</td>
</tr>
<tr>
<td>verticalSpace</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>horizontalIndent</td>
<td>0</td>
<td>the number of indentation pixels along the left side of the cell.</td>
</tr>
<tr>
<td>verticalIndent</td>
<td>GridData.BEGINNING</td>
<td>how controls will be positioned horizontally/vertically within a cell.</td>
</tr>
<tr>
<td>horizontalAlignment</td>
<td>GridData.CENTER</td>
<td></td>
</tr>
<tr>
<td>verticalAlignment</td>
<td>GridData.CENTER</td>
<td></td>
</tr>
</tbody>
</table>

FormLayout

A very flexible layout

FormLayouts

A very flexible layout

FormLayout Configuration Properties:

<table>
<thead>
<tr>
<th>Field</th>
<th>Default</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>marginHeight</td>
<td>0</td>
<td>the margin width/height</td>
</tr>
<tr>
<td>marginWidth</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>spacing</td>
<td>0</td>
<td>the number of pixels between the edge of one control and the edge of its neighbouring control.</td>
</tr>
</tbody>
</table>

FormLayouts (cont.)

Use FormData objects to configure the Control widgets in a FormLayout.

Use the setLayoutData() to set a FormData object into a Control Widget.

A FormData object has a FormAttachment object for each edge of the Control.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>width/height</td>
<td>the desired width/height in pixels.</td>
</tr>
<tr>
<td>top/bottom/left/right</td>
<td>Specifies the position of the control attachment.</td>
</tr>
</tbody>
</table>
FormLayouts (cont.)

- **Main FormAttachment Constructors:**
  - `public FormAttachment(Control control)`
  - `public FormAttachment(Control control, int offset)`
  - `public FormAttachment(int numerator)`
  - `public FormAttachment(int numerator, int offset)`

<table>
<thead>
<tr>
<th>Field</th>
<th>Default</th>
</tr>
</thead>
<tbody>
<tr>
<td>control</td>
<td>Parent Composite</td>
</tr>
<tr>
<td>numerator</td>
<td>100</td>
</tr>
<tr>
<td>denominator</td>
<td>100</td>
</tr>
<tr>
<td>offset</td>
<td>0</td>
</tr>
</tbody>
</table>

$$y = \frac{numerator}{\text{denominator}} \times x + \text{offset}$$

$$x = \text{control's width/height}$$