GUI I

IS 313

2.4.03
Outline

- Homework #1 solution
- Homework #2 continued
- GUI components
- Layout
Homework #1
Homework #2
Database Schema
Command Hierarchy
GUI
Standard UI program

- **Main program**
  - Creates a window
  - Adds components
  - Displays

- **Component maker**
  - Creates, customizes components
  - Sets event handlers
Application Architecture

User

UI

Application logic
Inside the UI

- UI Components
- Containment hierarchy
- Layout management
- Event handling
Top-Level Components

- Applet
- Dialog
- Frame
Intermediate Containers

- Panel
- ScrollPane
- SplitPane
- TabbedPane
- Toolbar
Basic Controls

- **Buttons**
- **ComboBox**
- **List**

- **Menu**
- **Slider**
- **TextField**
Non-editable Display

Label  ProgressBar  ToolTip

Label 1  

COW.
Editable Display

Table

<table>
<thead>
<tr>
<th>First Name</th>
<th>Last Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mark</td>
<td>Andrews</td>
</tr>
<tr>
<td>Tom</td>
<td>Ball</td>
</tr>
<tr>
<td>Alan</td>
<td>Chung</td>
</tr>
<tr>
<td>Jeff</td>
<td>Dinkins</td>
</tr>
</tbody>
</table>

Text

Verify that the RJ45 cable is connected to the WAN plug on the back of the Pipeline unit.

Tree

- tab3.gif
  - Tree View
    - drawing
    - treeview
Text Components

This is an editable JTextArea that has been initialized with the setText method. A text area is a "plain" text component, which means that although it can display text in any font, all of the text is in the same font.

This is an uneditable JEditorPane, which was initialized with HTML text from a URL.

This is an editable JTextPane, another styled text component, which supports embedded components... ...and embedded icons...
Examples

- Example 1
  - simple swing app
- Example 2
  - two-tiered architecture
- Example 3
  - specialized component design
Layout Manager

- Maintains topological relationship between components
- Follows a set of layout rules
- Uses constraints specified by the programmer
Border Layout
Border Layout

- **Topology**
  - Two-dimensional distribution

- **Rules**
  - Corner and edge components = preferred size
  - Center component = all extra

- **Constraints**
  - Component sector
Flow Layout
Flow Layout

- Topology
  - One-dimensional adjacency

- Rules
  - All components = preferred size

- Constraints
  - Order of insertion
Grid Layout

![Grid Layout Example](image1.png)

![Grid Layout Example](image2.png)
Grid Layout

- Topology
  - Two-dimensional distribution
- Rules
  - All components = same size
- Constraints
  - Order of insertion
  - Left-to-right
  - Top-to-bottom
GridBag Layout
GridBag Layout

- Topology
  - Two-dimensional distribution

- Rules
  - Components = preferred size
  - Extra space allocated using constraints

- Constraints
  - Grid location, cell span, padding
  - Fill dimension
  - Weight
Box Layout
Box Layout

- Topology
  - One-dimensional stacking

- Rules
  - Components = preferred size
  - Extra space allocated using special purpose “filler components”

- Constraints
  - Alignment
  - Order of insertion
Example Layout I

Would you like green eggs and ham?

Yes, please  No way!
Example Layout II

![Image of a software interface with file entries and revision information]

- CVS Entry: `todo.txt` 1.5 Mon May 20 11:11:16 2002
- CVS Entry: `conf`
- CVS Entry: `content`
- CVS Entry: `data`
- CVS Entry: `log`
- CVS Entry: `results`
- CVS Entry: `src`
Example Layouts III
Example Layout IV