Overview

- Java GUI Classes
- Creating a GUI
- Elements of a GUI
  - Component ⇒ items displayed (widgets)
  - Container ⇒ region containing components
  - Layout ⇒ arrangement of components
  - Event ⇒ interactions for GUI
Java GUI Classes

- AWT (Abstract Window Toolkit) (java.awt.*)
  - Old GUI framework for Java (Java 1.1)
  - Some reliance on native code counterparts
  - Platform independence problems

- Swing (javax.swing.*)
  - New GUI framework first introduced in Java 1.2
  - Includes AWT features plus many enhancements
  - Pure Java components (no reliance on native code)
  - Pluggable look and feel architecture

- SWT (Standard Widget Toolkit; from Eclipse)

Java GUI Classes

- GUI classes can be organized in three groups
  - Component classes
    - Items (widgets) displayed in GUI
      - JButton, JTextField, JDialog, JScrollPane...
  - Container classes
    - Hold GUI Components
      - JFrame, JPanel, JApplet...
  - Helper classes
    - Describe properties of other GUI components
      - Color, Graphics, Dimension...
How to Create a GUI

1. Define Frame or Applet to hold components
   - We'll just use frames for now

2. Add GUI components to Frame
   - Use layout manager to determine position

3. Add actions to GUI
   - By adding event listeners to GUI components

GUI Elements – Container

- Definition
  - Abstractions occupying space in GUI

- Properties
  - Usually contain one or more widgets
  - Can be nested in other containers

- Example
  - JFrame window containing
    - 1 menu (component)
    - 3 buttons (component)
    - 2 JInternalFrame windows (container)
Java Components

- JFrame
  - With three JButtons

![Image of JFrame with three buttons]

 JFrame Hierarchy

- Several super classes
  - As well as implemented interfaces

- Many, many member methods
  - Including inherited methods that allow
    - Resizing
    - Setting properties
    - Adding components,
    - Etc…

- Other top level containers
  - JDialog (dialog boxes)
  - JApplet (web applets)
  - JWindow (stripped down JFrame, no title bar or window buttons)
JFrame Structure

- Most things go into content pane
  - getContentPane()
- Use glassPane for pop up menus, some animations
- Methods
  - getRootPane()
  - getLayeredPane()
  - getContentPane()
  - getGlassPane()
- Can set…Pane explicitly

LayeredPane manages (optional) JMenuBar
LayeredPane contains contentPane

GUI Elements – Layout

- Definition
  - Arrangement of GUI components in container

- Layout specification
  - Logical terms (2nd row, 1st column, left)
    - Preferred approach
  - Actual coordinates (100 pixels, 5 inches)
    - Can be too rigid, limited to certain window sizes

- Layout manager
  - Entity translating layout specifications into actual coordinates at runtime, depending on conditions
Java Layout Managers

- FlowLayout
  - Lays out components from left to right

- GridLayout
  - Lays out components in a grid of user specified size

- BorderLayout
  - Designates portions of the container as North, South, East, West, and Center

- CardLayout
  - Adds components one on top of another

- GridBagLayout
  - Customizable manager that can use rows and columns of varying lengths

GUI Elements – Component

- Definition
  - Actual items (widgets) user sees in GUI

- Examples
  - Labels (fixed text)
  - Text areas (for entering text)
  - Buttons
  - Checkboxes
  - Tables
  - Menus
  - Toolbars
  - Etc…
Java Components

- JPanel

![Using JPanel with Borders](image)

Java Components

- JTree

![Creating a Simple JTree](image)
Java Components

■ JTable

Each JTable object
■ Gets its data from an object implementing
TableModel interface
■ Displays contents of TableModel object
■ DefaultTableModel class implements TableModel
■ Many different ways to use JTable to display data
GUI Elements – Events

Definition
- Action or condition occurring outside normal flow of control of program

Examples
- Mouse clicks
- Keyboard input
- Menu selections
- Window actions

In Java
- GUI events handled in event dispatching thread

Event Dispatching Thread

Background thread to process events
- From AWT graphical interface event queue

These events are mainly updates that
- Cause components to redraw themselves
- Represent input events

Swing uses a single-threaded painting model
- Event Dispatching thread is the only valid thread for updating GUI components
- Avoid updating GUI components from other threads
  - A source of common bugs
Event Dispatching Thread

Example code
- Allows current thread to execute GUI code in dispatching thread
- `createAndDisplayGUI`
  - Method that actually defines the GUI

```java
javax.swing.SwingUtilities.invokeLater(new Runnable() {
    public void run() {
        createAndDisplayGUI();
    }
});
```

Java Support For GUIs

- Several GUI code examples

Additional Resources
- Appendix C of textbook
- Javadoc for the JDK
- Swing tutorial
- Course slides and code handouts
- Java Ranch