


Principles of Computer Science II

Prof. Nadeem Abdul Hamid
CSC 121A - Spring 2005
Lecture Slides 4 -
GUI Components



Graphical User Interface

- GUI - provides user-friendly mechanism for interactive with an application
- Typical window (Internet Explorer)
 - Title bar
 - Menu bar
 - Buttons
 - Combo box (web site address)
 - Scroll bars
- GUIs built from *components* (window gadgets/widgets)

CSC 121A - Berry College - Spring 2005 2

Simple I/O with JOptionPane

- Prepackaged dialog boxes for input and output
 - showMessageDialog
 - showInputDialog
 - User inputs a String
 - If "cancel" clicked, returns null
- e.g.


```
JOptionPane.showMessageDialog( null, "Message", "Title",
JOptionPane.PLAIN_MESSAGE );
```
- Several overloaded versions of these methods are available

CSC 121A - Berry College - Spring 2005 3

Swing Components

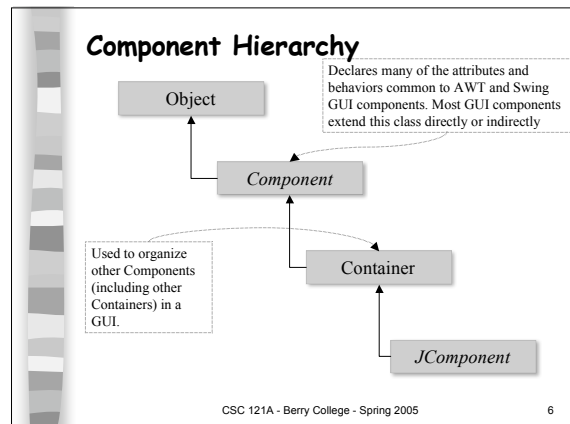
- Swing is a Java library of GUI components
- Provide a consistent "look and feel" across different computer platforms
- Some common components:
 - JLabel: displays uneditable text or icons
 - JTextField: enables user to enter data from keyboard
 - JButton: triggers event when clicked by mouse
 - JCheckBox: option can be selected or not
 - JComboBox: drop-down list of items from which user can choose, or enter a new item by typing
 - JList: list of items from which to select; may be able to select multiple elements
 - JPanel: an area in which components can be placed and organized; can also be used as a drawing area for graphics

CSC 121A - Berry College - Spring 2005 4

Swing vs. AWT

- Swing
 - Pure Java components - programmed, manipulated, displayed completely in Java
 - Part of the Java Foundation Classes (JFC)
 - <http://java.sun.com/products/jfc>
 - Uniform look and feel
 - Lightweight components: not tied to actual GUI components of the underlying OS
- AWT (Abstract window toolkit)
 - Earlier library of components
 - Display according to platform
 - Heavyweight components: rely on local OS's windowing system to determine functionality, look and feel

CSC 121A - Berry College - Spring 2005 5



JComponent

- Superclass of all (lightweight) Swing components
 - (Note, it is a subclass of container)
- Features supported
 - Pluggable look-and-feel (customizable appearance of components)
 - Shortcut keys (mnemonics) to access GUI components
 - Common event-handling capabilities
 - Tool tips (popup help/descriptions of components)
 - Support for assistive technologies
 - User-interface localization support (different languages, display formats, etc.)

CSC 121A - Berry College - Spring 2005

7

Steps To Use Window Frame

- Import classes and declare variables
 - `java.awt.*` and `javax.swing.*`
 - `JFrame` and `Container` variables
- Instantiate frame object and specify properties
 - Instantiate new `JFrame` object
 - Get content pane of the frame (store in `Container` variable)
 - Specify action when window is closed
 - Specify size of `JFrame`
 - Specify layout manager for content pane
- Put display objects into the frame
 - Create display objects and use `add` method
- Make frame visible on the screen
 - Use `setVisible` method
- Example: [lec04/PrintName.java](#)

CSC 121A - Berry College - Spring 2005

8

Layout Managers

- Used to control the positioning of components as they are added to a container
- Most basic layout is `FlowLayout`
- For laying out components in rows and columns, can use `GridLayout`
 - As components are added, they are placed row by row from left to right, and top to bottom
- Example: [lec04/DateFormats.java](#)

CSC 121A - Berry College - Spring 2005

9

Displaying Text and Images in Labels

- Windows can also be created by subclassing `JFrame`
 - Declare a subclass of `JFrame` with GUI specific to the application
 - Declare an application class with `main` method that instantiates and displays the window
- Identify the purpose of GUI components using `JLabel`
- Example:
 - [lec04/LabelFrame.java](#)
 - [lec04/LabelTest.java](#)

CSC 121A - Berry College - Spring 2005

10

Text Fields and Event Handling

- GUIs are event-driven
- When user interacts with a GUI component, the interaction (or event-- e.g. button click, key press) drives the program to perform a task
- Event handler: Code that performs a task in response to an event
- Event handling: Overall process of responding to events
- Example: `JTextField`, `JPasswordField`
 - [lec04/TextFieldFrame.java](#)
 - [lec04/TextFieldTest.java](#)

CSC 121A - Berry College - Spring 2005

11