

Swing

CSSE 221

Fundamentals of Software Development
Honors

Rose-Hulman Institute of Technology

Announcements

- HW3 due now
- Main course page has link to resolving SVN conflicts
- Capsule teams for phases 2 and 3 posted on schedule
- Exam 1 is this Thursday, 7:00 – 9:00 PM, G315 & G317
- Questions on Arrays?

This week: Fifteen assignment

- Monday:
 - Fifteen specification
 - GUIs using Java's Swing library
 - Intro to UML as a *design* tool
 - Start Fifteen
- Tuesday:
 - EventListeners: responding to user input
 - Shape classes
- Thursday:
 - Anonymous listeners
 - Exam 1

The Java Swing Library

- Use for GUIs.
- JFrame, JTextBoxes, JButtons, JScrollPanes... what's available?
- What components will I need...
 - ...now, for Fifteen?
 - ...later, for a term project?
- Browse the [Visual Index to the Swing Components](http://download.oracle.com/javase/tutorial/ui/features/components.html) in Sun's [Java Tutorial](http://download.oracle.com/javase/tutorial/ui/features/components.html). Follow this link to do so.
<http://download.oracle.com/javase/tutorial/ui/features/components.html>

Classes

- **JFrame**: top level. Can't draw on.
- **JComponents**: can draw on
- **Subclasses of JComponent**
 - **JPanel**: Use for organization
 - Use layouts to format layout
 - **JLabel**: non-interactive. `getText()`, `setText()`
 - **JTextField**: for data entry
 - **JTextArea**: like multi-line `JTextField`
 - **JButton**: generates *actions*
 - **JToggleButton**: superclass of `JCheckBox`, `JRadioButton`

JFrame

- A simple window that contains the other GUI components
- Make sure to set visible
`setVisible(true);`
- Setting the size
`setSize(width, height);`
- Background Color
`getContentPane().setBackground(Color);`

JLabel

- Displays text to user
- User cannot edit text
- Construction:

```
JLabel message = new JLabel(String text);
```

- Methods:

```
String text = message.getText();
```

```
message.setText(String text);
```

JTextField

- Lets user enter or modify a single line of text
- Construction:
`JTextField text = new JTextField(int nCharsWide);`
- Getting the Text:
`String input = text.getText();`
- Setting text in a Text Field:
`text.setText(String output);`

JTextArea

- Lets user enter or modify multiple lines of text.
- Constructors:
 - `text = new JTextArea(int rows, int columns);`
 - `text = new JTextArea(String output);`
 - `text = new JTextArea(String output, int rows, int columns);`
- Useful methods:
 - `String input = text.getText();`
 - `text.setText(String output);`
 - `text.setEditable(Boolean editable);`
 - `boolean editable = text.isEditable();`

JButton

- Used to let user execute action
- Construction:
`JButton button = new JButton(String text);`
- Method:
`button.setBackground(Color color);`

JPanel

- Simple container that groups objects
- Meant to subdivide
- Creation Syntax:
`JPanel panelName = new JPanel();`
- Useful attributes and methods
 - `.setLayout(new Layout());`
 - `.add(component);`
 - `.setBackground(Color color);`

Layout Managers

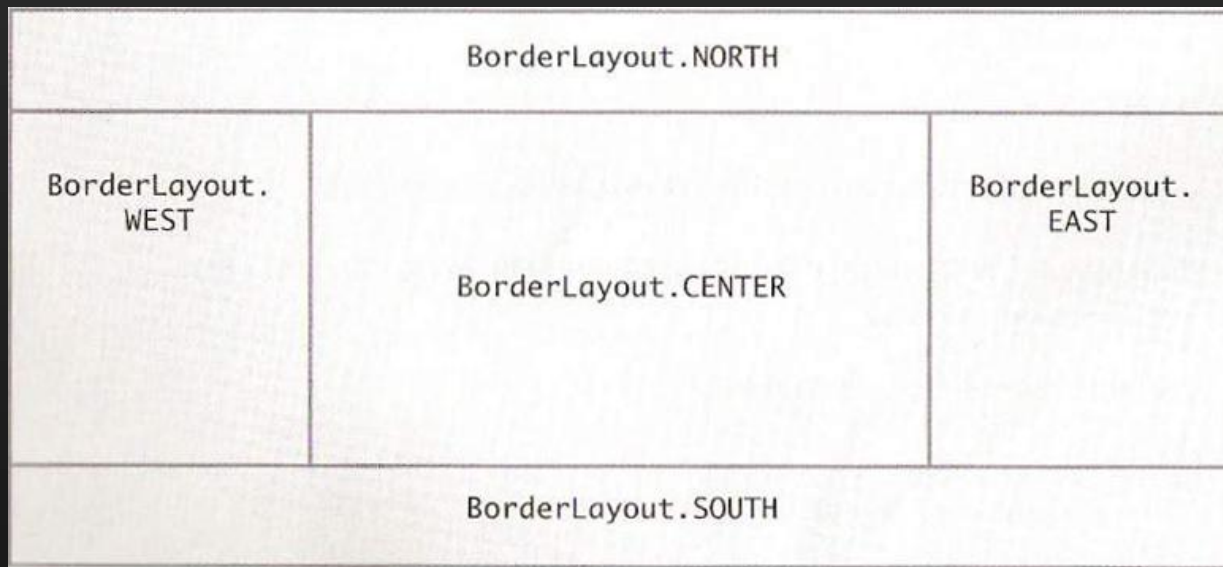
- Frames are organized by layout managers.
- Organizes the locations of the components within the frame
- Not only frames use layout managers, but Panels are also organized by layout managers.
- Allows for panels within panels.

Layout: Flow Layout

- Places the components from the left side to the right side.
- Wraps around the right side.
 - `.setLayout(new FlowLayout());`
- No special add syntax.

Layout: Border Layout

- North, South, East, West, Center
- `setLayout(new BorderLayout());`
- `.add(component, BorderLayout.NORTH);`



Layout: Grid Layout

- Similar to flow in the left to right manner, but the grid that you define limits the size of any one component
- You can leave out a definite definition of how many rows or columns are in the grid.
- `setLayout(new GridLayout(r,c));`
- Adds in components from left to right, top to bottom. Cannot specify which block.

PaintComponent

- Called automatically, no need to invoke it.
- Use `repaint()`; to force the `paintComponent()` method to execute.

```
public void paintComponent(Graphics g) {  
    super.paintComponent(g);  
    Graphics2D g2 = (Graphics2D)g;  
    this.setBackground(Color.RED);  
    g2.draw(new Rectangle2D.Double(10,40,20,20));  
}
```

From where are x and y
measured?

Alternative designs

- I want a component to appear. Question:
 - Should I paint it? Add a new component?



Press me!

- Answer: It depends...
 - On how much control you want over its appearance
 - On how you want it to respond to events
- It's usually a tradeoff.



Finish quiz

SwingDemo; can see detailed instructions online if you want

Demo together

Simplest Java Graphics Program

```
import javax.swing.JFrame;
/**
 * From Ch 2, Big Java.
 * @author Cay Horstmann
 */
public class EmptyFrameViewer {
    /**
     * Draws a frame.
     * @param args ignored
     */
    public static void main(String[] args) {
        JFrame frame = new JFrame();
        frame.setSize(300,400);
        frame.setTitle("An Empty Frame");
        frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        frame.setVisible(true);
    }
}
```

Creates a graphics
frame object

Configures it

Display the frame

Tells Java to exit
program when user
closes the frame

Other Shapes

- `new Ellipse2D.Double(double x, double y,
double w, double h)`
- `new Line2D.Double(double x1, double y1,
double x2, double y2)`
- `new Point2D.Double(double x, double y)`
- `new Line2D.Double(Point2D p1, Point2D p2)`
- `new Arc2D.Double(double x, double y,
double w, double h,
double start, double extent,
int type)`
- Try these!
 - Add an ellipse and both kinds of lines to MyComponent

Adding Text

- To add some text to a component:
 - `graphics2.drawString("some text", x, y);`
- You can change the font before drawing the text:
 - `Font f = new Font("Times New Roman",
Font.PLAIN, 72);`
`graphics2.setFont(f);`

Style. Other alternatives are:
Font.BOLD,
Font.ITALIC, and
Font.BOLD | Font.ITALIC

Font size in
points

Colors

- To change the Graphics2D object's "pen" color:
 - `Color c = ...; // see below`
`graphics2.setColor(c);`
- Lots of colors:
 - `new Color(red, green, blue)`, all from 0 to 255
 - `Color.RED`, `Color.WHITE`, etc. (see Javadocs)
 - `new Color(red, green, blue, alpha)`, all from 0 to 255. *alpha* is transparency
- To fill interior of shape:
 - `graphics2.fill(box);`

“Fifteen”

Arrays (especially
2D)

Creating GUIs
using Swing

Responding to
mouse clicks



Pair programming

- <http://www.researchgroup.org/pairlearning/educators.php#ppvideo>