

ITEC 120

Lecture 34
GUIs

Review

- Questions?
- Higher level design
- Homework

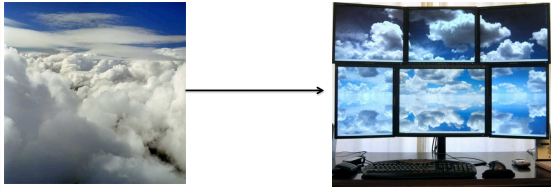
GUIs

Objectives

- Applying OO programming
- Learn basic GUI creation in Java
 - Conceptual model
 - How to make certain GUIs appear
 - No interactivity today

GUIs

Pixels



Real life

Virtual life

GUIs

How

Ends up with: 2,073,600 pixels

0,0
1920,1080

Lots of little dots make up the image

GUIs

Operating system

- Runs programs, allows access to hardware
- JFrame interfaces with the OS
- Allows you to ask for a window of a specific size, or one that will hold your app

Let me have a window of size X,Y please

GUIs

Example

```

import javax.swing.*;
import java.awt.event.*;
import java.awt.*;
JFrame frame = new JFrame("First GUI");
frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
frame.getContentPane().add(content);
frame.pack();
frame.setVisible(true);
    
```

Include the libraries

Window title

Shows it to the user

Tells the window What to display

Causes program to quit when you click the X

*Packing ensures it goes to the size you requested.

GUIs

Boxes

- Mental model

JPanel

JFrame

GUIs

Components

```

JPanel content = new JPanel();
JLabel aLabel = new JLabel();
aLabel.setText("JLabel");
content.add(aLabel);

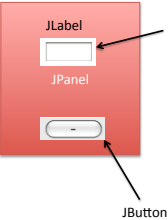
JButton minus = new JButton();
minus.setText("-");
content.add(minus);

JTextField jt = new JTextField(4);
content.add(jt);

```

You use `getText()` and `setText("Hi")` to access the contents of labels, buttons, and text fields.

Create GUI component
Add to content pane
Add one pane to the JFrame

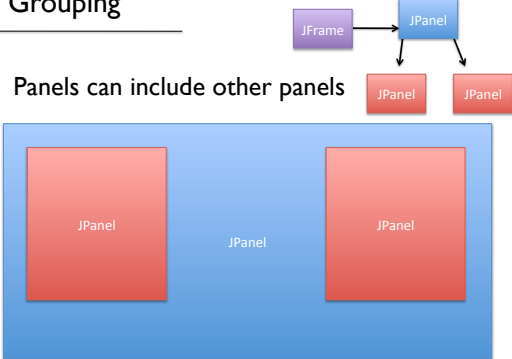


The diagram shows a red square representing a JPanel. Inside it, there is a white rectangular box labeled 'JLabel', a button labeled 'JButton' with a minus sign, and a white rectangular box with a border labeled 'JTextField'. Arrows point from the text labels to their respective components in the diagram.

GUIs

Grouping

• Panels can include other panels



The diagram shows a hierarchy of GUI components. At the top, a purple box labeled 'JFrame' has an arrow pointing to a blue box labeled 'JPanel'. This blue 'JPanel' box has two arrows pointing to two smaller red boxes, each labeled 'JPanel'. Below this, a larger blue box represents the 'JPanel' from the JFrame, containing three smaller red boxes, each labeled 'JPanel', arranged horizontally.

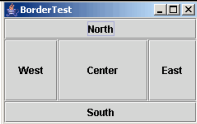
GUIs

Problem

- Arranging the components
 - What goes where
- No layout manager and set the X,Y position
 - Takes a while to get right
- LayoutManagers
 - Several different pre-built ways to order components

GUIs

Border Layout



The screenshot shows a window titled 'Border Test' with a standard Windows title bar. The window content is divided into five regions: 'North' at the top, 'West' on the left, 'Center' in the middle, 'East' on the right, and 'South' at the bottom.

- Specific layout
- Not all parts are required (scaling)

```

JPanel primary = new JPanel();
primary.setLayout(new BorderLayout());

JButton n = new JButton();
primary.add(n, BorderLayout.NORTH);

JButton c = new JButton();
primary.add(c, BorderLayout.CENTER);

frame.getContentPane().add(primary);

```

Tells the Panel to arrange its contents in a specific way

Tell the frame to display this panels content

GUIs

Grid Layout



- Rows and columns

```
JPanel primary = new JPanel();
primary.setLayout(new GridLayout(3,2));

JButton one = new JButton();
JButton two = new JButton();

primary.add(one);
primary.add(two);

frame.getContentPane().add(primary);
```

GUIs

Flow

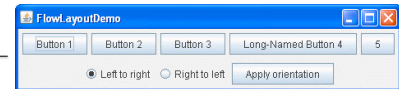
- Assures left to right or right to left

```
JPanel primary = new JPanel();
primary.setLayout(new FlowLayout());

JButton one = new JButton();
JButton two = new JButton();

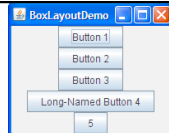
primary.add(one);
primary.add(two);

frame.getContentPane().add(primary);
```



GUIs

Box Layout



- Like the flow layout
- Allows for blank spaces in the GUI

```
JPanel primary = new JPanel();
primary.setLayout(new BoxLayout(primary,BoxLayout.PAGE_AXIS));

JButton one = new JButton();
JButton two = new JButton();

primary.add(one);
primary.add(two);

frame.getContentPane().add(primary);
```

GUIs

Documentation

- Reading it is good
- Using it for inspiration is better
 - Find out how to specify where extra space goes
 - Create spaces between components

<http://java.sun.com/docs/books/tutorial/uiswing/layout/box.html>

GUIs

Process

- Identify what the GUI does
- Draw boxes on paper
- Determine components for the page
- Figure out how many panels do you need
- Figure out the layout of the panels
- Code
 - Notice this is the last step!!!!

GUIs

Typical GUI

- Master panel
- Sub panels for each area of functionality
- Sub-sub panels if it is complicated
- Use layouts when possible

GUIs

Summary

- Pixels
- Frame
 - Panel (layout manager)
 - Components (Textfields / Buttons / Labels)

GUIs