CS 2334: Lab 9

Graphical User Interface

Eclipse Debugger

Demonstration…

Graphical User Interfaces

At the top level, our GUI should:

- Present information to the user
- Give the user a means for communicating with the program
- Respond appropriately to these user inputs

Graphical Components

- Components are pieces (widgets) found in a window
  - Determine how information is presented
  - In some cases, components accept user input
  - E.g., buttons, text, icons, menus
- Some components are containers:
  - Contain other components
  - Two special ones: JFrame (window) and JPanel (piece of a window)

Layout Managers

Determine how a set of components within a container are assigned to the screen

- Each container has its own Layout Manager
- Must attach this Layout Manager ***before*** components are added
- Useful managers:
  - GridLayout: regular grid of a specified size
  - BoarderLayout: can place components at the center, north, south, east and west
  - GridBagLayout: more flexible that GridLayout (see the book)

Events

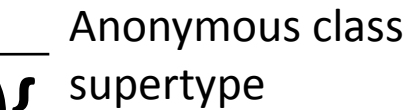
- Components generate an Event to indicate that something interesting has happened
  - In our case, events are raised by the Java graphics system
  - Events contain information about the type, source object and what the mouse buttons were doing at the time
- E.g.: A button when clicked by user will generate an ActionEvent

Listeners Wait for Events

- An event listener interface defines the methods that must be implemented by a specific type of listener
- Each event type will have at least one corresponding dispatch method in a listener interface.
- To listen for an event, a listener must implement the XXXListener interface and register itself with the component.
- When an event occurs the component will call the proper dispatch method.

Anonymous class

Anonymous classes helps to declare and instantiate a class at the same time. They are local (inner) classes except that they do not have a name.



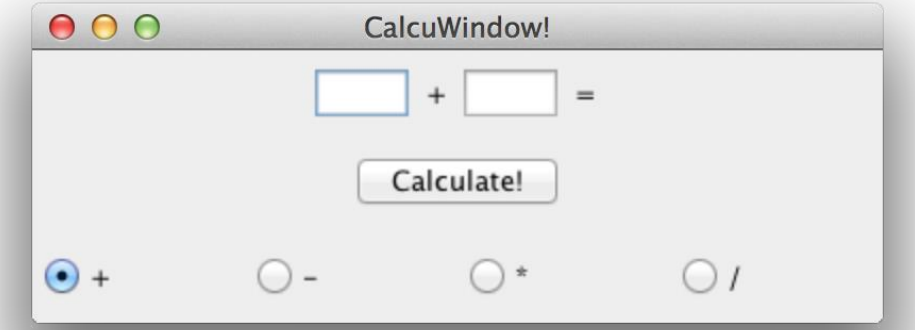

```
button.addActionListener(new ActionListener(){
        public void actionPerformed(ActionEvent e){
        opSign.setText("+");
        }
});
```

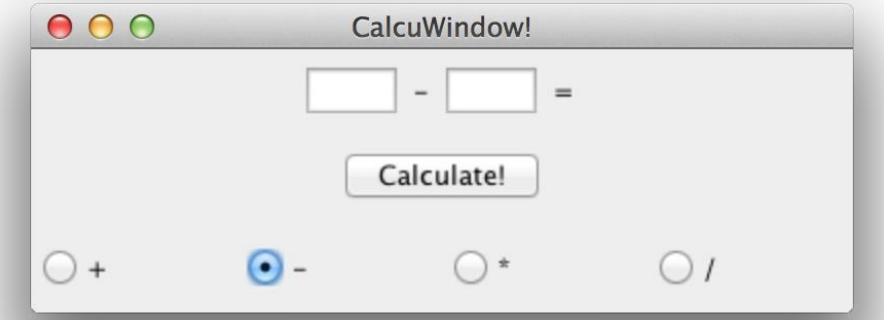
Graphical User Interfaces

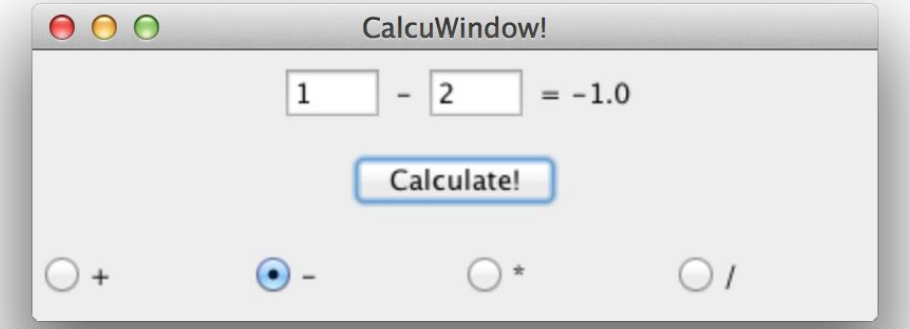
GUIs change the game: our code no longer executes in the order that we have written it

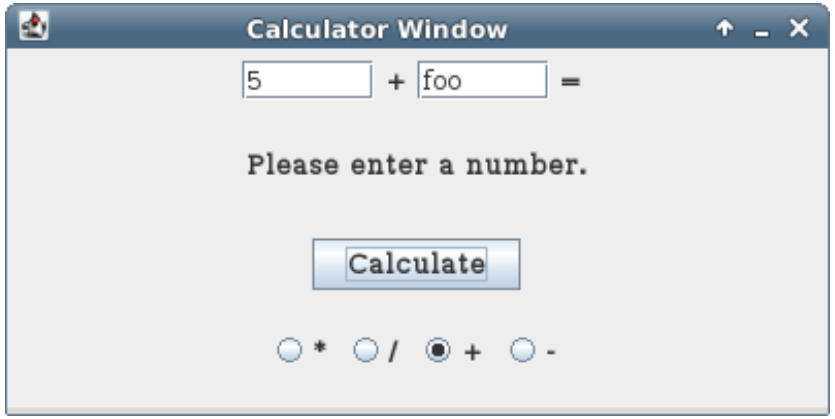
- Initialization phase is procedural: create all of the pieces of the GUI and initialize data structures
- After that: we move into an event-driven model of programming
  - Our code declares what should happen when an event happens
  - But: the Java graphics system determines when the events actually happen

Lab9: Simple Calculator







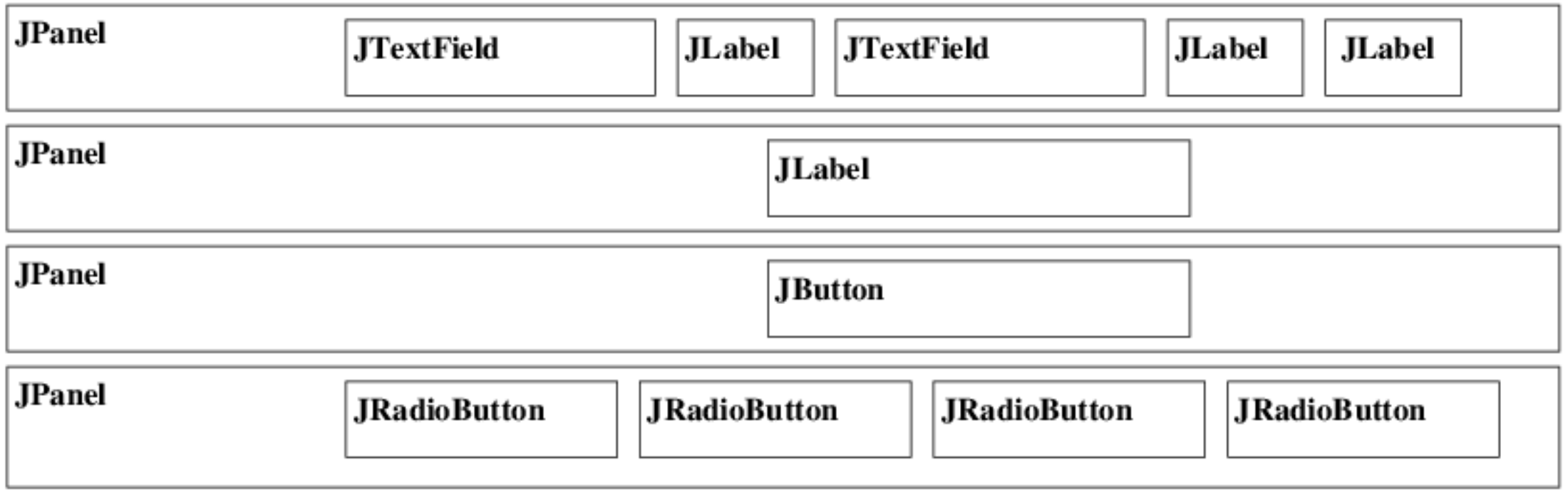


Demonstration…

Lab 9: Simple GUI-Based Calculator

- Labels: display text information
- Two text fields: input operands from user
- A set of radio buttons that allows the user to specify the operation to perform: +, -, *, /
- A button that triggers the calculation and display of the answer
  - Note: must be robust to non-numeric inputs

Calculator Component Structure



Your code must create all of these objects and attach them together

Submission

- Submit only one file: lab9.zip (casing matters)
- Due date: Sunday, October 23rd @11:59pm
- Submit to lab9 dropbox on D2L