

# Interactive GUIs

10-27-2010

# Opening Discussion

- Solutions to the IcP.
- Questions about the assignment?
- Minute essay comments:
  - Suggestions all went into the assignment.
  - Usefulness of matrices.

# Menus

- The frame has a MenuBar.
- Contents of a MenuBar should be Menus.
- Menus can hold the following:
  - MenuItem
  - Menu
  - CheckMenuItem
  - RadioMenuItem
  - Separator

# Interactive GUIs

- Last time we learned how to build a GUI with components.
- Our GUI wasn't interactive though.
- Buttons and MenuItem's are easy because we give them an Action.
- Simple read model of console apps is insufficient.
- Need a way to deal with input from many sources.

# Publisher/Reactor

- The basic model employed by `scala.swing` is Publishers and Reactors.
- A Reactor can `listenTo` events from a Publisher.
- Events are in `scala.swing.event`.
- To stop listening, set yourself to be `deafTo`.
- The Publisher can be the component or an object in the component.

# Partial Functions

- Scala has a construct called a partial function. It is a function that only works on some inputs.
- The brief syntax for them is like a match with no match. So it has curly braces with cases in them.
  - {
    - case 1 => doOption1()
    - case 2 => doOption2()
    - ...
  - }

# Events/reactions

- When a Reactor is listening to a Publisher, it needs to define reactions.
- You can add PartialFunctions into the reactions of the Reactor.
- They should respond to the events you are interested in.
  - `reactions += {`
    - `case e:ActionEvent => ...`
    - `case e:SelectionChanged => ...`
  - `}`

# Editing Pokemon

- Let's use this new knowledge to make it so that we are really editing our Pokemon.



# Minute Essay

- Questions?
- Interclass Problem:
  - Try to make your calculator (or whatever you wrote for today) work. At the very least put in some interactions. (Like clicking buttons changes the display.)