Opening Discussion

- Questions about the assignment?
- Minute essay comments:
  - GUI ideas
    - Class schedule maker/tester
    - Viewing physics simulations
    - GUI displaying your spring break activities
    - Something easy
    - GUI for a website
    - Alternate sports stats
    - Concert ticket reservations
    - Ray tracing geometry editor and viewer
Music GUI

Text adventure in a GUI

GUIs seem to have lots of typing.

Is there a panel type where East and West behave the way North and South normally do? Can you make your own panel types?

Adding pictures/icons/symbols to a GUI.
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 MenuItems 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.
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.
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()  
  - ...  
  - }  

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.

```scala
reactions += {
  case e:ActionEvent => ...
  case e:SelectionChanged => ...
}
```
Let's use this new knowledge to make it so that we are really editing our running information.
Questions?

What are your plans for Spring Break?