PA2 Discussion

By Jake

Two Programs

- TwoSmallest.java
  - Command Line
  - Takes in numbers
    - Outputs smallest and second smallest
- Mickey.java
  - GUI (Graphical User Interface) Applet
  - A click creates a Mickey
  - Mickey can be dragged around

README, not README.txt

- MUST be name “README”, not “README.txt”, “README.doc”, or anything else
- NO FILE EXTENSIONS (we won’t accept anything else)

README - Two Parts

- How to Run / High-Level Description
  - What kind of inputs/events does your program take?
  - What are the expected outputs/responses of your program?
  - Don’t assume the reader is a CS Major
- Short Response
  - Each question is 2 points
  - We are looking for specific answers

Style

- Read the Style Guidelines carefully:
  - No Magic Numbers
  - File Headers and Method Headers
  - Inline Comments
  - Meaningful Variable Names
  - Use of Blank Lines
  - Lines <= 80 characters (README too!)
  - Consistent Indentation

Use Vim (or else...)

- DO NOT use Eclipse or an IDE
  - Rick asks VIM questions on READMEs and tests! So you should know it!
- Your program MUST work on the lab computers
- Your program MUST compile to be able to turn in

TwoSmallest.java

- User inputs a series of numbers (variable size), and then ends with the EOF character (via ctrl+d (note that ctrl+d is not the same as ctrl+c or ctrl+z)
- Program outputs the smallest and second smallest numbers
- Example

TwoSmallest - Input

- Use a Scanner
  - Start by checking out the Java API
- Useful functions:
  - hasNext()
  - nextInt()
- Creating the Scanner
  - x = new Scanner(System.in);

TwoSmallest - Logic

- Loop while the scanner hasNext()
- Only hold onto three numbers:
  - The Smallest Number (so far)
  - The Second Smallest Number (so far)
  - The Current Number
- To the Chalkboard!

Mickey.java

- Display Instructions on screen to the user
- When they click, put Mickey there (there should only ever be one Mickey)
- When they click and drag on Mickey, move Mickey
- When the mouse leaves the canvas, clear Mickey.
- Example

Mickey - Outline

```
import objectdraw.*; //libraries
import java.awt.*;

public class Mickey extends WindowController //handles mouse events and drawing graphics
{
        // declare variables ......

        // initialize boxes and text at beginning of program
        public void begin()
        {
            // Add code here.
        }

        public void onMouseClick(Location point)
        {
            // Add code here.
        }

        public void onMouseEnter(Location point)
        {
            // Add code here.
        }

        // And so on...

    }
```

Mickey - ObjectDraw

- `public void begin()`
- `myText = new Text(string, x, y, canvas)`
- `myOval = new FilledOval(x, y, width, height, canvas)`
- `public void onMouseEnter(Location point)`
- `public void onMouseClick(Location point)`
- `public void onMousePress(Location point)`
- `public void onMouseDrag(Location point)`
- `public void onMouseRelease(Location point)`
- `public void onMouseExit(Location point)`

Turnin and Verify

- To turn in the assignment, type “turnin pa2”
- To verify that your turn in was successful, type “verify pa2”

- Make sure you turn in and verify before midnight on the due date. NO Late assignments will be accepted, NO EXCEPTIONS.

No PA2 Extra Credit

- Future assignments have LOTS of opportunities for extra credit.

Questions?

START EARLY!!!