Lecture 5: Listeners and assertions

CS 62
Fall 2018
Alexandra Papoutsaki & William Devanny
Inner Classes

• **DrawingCanvas** extends JPanel
  • Associates listeners for mouse actions on the canvas
  • Responsible for repainting the screen

• **DrawingMouseListener** and **DrawingMouseMotionListener**
  • Responsible for responding to mouse actions by changing the items in the ArrayList.
Handling Mouse Events

• If you want your program to react to mouse press, click, or release on a component
  • send `addMouseListener(mlo)` to component (usually in the constructor of the component)
  • See `PostItApplication.java` and `Demo.java`
  • For motion or drag, send `addMouseMotionListener(mlo)`

• When user presses mouse on a component
  • Computer looks for registered `MouseListener` for component or its containers.
  • If found, sends `mousePressed(evt)` to listener
Listener

- Object designated as mouse listener must
  - implement `MouseListener` (& implement `mousePressed, mouseReleased, & mouseClicked`) or
  - extend `MouseAdapter` (which has default implementations of all 3)

- Second is easier unless a class already extends another.
  - *Can only extend one class in Java*

- Similarly, for mouse motion listener
  - implement `MouseMotionListener` or
  - extend `MouseMotionAdapter`
Listeners in PostItApplication

• Main class (this) is listener for button and choice. Set up when GUI items constructed

• Special listener objects for mouse actions. Set up by DrawingCanvas since listening for actions on that object.

• When button or combobox are clicked the actionPerformed() is triggered.
Pre and Post conditions

• Pre-condition: Specification of what must be true for method to work properly

• Post-condition: Specification of what must be true at end of method if precondition held before execution.

• See Ratio class example
Assertions in Java

• Won’t use `Assert` class from Bailey.
• Command to check assertions in standard Java
• Two forms:
  • `assert boolExp`
  • `assert boolExp: message`
• Article on when to use `assert`:
  https://docs.oracle.com/javase/8/docs/technotes/guides/language/assert.html
Assertions help...

- Defensive programming
- Little cost to executing assertions ... and can turn off checking
- Extremely useful in debugging in tracking down what is going wrong - can be better than inserting println’s.
- Also useful in checking cases that should not occur
  - e.g., defaults in `switch`, other control paths not taken.
- Do NOT use argument checking in public methods
  - Should throw an exception instead
- Do NOT use to perform action that is critical for the program
  - Instead perform the action before the assertion and then assert that the action succeeded
Turning on assert

• Turn on assertions when run program, by adding `-ea` as virtual machine argument in arguments tab in Eclipse when set up runtime configuration.
• If you leave it off, then it ignores assert statements.
• If on and the assertion is false, then it will raise an `AssertionError` exception and will print associated message.
• They should not be caught as represents a program error.