Java SWING Event Handling

CSCI 2300

Input

So we now know how to present widgets on the screen

A program also needs to react to the user's actions

Examples:

- When the user presses a button we want to save a file
- When the user closes the program we want to ask “are you sure?”
- ...

Swing mechanism: Events and Listeners

Events, Listeners

Swing defines all sorts of Listener interfaces

- E.g.: `ActionListener`, `MouseMotionListener`, ***`WindowListener`***, ...

```
public interface ActionListener extends EventListener {
  public void actionPerformed(ActionEvent e);
}
public interface MouseMotionListener extends EventListener {
  public void mouseDragged(MouseEvent e);
  public void mouseMoved(MouseEvent e);
}
```

There are default (empty) implementations for many of the listeners

- E.g.: `MouseMotionAdapter`, ***`WindowAdapter`***

Events, Listeners (cont.)

A listener is an object that implements a listener interface

If we need to react to an event (on a certain widget) we register a listener object with that widget

E.g.: `addActionListener()` registers an action listener with its receiver:

```
JButton button = new JButton();
ActionListener listener = ...;
button.addActionListener(listener);
```

When an event occurs, all registered listeners are notified

- The appropriate listener method (e.g: `actionPerformed()`) is invoked
- An object describing the event is passed as a parameter

4

Event Demo: GUI

Code posted online: EventDemo.java

5

Inner Classes

Nested within another classes

Instance specific:

- Has access to methods & fields of the object that created it
- => An inner class has TWO **this** variables

Can be static

- Can access only static members and methods only
- A static method cannot create a non-static inner class

6

Local Classes

Same as inner classes but defined inside a method

Has access to local variables of the enclosing method
- Only if the variable is defined as final

Can be anonymous
- Doesn't have a name.

7

Event Handling Demo: Local Class

Code posted online: EventDemoLocal.java

8

Accessing Fields of Enclosing Object

```
public class A {
  int x = 0;
  public void f() {
    B b = new B();
    b.g();
    System.out.println(x);   // Output: 5
  }

  public class B {
     public void g() { x = 5; }
  }

  public static void main(String[] args) {
    new A().f();
  }
}
```

9

Using the Second this Variable

```
public class A {
  public void f() {
    B b = new B();
    System.out.println(b.g());   // Output: 1024
  }

  public int g() { return 512; }

  public class B {
    public int g() { return A.this.g() * 2; }
  }

  public static void main(String[] args) {
    new A().f();
  }
}
```

10