File I/O

18 Feb 2008
CMPT166
Dr. Sean Ho
Trinity Western University

Quiz3

- Given:

```
public class Ferrari extends Car {}
Car sentra = new Car();
Ferrari f430 = new Ferrari();
```

which of the following are legal, and why? [6]

(a) sentra = f430 (b) f430 = (Ferrari) sentra

(c) sentra = f430; f430 = (Ferrari) sentra

- Contrast abstract superclasses with interfaces [4]
- Contrast JFrames with JPanels [4]
- Pseudocode a simple event-based Swing program as a subclass of either JFrame or JPanel [6]

Quiz 3: answers #1-2

- sentra = f430
  - Okay: f430 is also a Car
- f430 = (Ferrari) sentra
  - Not okay: sentra can never be a Ferrari
- sentra = f430; f430 = (Ferrari) sentra
  - Okay: sentra refers to a Ferrari (downcast)
- Contrast abstract superclasses with interfaces
  - Superclass: identity; inherit variables/methods
    - No multiple inheritance in Java
  - Interface: capability; multiple interfaces okay

Quiz3: answers #3-4

- Contrast JFrames with JPanels [4]
  - JFrame: Swing window
  - JPanel: Swing container for widgets; has a layout manager
- Pseudocode a simple Swing program as a subclass of either JFrame or JPanel [6]
  - (see, e.g., Histogram.java)
  - public class MyProgram extends JPanel
  - main(), createAndShowGUI(), constructor, event handler

java.io classes

- Byte-based streams:
  - FileInputStream, FileOutputStream
- Character-based streams:
  - FileReader, FileWriter
- Object-based streams:
  - ObjectInputStream, ObjectOutputStream
- Standard streams:
  - System.in, System.out, System.err
- Object holding pathname information:
  - File

File methods

- Constructor:
  - File oFile = new File(“output.txt”);
- Check if exists, can read/write:
  - if (oFile.exists() && oFile.canRead())
- Check file type:
  - If (oFile.isFile() || oFile.isDirectory())
- Get parent directory:
  - oFile.getParent()
- Get just the filename:
  - oFile.getName()

JFileChooser

- Instantiate and set selection mask:
  - JFileChooser chooser = new JFileChooser();
  - chooser.setFileSelectionMode(JFileChooser.FILES_ONLY);
- Pop-up dialog:
  - int result = chooser.showSaveDialog(this);
- Check if user pressed Cancel:
  - if (result == JFileChooser.CANCEL_OPTION);
- Get the filename the user chose:
  - File fileName = chooser.getSelectedFile();

Formatted text stream I/O

- java.util.Formatter:
  - Outputs formatted text to a file
    - Formatter output = new Formatter(“out.txt”);
    - output.format(“%d %s\n”, id, name);
- java.util.Scanner:
  - Reads formatted text from a stream
    - Scanner input = new Scanner(new File(“in.txt”));
    - new Scanner(System.in);
    - id = input.nextInt();
    - name = input.next();
- Remember to close()

Exceptions in file I/O

- An instance of the class SecurityException is raised if file permissions fail:

```
try {
    out = new Formatter( “out.txt” );
} catch ( SecurityException e ) {
    System.err.println( “No write permissions!” );
}
```

- FileNotFoundException, IllegalStateException
- Scanner raises NoSuchElementException if the data is in the wrong format
- EOFException, IOException