# §14.1-14.5: Sequential-Access Files

19 March 2007
CMPT167
Dr. Sean Ho
Trinity Western University

# Review last time

- Event handling
  - Delegate classes
  - Subclasses of ActionEvent
  - Sub-interfaces of EventListener
- Swing widgets
  - JLabel
  - JTextField, JPasswordField
  - JButton
  - JCheckBox, JRadioButton, JComboButton
    - ItemListener interface and ItemEvent class

# 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()