CSC108H lab – week 8

This document contains the instructions for the week 8 CSC108H lab. To earn your lab mark, you must actively participate in the lab. You don’t need to finish in the time allotted, you just need to try hard.

In this lab, you will work with your partner to learn more about file reading, panels, and loops. All file reading is done in package java.io, so remember to import.

1 Starting up

Sit down with your partner. The rest of these instructions call you two s1 and s2. Pick which one is which. s1 should log in and start up DrJava, and be the first driver.

2 File copying

This opens a file for reading: BufferedReader br = new BufferedReader(new FileReader("f.txt"));
To read a line, use br.readLine(). When there is no more input, br.readLine() returns null.
The following code opens a file for writing; p has methods print and println, just like System.out:

PrintStream p = new PrintStream(new FileOutputstream("file.txt"));

Most of these I/O operations might throw an IOException. You must add throws IOException to any method header that uses any of this code, or use a try/catch block. (In 108, you only need to know how to use the throws IOException, but some of you have been curious about try/catch.)

Write a class FileCopier with one static method copy that takes the name of the input file and the name of the output file, and copies the contents of the input file to the output file. As a warning, do not use an important file as the output file; you will lose the contents!

Save FileCopier.java directly in your H: drive. This will be important in the last section.

Compile it, test it, and fix any errors. You can, of course, open the output file with DrJava.

3 Showing a file

Switch roles: s2 drives and s1 navigates.

Write a subclass of JFrame called FileDisplay that has one 40 x 80 JTextArea (40 rows, 80 columns) in the center. The constructor takes the name of a file as a String, and displays the contents of the file in the text area. For example, new FileDisplay("FileDisplay.java").show() makes a new window showing your class.

Make sure you pack the window at the end of the constructor.

Save FileDisplay.java directly in your H: drive. This will be important in the next section.
Both the FileReader constructor and method readLine might throw an IOException. You must add throws IOException to your FileDisplay constructor method header, or use a try/catch block. If you don’t know how to append text to a JTextArea, look it up in the APIs. Compile it, test it, and fix any errors. Demonstrate to your TA that it works, if they’re not totally swamped. :-)

4 JavaDoc!

Switch roles: s1 drives and s2 navigates.
In this section, you’ll see JavaDoc in action. Add a comment like this to FileDisplay:

    /** A window showing a file. */

Add a comment like this to the FileDisplay constructor (assuming you named your parameter f):

    /**
     * A new window showing the contents of file f.
     * @param f the name of the file.
     * @throws IOException if there is a problem reading from f.
     */

In the Start menu, select “Run”. Type javadoc *.java in the window that appears. This runs JavaDoc on every Java file you have at the top of your H: drive. You will see lots of output.

In Windows Explorer, navigate to your H: drive. You’ll see a file called index.html. Double-click on it, and gasp in astonishment at how cool this is. (Note: make sure your TA hears you gasp.)

Now do the same for FileCopier.java: add good comments, including using @param and @throws tags. Here, use one @param tag for each parameter. Then run javadoc *.java again, and reload index.html in your browser.