COMP101 Lab 1: First Java Programs

Not Marked, Not Assessed

The module web page is at `http://www2.csc.liv.ac.uk/∼martin/teaching/COMP101/`.

If you have any problems using the Departmental Computer Systems please have a look at the documents "Using the Departmental Windows and Linux Systems" linked from the module web site (see the section under "Lab" on the "Course Schedule" webpage `http://www2.csc.liv.ac.uk/ martin/teaching/comp101/schedule.php`).

I would expect you to open a *command terminal* and use an editor to create/edit your files, e.g. *WordPad* or *Notepad++* on Windows, or *vim* or *emacs* on a Unix system.

I would recommend you to create a directory for your COMP101 work. Navigate to that directory using the commands: *cd directoryName* (change directory), *pwd* (state the current directory), *dir* list the contents of the current directory. If required, cut and paste sample files below into the editor you are using and save them or save directly into the directory you require from the browser as a Java source file.

Your Java classes should successfully compile and run on ANY departmental computer system, both Windows and Unix. For this reason, the use of interactive development environments (IDEs) like NetBeans or Eclipse is discouraged.

1. Download the `HelloWorld.java` program from the module web site and save as "HelloWorld.java" in a suitable directory. Compile the program and then run it by typing

   ```
   javac HelloWorld.java
   java HelloWorld
   ```

   Make certain that you know how to successfully compile and run a Java program in this manner, as this is how your programs will be executed when they are assessed. (In other words, you should test your programs in this fashion before you submit them to be assessed.)

2. Copy `HelloWorld.java` into a new file `FirstProgram.java`. Note you must change the class name to be FirstProgram. Instead of printing "Hello World", change the text to say the following where "MyName" is your own name.

   ```
   My first Java program.
   Created by MyName
   ```

   Compile and run the program.

3. Download the `HelloWorld1.java` program from the module web site and try to compile this. There will be a number of errors. What are these? Fix them until the program complies and runs.

4. Which of the following is <u>not</u> a correct Java comment?

```
/** comment */

/* comment */

/* comment

 // comment
```

If you are unsure, change one of the programs you have written and try them out.

5. Write a Java program that declares some variables to be integers and doubles. Assign values to these variables. Try applying some of the numeric operations (+, -, *, /, %) discussed in lectures using some of these variables and print out the results. Make sure the values are as you would expect. Has promotion of variables occurred in any case that you tried out?

6. The exchange rate at the time of writing is £1 is €1.16771 and €1 is £0.85624. Write a Java program that assigns a value in pounds to a variable and then calculates the amount this is in Euros. Print out the value in pounds and also in Euros. Also assign a value to a variable in Euros, calculate its value in pounds and print out both values.

7. Many modules in the Department of Computer Science have a coursework mark with weight 25% and the exam mark with weight 75%. The coursework mark and exam mark will always be a whole number. From these values we can calculate an overall mark for the module. Write a Java program that declares variables and/or constants to capture this data. Assign values to the variables denoting the coursework and exam marks, print out the coursework and exam marks, and calculate and print the overall mark.

8. Experiment to see what happens if you try assign a value to a variable that is too large for the data type.

9. Read the "Getting Started" material from the Java Tutorials (see
http://docs.oracle.com/javase/tutorial/getStarted/index.html).

10. Read the "Object-Oriented Programming Concepts" and "Language Basics" material from the Java Tutorials (see
http://download.oracle.com/javase/tutorial/java/index.html).

Any questions not finished in your lab slot should be completed as self-study.