Java Tutorial
MSc/MRes Web Science & BDA

Jun Wang and Ying Wen
j.wang@cs.ucl.ac.uk and ying.wen@cs.ucl.ac.uk

September 27, 2016

Candidates for the MSc/MRes in Web Science and Big Data Analytics must be sufficient programmers in order to meet the requirements of some of the modules, and the programming language of choice is usually Java. The purpose of these exercises is to ease you into programming with Java and have been designed for students with different programming ability.

The exercises are split into three difficulty sections, you should start from the section that you are most comfortable with and aim to complete all exercises from that point onwards:

Beginner: For students who have little or no programming experience, or who have never used Java before.

Intermediate: For students who are somewhat familiar with programming languages, but with little experience with Java.

Advanced: For students who are experienced with programming in Java.

You will have 2 hours to work on these exercises in the lab with the assistants, you are not expected to complete them all in this time. Please read over and attempt the exercises beforehand so that you can make the most of your lab time with the assistants. Please make sure that you complete all of the exercise sheet in your own time after the lab.

1 Environment Setup

For the purposes of this assignment, we will be using the NetBeans IDE (Integrated Development Environment) which is already installed on all of the lab computers. If you'd like to work on your own computer, you can download the latest version of Java and NetBeans at http://www.oracle.com/technetwork/java/javase/downloads/jdk-netbeans-jsp-142931.html.

2 Exercise

2.1 Beginner

1. To familiarise yourself with Java, NetBeans and running programs, please read through Section 1 on http://www.homeandlearn.co.uk/java/java.

html

2. To run a simple "HelloWorld" Java program in NetBeans, follow the instructions here http://docs.oracle.com/javase/tutorial/getStarted/cupojava/netbeans.html

3. For help with the following exercises, please read through Section 2 on http://www.homeandlearn.co.uk/java/java.html.

   (a) Create a program that displays your name in the Output Window.

   (b) Create a program that asks a user for their name and then displays it in the Output Window.

   (c) Create a program that asks a user for two different numbers and then multiplies them and displays their product in the Output Window.

4. For help with the following exercises, please read through Section 3 on http://www.homeandlearn.co.uk/java/java.html.

   (a) Modify your program in Exercise 3a to only multiply numbers between 10 and 100.

   (b) Create a program that allows a user to enter three numbers a, b and c. If $a^2 \times b^2 = c^2$, then display the message "This is a Pythagoras Triple".

   (c) Create a program that allows a user to enter two numbers, and then displays every even number that lies between those two numbers.

2.2 Intermediate

For help with the following exercises, please read through Sections 6, 7 and 8 on http://www.homeandlearn.co.uk/java/java.html

1. Create a Java Class called *Calculator*.

2. In that Class, create a method call *plus* that takes two integer parameters and returns those two parameters added together.

3. Create three similar methods called *subtract*, *multiply* and *divide* which perform the corresponding operations.

4. Write a new program that creates a new Object of type *Calculator*. Write code that asks a user to input two numbers, then a String representing an operation (such as "+" or "plus"), then outputs the result of that operation on the two numbers using the methods in the *Calculator* class.

5. Modify the *divide* method to throw an appropriate error if a division by zero occurs.

6. Use inheritance to create a new Class called *ScientificCalculator* that extends *Calculator*. In this Class, create a new method *power* that returns the first parameter a to the power of the second parameter b i.e. a^b. Change your program to allow users to use this new operation.

7. Create a new method called *factorial* that uses recursion to find the factorial of a number. Please read http://danzig.jct.ac.il/java_class/recursion.html if you would like to know more about recursion.

2.3 Advanced

1. Print out a list of the first 1000 prime numbers to a file called *prime.txt*.
2. Download the 'Alice in Wonderland' Plain Text file at http://www.gutenberg.org/cache/epub/11/pg11.txt and create a program that counts the total number of words (tokens separated by spaces, don't worry about punctuation in this exercise).
   (a) Output to a file each unique word and how many times it occurs.
   (b) Modify this program to list the words in decreasing order of frequency.

3 Further Reading

There are some official tutorials by Oracle are available at http://docs.oracle.com/javase/tutorial/, which are practical guides for programmers who want to use the Java programming language.

An alternative IDE to NetBeans is Eclipse http://www.eclipse.org/downloads/ which is generally more powerful, but more complicated to use.

If you are interested in more challenging exercises, then consider visiting Project Euler http://projecteuler.net/

3.1 Textbooks

Some good Java text books include:

- 'Introduction to Programming in Java' is a free ebook available online at http://introcs.cs.princeton.edu/java/home/ which also has a corresponding 'Introduction to Computer Science'.
- 'Head First Java' by Kathy Sierra, which is good book for beginner.
- Some good books available in the Science library:
  - 'Thinking In Java' by Randall Newton.
  - 'Java in a nutshell : a desktop quick reference' by David Flanagan.
  - 'Java examples in a nutshell : a tutorial companion to Java in a nutshell' by David Flanagan.
  - 'Java for programmers' by Paul J. Deitel and Harvey M. Deitel