Readme

Welcome in the beautiful world of programming. Maybe some of you already have some ambitions, or maybe you want to discover your potential. Regardless your mission goal, this guide will lead you through the crash course of JAVA programming which will broaden your vision.

Course contents

The crash course will consist of several items, namely: (in the order of occurrence)

- Primitive datatypes (like string, doubles, integers etc)
- Conditionals (if and else statements)
- Loops (do…while, while, for loops)
- Methods
- String methods
- Error handling
- Dealing with in- and outputs (e.g streamreader and streamwriter)
- Object Oriented Programming (OOP paradigm)

And if there is any time left, I might implement a small assignment about making a graphical user interface.

Before the fun

Before you start cranking code, the first things you need to install are an IDE (Integrated Development Environment) and the JAVA SDK 1.6 (or lower if you prefer that). The choice of IDE is free, but it is recommended to use Netbeans.

- Java SDK6 U29: http://www.oracle.com/technetwork/java/javase/downloads/jdk-6u29-download-513648.html Select your platform.
- Netbeans: http://netbeans.org/downloads/index.html Select either the Java SE, Java EE or All version

Now you finished the first steps, it is very important to learn the proper habits of programming. There is nothing worse than learning something on the bad manner and it should thus be prevented. As a result, two coding guidelines were uploaded and I strongly encourage you to **read these thoroughly** and applying them. The guidelines can be found at the following locations:

- The folder Information. Read the pdf from slide 15 to 21, mainly studying slides 15 till 19
- https://docs.google.com/View?id=dhqzrbwz_42g8k89pf7&pli=1

General programming tips

One of the good practices in programming is proper indenting. This not only enhances the readability of the program you're writing, it also enhances the maintainability and the reviewability of the program. Nothing is worse than having an improper indenting and having me (or even worse YOU) to find out what's wrong with either the program or what you are actually doing.

A second good practice is the use of proper commenting. Comments should be descriptive and **not** a repetition of the statement. An example:

//Add a + b and assign to variable C
C = a + b;

This is a bad comment as it basically repeats the statement below. This is an unnecessary comment, as I (and you hopefully) can deduce this already. An example of a good comment would be:

//The variables a and b are added to the total cost C, as a and b represent the price of an apple and
//pears respectively.
C = a + b;

This is a good comment as it describes WHY you are doing this. This is a so called constructive comment.

Note that there is no such thing as over-commenting. Although it is not widely used (commenting everything takes simply too much time), I strongly encourage you to comment the most important part of the program, such that I can understand what you are doing or trying to do in your program.

The third and last thing a programmer should always keep in mind: the code you are producing must be understandable for any outside programmer. In this way, not only I and others can understand it, but it will keep your own comprehension of the program at a high level when you review it.

Logistics

Additional information
As I was not able to scan in Chapmans book (over 40 pages per time, it takes too long to scan), I found another suitable e-book. As I am against boring texts for programming, I made the Java for Dummies (ebook) available.

File logistics
Dropbox is used in combination with git, which brings me to the following point. In the Dropbx, a hidden folder .get can be found. I strongly emphasize that this folder should not be touched or be moved (let alone deleted) as it is being used by me to upload all the files to my gitserver.

Files should be uploaded in Assignments<x>\<Your name>. As an example: suppose I want to make assignment 1 and my name is Tyler Skidmore. Then, the map where I should upload the files can be found in the folder assignment 1 and voila, my name is there too.

As GIT is being used to synchronize all the made files with my Gitserver, it is **mandatory** to upload only the .java files which is due to the fact that each compiled program is computer bounded.

This means that you have to make the assignments at your own desktop computer. Just make the assignments at a convenient location somewhere at your desktop and when you are finished… Just upload the .java file into the Dropbox. If you want to keep backups of your .java files, again upload the files at Dropbox, as each day the files are synchronized with my gitserver.
The last thing I have to emphasize: **Do not compile in the Dropbox as this will mess up things badly!**

The files will be synchronized each day at approximately 10.00 PM till 12.00 PM (GMT+1, so this corresponds with 04.00 PM till 06.00 PM at your local time) with my github server.
In case of an emergency (e.g. you messed up something in your desktop programming folder) I can get you a backup of the file(s) from a day before. Do not hesitate to contact me when such a situation occurs.

Course set up
The setup of the course is as following: each week an assignment needs to be made, which has to be handed in somewhere in the subsequent week (so you have one week to make your homework). During that time I will check the file on several criteria (to be determined) and set it back (by putting it back in the folder with my comments in the file). You then have a week to improve your assignment and making the next assignment. A detailed schedule can be found in the Dropbox folder.

File criteria
The programmed files need to comply with several criteria, namely:

- They need to conform according to the coding guidelines I posted
- Work with JAVA SDK 1.6 or older (do **not** use Java 1.7)
- Is self explanatory

In addition, uploads should only be the .java files in your **own folder**.

Fraud
Although I cannot report fraud to some committee, I strongly discourage the use of it. As I know that there are possibilities to edit someone else its files, I can monitor this and will result in a warning.

I encourage the use of cooperation, but all code should be written by own means.

Contact
In case of (programming) questions related to the assignments, you can contact me at karabasf@gmail.com with the subject: JAVA crash course: <Insert your subject/question>. As feedback is much appreciated, feel free to mail me if you have something as a piece of advice or whatsoever!

Please note that I am a student and I will not be able to respond at all times.

Acknowledgements
All material was made available by friends and the University of Technology Delft

The last thing what is left to say is:

Happy Coding ^^