

A Java Crib Sheet

- Unlike JavaScript, which is pretty much ready-to-go on any computer with a modern Web browser, Java might be a more complex affair
- However, logging some time with Java can be fairly valuable, if only to experience the typical programming cycle of *code-compile-execute*
- This handout hopes to hit the salient points involved with getting Java up and running on your machine; as always, individual variations may emerge, so never hesitate to ask if something unexpected comes up

First: Find the Command Line

- We will be working from a *command line* interface, so the first order of business is to find it on the computer that you'll be using for Java
- On Windows, the quickest way to the command line is to choose *Run...* from the *Start* menu then enter *cmd* as the program to run
- On Mac OS X, you get to the command line through the *Terminal* application, which can be found in the *Utilities* folder inside the *Applications* folder

Command Line Basics

- To invoke a command, type it into the command line window then press the Enter or Return key; the computer then displays some response to that command (which may or may not make sense to you)
- To get out of the command line in both Windows and Mac OS X, type *exit* and press Enter or Return
- On Mac OS X, the *Terminal* program remains running, in case you want to open a new command window; you'll need to *Quit* as well if you're sure you won't need another command window anytime soon

Check for Java

- At the command line, enter *javac* (the Java compiler); if the Java development kit is properly installed on your machine, you should see a long-ish message describing how to use the *javac* program
- If you don't get this message, then you'll need to install the Java Development Kit (JDK); if you're sure you've already installed it, you may need to contact me at this point so we can figure out what went wrong
- Mac OS X typically comes with JDK "out-of-the-box;" it's Windows that might not have it preinstalled

Downloading and Installing Java

- If you conclude that you *do* need to download and install Java, you can find it here:

<http://java.sun.com/javase/downloads>

- Notice that there is no Mac OS X JDK download — as mentioned, modern Macs all have JDK preinstalled (so if you're using Mac OS X and are having trouble invoking *javac*, let me know)
- Make sure to download the latest *JDK*, not *JRE* (this is a smaller version of Java that is meant for end-users instead of programmers)

Let's Work on the Desktop

- You can really work anywhere on your computer's disk drive, but for convenience and visibility, let's work on your account's desktop
- On Windows, enter *cd Desktop* if the command line prompt says *C:\Documents and Settings\your-username*; if not, enter *C:* first, then *cd \Documents and Settings\your-username\Desktop*
- On Mac OS X, enter *cd ~/Desktop* to get there
- The nice thing about working on your desktop is that you'll immediately see the files that you create as icons

Create Your Java Source File

- As with XHTML, Java source files are simply plain text files — so they can be created with the likes of Windows *Notepad* or Mac OS X *TextEdit*
- Type up the code, then save it onto your desktop
- Java source files follow a strict naming system: they should bear the name of the *class* that is in the source (you'll find it in *public class name-of-class*, a few lines from the top) and must end with *.java*
- Again, you can really save the file anywhere, but we're just using the desktop for convenience

Compile Your Program

- We'll use the command line to compile your program
- In case you're wondering, yes, there *are* development tools that have more “modern” user interfaces — still, in the end, it boils down to invoking a *compiler*
- Go to your computer's desktop on the command line (see above), then enter *javac name-of-file.java*
- If all goes well, *javac* will simply finish as if nothing happened — but you should see something new: a file called *name-of-file.class* — this is the translated code

If Problems Arise...

- Don't feel bad if *javac* does complain — it happens to everyone, including professional programmers
- The first thing to check is to make sure that you typed in your program exactly as provided; the error message(s) provided by *javac* should give you a hint as to where something went wrong
- If you can't get *javac* to compile your file properly, just contact me — the problem is usually easy to fix, but only for someone who writes code for a living :)

Run Your Program

- Once your class file is successfully created (i.e., you have a *name-of-file.class* file on your desktop), you can invoke it through *java* (without the *c*)
- Enter *java name-of-file options* (no extensions this time — no *.java* nor *.class*); *options* includes any information that you would like to provide to the Java program
- If all goes well, the program should do its work and show you some kind of a response
- Note that errors may happen at this stage too — when in doubt, ask me

Have Fun!

- Remember, the point of this exercise isn't really to learn parts of Java as a language; it's mainly to get a feel for how compiled programming goes
- Never, never hesitate to ask — this may be your first time out, so there are sure to be glitches