Getting Started with Java

Maria Litvin
Phillips Academy, Andover, Massachusetts

Gary Litvin
Skylight Publishing

1. Questions and Answers

What software do I need?

You need a Java compiler and an IDE.

What is a compiler?

A compiler translates Java source code (program text written in Java) into bytecode (instructions for Java Virtual Machine).

Where do I get a Java compiler?

There is essentially only one Java compiler. It is part of JDK (Java Development Kit) and it is available free for downloading from Oracle's web site (see instructions below). The latest version, as of July 2015, is JDK 8u51, but an earlier version (for example JDK 7) will serve your needs, too. JDK also includes an interpreter (which interprets bytecode on a particular system and runs Java programs), Java libraries, and a few utility programs (in particular *javdoc*, for generating program documentation automatically, and *jar*, for creating your own Java libraries and runnable "jar" files).

What is an IDE?

An IDE (Integrated Development Environment) serves as a front end for the Java compiler and interpreter. It also includes an editor for writing Java code, a project maker, sometimes a debugger, all conveniently packaged into one program with a graphical user interface.

Install the JDK first, then an IDE.

Where do I get an IDE?

Several free IDEs are available for downloading. One, called *NetBeans*, is available from Oracle in a "cobundle" with the JDK. However, it may be too cumbersome for a beginner programmer. Some of the popular choices are *jGRASP*, *DrJava*, and *BlueJ*.

(We used to recommend *JCreator LE* from Xinox software, but it doesn't run on Macs, and Xinox started charging license fees for it. The older version, LE 4.5, may be still available as a free download at various web sites.)

On the opposite end of the spectrum is *Eclipse*, a professional IDE, highly configurable and loaded with convenience features, also available free. It might take some getting used to, though. Eclipse runs on both PCs and Macs. We have detailed instructions for downloading and using *Eclipse* in a separate document.

Some of the "light" IDEs do not have a debugger, which we consider a plus (see Top Ten Reasons Not to Use a Java Debugger in School).

What else do I need?

Get very proficient with viewing, moving, deleting, and renaming files and folders in your operating system.

2.  Adjusting Windows Setup

The following *Windows* configuration changes are not mandatory but will make working with Java easier:

- In any folder on the `Tools` menu choose `Folder options`. Click on the "View" tab, find and uncheck the "Hide extensions for known file types" box.
- You might find it more convenient to view folder contents as a list or a detailed list, rather than icons.
- You will use *Windows Explorer* often, so place shortcuts to your frequently used folders on your desktop.
- If you plan to run your Java programs from *Command Prompt*, find *Command Prompt* under `All programs/Accessories` and put a shortcut to it on the desktop.

3. Downloading and Installing the JDK

To download the JDK, go to http://www.oracle.com/technetwork/java/javase/downloads/index.html.

Click on the "Download" button. Click the radio button to accept the license agreement, then click on the installation file appropriate for your system, for example, `jdk-8u131-windows-i586.exe` for 32-bit Windows systems or `jdk-8u131-windows-x64.exe` for 64-bit Windows systems. Save the JDK installation file to the `Downloads` folder or a folder of your choice.

The JDK includes the compatible version of the JRE (Java Run-Time Environment).

To install the JDK, run the downloaded installer file and follow the instructions. Accept all the defaults.

 In Windows, the JDK is installed by default in the `C:\Program Files\Java\jdk-...` folder.

4. Downloading and Installing an IDE

We have instructions for downloading, configuring, and using Eclipse, jGRASP, and JCreator at skylit.com/javamethods/faqs:

- Eclipse
- jGRASP
- JCreator

For other IDEs, see their own web sites for instructions or search for instructions and tutorials on other web sites.

5. Accessing the Java API Documentation

"API" stands for Application Programming Interface. In the context of Java it refers to the detailed description of the Java library packages, classes and interfaces. The API docs are online at http://docs.oracle.com/javase/8/docs/api/index.html. You might want to add a shortcut to this page to your desktop.

It is also possible to download the documentation files to your computer and work with them offline. The Oracle's documentation download page is not easy to find in the latest release: it is hidden at http://www.oracle.com/technetwork/java/javase/documentation/jdk8-doc-downloads-2133158.html.

After downloading the zipped file, (for example, `jdk-8u51-docs-all.zip`) extract the `docs` folder from it and move it to the `jdk...` folder (for example, `jdk1.8.0_51`), at the same level as `bin` and `lib`. (Extracting the `docs` folder may take a long time, up to an hour.) Many IDEs (including Eclipse) provide context-sensitive help: a user can position the cursor over a library class name in the editor and press a key (`Shift+F2` in Eclipse) to bring up the Java API page for the class.

The API page for each Java class describes the package this class belongs to, the class it extends (the *superclass*), the interfaces it implements, if any, and the public "fields", constructors, and methods defined in the class.

Reading Java API docs is a useful skill but it might be daunting, since the complete API includes hundreds of classes with thousands of entries. A beginner Java programmer may occasionally need to refer to the APIs for the `Integer`, `Double`, `String`, `Math`, `ArrayList`, `Arrays`, `Collections`, `Color`, `Graphics`, and `Random` classes. A programmer working on a GUI (Graphical User Interface) will also need to use APIs for the classes defined in the `javax.swing` package.