
How to Run Java Programs with JDK and a Text Editor

by C. Thomas Wu

In this document, we will explain how to run Java programs using a plain JDK and Notepad. Although there are many possible variations, to make things simple, we will explain here one particular way that should apply to almost everybody. For this handout, we assume the Windows platform. If you are taking a class, you should check with your instructor or T.A. for instructions specific to your case.

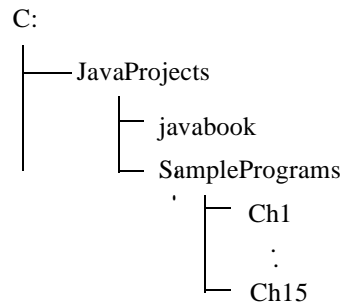
1.0 Location of JDK and JavaBook package

You must download the JDK from Sun Microsystems (www.javasoft.com/products/jdk/1.2/) and the javabook package from our web site (www.drcaffeine.com). We strongly recommend that you download the sample programs also from our web site. We assume the most recent JDK, which is Java 2 Platform (formally called JDK 1.2). Most of the discussions presented here will apply to older versions of JDK, but please be aware that the javabook package requires JDK 1.1.5 or above.

2.0 Directory Structure

We assume the JDK is installed under the directory `jdk1.2` in the C: drive.

For the javabook package and sample files, we assume the following directory structure:



Notice that the Java package and the OS directory have one-to-one correspondence. Make sure that the javabook directory contains all the bytecode files of the javabook package (MainWindow.class, InputBox.class, etc.)

3.0 Modify the Autoexec.bat File

You must let Java compiler know where the packages and other necessary files can be located. The most straightforward way to do this is to modify the autoexec.bat file. First, you should modify the PATH environment variable, so you can invoke the Java commands from any directory. Add the following line:

```
SET PATH=%PATH%;C:\JDK1.2\BIN
```

Second, to let the Java compiler find the necessary packages, add the following line:

```
SET CLASSPATH=.;C:\JavaProjects
```

Notice that there's a dot (period) right after the equal symbol.

4.0 Entering a Java Program

Using Notepad or any other text editor, enter your Java program. When you save the program to a file, name this file with the class name you are defining and add a suffix .java. For example, suppose the class is FunTime, then you save this program to a file named FunTime.java. We recommend you create a separate directory (folder) for each project (program). For example, you may want to create a directory named Test1 under the JavaProjects directory and place the source file FunTime.java in Test1.

5.0 Compiling a Java Program

Now open the DOS Prompt window and enter the command **javac** to compile a program. For example,

```
> javac FunTime.java
```

6.0 Running a Java Application Program

After a successful compilation, you can run it by entering

```
> java FunTime
```

at the DOS prompt.

7.0 Quick Summary

Here's a quick summary of the steps you take to enter, compile, and run a Java program.

1. Modify the file **autoexec.bat**. (You do this only once.)
2. Enter the program using a text editor.
3. Compile the program with **javac**.
4. Run the program with **java**.

8.0 Running Applets

Steps for compiling and executing applets are similar to those for applications. The main difference is that an applet requires an html file. The role of html files is explained in Chapter 2. Here are the steps to compile and run an applet. I will use **MyFirstApplet** as an example.

1. Using a text editor, enter the applet code and save it in the file **MyFirstApplet.java**.
2. Compile **MyFirstApplet.java** using **javac**.
3. Using a text editor, create the corresponding html file **MyFirstApplet.html**.
4. Run the applet using **appletviewer**:

```
> appletviewer MyFirstApplet.html
```