

Java Program Development in Linux

The procedures below provide information general to Linux environment-based Java program development. The instructions assume you are using a command-line environment.

0. If necessary (i.e., you are in a windowed environment), open a terminal or console window. If logged in using PuTTY, this step is not needed.
1. Create a source (`.java`) file by choosing the appropriate editor and typing in the program text. **Note: the file name MUST be the same as the public class name in the file.** Also, single (`.java`) file programs may not require other files but multi-file programs will need to be compiled separately. Save often when typing in your program and keep backup copies.
2. In the command window, compile your program using the Java compiler (`javac`). A typical command might be:

```
$ javac prog1.java
```

This command tells the Java compiler to compile the source file `prog1.java` and if successful, create an object (byte code) file called `prog1.class`. Also note that you do NOT compile the standard import files - they are automatically “included”. The byte code file is the “executable” file for Java. If the compile command is **unsuccessful** (i.e. you have syntax errors), then the error messages will appear on the screen and you must correct your errors (no class file will be created – note that an “old” version of the class file may exist from previous compiles so be careful – if your corrections do not appear to be working, you might want to remove the old class file).

3. If the compile succeeds, the `.class` file will be created. The traditional link operation will normally take place automatically (either at compile time or at “run” time). No executable file will be created (the `.class` file is the “executable”).
4. The program is now ready to be run. This is accomplished by using the command below, typed at the command prompt:

```
$ java prog1
```

Note that the `.class` extension is NOT included. This command invokes the interpreter to “run” the program. Either the program will run at this point or will produce an error message (from your program). There is no automatic prompt for data input unless you wrote it in your program.

5. You should save (for submission) a listing of your program. **Save and close** all of your files.