**Compilation and Execution process**

For most computer languages, the name of the file that holds the source code to a program is immaterial. However, this is not the case with Java. The first thing that you must learn about Java is that the name you give to a source file is very important. For this example, the name of the source file should be Simple.java.

If your program has compile errors, you have to modify the program to fix them, then recompile it. If your program has runtime errors or does not produce the correct result, you have to modify the program, recompile it, and execute it again.

The Java language is a high-level language, but Java bytecode is a low-level language. The bytecode is similar to machine instructions but is architecture neutral and can run on any platform that has a Java Virtual Machine (JVM).Rather than a physical machine, the virtual machine is a program that interprets Java bytecode. This is one of Java’s primary advantages: Java bytecode can run on a variety of hardware platforms and operating systems. Java source code is compiled into Java bytecode and Java bytecode is interpreted by the JVM.

While converting the source code into the bytecode, the compiler follows the following steps:

* **Parse**: Reads a set of \*.java source files and maps the resulting token sequence into AST (Abstract Syntax Tree)-Nodes.
* **Enter**: Enters symbols for the definitions into the symbol table.
* **Process annotations**: If Requested, processes annotations found in the specified compilation units.
* **Attribute**: Attributes the Syntax trees. This step includes name resolution, type checking and constant folding.
* **Flow**: Performs dataflow analysis on the trees from the previous step. This includes checks for assignments and reachability.
* **Desugar**: Rewrites the AST and translates away some syntactic sugar.
* **Generate**: Generates ‘.Class’ files.

Your Java code may use the code in the Java library. The JVM exe- cutes your code along with the code in the library.

To execute a Java program is to run the program’s bytecode. You can execute the bytecode on any platform with a JVM, which is an interpreter. It translates the individual instructions in the bytecode into the target machine language code one at a time rather than the whole pro- gram as a single unit. Each step is executed immediately after it is translated.