**Java Compilation Process**

There are two stages that need to be followed for a java file to run. The first stage is compilation and the second stage is execution.

In compilation stage, the ‘.java’ file is passed through a compiler. In the compiler, there are steps that need to be followed which is parse, enter, process annotations, attribute, flow, desugar and generate. These steps convert the source code to bytecode.

The next stage is the execution. After the source code (.java) is encoded to bytecode(.class), the class file goes through three stages; class loader, bytecode verifier, and just-in-time compiler in the JVM (Java Virtual Machine).

First, the class file is loaded into the memory which is called the class loader. After the bytecode of class is loaded by the class loader, the bytecode verifier checks the code to ensure that the instructions don’t perform damaging actions, such as errors and bugs. Then, loaded bytecode is converted to machine code with the help of JIT (Just-In-Time) Compiler and Interpreter. The JIT Compiler compile and remove repeated code that is in the loaded bytecode, after that the interpreter translates it one line at a time. Then, it will be transferred to the OS (Operating System) and hardware to execute the code.