Java Basics

Problems with languages before Java:-

1. FORTRAN :- Useful for writing efficient programs for scientific applications but not good for system code.
2. BASIC :- Easy to learn but it wasn't very powerful.
3. ASSEMBLY :- Can be used to build highly efficient Programs but it is not easy to learn or use effectively.
4. PASCAL :- Structured but not efficient.
5. C :- Structured, easy but not able to build large complex programs (as far as I remember, once a C program length exceeds 30k lines of code, it starts creating problem.

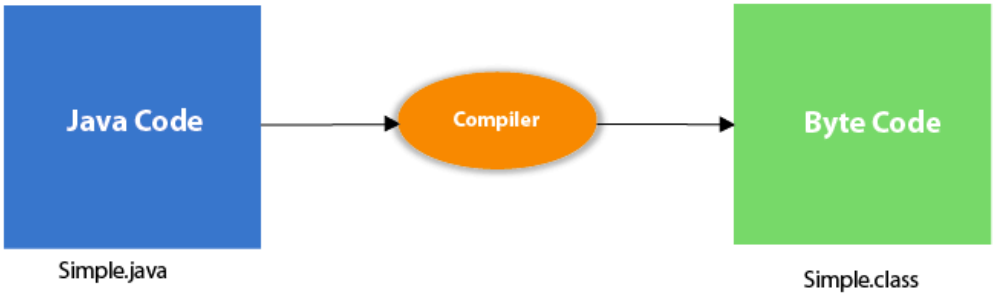
Requirement :-

Portable, platform-independent language than can be used to produce code that would run on a variety of CPU's under different environments & for Internet Programming as well.

Java Program Internals :-

Compilation Flow:

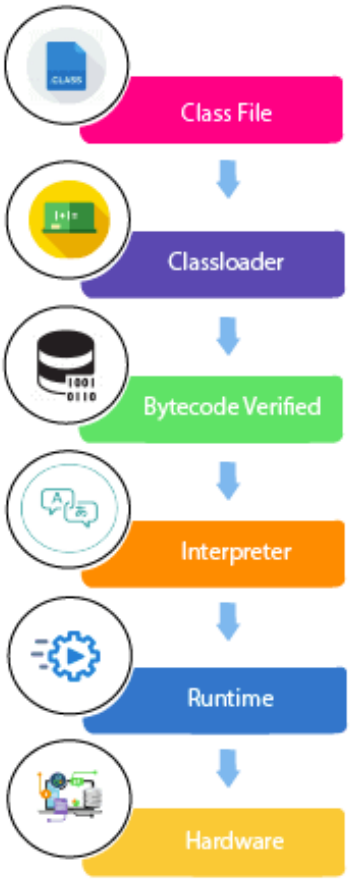
When we compile Java program using javac tool, the Java compiler converts the source code into byte code.



- Bytecode is a highly optimized set of instructions designed to be executed by the JVM
- Java Virtual Machine (JVM) is platform dependent.

What happens at runtime?

At runtime, the following steps are performed:

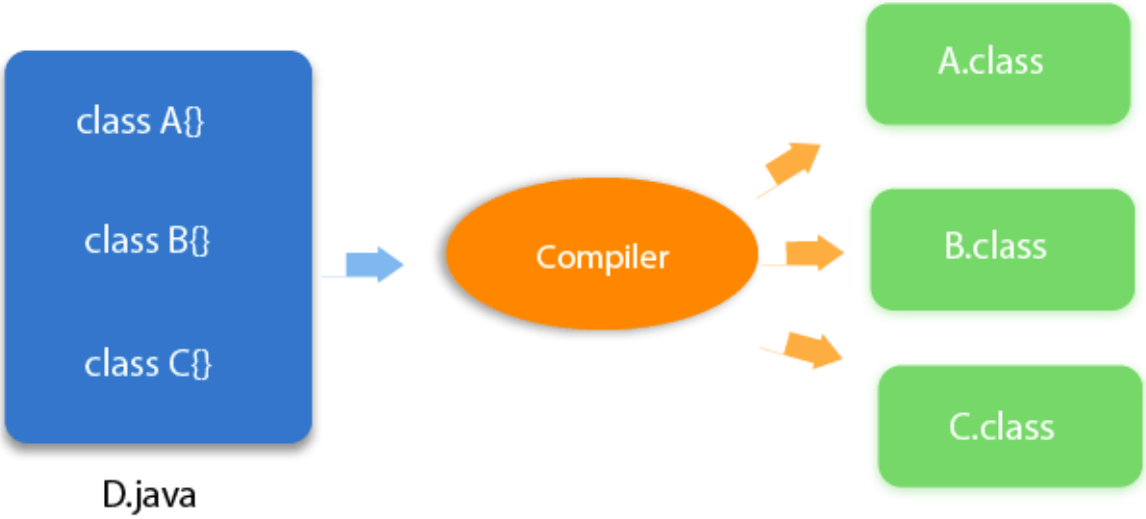


Classloader: It is the subsystem of JVM that is used to load class files.

Bytecode Verifier: Checks the code fragments for illegal code that can violate access rights to objects

Interpreter: Read bytecode stream then execute the instructions.

- Hotspot provides a Just-in-time (JIT) compiler for bytecode execution. When JIT is a part of JVM, selected portions of bytecode are compiled into executable code in real-time on a piece-by-piece demand basis. The remaining code is simply interpreted by JVM.



After source code compilation, each class will have its own .class file.

- Java is a strictly typed language: It checks your code at compile time as well as at run-time.
- Java provides garbage collection facility for unused object, thus memory management is done by Java itself.

Things to remember :-

- The true literal in Java does not equal to 1, similarly false is not equal to 0.
- In Java, we can not declare a variable to have the same name as one in the outer scope.
- Java automatically promotes each byte, short or char operand to int when evaluating an expression.

```
Ex:- byte b = 50;
     b = b * 2; //Error
```

- The elements in the array allocated by new will get automatically initialized to 0 (for numeric type), false (for boolean type) or Null (for Reference Type).
- When we do right shift, the top bits exposed by the right shift are filled with the previous content of the top bit.

```
Ex:- Initial bit representation:- 11010101
     right shifting by 3 (>>) :- 11111010
     Unsigned right shift by 3 (>>>) :- 00011010
```

- Switch runs faster than if-else-if blocks.

public static Void main() (String args[]) :-

- main represents the starting point of the program.
- main() must be declared public, since it must be called by the code outside of its class when the program is started.
- The keyword static allows main() to be called without having to instantiate a particular instance of the class. This is necessary since main() is called by JVM before any objects are made.
- Java compiler will also compile classes that do not contain main() but java has no way to run these classes.
- String args() is used to pass command line arguments.

System.out.println() :-

- System is a predefined class that provides access to the system.
- Out is an object of PrintStream class
- println() is the method of PrintStream class.

Thus, all Java applications begin execution by calling main()