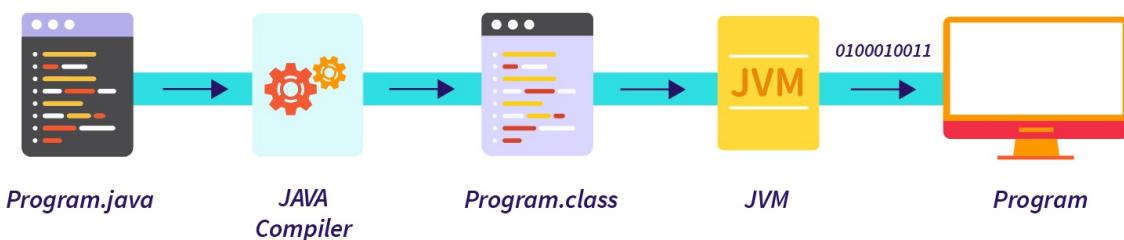


How Java Program Works

The execution of a Java program consists of five steps which are

1. Creation of a Java Program
2. Compiling a Java program,
3. Loading the program into the memory by Java Virtual Machine (JVM),
4. Java Virtual Machine verification for bytecode,
5. Java Program execution

How Does Java Programming Language Work?

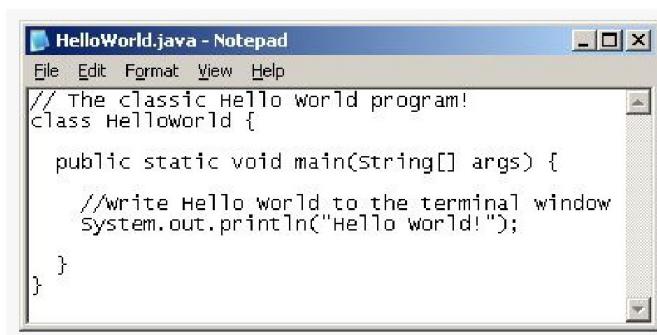


1. First we need to have a java source code, we need to save it with the *program.java* extension.

2. Secondly, we need to use a compiler so that it compiles the source code which in turn gives out the **java bytecode** and that needs to have a **program.class** extension.(Java bytecode is a redesigned version of the java source codes), and this bytecode can be run anywhere irrespective of the machine on which it has been built.
3. Java Virtual Machine which is an interpreter that reads all the statements thoroughly step by step from the java bytecode which will further convert it to the machine-level language so that the machine can execute the code.

Execution Process of Java Program

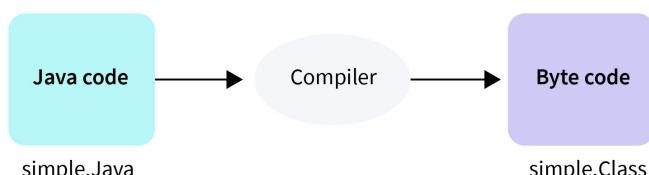
1.



```
// The classic Hello world program!
class HelloWorld {
    public static void main(String[] args) {
        //write Hello world to the terminal window
        System.out.println("Hello World!");
    }
}
```

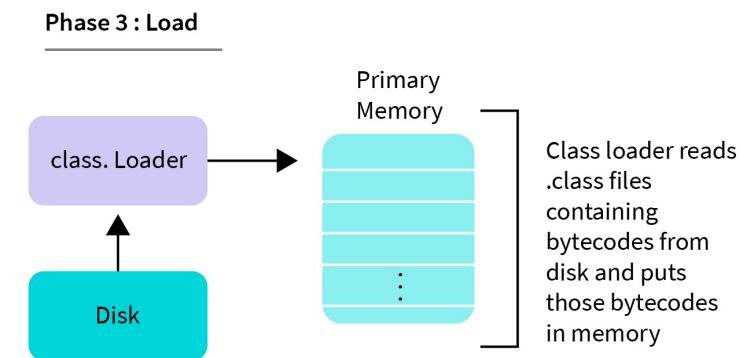
Creation of a Java Program

2.



Compiling a Java Program

3.



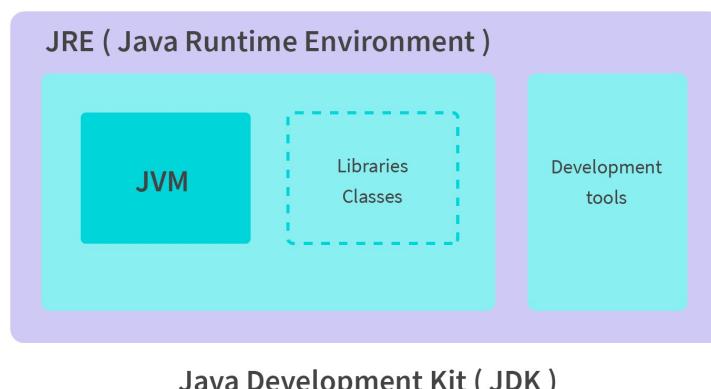
Loading the Program into the Memory by Java Virtual Machine

4. Java Virtual Machine verification for bytecode

5. Java Program Execution

The tools that come in handy when we code in the java programming language.

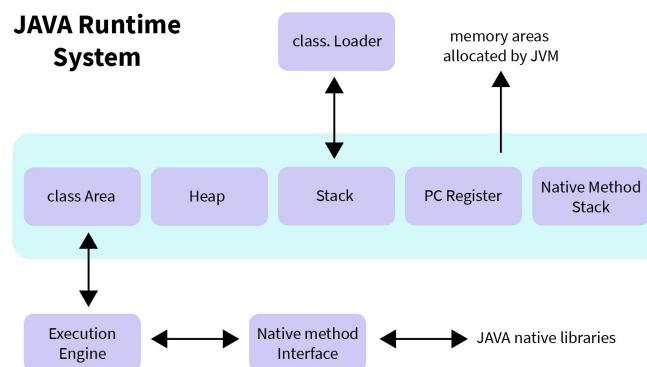
1. Java Development Kit : Java development kit is a full-time kit that has a **compiler, Java Runtime Environment(JRE), Debuggers, Java documents** inclusive in it. For further execution in java, we need to have JDK installed on our computers to further lead on to the creation, compilation, and running of the java program. Here, as we use JDK we need an environment to run the programs. We use JRE Java Runtime Environment which provides the least requirements to execute the java program. It provides the JVM, Core classes and supporting files.



Java Development Kit (JDK)

2. Java Virtual Machine : When we run a program using the JRE and JDK it also goes to the JVM as it is required to run the java program and it also interprets the program. The phases are as follows: **Compile the Code:** The Java Development Kit(JDK) provides us

with the JAVAC compiler to get through with this step. **Run the Code:** JVM runs the bytecode that is provided by the compiler.



3. Java Runtime Environment: JDK includes JRE.

- Java virtual machine (JVM)
- Java class libraries
- Java class loader when our software tends to execute a particular program, it requires some environment to run in. Here our JRE acts as a translator and also a facilitator between the java program and the operating system.

Is Java a Compiler or Interpreter Based Language?

- When we talk about compilers we mean that- A tool that converts source code into machine language making it easier for the computer to interpret it is called a compiler.
- Let's have a look at the Java interpreter- As a compiler, an interpreter is also a programming tool that converts our source code into readable(for computers) machine code but it doesn't work in the same way that the compiler does. It converts each statement going through it thoroughly before moving on to the next one and then executes it. No executor is needed for an interpreter.

Hence we can conclude that we can consider java to be both **compiler and interpreter** based language because the source code first gets compiled into binary code and then the same binary code runs on the JVM (Java Virtual Machine), and this is mostly a software-based interpreter

Summary

- During programming, there are 5 steps that we undergo to execute our programs which are

- Creation of a Java program
 - Compiling a Java program
 - Loading the program into the memory by Java Virtual Machine
 - Java Virtual Machine verification for bytecode,
 - Java Program execution.
- There are 4 tools(JDK, JVM, JRE, JIT) that help us with the smooth working of our programs.

▼ Understand Byte code

Bytecode is an intermediate code representation that is generated by a compiler or an interpreter and is designed to be executed by a virtual machine. In the context of Java, when you compile a **Java source code file, it is translated into bytecode rather than native machine code.**