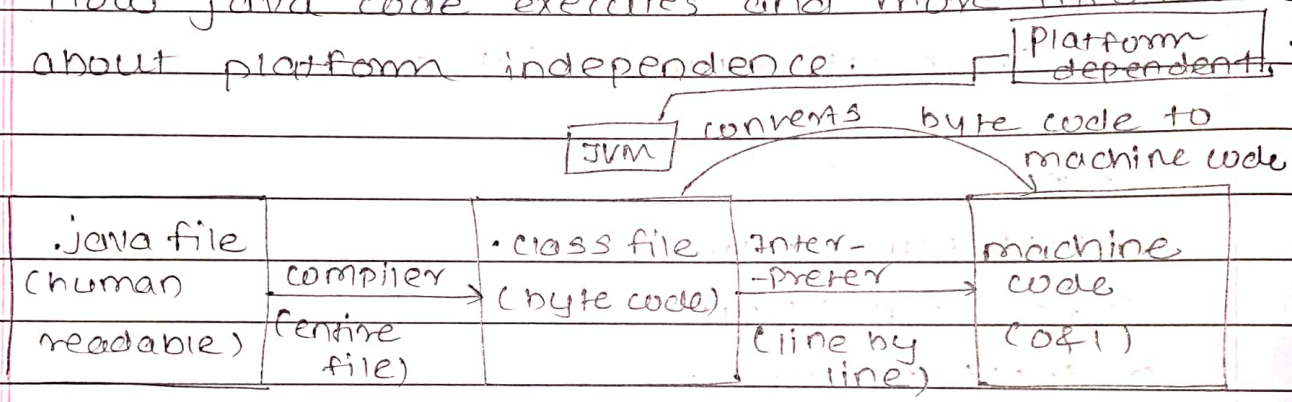
Introduction To Java

* why do we use programming language?
- machine only understand 0's & 1's for humans it is very difficult to instruct computer in 0's and 1's so to avoid this issue we write code in human readable language i.e. (programming language).

* "Java is one of the programming language".

* How java code executes and more information about platform independence.



1] - code written in java is human readable & saved using extension .java → called source code.

2] Byte code :-
- It can run on all OS.
- This code doesn't run directly on a system, for this we need JVM {JVM is platform dependent}
→ * Therefore, java is platform independent. *

→ we can provide this byte code to any system means, we can compile the java code on any os.
→ But JVM is platform dependent means for every os · the executable file that we get, it

has step by step set of instruction dependent on platform.

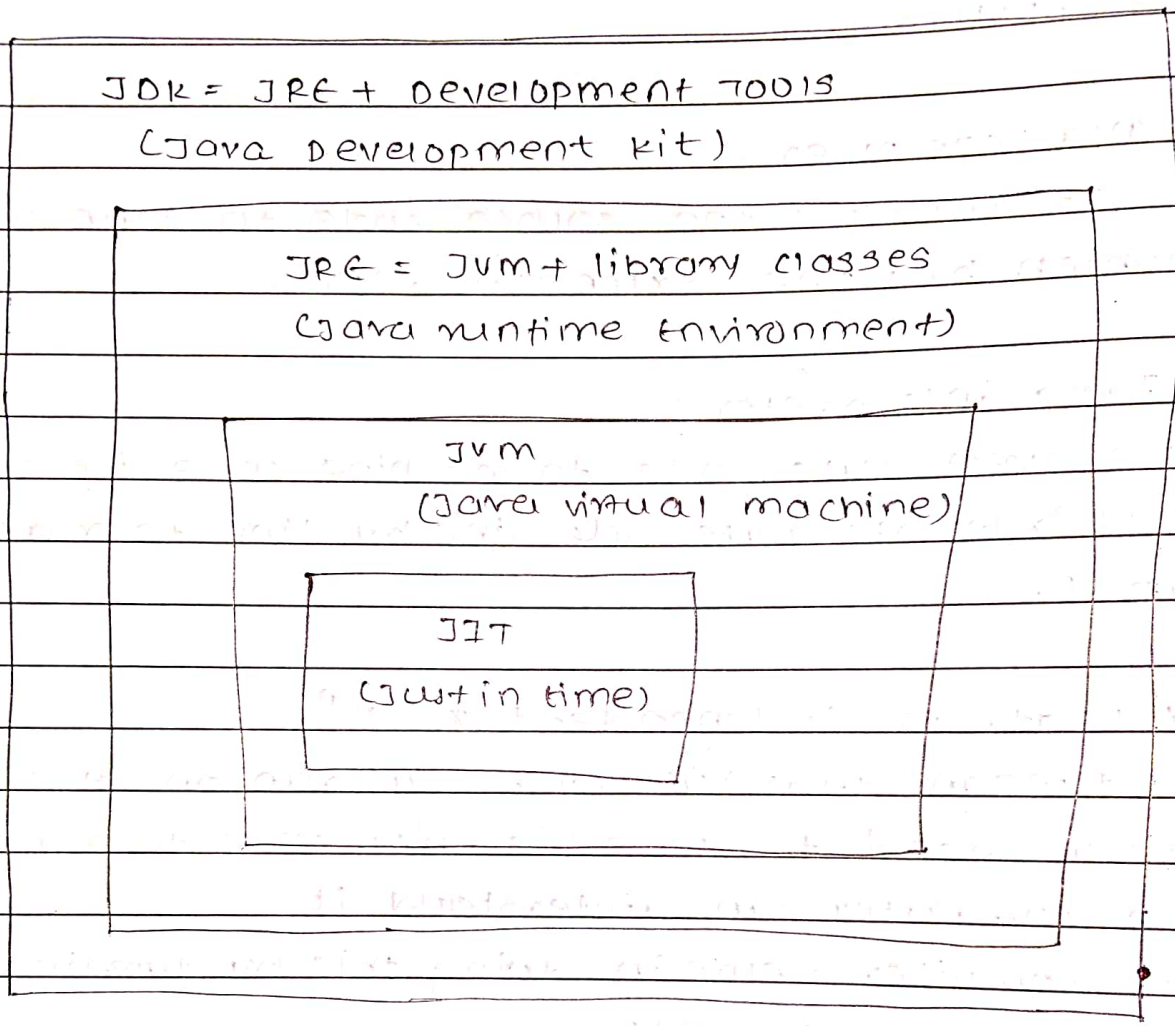
2] Java compiler :-
- It converts the source code to byte code which has the extension .class

4] Java interpreter :-
- convert byte code to machine code i.e. 0's & 1's
- It translate byte code line by line to machine code.

* platform independent * Imp
→ It means that byte code can run on all OS.
→ * we need to convert source code to machine so computer can understand it.
→ • compiler helps in doing this by turning it into executable code.
→ This executable code is a set of instruction for computer.
→ After compiling C/C++ code we get .exe file which is platform dependent.

→ In Java we get byte code, JVM converts (byte code) to machine code.
→ Java is platform independent, but JVM is platform dependent.

* Architecture of Java :



1) JDK :

→ It provide environment to develop and run the java program.

→ It is a package that includes :

- development tools : To provide an environment to run your program.
- JRE : To execute your program
- A compiler : Javac
- Archiver : Jar
- Docs generator : Javadoc
- Interpreter / loader

2] JRE :

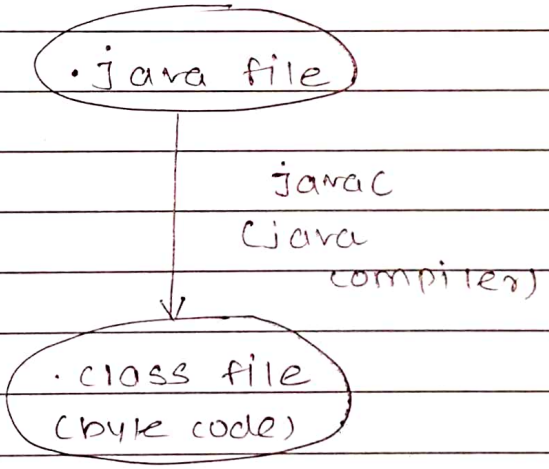
→ It is an installation package that provides environment to only run the program.

→ It consist of :-

- Deployment technology
- User interface toolkit
- Integration libraries
- Base libraries
- JVM : Java virtual machine.

* Java compile time & Run time Environment :

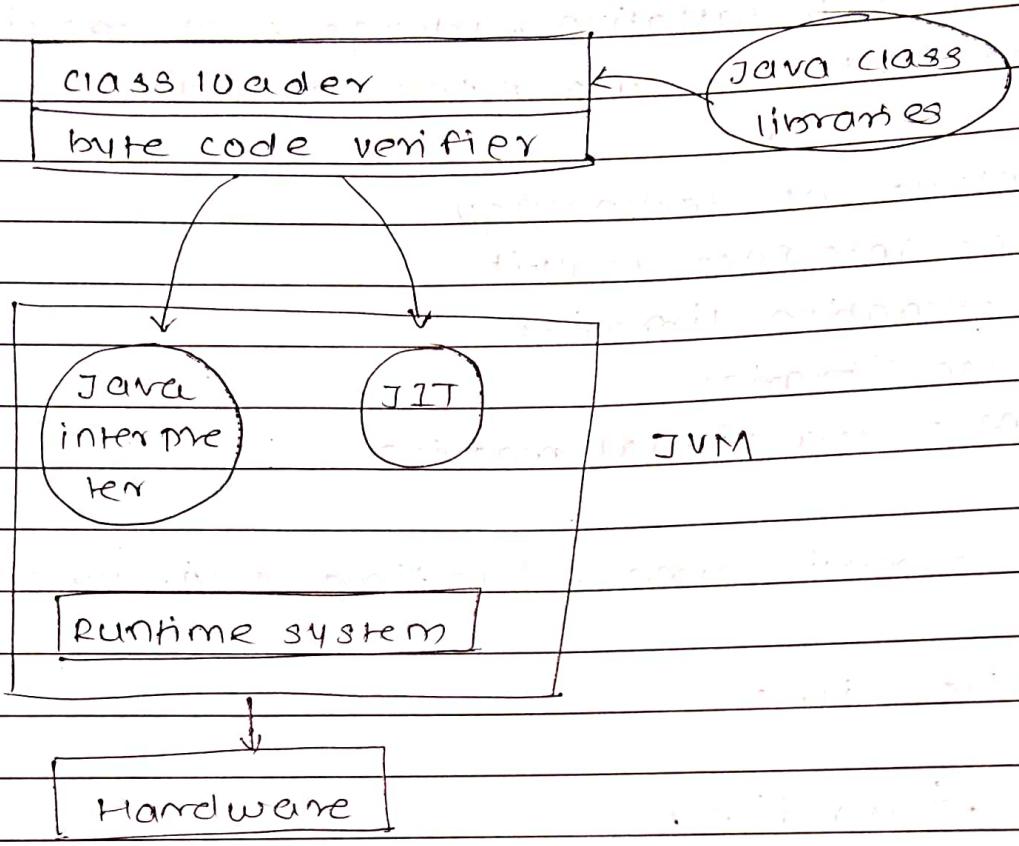
compile time



→ After we get .class file next thing happen at runtime.

1) Class loader loads all classes needed to execute the program.

2) JVM sends code to bytecode verifier to check format of code.



* How JVM works?

⇒ class loader

1] Loading :
- Read .class file & generate binary data.
- an object of this class is created in heap memory.

2] Linking :
- JVM verifies the .class file
- allocates memory for class variables & default values.

- replace symbolic references from the type with direct reference.

3) Initialization:
- All static variables are assigned with their values defined in the code & static block.
- JVM contains the stack & heap memory locations.

⇒ JVM Execution
1) Interpreter
- Line by line execution
- when one method is called many times it will interpret again and again.

2) JIT
- Those methods that are repeated, JIT provides direct machine code so that interpretation is not required.
- makes execution faster.

3) Garbage collector.