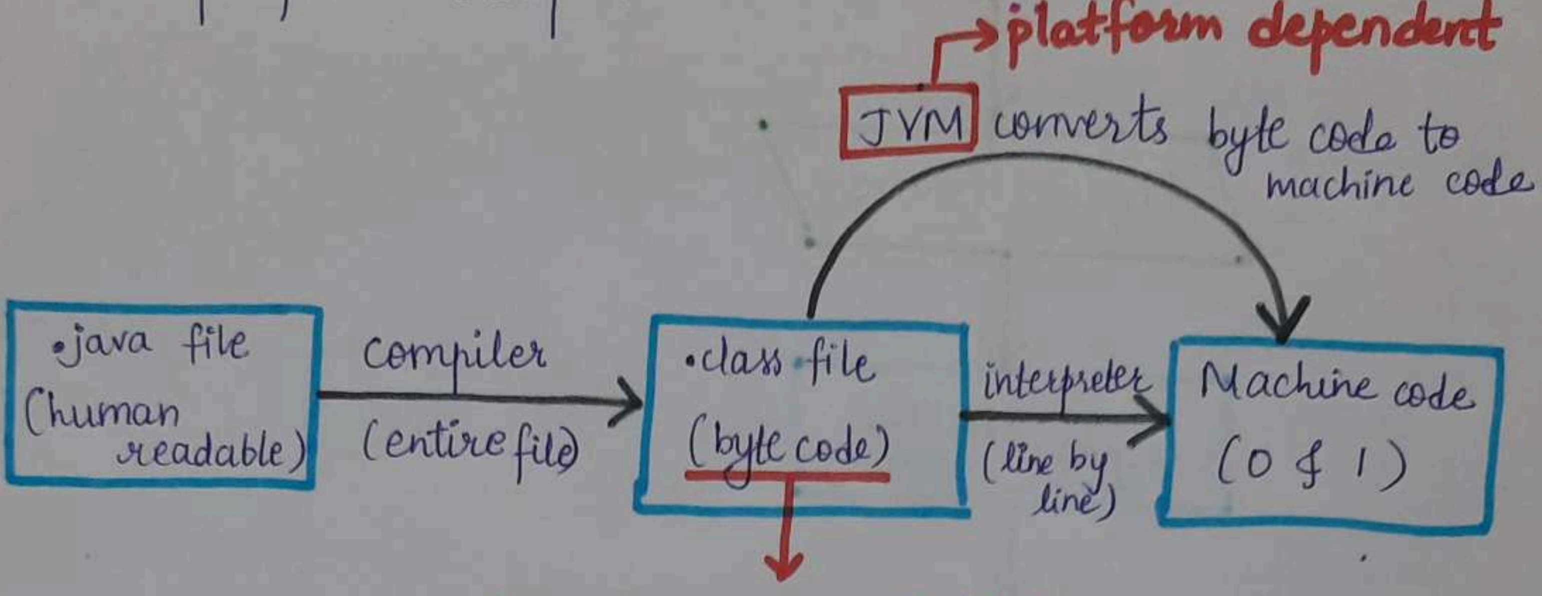
3/8/21

Introduction to Java ♥

★ How Java code executes and more information about platform Independence...



platform dependent
JVM converts byte code to machine code
.java file (human readable)
compiler (entire file)
.class file (byte code)
interpreter (line by line)
Machine code (0 & 1)

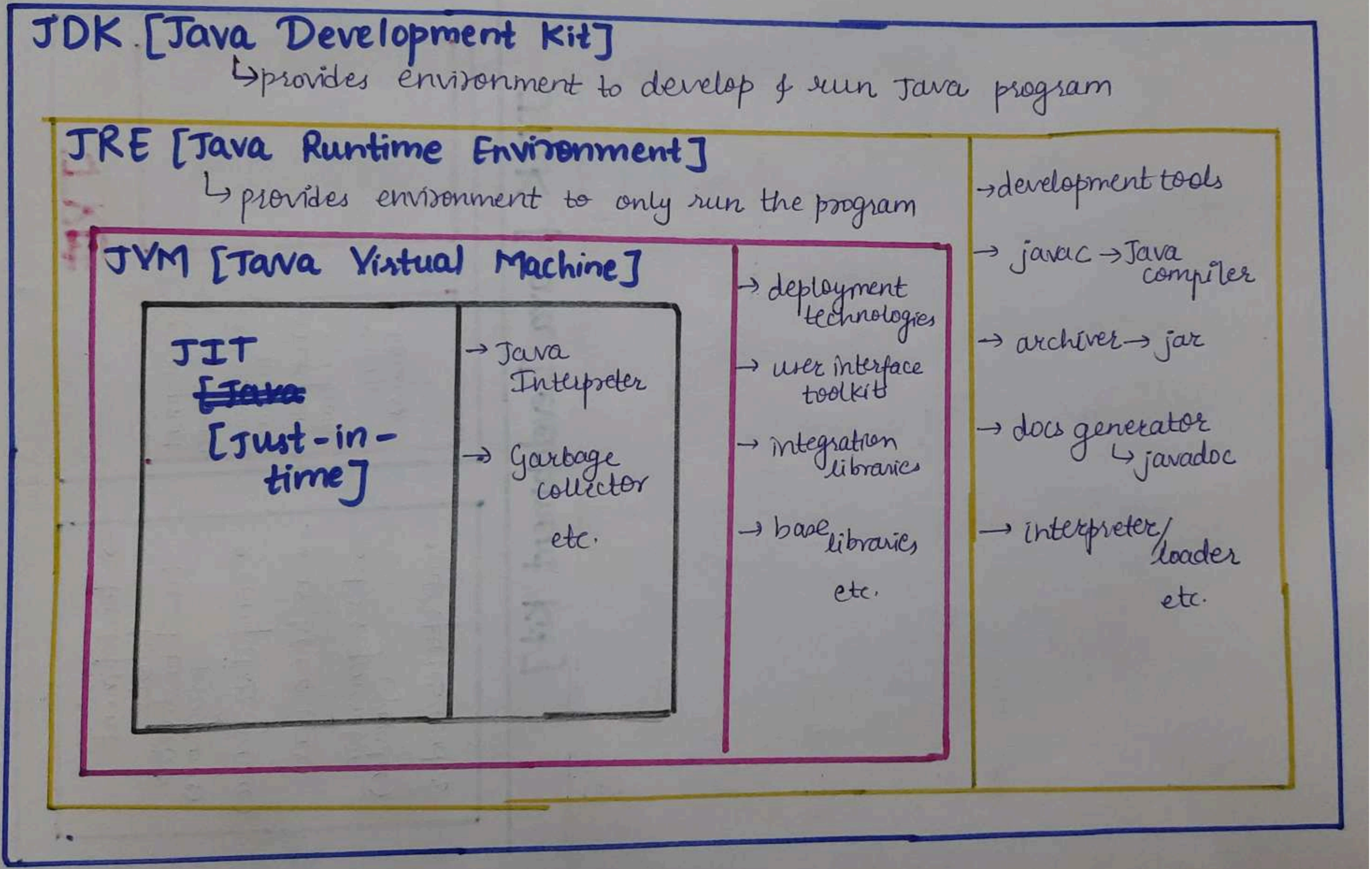
- can run on all O.S.
- this code doesn't run directly on a system, for this we need JVM

★ Therefore, Java is platform independent ★

⟹ We can provide this byte code to any system means we can compile the java code on any system.

⟹ But JVM is platform dependent means for every O.S. the executable file that we get, it has step by step set of instruction dependent on platform.

✯ JDK vs JRE vs JVM vs JIT



JDK [Java Development Kit]

↳ provides environment to develop & run Java program

- → development tools
- → javac → Java compiler
- → archiver → jar
- → docs generator ↳ javadoc
- → interpreter/loader
- etc.

JRE [Java Runtime Environment]

↳ provides environment to only run the program

- → deployment technologies
- → user interface toolkit
- → integration libraries
- → base libraries
- etc.

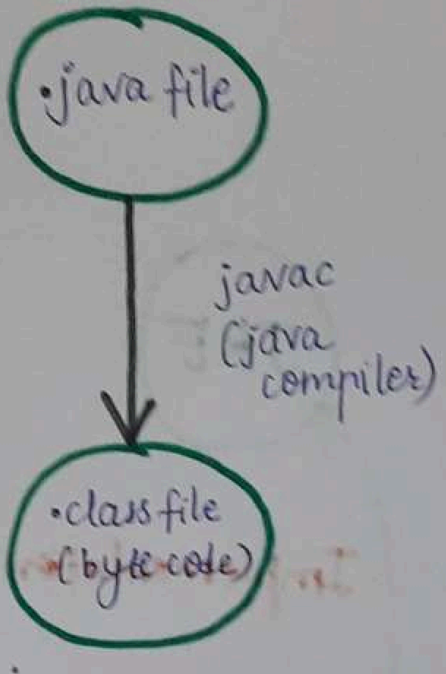
JVM [Java Virtual Machine]

- → Java Interpreter
- → Garbage collector
- etc.

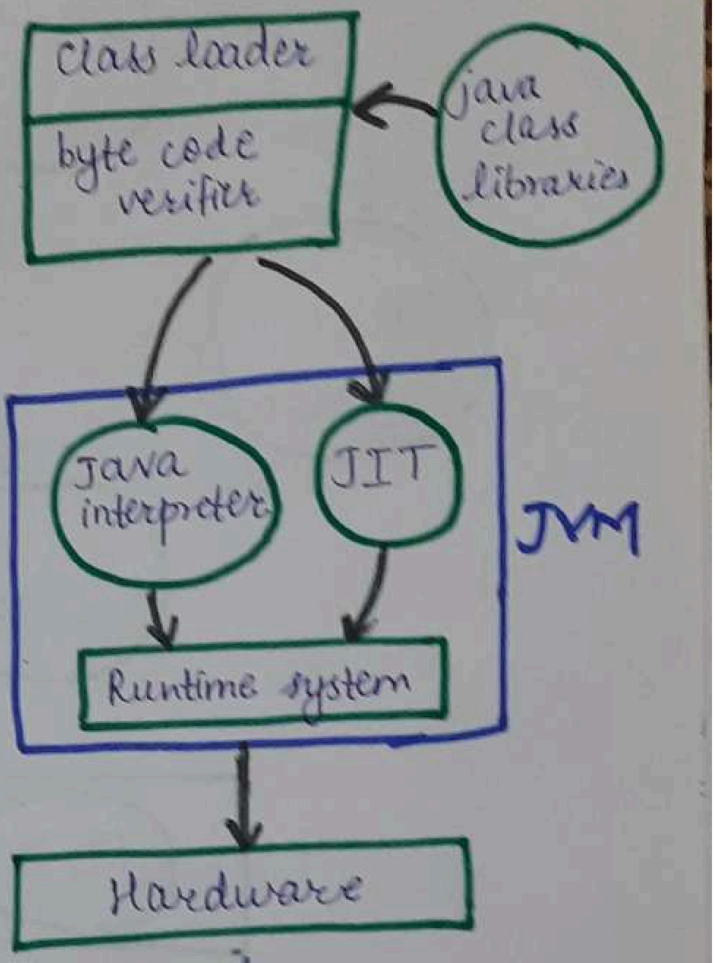
JIT ~~[Java~~ [Just-in-time]

✱ Java Development and Runtime Environment

Compile time



Runtime



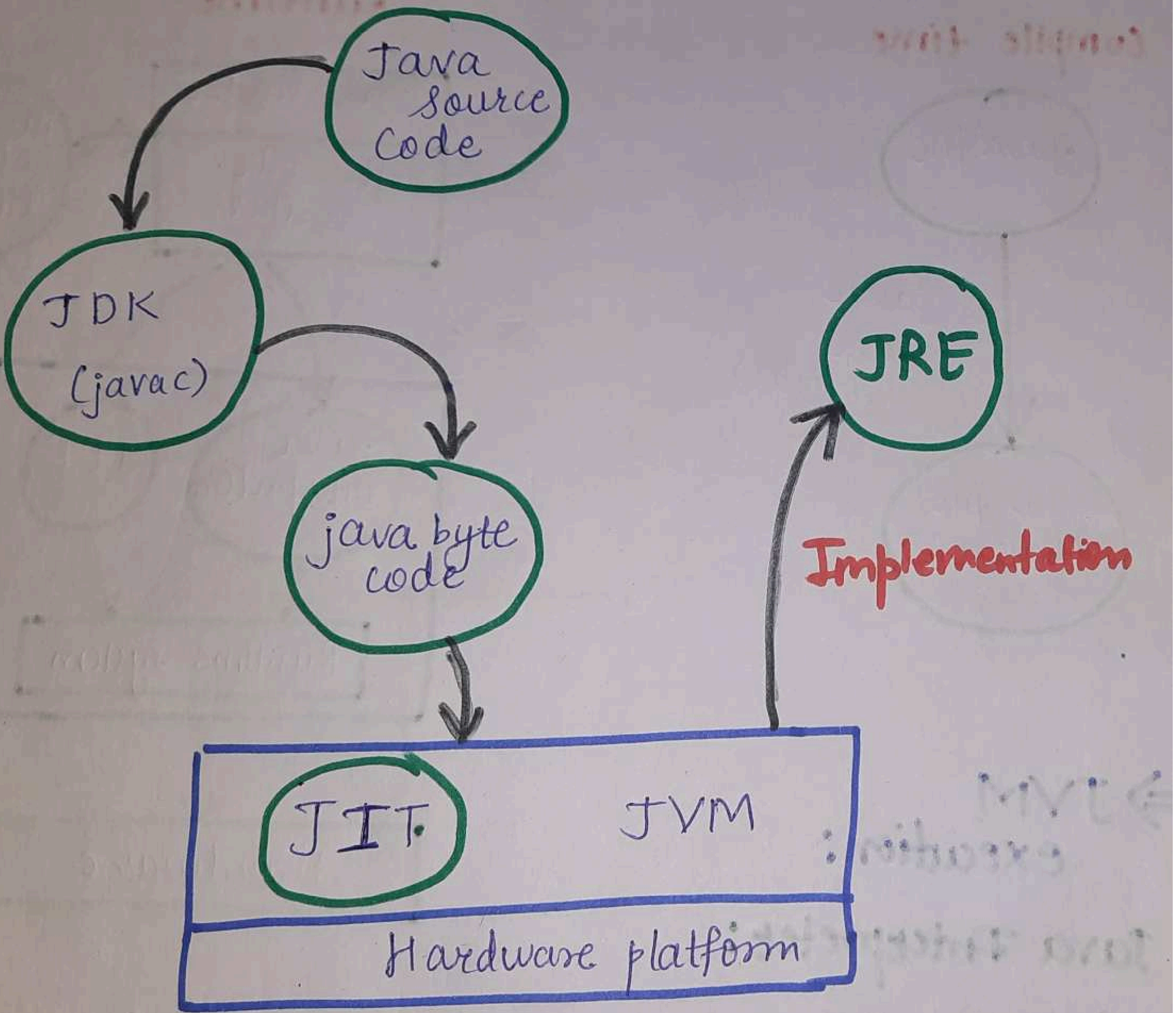
⇒ JVM execution:

- **Java Interpreter:**
  - → line by line execution
  - → when one method is called many times, it will interpret again & again
- **JIT:**
  - → method that are repeated, JIT provide direct machine code so re-interpretation is not required
  - → makes execution faster
- **Garbage Collector**

✱ Class loader:

- **Loading**
  - → reads byte code file & generates binary data
  - → an object of this class is created in heap
- **Linking**
  - → JVM verifies .class file
  - → allocates memory for class variables & default values
  - → replace symbolic references from the type with direct references
- **Initialization**
  - → all static variables are assigned with their values defined in the code & static block.

✯ Summary:


Java source code
JDK (javac)
java byte code
JIT
JVM
Hardware platform
JRE
Implementation