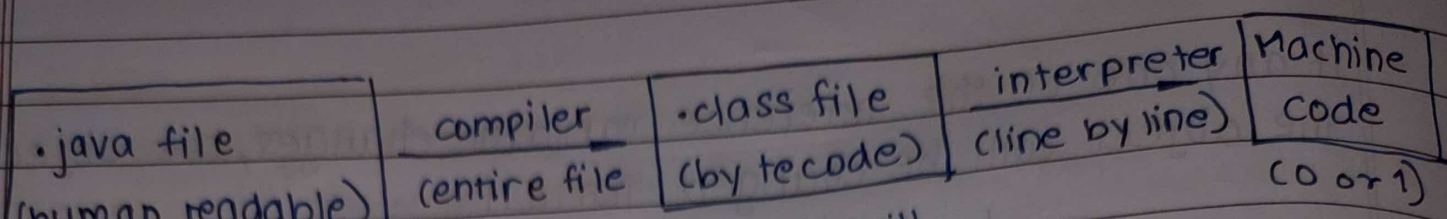
* LECTURE 3:- Introduction to Java.



this is the source code

- this code will not directly run on a system
- we need JVM TO run this
- Reason why JAVA is platform indep.

- PLATFORM INDEPENDENCE

- It means that byte code can run on all operating systems
- JVM converts this to machine code.
- Java is platform-independent but JVM is platform dependent
- C|C++ we get .exe which is platform dependent.

COMPILE TIME

JAVA FILE

javac (compilation)

.class file

RUNTIME

class Loader

Byte code verifier

Interpreter

Runtime

Hardware

JVM Execution

Interpreter :-

- Line by line executions
- when one method is called many times, it will interpret again

JIT

- those methods that are repeated, JIT provides direct machine code so reinterpretation is not required.
- makes execution faster
- Garbage collector

(How JVM works) class Loader

- LOADING
  - reads .class file and generate binary data
  - an object of this class is created in loop heap.
- LINKING
  - JVM verifies the .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 and static block

JVM contains the stack & Heap memory allocation