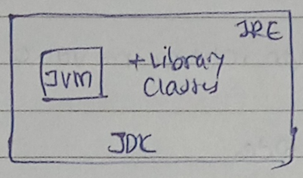
Assignment No. 01

M T W T F S S
Page No.:
Date:
YOUVA

1. What is JDK?

i) It is distribution of java technology of oracle operation.
ii) Provides collection of tools & libraries
iii) JDK along with JVM & JRE is used as core package in java



components of JDK

1. JRE : Java runtime environment
consists of JVM, library classes & JDK

2. Interpreter/loader (Java)
A computer program that converts high level program statement into Assembly-level language.

3. Compiler :- reads source file written in java programming language convert it into machine code & then that file is then executed.

4. An Archiever :- A JAR (Java Archieve) is a package file used to aggregate java class files, metadata & resource into one file

2. Comparison between

	JDK	JVM	JRE
1)	Java development kit	Java ~~runtime~~ ~~environment~~ virtual machine	Java runtime environment
2)	It develops application in java Along with JRE, JDK	platform indepent machine. This describes JVM implementation	It is software package that provide class libraries of java & JVM & other

3) Role of JVM in java? How JVM executes java code?

→ i) JVM is java virtual machine
ii) JVM starts execution of byte code which it gets from compiler
iii) JVM starts execution of code from main method in the program.
iv) In void main method JVM returns no value to the JVM in machine code or byte code.

JVM role:- i) verifies code
ii) Executes code
iii) provides runtime environment.

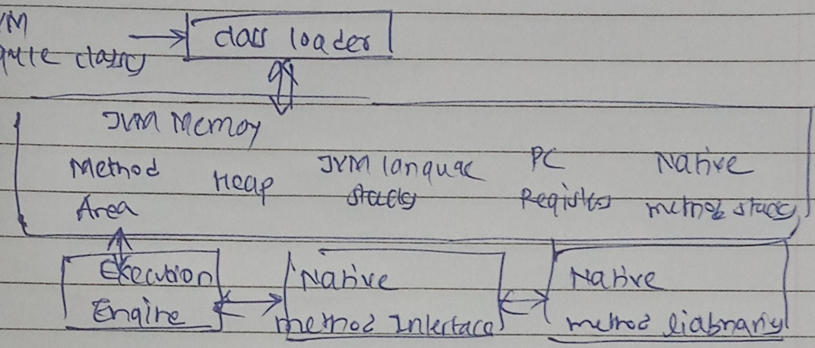
4) memory management systems of Java?
i) JVM manages memory in java
ii) Java is block structure in which devices memory in stack & heap
iii) When method is entered or exited request is made to the JVM for an amount of memory, which size is known at runtime while creating object

5) JIT compiler & role in JVM, bytecode & role
JIT compiler is enabled by default, JIT compiler requires processor time & memory usage. JVM starts up thousands of methods are called. Compiling all of these can significantly startup time even if program eventually achieves good peak.
bytecode it is the code in the form of one's & zero generated by compiles in java. JVM only understand machine code for execution.

6. Architecture of JVM

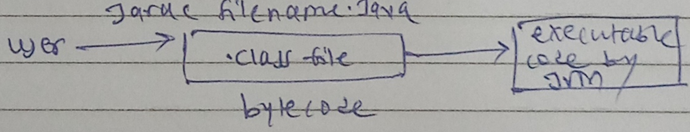
M T W T F S S
Page No.:
Date:
YOUVA

6. Architecture of JVM



7) How JVM is platform independence

→ i) In java compiler creates byte code from the source code
ii) which .class file by java compiler
iii) Then class loader subsystem loads all reference classes into memory
iv) In execution, bytecode instructions are executed one by one
v) execution by JVM & execution engine with JIT (Just in time compiler
vi) In Runtime JVM manages memory, thread execution & other runtime aspects
vii) as bytecode is independent of platform i.e. .class file
viii)



8) Significance of class loader & garbage collection
class loading is the process of loading reference classes into the memory. This done by JVM.
There are, 1) Bootstrap classloader
2) Extension classloader
3) System class loader

garbage collection is the automated process of deleting code in java.