Assignment no. 2.

Q.1] what is the diffrence between JDK, JRE & JVM?

	JDK	JRE	JVM
1)	JDK stands for: Java Development kit.	JRE Stands for Java Runtime Environment	JVM = Java virtual machine
2)	It is often called as superset of JRE	It is a set of soft-ware tools.	JVM loads, verify & executes Java bytecode
3)	It is the foundational component that enables Java Application	It uses heap space for dynamic memory allocation for java objects	It is known as interpreter.
4)	JDK contains all the tools required to compile, debug & run a program developed using java platform	JRE is composed of a variety of other supporting software tools & features to get most of java app"	It is specially responsible for converting bytecode to machine specific code & is necessary in both JDK & JRE.

Q.2] What is JIT compiler?

→ 1. JIT is a integral part of JVM (Just in Time).
2. It is a long-running, computer-intensive program that provides the best environment performance.
3. It optimizes the performance of java application at compile or runtime.

* Advantages:-

1. It requires less memory usages.
2. The code optimization is done at run time.
3. It uses diffrent level of optimization.
4. It reduces the page fault.

* Disadvantages:-

1. It increases complexity of program.
2. Program with less line of code dose not take the benifit of JIT compiler.
3. It uses a lots of CACHE Memory

Q.3] what is class loader ?

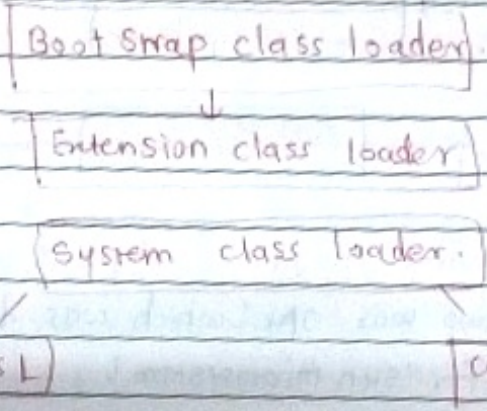
→ 1. Java Class loader is an abstract class
2. It belong to a Java.lang package.
3. It is used in loading classes at runtime.
4. Java class loader is based on 3 principles.

a) Delegation :- It forwards the request for class loading to parent class loader.

b) Visibility :- It allows child class loader, to see all the classes loaded by parent classloader, but parent class loader cannot see classes loaded by child class loader.

c) Uniqueness :- It allows to load a class once it is achieved by delegation principle. It ensures that child classloader dosent reload the class which already loaded by the parent.

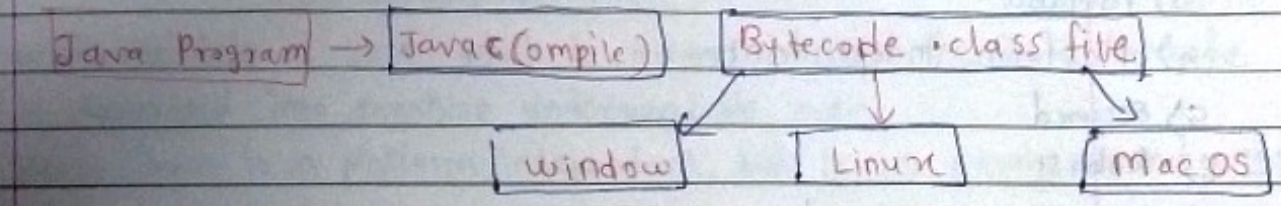
Types of class loader :-



Hierachry of class loader

Q.4] What gives Java its "Write Once & Run anywhere' nature ?

→ • Java applications are called WORA i.e. (write one Run Anywhere)
• This means programmer can develop java code on one system & can expect it to run on any other Java enabled platform
• This is all possible because of JVM.

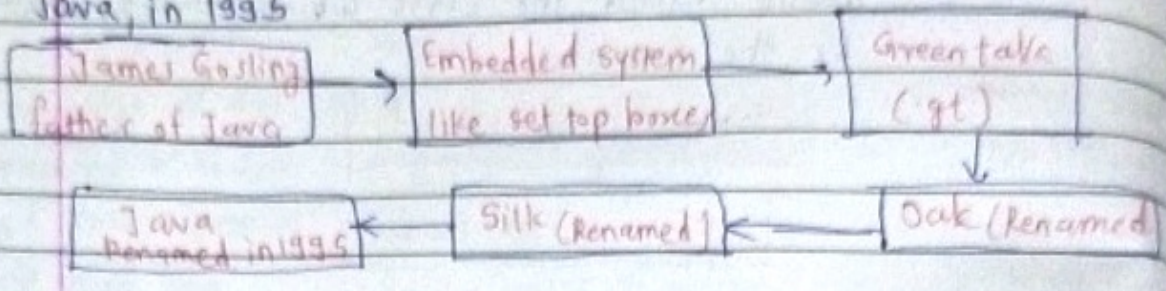


• In java the program is not converted to code directly understood by hardware rather it is converted to bytecode (.class file) which interpreted by JVM so once compiled it generates byte file. which can be run anywhere (any machine) which has JVM.

& hence it gets the nature of write once & Run Anywhere!

Q.5] Explain History of Java & who invented Java?

→ 1) History of Java starts with Green team.
2) The principles for creating Java programming were "simple, Robust, Portable, platform - independant, secured, High performance etc.
3.) Java is used in Internet programming, mobile devices, games, e-business solutions etc.
4) James Gosling / mike sheridan, Patrick Naughton initiated Java language project in June 1991 this small team of Sun engineers called "Green Team".
5) Java was developed by "James Gosling" who is known as father of Java, in 1995.



Q6] What was the original name of Java & why it was renamed?

→ 1) The original name of Java was 'OAK' which was developed by small team of engineer working for 'Sun microsystem'.
2.) They called themselves the 'Green team'.
3) 'Oak' name was renamed due to the fact that oak was already registered as part of another trademark.

Q7] List the features of Java?

→ 1) Simple
2) object - oriented
3) Portable.
4) Platform independant.
5) secured.
6) Robust.
7) Architecture neutral.
8) Interpreted.
9) High performance.
10) Multi thread.
11) Distributed.
12) dynamic.

Q8] ... in Java?

Datatypes

- Premitive
  - Boolean
  - Numeric
    - Integer
      - char (2 bytes)
      - Integer
        - Byte - 1 byte
        - Short - 2 byte
        - Integer - 4 bytes
        - long - 8 bytes
    - floating
      - float (4 bytes)
      - double (8 bytes)
- Non-Premitive (user defined)
  - class
  - Interface
  - Arrays
  - String

Q9] What is the difference between System.out.print(); & System.out.println(); System.out.err.print();

→ •) System.out.print(); ;→ Cannal or cursor remains on the same line.

•) System.out.println(); Cannal or cursor moves to the next line. after printing.

•) System.err.print(); used to display error mesrage., The output is displayed in red color.

Q10] How is Java platform independant?

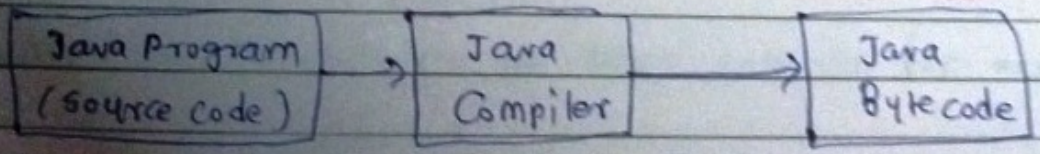
→ • When you compile Java program using javac file. compiler it generates bytecode.

• we can execute this code using bytecode in any other platform with the help of JVM which is present in JDK. which had JDK. installed i.e. Java Development kit.

• with the help of JVM which is present in JDK the Java bytecode is translated into machine understandable code.

• Hence Java is a platform independant but it is purely depend on JDK

Q11] What is bytecode? How it is diffrent from machine code?



1) Bytecode is a set of command that ... for software transition operation.
2) Commonly known as 'p-code' due to portability that it provides.
3) It is a intermidiate code compiled into a low-level code from the source code for efficient execution by software interpreter.

Bytecode :-

1) It is an intermediate code designed to run on virtual machine instead of central processing unit (CPU).

2) ~~Machine code is language which all machines~~ Programs

2) The function of bytecode is to be format that can be executed efficiently by virtual machines interpreter.

3) It is platform independant because it can be executed on any platform using virtual machine

machine code

1) It is a computer program made up of native instructions associated with particular computer.

2) machine code is the language which all programs must be converted into before they can be run.

3) It is not platform independant means it cannot be run on just any platform with same operating system.

Q.12] Explain various memory logical Partitions?

1) A logical partician is the division of a computers memory, & storage into multiple sets of resourses so that each ~~con~~ set of resources can be operated independantly with its own operating system instance & applications.

2) Number of logical partitions are used for diffrent purposes such as a database operative or client/server operation or seprate test of production environment.

3) Each partitions can communicate with other partitions as if other partitions is in a seprate machine.

Q.13] What is the difference between Jar file & Runnable jar file?

	Jar file	Runnable Jar file
1)	Jar file is a java appln which requires a command line to run. a runnable JAR file can be directly executed by double clicking.	1) Runnable jar file allows a user to run java classes without having to know class names and type them in command prompt, rather the user can just double click on the jar file & the program will fire up.
2)	A jar (Java Archive) is a package file format typically used to aggregate many java class files associated metadata & resources into one file to distribute application software or libraries on the java platform.	2) A runnable jar file allows java classes to be loaded just like when a user clicks on exe. file.

Q.14] What is the diffrent bet^n Runnable jar file & exe. file?

	Runnable Jar file	exe. file
1)	Jar file are like dead body.	1) exe. file are like living man.
2)	Jar file is a combination of compiled java classes.	2) Executable jar file is also combination of compiled java classes with main class.

Q.15 How is 'C' platform dependant Language?

→ 1.) C is a portable programing language because it is not tied to any hardware or system.

2) We can say, it is a hardware independant language or platform independant language.

3) That is why C is called 'portable language'.

4) 'C' programs does not depend on actually but the executable file that is generated at the end for running the 'C' program. many depend on platform.

5) When you use O.S. you get other extension for executable files.

Q16] What is difference between path & class path?

→

Path	class path.
1) Path variable is used to set the path for all Java software tools like javac.exe, java.exe, javadoc.exe, & so on.	1) classpath variable is used to set the path for java classes
2) Variable name :- PATH variable value :- C\program files\Java\JDK1.7.0_21\bin;	2) Variable name: classpath Variable value: C\program files\Java\jre.1.6.0\jre\lib\rt.jar.