Assignment - 02

M T W T F S S
Page No.:
Date: 04 |09| 24
YOUVA

1) Expai the components of JDK.

→ JDK has a private virtual machine (JVM) and a few other resources necessary for the devlopment of a Java Application:

JDK contains:

- Java Runtime Enviornment (JRE)
- A interpreter / loader (Java)
- A compiler (javac)
- An archiver (jar) and many more.

Following are the components of JDK.

Component	Use
javac	The compiler converts sourcecode into Java bytecode
java	the loader of java app
javap	The class file disassembler
javadoc	Documentation genrator
jar	Java archiver help manage JAR files.
appletviewer	Debuging of java applets without a web browser
xjc	Accepts an XML schema & genrates java classes
apt	annotation processing tool.
jdb	Debugger

2) Diffrence between JDK, JVM, JRE?

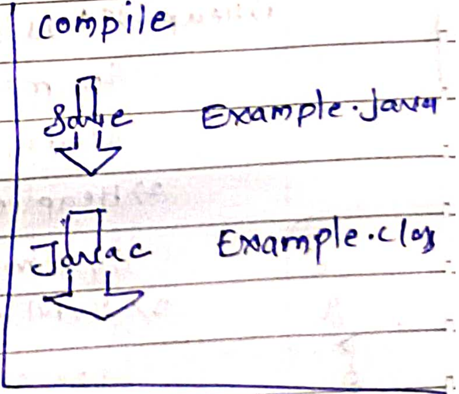
→ 1. JDK: Java Devlopment kit is a kit that provids the enviornment to devlop and execute (run) the java program. JDK is a kit (or package) that includes two things
- Devlopment: Tools (provide an enviornment to devlop your java program)
- JRE (to execute your java program

2. JRE: (Runtime Enviornment) is an installation package that provides an enviornment to only run (not devlop) the (not to devlop) the Java program or application onto your machine. JRE is only used by those who only want to run java program that are end-users of your system.

3. JVM (Java virtual machine): Important part of JDK & JRE because it is ~~[illegible]~~ contained or inbuild in both. what ever java program you run using JRE & JDK goes into JVM & JVM is responsible for executing the java program line by line, hence it is interpeter

3) What is role of JVM in Java? How does JVM execute the Java code?

→ > JVM is the componet of JRE
> JVM becomes an instance of JRE at the runtime of a Java program.
> It is widely known as runtime interpreter.
> Main responsible 3 activities.
- Loading.
- Linking.
- Initialization.



> Following actions occurs at runtime as listed below.
- class loader
- Byte code verifer
- Interpreter
  > Execute the byte code
  > Make appropriat calls to the underlyning harding.

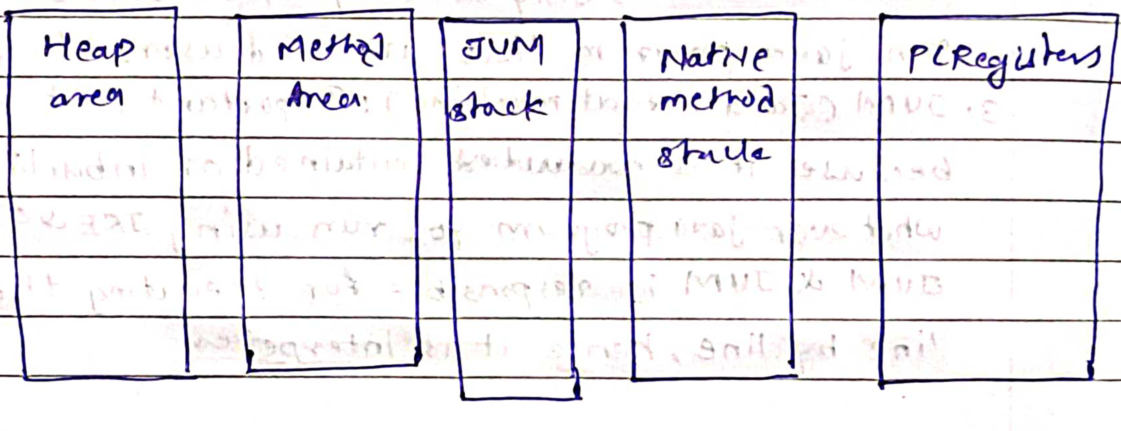
4> Explain the memory management system of the JVM.

→ > Two major concepts in Java memory Management:

- JVM Memory structure.
- Working of Garbage collector.

1> JVM memory structure:

JVM defines various runtime data used during execution of program. Some of the JVM whereas some are created by threads that are used in program However, the memory area created by JVM is destroyed only when JVM exits.



Java memory Area part

1> Heap :1> It is shared runtime data & stores the actual object in memory. It is instantiated during virtual machine startup:

2> Heap can be of fixed or dynamic size depending upon the system configuration.

3> JVM provides the user control to initialize or vary the size of zip as per the requirement.

When the new keyword is used, object is assigned a space in heap, but the refrence of the same exists onto the stack.

- There exists one and only one heap for a running JVM process.

Scanner SC = new Scanner (System. in)

The above statement creates the object of Scanner class which get allocated to heap. & "SC" refrence is pushed to stack.

2) Method Area:
- It is logical part of heap area & created on virtual machine startup.
- This memory is allocated for class structures, method data & constructor field data, & for interface or speial method used in classes.
- It memory can be fixed or dynamic size depending on system configuration.

3) JVM Stacks:
- Stack is created at the same time when the thread is created & is used to store data & partial result which will be needed while returning value.
> for method performing dynamic linking.
> Memory for stack need not to be contigious.

4) Native method Stack:
> Called as C stack.
> written in java languge.
> This memory is allocated for each thread & it can be fixed or dynamic

5) Program counter (PC) registers:
> each JVM thread out task of a specific method has a program counter register associated with it.
> non-native method has a PC which stores the adress of the available JVM instruction whereas in native method, value of PC is undefined.
> PC is capable of storing return adrs or native pointer on specific platform.

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

• Working of a Garbage counter:

> JVM triggers this process and as per the JVM garbage ~~process~~ collection process is done or else withheld. It reduces the burden of programmer by automatically performing the allocation or dellocation of memory.

> Garbage collection process causes the rest of the processor or threads to be paused and thus is costly in nature. This problem is unacceptable for the client but can be eliminated by applying several garbage collector turning & is important for improving the preformance of a program.