Assignment 2:-

1) components of JDK -

It stands for Java development kit. And it is the software development environment which is used for developing java application.

Java Development kit are consists of the component like loader, compiler, an Archive file, a documentation generator with many other components.

2) Difference between JVM, JRE and JDK.

JVM - JVM stands for the Java virtual Machine which provides the Runtime environment to the Java programs. In the JVM the java applications can be executed.

JRE - JRE stands for Java Runtime Machine JVM uses JRE for its implementation.

JDK - JDK stands for Java Development kit And it contains JRE and development tools.

3) Role of the JVM in Java? How does the JVM execute Java code.

JVM stands for Java virtual Machine that compiles the Java program and produces the .class file.
JVM executes the code which include load code, verify code, Execute code and provides the runtime environment.

4) Memory Management of the JVM.

As JVM has various data area which are used during execution of program. Some of the Memory Management created and by JVM and some are created by thread.
Some of JVM memory Areas areas are Heap area, Method area, JVM stack, Native Method stack, PC Registers.

Page No.
Date

5) JIT compiler and its role in JVM.
What is bytecode and why it is important.

JIT stands for Just In Time
It is runtime environment component and used for the improvement of the Java code or application. It compile the Bytecode in the Machine code at runtime
At the runtime JVM loads the class files and determines the each individual of the bytecode and performs the necessary computation
JIT compiler is enabled by default.
When the method is compiled the JVM calls the code directly and compilation doesnot take and extra time or memory.

6) How JVM Works.
JVM stands for Java Virtual Machine
JVM calls the main method present in the code. We can write java code once and can run it on any platform which is all possible due to JVM.
As the code is compiled this compilation is done by JVM and class file is created and the code is compiled step by step this step by step compilation decribes JVM.

7) How does java achieve platform dep Indepe through the JVM.

As Java follows the WORA concept which means Write once and Run Anywhere this is supported by the JVM. once the code is written then it can be run on any device that has through JVM. The code is converted into the Machine understandable language through the compiler with the help of JVM. So the code once written can be run on any operating system without any changes by JVM.

8) Significance of class loaders in Java and process of garbage collection in Java.

class loader performs the task of loading Java class in to the JVM during runtime. class loaders are also the part of JRE. JVM does not need to know about the files and files structure because of class loader.

Due to the Garbage collector there is no need to allocate or deallocate the memory dynamically like C++ It main purpose is to enhance speed of application and memory leaks.