Java assingment interview questions

Q.1) Explain the components of JDK.

→ JDK is a software development environment used for developing java application.
It is neccessary to write compile Debug and run java

1) Java compiler :-

Translate java source code (.java files) into bytecode (.class file can be executed by java virtual machine)

2) Java Runtime enviroment :-

1) A subset of the jdk that includes the libraries and components neccessary to run java application

2) It is the part of the JDK but can be downloaded seprativ

3) provide libraries, the JVM and other components to run application.

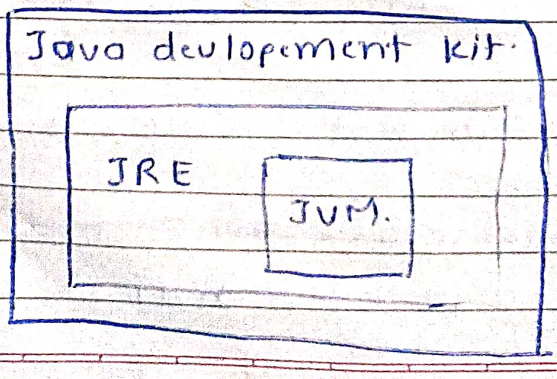
~~4) Does not have tools~~

3) JVM.

1) It is apart of JRE resposible for executing bytecode

2) It ensures the write ~~con~~ once run anywhere capabillity

3) Not platform independent a diffrent JVM is needed for each type of os.



Teacher's Signature:.....

2) Diffrence between JDK, JVM, JRE.

JDK	JRE	JVM
It stands for Java Development kit	It stands java Runtime enviroment	It stands for java virtual machine
It is the tool neccesary to compile document and Package Java program	JRE refers to a runtime enviroment In which Java bytecode can be executed	It is a abstract machine It is a specification that provides runtime enviroment which java bytecode can be executed.
~~3) It is sub~~		
3) It is superset of of JRE & JVM	JRE is a subset of JDK	JVM is a subset of JRE & JDK

3) what is the Role of JVM in java? How does the JVM execute java code?

→ 1) It is the part of JRE resposible for executing the bytecode.
2) Ensure java's write once run anywhere capabillity
3) It is platform dependent diffrent for the diffrent platforms.

4) Explain memory management system ~~in~~ of the JVM

→ In the JVM memory management is handled by Garbage collection (G.C) which automatically free up memory used by unrefrenced objects memory divided into.

Teacher's Signature:.....

Heap memory

- Young genration :- for new objects.
- old genration :- for long-lived objects.

Garbage collection :-

- Minor GC :- cleans the young genration.
- Major GC :- cleans the old genration.

stack memory :- Holds local variables and method calls for each thread.

This automatic proccess optimizes the automatic process optimizes memory usage and prevents memory leaks.

5) What is JIT compiler and its Role in the JVM ? what is the bytecode and why it is important for Java.

→ Just-In-time compiler is a part of the Java virtual machine that improves the performance of java application

During execution JVM uses the JIT to translate java bytecode

Performance boost :- by compiling bytecode ~~It red~~ into a native code during runtime

It Reduce overhead of interpreting bytecode multiple times.

Bytecode :-

Bytecode is the intermediate code genrated by the java compiler from java source code

It is platform independent and it can be run on any machine

6) Describe the architecture of JVM.

→ 1) class loader :- loads java classes into memory during runtime

2) Memory area :-
- Method area :- stores class level data like method defination and static variable
- Heap :- Allocates memory for java objects
- stack :- Manages method calls and local variable foreach thread

3) Execution Engine :- Executes bytecode using components like
- interpreter :- Interprets bytecode line by line
- JIT compiler :- compiles frequently used bytecode to native machine code for better performance

4) Garbage collector :- Automatically manges memory by freeing unused objects

5) Native method interface :- Allows Java code with interact with Native (Non java) application.

7) How does java achieve Platform independence through JVM

→ Java achieves platform independence through the java virtual machine (JVM) By compiling the source code into byte code which is platform independant the bytecode can be executed on any system with JVM regardles underlying Hardware and opreating System The JVM interprets OR compiles bytecode into native machine code. and enable write eonce Run anywhere

8) What is the significance of the class loader in Java? what is the process of garbage collection in java?

→ The class loader:- In Java dynamically loads classes into memory during runtime, enabling Java's ability to load classes as needed

Process of garbage collection in java:-
In java grobage collection automatically frees memory by identifying and removing objects that are no longer in use.

9) what are the four access modifiers in java and how do they differ from each other?

→ Public:- Accesible in any class.
Protected:- Accesible within the same Package and subclasses in other Package.
Private:- accesible only within the same class.
Default:- Accesible only within the same package.

10) what is the difference between Public protected and default access modifiers

→ Public: members are accesible from any class, regardles of the package
Protected:- members are accesible with in the same package and to the subclass in other package
Default:- members are accesible within same package or subclass

11) can you override method with a diffrent access modifier in a subclass

→ No we cannot override a method with a more restrictive access modifier in a subclass

12) what is diffrence between protected and default (Package - private access)

→ The protected access modifier allows a member to be accesible in the same package and by subclasses in any package
Default access allows a member to be accesible only within same package but not by the subclasses outside package.

13) It is possible to make a class private in java? If yes where can it done.

→ In java class cannot be made private at the top level However inner (nested) classes can be declared private within another class.
limitations :-
1) A top level class cannot be private : It can only have public OR package-private (default)
2) A private inner class cannot be instantiated OR accessed from outside outer class.

14) can top-level class in java be declared as protected OR private

→ No top level class in java cannot be declared as protected OR private this is because top level classes need to be accesible from other classes.

Teacher's Signature:.....

15) What Happens if you declare variable or method as private in a class and try to access from it another class within same package?

→ If you declare a variable or method as private in a class and try to access it from another class within the same package will Result compile error.

The private access modifier restricts the access to the member only within the class where it declared meaning it cannot be access from any other class even if they are in same package.

16) Explain the concept of "package-private" or default access. How does it affect the visibility of class member?

→ ~~Visibility Rules:-~~

Package private access in java means that a class member (field, method or constructor) is accesible only to ~~ther~~ other classes within the same package. It No access modifier the member implicitly given package-private access.

Visibility Rules:-

- Accesible:- to all classes within same package.
- Not accesible to the classes in diffrent packages, even if they are subclass.

Teacher's Signature:.....