ANS:

Java Introduction :. (GNS (Theories))

Java is a high-level, object oriented programming language developed by James Gosling and his team at Sun Microsystems in the mid-1990s. It was designed to be platform independent and could run on any system that had a Java Virtual Machine (JVM) installed. Java is known for simplicity, readability and ease of use, making it one of the most popular programming languages in the world.

(10) class:

class is a blueprint or template that defines the properties and behaviours of an object.

Method:

A method in java is a set of instructions or code that performs a specific tasks or action. Methods are defined within a class and can be called or invoked from within the same class or from other classes.

object:

An object in java is an instance of a class. It is created using the constructor of the class and can access the properties and methods of the class.

QUESTIONS (Theory)

(1.) **platform independent ?**

platform independent refers to the ability of a program or software to run on multiple platforms or operating system without modification.

Eg. Java is platform independent language.

(2.) **open source :**

open source refers to a type of software development model in which the source code of a program or application is made available to the public.

Any one can view, modify and distribute the code without any kind of cost.

(3) JDK :→

Java Development kit. JDK is a Software development kit used for developing Java applications. It includes a compiler, debugger and other tools needed for Software Development.

JRE :→

Java Runtime Environment JRE is a Software package that provides the ~~Machine~~ environment in which java programs run.

JVM :→

Java Virtual Machine. JVM is the runtime environment in which java bytecode is executed. It provides a layer of abstraction between the Compiled java code and the underlying operating System and hardware.

(4) Why we go for Java? →

There are several reasons why Java is popular programming language and why developers choose to use it for developing applications. Here are some of the key reasons -

* Platform Independence, * OOPS,
* large community and extensive libraries, * Security, * Scalability.

(5) latest version of JDK? →

As of Sept 2021, latest version of JDK is JDK 17.

The version I have used is JDK, "1.8.0_361"

(6) latest version of Eclipse? →

Eclipse 4.26 (2022-12) released on Dec 7, 2022.

Eclipse version I have used is →

Eclipse 4.27 (2023-03)

(7) Difference between C++ and Java?

C++:

(i) Supports manual memory management, allowing for more fine-grained control over memory allocation and deallocation

(ii) Requires platform-specific compilation, making it less portable across different Operating systems.

Java:

(i) Uses automatic memory management, allowing the Java Virtual Machine (JVM) to handle memory allocation and deallocation automatically.

(ii) Runs on any platform that has a JVM installed, making it highly portable across different OSs.

(8) Features of Java:

(i) platform independent.
(ii) object Oriented programming
(iii) Automatic memory management
(iv) security
(v) large community and extensive libraries.
(vi) scalability.

(9) I have using the tool for Java execution is Eclipse.

(11) Where object stored?

objects are stored in

heap memory:

(12) How to access one class method into different package class?

Ans: By Creating Object reference and it will get import automatically

(15) what gives Java it's "write once, runs anywhere" natur?

Ans: platform Independent.

(14) coding standard to create :-

Project :

Every word first letter should be caps without any spaces.

Eg: ProjectOne

class:

same as project

Eg: StudentDetails

method:

Every first letter of word should be in small and first letter of second

word should caps (if any)

for eg:. empId()

Package:.

all letters (or) word should be in small case.

Syntax :.

org.student.

object:.

It should be in first letter of word (or) first letters of pascal notation words of your class file name.

for example.

If your class name is Animal,

The syntax of ~~put~~ Creating object is

Animal A = new Animal

If your class name is StudentDetails,

The object is

~~Student S~~

StudentDetails SD = new StudentDetails();