Oops Fundamentals

Q1- How to create an object in java?

Ans-The **object** is a basic building block of an OOPs language. In **Java**, we cannot execute any program without creating an **object**. There is various way to **create an object in Java** that we will discuss in this section, and also learn **how to create an object in Java.**

Java provides five ways to create an object.

- Using **new** Keyword
- Using **clone()** method
- Using **newInstance()** method of the **Class** class
- Using **newInstance()**method of the **Constructor** class
- Using **Deserialization**

Q2- what is the use of a new keyword in java?

Ans-The Java new keyword is used to create an instance of the class. In other words, it instantiates a class by allocating memory for a new object and returning a reference to that memory. We can also use the new keyword to create the array object.

Syntax

```
1. NewExample obj=new NewExample();
```

Points to remember

- It is used to create the object.
- It allocates the memory at runtime.
- All objects occupy memory in the heap area.
- It invokes the object constructor.
- It requires a single, postfix argument to call the constructor

Q3-What are the different types of variables in java?

Ans- There are three different types of variables in OOPs in Java.

- Instance variable
- Static variable
- Local variable

Q4-What is the difference between instance variable and local variable?

Ans-

Instance Variable	Local Variable
They are defined in class but outside the body of methods.	They are defined as a type of variable declared within programming blocks or subroutines.
These variables are created when an object is instantiated and are accessible to all constructors, methods, or blocks in class.	These variables are created when a block, method or constructor is started and the variable will be destroyed once it exits the block, method, or constructor.
These variables are destroyed when the object is destroyed.	These variables are destroyed when the constructor or method is exited.
It can be accessed throughout the class.	Its access is limited to the method in which it is declared.

It includes access modifiers such as private, public, protected, etc.	It does not include any access modifiers such as private, public, protected, etc.

Q5-In which area memory is allocate for instance variable and local variable?.

Ans-Instance variables are declared in the class, but outside of the constructors, methods, or blocks of the particular class. They are used to represent the state of an object and are stored in the **heap section** of the memory.

Primitive local variables are only allocated to **Stack Memory** blocks that contain their methods.

Q6-What is method Overloading?

Ans-Method Overloading allows different methods to have the same name, but different signatures where the signature can be represent by the number of input parameters or type of input parameters.

In java two ways for overloading method.

1. **By changing the number of arguments.**
2. **By changing the type of datatype.**