# OOPs Fundamental

## Day 10 Assignment

1. How to Create an Object in Java?
2. What is the use of a new keyword in Java?
3. What are the different types of variables in Java?
4. What is the difference between Instance variables and Local variables?
5. In which area memory is allocated for instance variable and local variable?
6. What is method overloading?

### 1. How to Create an Object in Java?

Ans: Five ways to create an object in java:

1. Using new Keyword
2. Using clone() method
3. Using newInstance() method of the Class keyword
4. Using newInstance() method of the Constructor class
5. Using Deserialization

### 2. What is the use of a new keyword in Java?

Ans: Use of new Keyword

- 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

### 3. What are the different types of variables in Java?

Ans: There are three types of variables in Java (i.e **Local, Instance, and Static**.).

### 4. What is the difference between Instance variables and Local variables?

Ans:

| Instance variables | Local variables |
|---|---|
| The variables that are declared inside a class but not within a method are called Instance variables. | The variables that are declared within a method are called local variables. |
| Instance variables have default values. | Local variables do not get default values. |
| If we call instance variables that are not initialized, we will get the default values of the particular variable type. | If we call local variables before they are initialized, compiler will throw a compile error. |

## 5. In which area memory is allocated for instance variable and local variable?

Ans: For instance variable, the memory is allocated in **Stack** & For local variable, the memory is allocated in **Heap**.

## 6. What is method overloading?

**Ans:** Method overloading is a feature that allows a class to have more than one method with the same name, but with different parameters.
Java supports method overloading through two mechanisms:

1. By changing the number of parameters
2. By changing the data type of parameters Overloading by changing the number of parameters A method can be overloaded by changing the number of parameters.