Constructor

Q.1) What is a Constructor?

Ans) A Constructor is a special method that is used to initialize objects of a class. It is called automatically when an object is created, and it has the same name as the class in which it is defined.

Q2) What is Constructor chaining?

Ans) Constructor chaining is a technique in Java that allows one constructor to call another constructor in the same class or in the parent class. It is achieved using this() and super() keyword.

Q3) Can we call a subclass Constructor from a superclass Constructor?

Ans) No, because a subclass Constructor requires the superclass Constructor to be called first in order to initialize the inherited variables & set up the superclass state properly.

Q4) What happens if you keep a return type for a Constructor?

Ans) Constructor do not have return type. If you try to define a return type, it will be treated as normal method and not as a Constructor. This means no object will be created & returned by the method.

Q5) What is No-arg Constructor?

Ans) A no-arg Constructor, also known as a default constructor

A no-arg Constructor is a Constructor in Java that takes no arguments. It is a special type of Constructor that is provided by default by java Compiler if no other Constructor is explicitly defined for a class. It is also called default Constructor.

Q6) How is a No-arg different from the default Constructor?

Ans) Default Constructor :- it is given by Complier,

- It's accessibility modifier is same as class accessibility modifier. So the only allowed accessibility modifiers are default & public,
- It does not have logic except the super() call.

No-arg Constructor - It is given by developer.

- It Can have all four accessibility modifiers as it is defined by the developer. So that allowed accessibility modifiers are private, default, Protected & public.
- It Can have logic including super() Call.

Q7) When do we need Constructor Overloading?

Ans) Constructor overloading is used in Java when a class needs to have multiple Constructors with different parameters. The main reason for using Constructor overloading is to provide flexibility in object creation by allowing different ways to initialize an object.

Q8) What is Default Constructor, Explain with an Example?

Ans) A Constructor that take no arguments

Eg - Public class Person {
    private String name;
    private int age;

    public Person(){ //default
        this.name = "unknown"; Constructor.
        this.age = 0;
    }