Smart Programming : YouTube Channel

**An investment in Knowledge pays the best interest….**



# Polymorphism (Method Overloading) in Java

## => Polymorphism :-

-> Poly (many) + Morhism (forms, structure)

-> Real world example : sound, water, many brand clothes in one shop, person, single institute has multiple trainer etc

-> Advantage :- It provides the flexibility to develop an application i.e. it allows us to perform a single task by different ways.

-> Types of Polymorphism :-

1. Compile Time Polymorphism

2. Runtime Polymorphism

## => Compile Time Polymorphism :-

-> It is also known as Static Polymorphism or Early Binding

-> If the polymorphism is achieved at compile time then it is compile time polymorphism

-> Compile Time Polymorphism can be achieved by 2 ways :-

1. Method Overloading

2. Operator Overloading (is not supported in java except + symbol)

**=> Runtime Polymorphism :-**

-> It is also known as Dynamic Polymorphism or Late Binding

-> If the polymorphism is existed at runtime then it is known as Runtime Polymorphism

-> Runtime Polymorphism can be achieved by "Method Overriding"

## => Method Overloading :-

-> The process of compiler trying to resolve the method call based on reference type is known as method overloading

-> Rules for method overloading :-

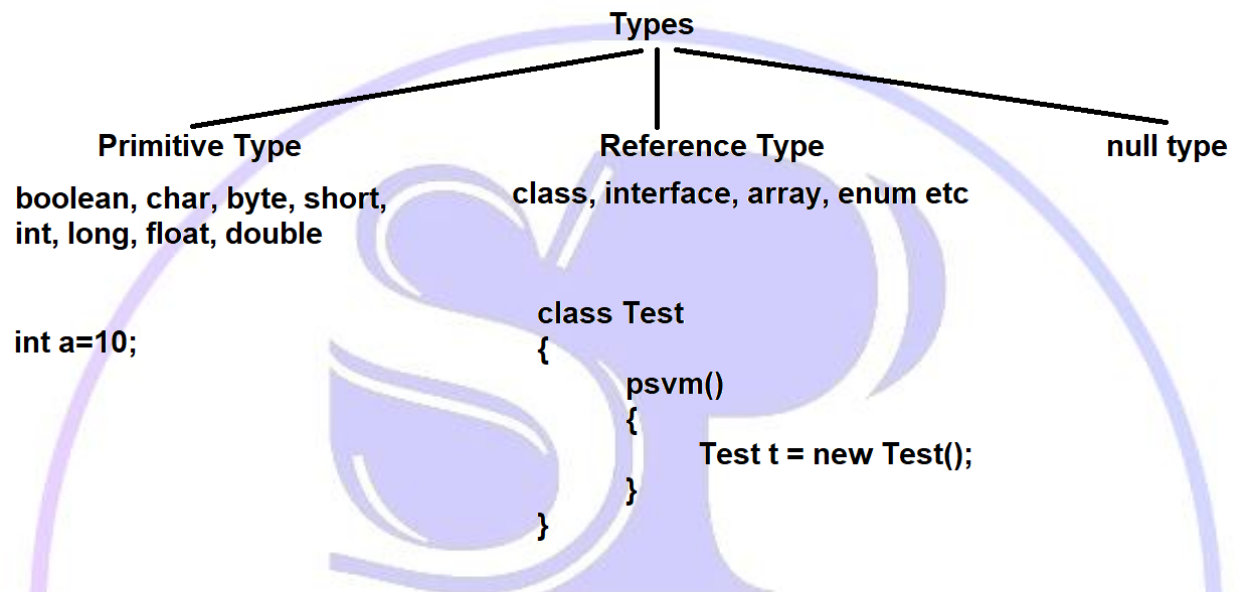1. Same name

2. Within same class

3. Different parameters

-> No of parameters

-> Type of parameters

-> Sequence of parameters

# What is reference type ?

```
                                Types


        Primitive Type              Reference Type              null type

    boolean, char, byte, short,   class, interface, array, enum etc
    int, long, float, double


                                class Test
    int a=10;                   {
                                    psvm()
                                    {
                                        Test t = new Test();
                                    }

                                }
```

# What is parameter and argument ?

```
class Test
{
    void show(int a)          → Parameter
    {
        //coding
    }
    public static void main(String[] args)
    {
        Test t=new Test();
        t.show(20);
    }                              → Arguments
}
```

# Interview Questions :-

1. Operator Overloading is not supported in java but + operator is overloaded

   -> Operator Overloading concept is achieved only by java designers but it cannot be achieved by developers like us.

2. What is difference between Compile Time Polymorphism & Runtime Polymorphism

3. What is varargs

4. Can we overload main method -> Yes

5. Can we overload constructors -> Yes

# Company Links & Contacts

**Company Name:** Smart Programming (+91 62838-30308)

**Address :** Chandigarh & Mohali (Punjab), India

**Websites:** https://www.smartprogramming.in/

https://courses.smartprogramming.in

**Android App:**
https://play.google.com/store/apps/details?id=com.smartprogramming

**YouTube Channel:**
https://www.youtube.com/c/SmartProgramming