**Method Overloading in Java**

If a [class](https://www.javatpoint.com/object-and-class-in-java) has multiple methods having same name but parameters of the method should be different is known as **Method Overloading**.

If we have to perform only one operation, having same name of the methods increases the readability of the [program](https://www.javatpoint.com/java-programs).

Suppose you have to perform addition of the given numbers but there can be any number of arguments, if you write the method such as a(int,int) for two parameters, and b(int,int,int) for three parameters then it may be difficult for you to understand the behavior of the method because its name differs.

**Parameters should be different means**

1. **Type of parameter should be different**

**Eg:**

**void add(int, int);**

**void add(double,double);**

class Adder{

void add(int a, int b){

System.out.println(“sum =”+(a+b));

}

void add(double a, double b){

System.out.println(“sum=”+(a+b));

}

public static void main(String[] args){

Adder ad=new Adder();

ad.add(5,6);

ad.add(5.4,7.2);

}}

1. **Number of parameter should be different**

**Eg:**

**void add(int , int);**

**void add(int , int, int);**

class Adder{

void add(int a, int b){

System.out.println(“sum =”+(a+b));

}

void add(int a, int b,int c){

System.out.println(“sum=”+(a+b+c));

}

public static void main(String[] args){

Adder ad=new Adder();

ad.add(5,6);

ad.add(5.4,7.2);

}}

1. **Sequence of parameter should be different**

**Eg:**

**void add(int , double);**

**void add(double , int);**

class Adder{

void add(int a, double b){

System.out.println(“sum =”+(a+b));

}

void add(double a, int b){

System.out.println(“sum=”+(a+b));

}

public static void main(String[] args){

Adder ad=new Adder();

ad.add(5,6.2);

ad.add(5.4,7);

}}

**Q: Write a class Volume in Java program to find volume of cube, cylinder and rectangular box using the concept of method overloading.**

**Volume of cube=s3; volume of cylinder=pi\*r2h; and volume of rectangular box=l\*b\*h**

**Q: Write a class Draw in Java program with three draw methods. The output of that draw method should be as follows:**

**draw() draw(int) draw(int,char)**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  | **\*** |  |  |
|  | **\*** |  | **\*** |  |
| **\*** |  | **\*** |  | **\*** |
| **\*** | **\*** |  | **\*** | **\*** |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  | **\*** |  |  |
|  | **\*** |  | **\*** |  |
| **\*** |  | **\*** |  | **\*** |
| **\*** | **\*** |  | **\*** | **\*** |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  | **A** |  |  |
|  | **A** |  | **A** |  |
| **A** |  | **A** |  | **A** |
| **A** | **A** |  | **A** | **A** |