There are two ways to overload the method in java

1. By changing number of arguments
2. By changing the data type

❖ **Program By changing number of arguments**

```
class Moverloading
{

        public int sum(int x, int y)
        {
                return (x + y);
        }

     public int sum(int x, int y, int z)
     {
          return (x + y + z);
     }

     public double sum(double x, double y)
     {
          return (x + y);
     }

     public static void main(String args[])
    {
           int answer;
           double answer1;
           Moverloading s = new Moverloading();

           answer = s.sum(10, 20);
           System.out.println("The answer of addition of two Integer numbers is =" +
answer);

           answer = s.sum(10, 20, 30);
       System.out.println("The answer of addition of three Integer numbers is =" +
answer);

        answer1 = s.sum(10.5, 20.1);
        System.out.println("The answer of addition of two Double numbers is =" +
answer1);
   }
}
```

❖ **Program By changing Datatypes**

```
public class Moverloading1
{

        public int sum(int x, int y)
        {
                return (x + y);
        }

    public double sum(double x, double y)
    {
         return (x + y);
     }

   public static void main(String args[])
   {
        int answer;
        double answer1;
        Moverloading1 s = new Moverloading1();

        answer = s.sum(10, 20);
        System.out.println("The answer of addition of two Integer numbers is
                        =" + answer);

            answer1 = s.sum(1.2, 0.8);
         System.out.println("The answer of addition of two double numbers is
                          =" + answer1);

   }

}
```

❖ **Constructor Overloading**

```
public class Student
{
    int id;
   String name;
    Student()
   {
         System.out.println("this a default constructor");
    }
     Student(int i, String n)
    {
       id = i;
       name = n;
     }

    public static void main(String[] args)
   {
      Student s = new Student();
      System.out.println("\nDefault Constructor values: \n");
      System.out.println("Student Id : "+s.id + "\nStudent Name : "+s.name);

      System.out.println("\nParameterized Constructor values: \n");
      Student student = new Student(10, "David");
      System.out.println("Student Id : "+student.id + "\nStudent Name : "
                                    +student.name);
    }
}
```