Experiment No. 4
Implement a program on method and constructor overloading.
Date of Performance:
Date of Submission:

Aim: Implement a program on method and constructor overloading.

Objective: To use concept of method overloading in a java program to create a class with same function name with different number of parameters.

Theory:

Method Overloading is a feature that allows a class to have more than one method having the same name, if their argument lists are different. It is similar to constructor overloading in Java, that allows a class to have more than one constructor having different argument lists.

Example: This example to show how method overloading is done by having different number of parameters for the same method name.

```
Class DisplayOverloading
{
   public void disp(char c)
   {
       System.out.println(c);
   }
   public void disp(char c, int num)
   {
       System.out.println(c + “ “+num);
   }
}
Class Sample
{
  Public static void main(String args[])
  {
      DisplayOverloading obj = new DisplayOverloading();
      Obj.disp(‘a’);
      Obj.disp(‘a’,10);
  }
}
```

Output:

A

A 10

Java supports Constructor Overloading in addition to overloading methods. In Java, overloaded constructor is called based on the parameters specified when a new is executed.

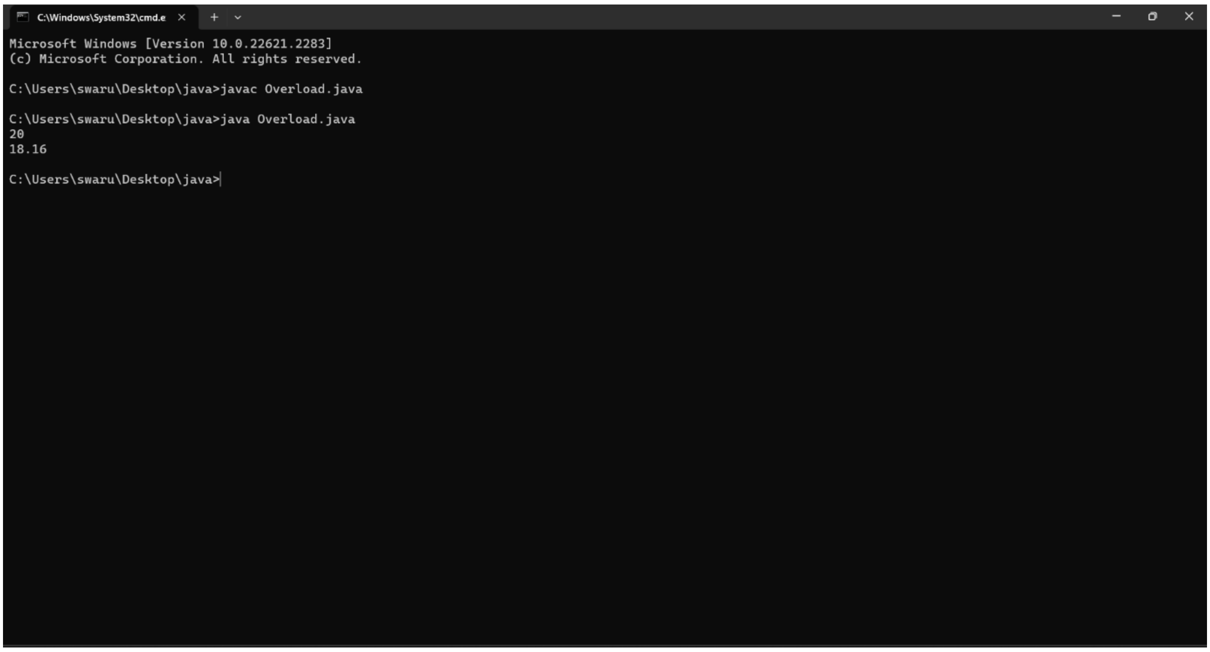
Sometimes there is a need of initializing an object in different ways. This can be done using constructor overloading.

For example, the Thread class has 8 types of constructors. If we do not want to specify anything about a thread then we can simply use the default constructor of the Thread class, however, if we need to specify the thread name, then we may call the parameterized constructor of the Thread class with a String args like this:

Thread t= new Thread (" MyThread ");

Code:

```
class Overload2
{
  public static void main(String args[])
   {
     System.out.println(Add.add(5,4));
     System.out.println(Add.add(2.80,3.12,9.00));
   }
}
class Add{
static int add(int a,int b) {return a+b;}
static double add(double a,double b,double c) {return a+b+c;}
}
```



```
Microsoft Windows [Version 10.0.22621.2283]
(c) Microsoft Corporation. All rights reserved.

C:\Users\swaru\Desktop\java>javac Overload.java

C:\Users\swaru\Desktop\java>java Overload.java
20
18.16

C:\Users\swaru\Desktop\java>
```

Conclusion:

Function and constructor overloading in Java involve a technique where you create multiple methods or constructors within a class, all bearing the same name but differing in their parameter configurations.

Function Overloading:
When you employ function overloading, you define multiple methods with identical names in a class, but each of them takes different sets of parameters, which may vary in terms of the number or types of parameters. This approach allows you to offer distinct behaviors for a particular operation, depending on the input values provided. During compilation, the appropriate method to execute is determined based on the arguments passed.

Constructor Overloading:
Similarly, constructor overloading involves defining multiple constructors within a class, all sharing the same name but differing in their parameter lists. Constructor overloading empowers you to create objects in various ways, contingent on the arguments supplied during the instantiation of objects. Much like function overloading, constructors can vary in parameter types and numbers, ensuring that each constructor is uniquely identified.