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.

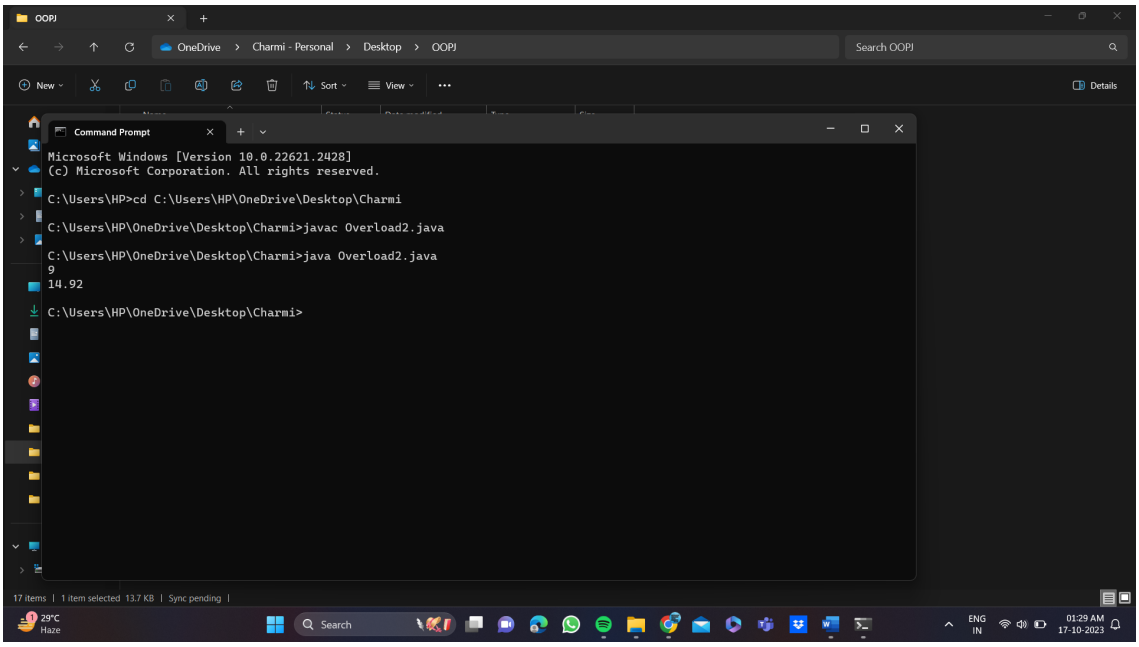
Sometimes there is a need of initializing an object in different ways. This can be done using constructor overloading.

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;}
}
```



Conclusion:

Comment on how function and constructor overloading used using java

In Java, both function overloading and constructor overloading revolve around the idea of creating methods or constructors with the same name but different parameter configurations within a class.

Function Overloading:
Function overloading empowers you to define multiple methods in a class under the same name, differing only in the type or number of parameters they accept. This mechanism allows you to tailor the behavior of a method to accommodate a variety of input scenarios. The Java compiler, during the compile-time phase, decides which method to invoke based on the provided arguments.

Constructor Overloading:
Similarly, constructor overloading involves the creation of multiple constructors within a class, all sharing the same name but distinct parameter lists. This feature opens the door to diverse ways of instantiating objects, depending on the arguments supplied during object creation. While constructors can differ in parameter types, it's crucial that the number and types of parameters vary between them to ensure distinction.