Vidyavardhini's College of Engineering and Technology

Department of Artificial Intelligence & Data Science

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

Experiment No. 4

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 ");

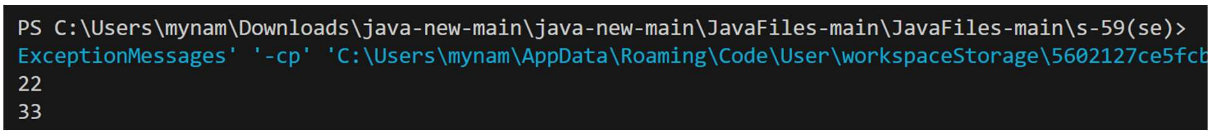
Code:

```
class Testoverloading1 {
    public static void main(String[] args) {
        System.out.println(Adder.add(11, 11));
        System.out.println(Adder.add(11, 11, 11));
    }
}

class Adder {
    static int add(int a, int b) {
        return a + b;
    }

    static int add(int a, int b, int c) {
        return a + b + c;
    }
}
```

OUTPUT:



```
PS C:\Users\mynam\Downloads\java-new-main\java-new-main\JavaFiles-main\JavaFiles-main\s-59(se)>
ExceptionMessages' '-cp' 'C:\Users\mynam\AppData\Roaming\Code\User\workspaceStorage\5602127ce5fcb
22
33
```

Conclusion:

Comment on how function and constructor overloading used using java

Function and constructor overloading in Java are used to create multiple methods or constructors with the same name but different parameters. This allows developers to create more flexible and intuitive code that can perform similar operations on different types of data or with different combinations of parameters.

Function overloading enables the creation of multiple functions with the same name but different parameter lists. Java determines which function to call based on the number and types of parameters provided

Constructor overloading allows the creation of multiple constructors within a class, each with a different parameter list. This enables the creation of objects in different ways, depending on the provided parameters.