

Experiment-2

NAME- Rajdeep Jaiswal	UID- 20BCS2761
CLASS & GROUP- 13-A	DATE-
SUBJECT- Java Programming	

1. Aim/Overview of the practical:

Write a program to create classes and use of different types of methods.

2. Task to be done/Which logistics used:

Show the use of different types of Method, i.e., Static and Non-Static method. Also, we have to create custom classes and use its methods in Main method by instantiating it through objectmethod.

3. Algorithm/Flowchart:-

1. Creating a custom class.
2. Creating a static method inside the custom class.
3. Creating a non-static method with same name and different arguments in order to show method overloading.
4. Writing the syntax of Main class and writing main method inside it.
5. Creating a static method inside the Main class with some arguments.
6. Creating a non-static method with same name and different arguments in order to show method overloading.
7. Creating objects of both Custom class and Main class inside the main method.

8. Calling static method of main class by their name and calling non-static method through object in main method.
9. Calling static method of custom class by class name and calling non-static method through object in main method.

4.Steps forexperiment/practical/Code:-

```
class ShowUse{

public static void ShowUse(){
System.out.println("->This is the static method of the custom
class.");
}
public void showUse(int a)
{
System.out.println("->This is the overloaded method of custom
class and the parameter is "+a);
}
}
public class Main
{
public static int addNum(int a,int b){return a+b;
}

public int addNum(int a,int b,int c){ return a+b+c;
}
}
```

```
public static void main(String[] args){
Main obj=new Main();
//object of Main class
System.out.println("Sum of the two numbers are:"+addNum(2,3));

//Static method of Main class
System.out.println("Sum of the three numbers
are:"+obj.addNum(2,3,4));

ShowUse obj1=new ShowUse();

obj1.showUse(2);
}

}
```

5. Observations/Discussions/Complexity Analysis:-

Static methods can be called by class.method() and we can call non-static method only by object.method().Method overloading can be done by giving different parameters to method sharing same name.

6. Result/Output/Writing Summary:-

Write your code in this editor and press "Run" button to execute it.

```
*****/
class ShowUse{
    public static void showUse(){
        System.out.println("-> This is the static method of the custom class.");
    }
    public void showUse(int a){
        System.out.println("-> This is the overloaded method of custom class and
the parameter is "+a);
    }
}
public class Main
{
    public static int addNum(int a, int b){
        return a+b;
    }
    public int addNum(int a, int b, int c){
        return a+b+c;
    }
    public static void main(String[] args) {
        Main obj=new Main(); //object of Main class
        System.out.println("Sum of the two numbers are: "+addNum(2,3));
        // Static method of Main class
        System.out.println("Sum of the three numbers are:
"+obj.addNum(2,3,4));
        ShowUse obj1=new ShowUse();
        ShowUse.showUse();
        obj1.showUse(2);
    }
}
```

```
Sum of the two numbers are: 5
Sum of the three numbers are: 9
-> This is the static method of the custom class.
-> This is the overloaded method of custom class and the parameter is 2

...Program finished with exit code 0
Press ENTER to exit console. █
```

Learning outcomes(What I have learnt):-

1. Creation of static method
2. Creation of custom class
3. Creation of object of classes.
4. Method overloading.
5. Calling static methods using class.method and non-static method using object.method.