Laboratory Report

OBJECT ORIENTED PROGRAMMING THROUGH JAVA

(CSE2016L)

School of Engineering

Department of Computer Science and Engineering

Submitted By	
Student Name	**Prashant Dagar**
Roll Number	**190020203031**
Section/Group	A- T2 group
Department	Computer Science and Engineering
Session/Semester	2020-21/ Odd Semester
Submitted To	
Faculty Name	**Ms. Neha Kapoor**

G D Goenka University

Gurgaon, Haryana

INDEX

10	16th September 2020	WAP in which class “A” extends class “B”. Show the order of execution of the constructors if object of class B is created.		
11	21st October 2020	WAP in which a class inherits two interfaces		

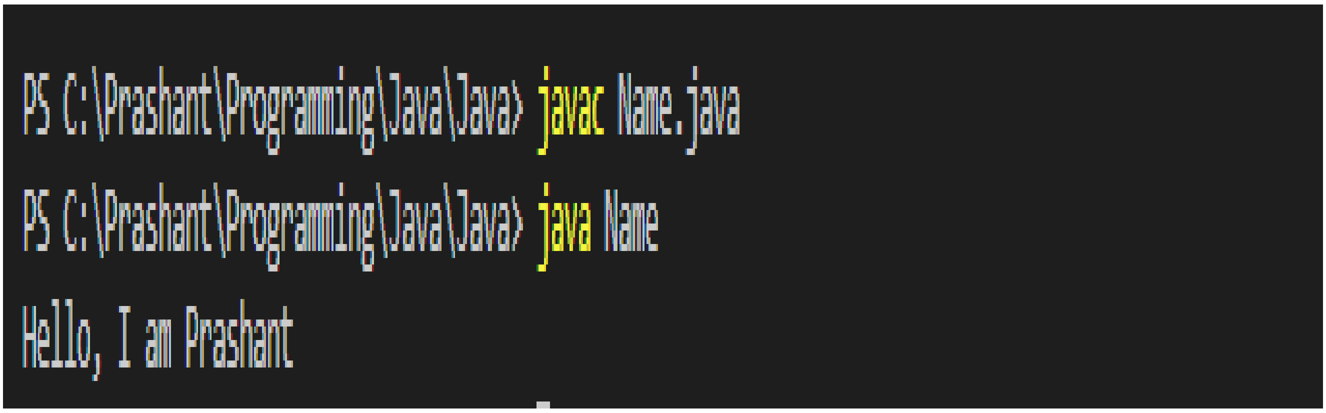
Experiment No. 1

AIM : Write a program to print your name.

Program:

```
public class Name {
public static void main(String[] args) { System.out.println("Hello, I
am  Prashant");}

}
```

Output



```
PS C:\Prashant\Programming\Java\Java> javac Name.java
PS C:\Prashant\Programming\Java\Java> java Name
Hello, I am Prashant
```

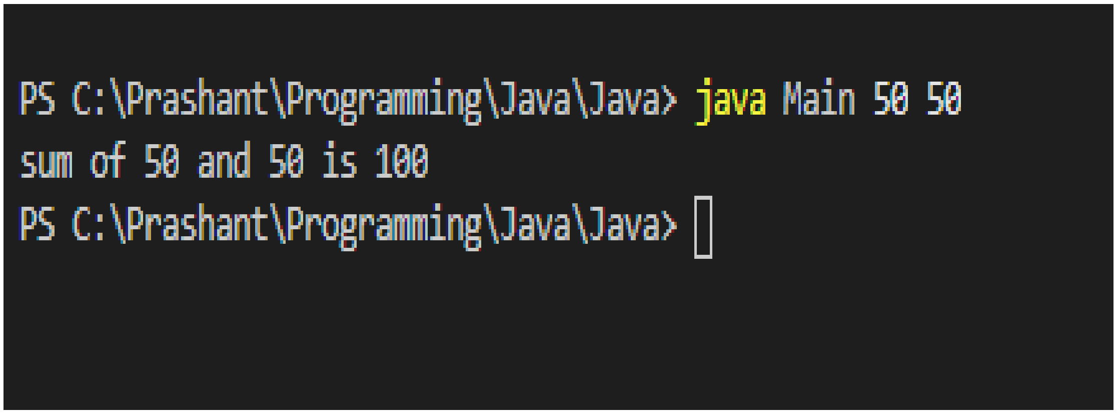
Experiment No. 2

AIM :WAP in java to accept two numbers (int) as command line arguments and print their sum.

Program:

```
public class Main {

public static void main(String ar[]) { int x,y,s;
x=Integer.parseInt(ar[0]); y=Integer.parseInt(ar[1]); s=x+y;
System.out.println("sum of " + x + " and " + y +" is " +s); }

}
```

Output:



```
PS C:\Prashant\Programming\Java\Java> java Main 50 50
sum of 50 and 50 is 100
PS C:\Prashant\Programming\Java\Java> 
```

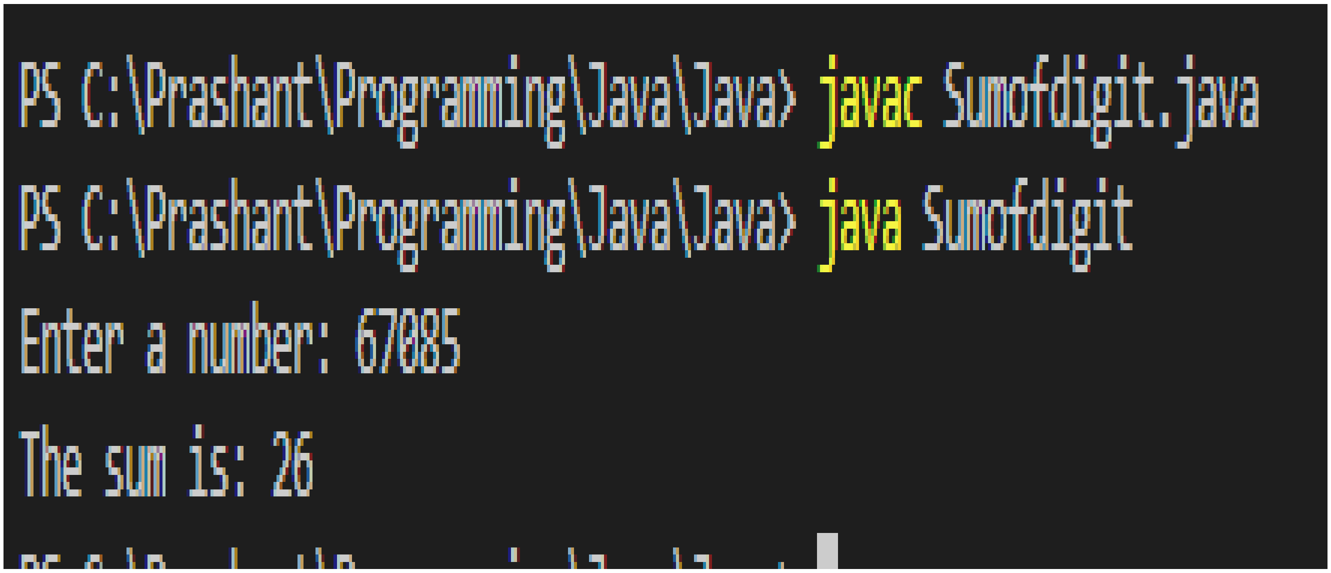
Experiment No. 3

AIM : WAP to find the sum of the digits of a number.

Program:

```
import java.util.Scanner;

public class Sumofdigit {

    public static void main(final String[] args) {
        int num, sum;
        final Scanner m = new Scanner(System.in);
        System.out.print("Enter a number: ");
        num = m.nextInt();

        sum = sum_of_digits(num);

        System.out.print("The sum is: " + sum + "\n");
        m.close();
    }

    public static int sum_of_digits(int a) {
        int s = 0;
        while (a > 0) {
            final int m = a % 10;
             s= s + m;
             a = a/10;
         }
         return s; }
 }
```

Output:



```
PS C:\Prashant\Programming\Java\Java> javac Sumofdigit.java
PS C:\Prashant\Programming\Java\Java> java Sumofdigit
Enter a number: 67085
The sum is: 26
PS C:\Prashant\Programming\Java\Java>
```

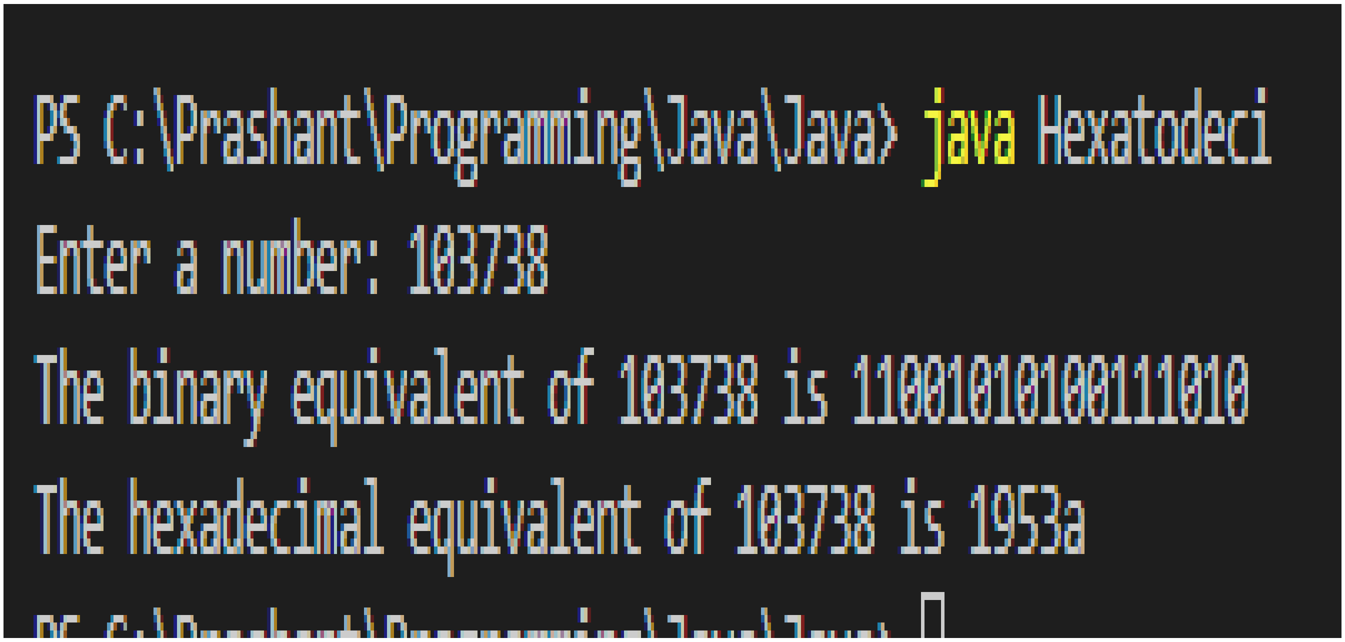
Experiment No. 4

AIM : WAP to convert an int value into its hexadecimal and binary equivalent.

Program:

```
import java.util.Scanner;

public class Hexatodeci {

    public static void main(String[] args){
        int num;
        String i,j;
        Scanner m = new Scanner(System.in);
        System.out.print("Enter a number: ");
        num =m.nextInt();

        i = binary_equivalent(num);
        j = hexadecimal_equivalent(num);
         System.out.print("The binary equivalent of "+num+ " is " +i+
 "\n");
         System.out.print("The hexadecimal equivalent of "+num+ " i
s " +j+ "\n");
         m.close();

        }
     public static String binary_equivalent(int a){
         String c = Integer.toBinaryString(a);
         return c;
    }
     public static String hexadecimal_equivalent(int b){
         String d = Integer.toHexString(b);
         return d;
        }
    }
```

Output:



```
PS C:\Prashant\Programming\Java\Java> java Hexatodeci
Enter a number: 103738
The binary equivalent of 103738 is 11001010100111010
The hexadecimal equivalent of 103738 is 1953a
PS C:\Prashant\Programming\Java\Java>
```

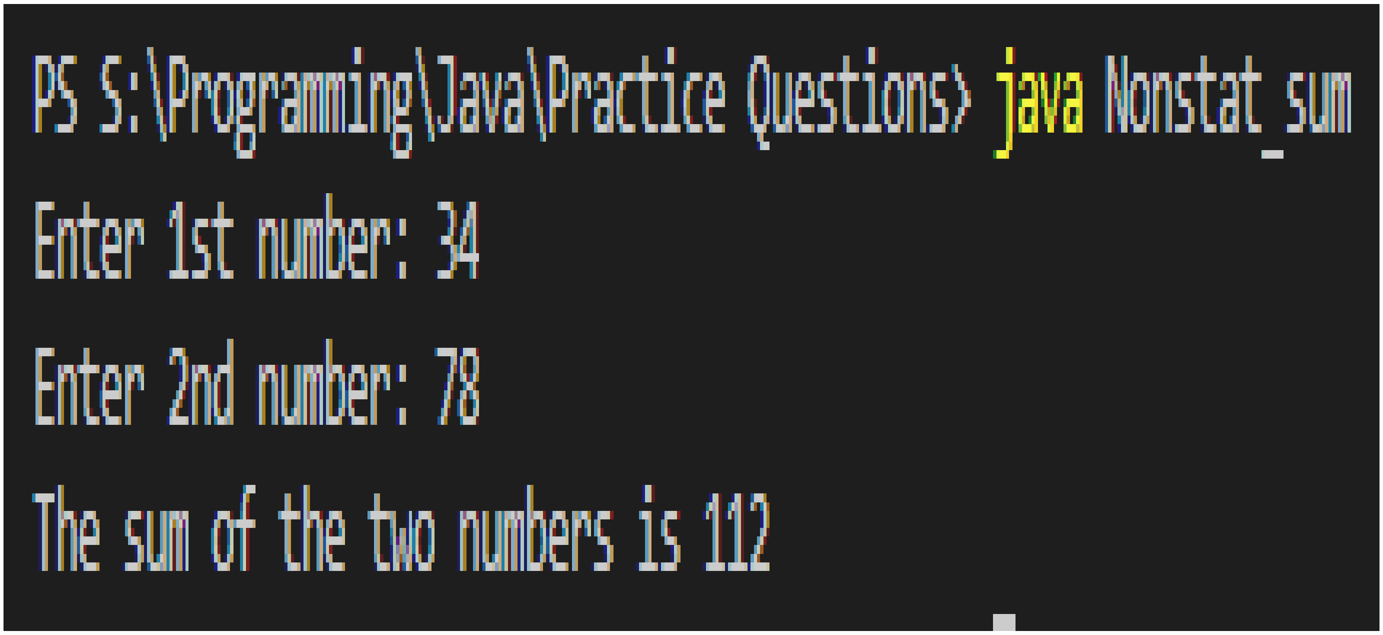
Experiment No. 5

AIM : WAP to create a non- static function that prints the sum of two numbers.

Program:

```
import java.util.Scanner;
public class Nonstat_sum {

    public static void main(String args[]){
        Scanner s = new Scanner(System.in);
        System.out.print("Enter 1st number: ");
        int n1 = s.nextInt();
        System.out.print("Enter 2nd number: ");
        int n2 = s.nextInt();
        sum m = new sum(n1,n2);
        m.display();
        s.close();
        }
}

class sum{
    int a,b,ans;
    sum(int i,int j){
        a=i;
        b=j;
        ans=i+j;
    }
    void display(){
        System.out.print("The sum of the two numbers is "+ans+"\n")
;
    }
    }
```

Output:



```
PS S:\Programming\Java\Practice Questions> java Nonstat_sum
Enter 1st number: 34
Enter 2nd number: 78
The sum of the two numbers is 112
```

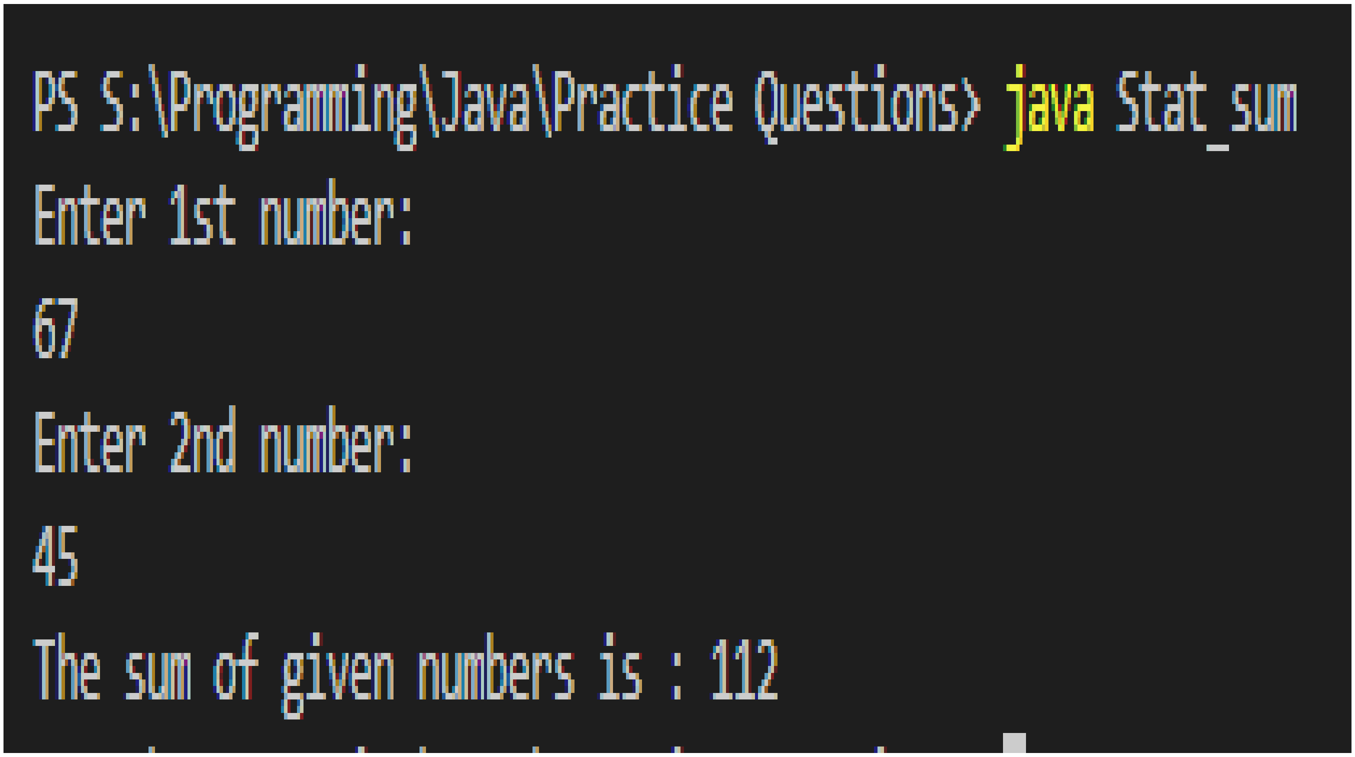
Experiment No. 6

AIM : WAP to create a static function to find the sum of two numbers.

Program:

```
import java.util.Scanner;
public class Stat_sum {

    static int a,b,sum;
    static Scanner s1 = new Scanner(System.in);
    public static void main(String[] args){
        Input1();
        Input2();
        sum = SumCal(a,b);
        Display(sum);
    }
    public static int Input1(){
      System.out.println("Enter 1st number: ");
      a = s1.nextInt();
      return a;
    }
    public static int Input2(){
      System.out.println("Enter 2nd number: ");
      b = s1.nextInt();
      return b;
    }
    public static int SumCal(int a, int b){
        return(a+b);
    }
    public static int Display(int sum){
       System.out.println("The sum of given numbers is : "+sum);
        return 0;
    }
}
```

Output:



```
PS S:\Programming\Java\Practice Questions> java Stat_sum
Enter 1st number:
67
Enter 2nd number:
45
The sum of given numbers is : 112
```

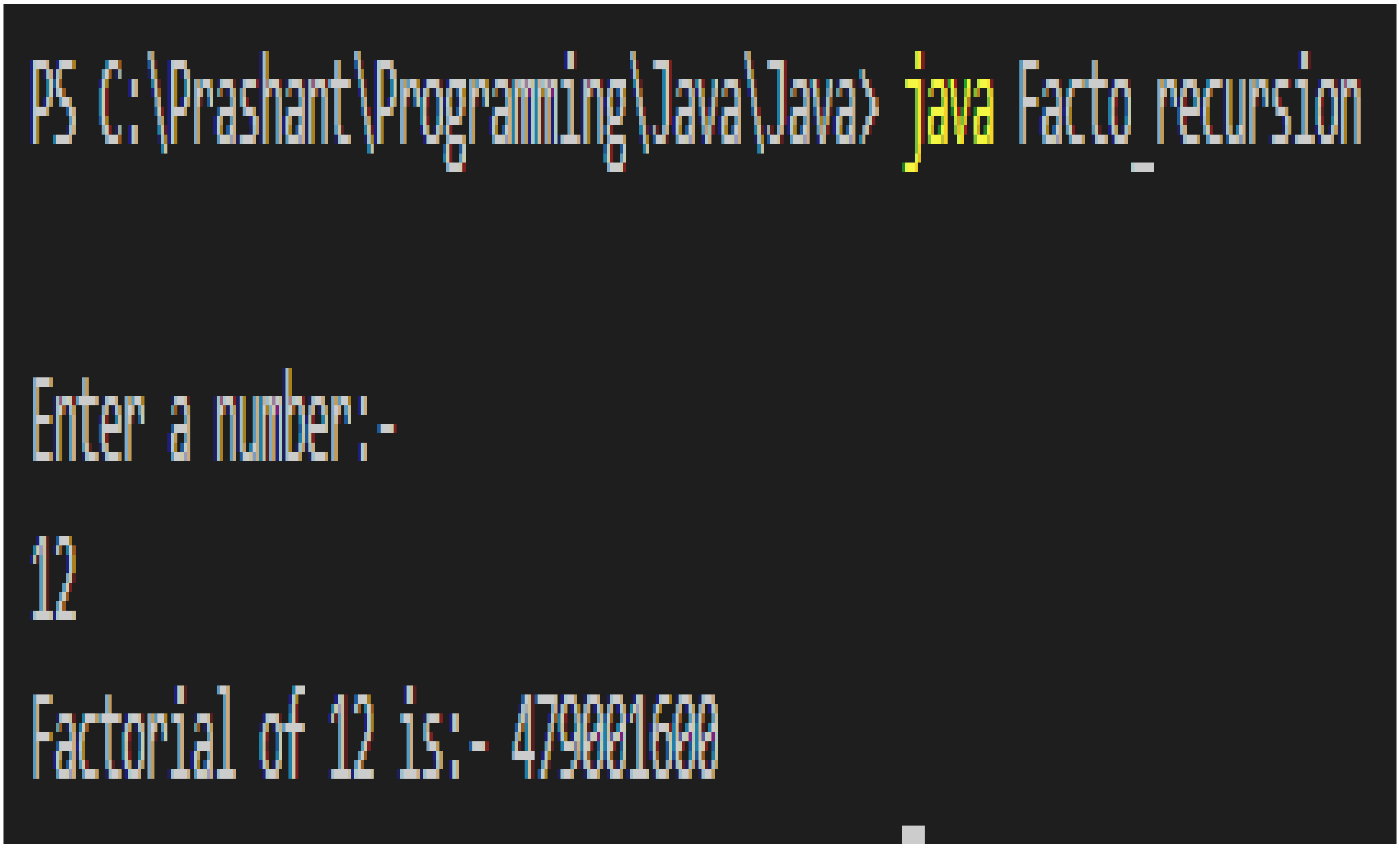
Experiment No. 7

AIM : WAP to print factorial of a number using recursion.

Program:

```
        import java.util.Scanner;

        public class Facto_recursion{

            public static void main(String args[]){
                System.out.println("\nEnter a number:- ");
                Scanner n = new Scanner(System.in);
                int num = n.nextInt();
                Facto_recursion r = new Facto_recursion();
                int fact = r.factorial(num);
                r.Display(num,fact);
                n.close();
            }

        int factorial(int n){
            if (n==0)
                return 1;
            else
                return(n*factorial(n-1));
        }

         void Display(int n, int fact){
             System.out.println("Factorial of "+n+" is:- "+fact);
         }


}
```

Output:



```
PS C:\Prashant\Programming\Java\Java> java Facto_recursion


Enter a number:-
12
Factorial of 12 is:- 479001600
```

Experiment No. 8

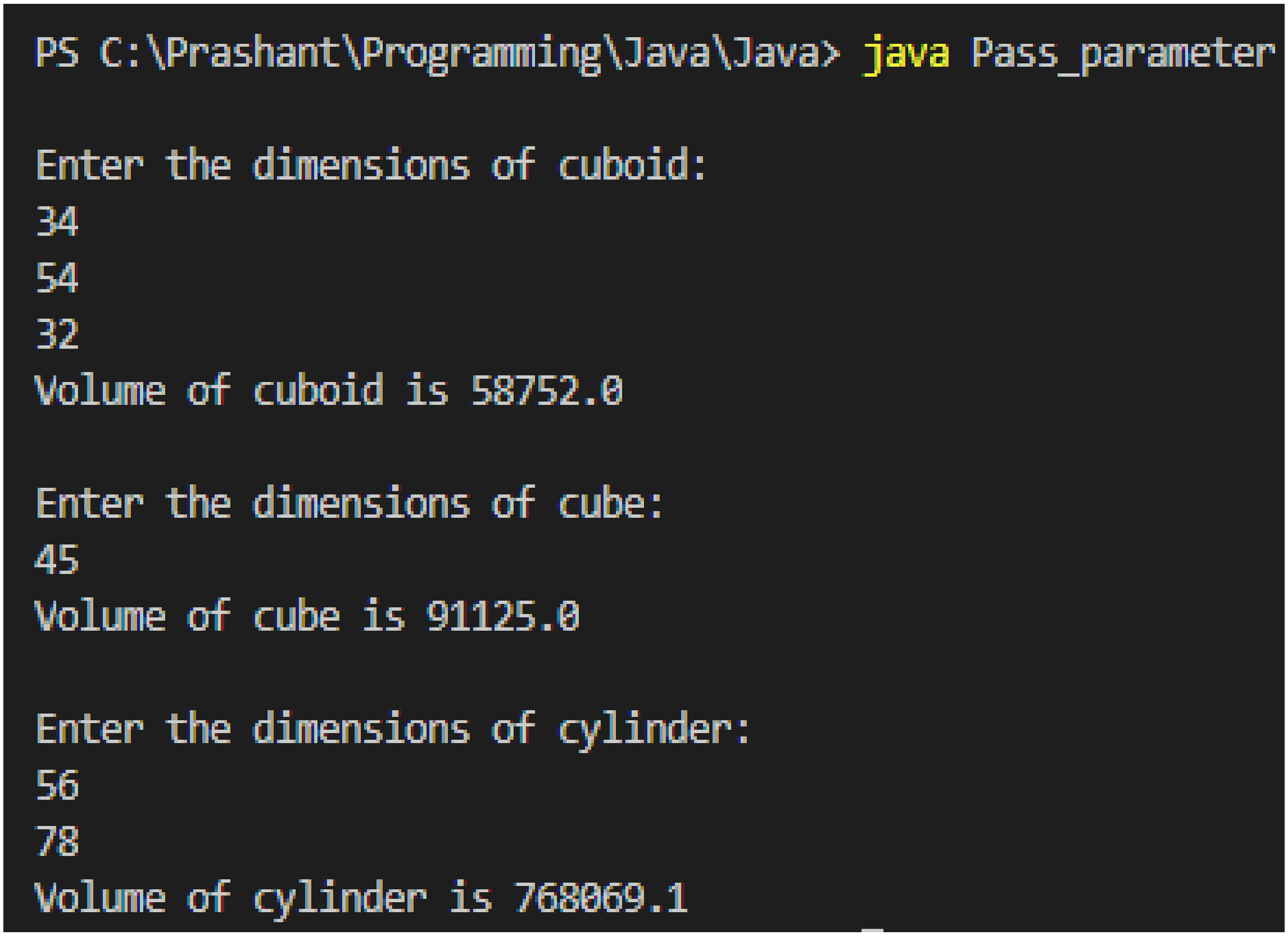
AIM : WAP to to implement constructor overloading by passing different number of parameters of different types.

Program:

```
import java.util.Scanner;

public class Pass_parameter {
    public static void main(String args[]){
        System.out.println("\nEnter the dimensions of cuboid:");
        Scanner n = new Scanner(System.in);
        int c1 = n.nextInt();
        int c2 = n.nextInt();
        int c3 = n.nextInt();
        Volume v1 = new Volume (c1,c2,c3);
        v1.displayvolofcuboid();

        System.out.println("\nEnter the dimensions of cube:");
        int p1 = n.nextInt();
        Volume v2 = new Volume (p1);
        v2.displayvolofcube();

        System.out.println("\nEnter the dimensions of cylinder:");
        int y1 = n.nextInt();
        int y2 = n.nextInt();
        Volume v3 = new Volume (y1,y2);
        v3.displayvolofcylinder();

        n.close();
    }
}
```

```
\\ in another file in same folder

public class Volume {

    int l;          // length of the object
    int b;      // breadth of the object
    int h;      // height of the object
    int r;      // radius of the circle/object
    float vol;  // voulume of object

    void volOfCuboid(int length, int breadth, int height){
        vol = length*breadth*height;
    }
    void volOfCube(int length){
        vol = length*length*length;
    }
    void volOfCylinder(int radius, int height){
        final float PI = (float) 3.14;
        vol =(PI*(radius*radius)*height);
    }

    Volume(int length, int breadth, int height){
        l = length;
        b = breadth;
        h = height;
        this.volOfCuboid(length, breadth, height);
    }
    Volume(int length){
        l = length;
        this.volOfCube(length);
    }
    Volume(int radius, int height){
        r = radius;
        h = height;
        this.volOfCylinder(radius, height);
    }

    void displayvolofcuboid(){
        System.out.print("Volume of cuboid is "+vol+"\n");
    }
    void displayvolofcube(){
        System.out.print("Volume of cube is "+vol+"\n");
    }
    void displayvolofcylinder(){
        System.out.print("Volume of cylinder is "+vol+"\n");
    }
}
```

Output:



```
PS C:\Prashant\Programming\Java\Java> java Pass_parameter

Enter the dimensions of cuboid:
34
54
32
Volume of cuboid is 58752.0

Enter the dimensions of cube:
45
Volume of cube is 91125.0

Enter the dimensions of cylinder:
56
78
Volume of cylinder is 768069.1
```

Experiment No. 9

AIM : WAP to Create an abstract class Shape which has a field PI=3.14 as final and it has an abstract method Volume. Make two subclasses Cone and Sphere from this class and they print their volume.

Program:

```
        public abstract class Shape {

            final double pi = 3.14;
            abstract void volume();
}



// in same folder but in different file for volume of cone

        import java.util.Scanner;
        public class Cone extends Shape{

            void volume(){
                Scanner n=new Scanner(System.in);
                System.out.println("\nEnter radius of cone:- ");
                double radius= n.nextDouble();
                System.out.println("\nEnter height of cone:- ");
                double height=n.nextDouble();
                double v=0.33333*pi*radius*height;
                System.out.println("\nVolume of cone is:-  "+ v);
                n.close();
            }
}
```

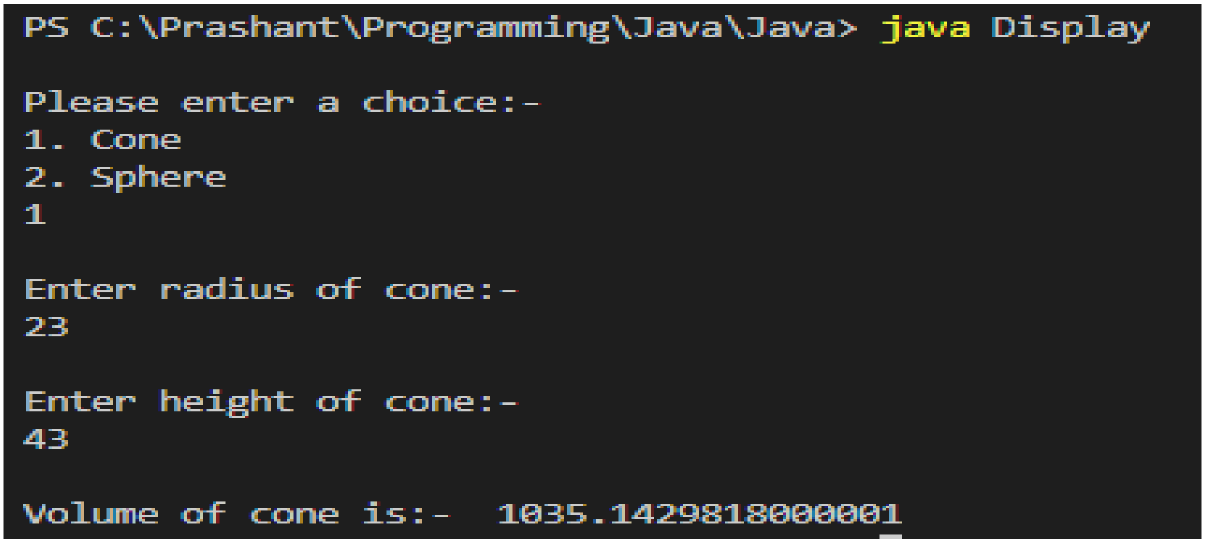
// again for volume of sphere creating a class file in same folder but in different file

```
    import java.util.Scanner;

    public class Sphere extends Shape{

        void volume(){
            Scanner n=new Scanner(System.in);
            System.out.println("\nEnter radius of Sphere:- ");
            double radius=n.nextDouble();
            double v= 1.3333333*pi*radius*radius*radius;
            System.out.println("\nThe volume of sphere is :- "+ v);
            n.close();
    }
}
```

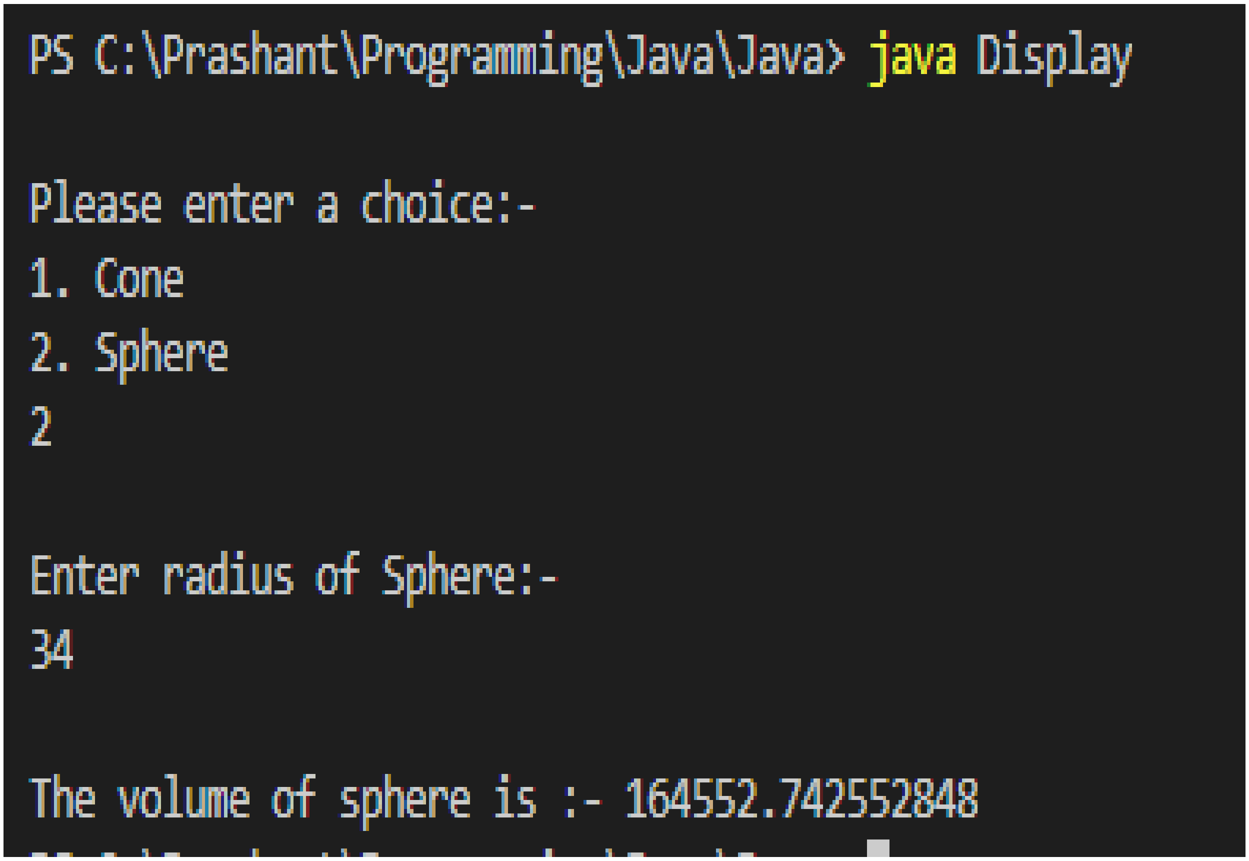
// to display the results

```
    import java.util.Scanner;
    public class Display {

        public static void main(String args[]){
            Scanner n=new Scanner(System.in);
            System.out.println("\nPlease enter a choice:-\n1. Cone\n2.
    Sphere");
            int op= n.nextInt();
        switch(op){
          case 1:{
            Cone c = new Cone();
            c.volume();
            break;

          }
          case 2:{
            Sphere s = new Sphere();
            s.volume();
            break;
          }
          default:System.out.println("Invalid choice entered");
        }
        n.close();
        }
}
```

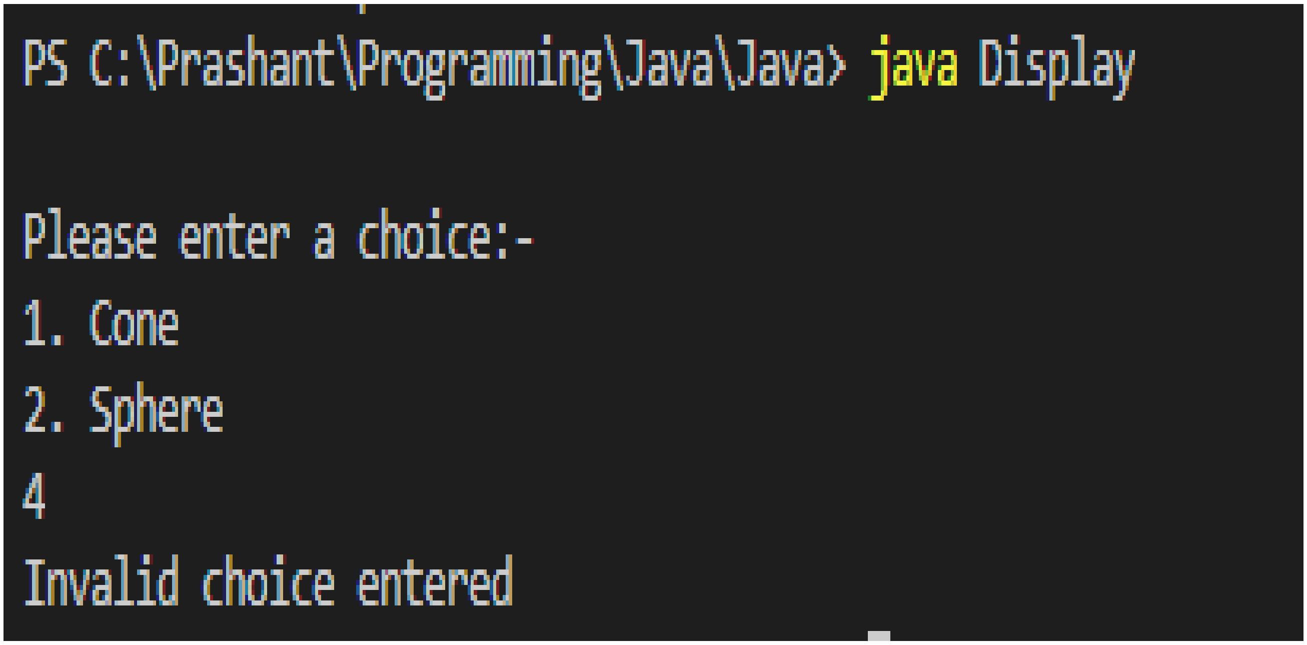
<u>Output:</u>



```
PS C:\Prashant\Programming\Java\Java> java Display

Please enter a choice:-
1. Cone
2. Sphere
1

Enter radius of cone:-
23

Enter height of cone:-
43

Volume of cone is:-  1035.1429818000001
```



```
PS C:\Prashant\Programming\Java\Java> java Display

Please enter a choice:-
1. Cone
2. Sphere
2

Enter radius of Sphere:-
34

The volume of sphere is :- 164552.742552848
```



```
PS C:\Prashant\Programming\Java\Java> java Display

Please enter a choice:-
1. Cone
2. Sphere
4
Invalid choice entered
```

}

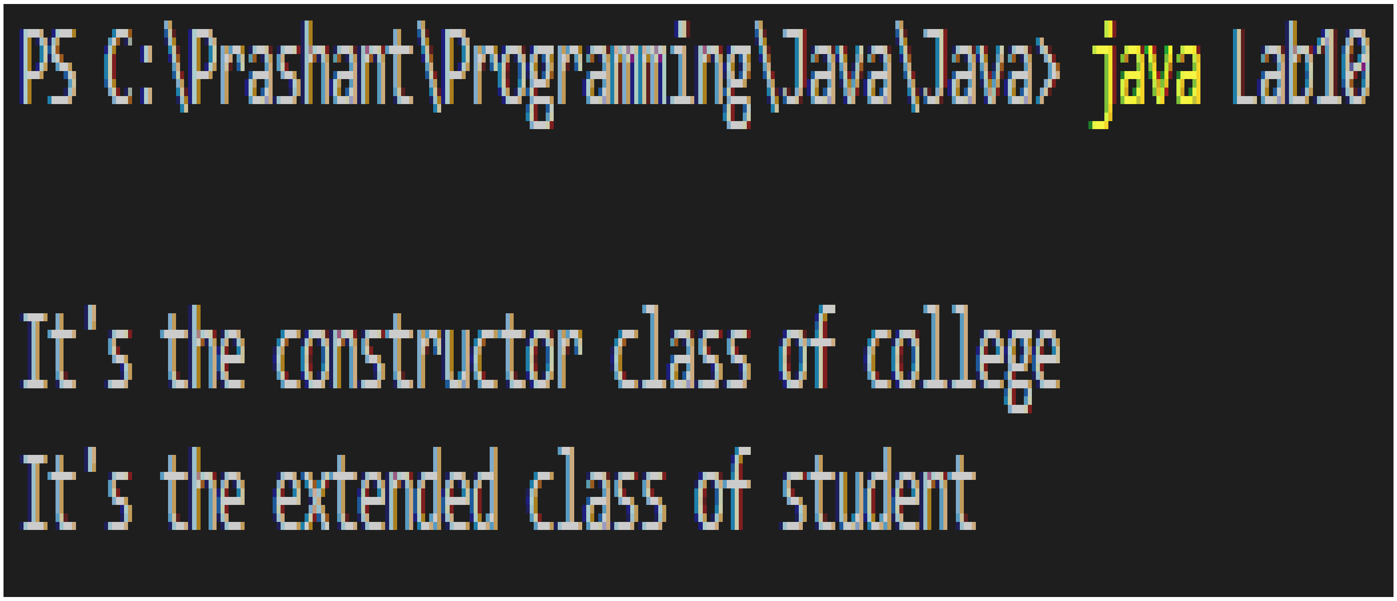
Experiment No. 10

AIM : WAP in which class “A” extends class “B”. Show the order of execution of the constructors if object of class B is created.

Program:

```
class College {

   College(){

      System.out.println("It's the constructor class of college");
   }
}

class Student extends College {

   Student(){
      System.out.println("It's the extended class of student");
   }
}

public class Lab10 {

   public static void main (String args[]){

      Student obj = new Student();

}
```

Output:



```
PS C:\Prashant\Programming\Java\Java> java Lab10


It's the constructor class of college
It's the extended class of student
```

Experiment No. 11

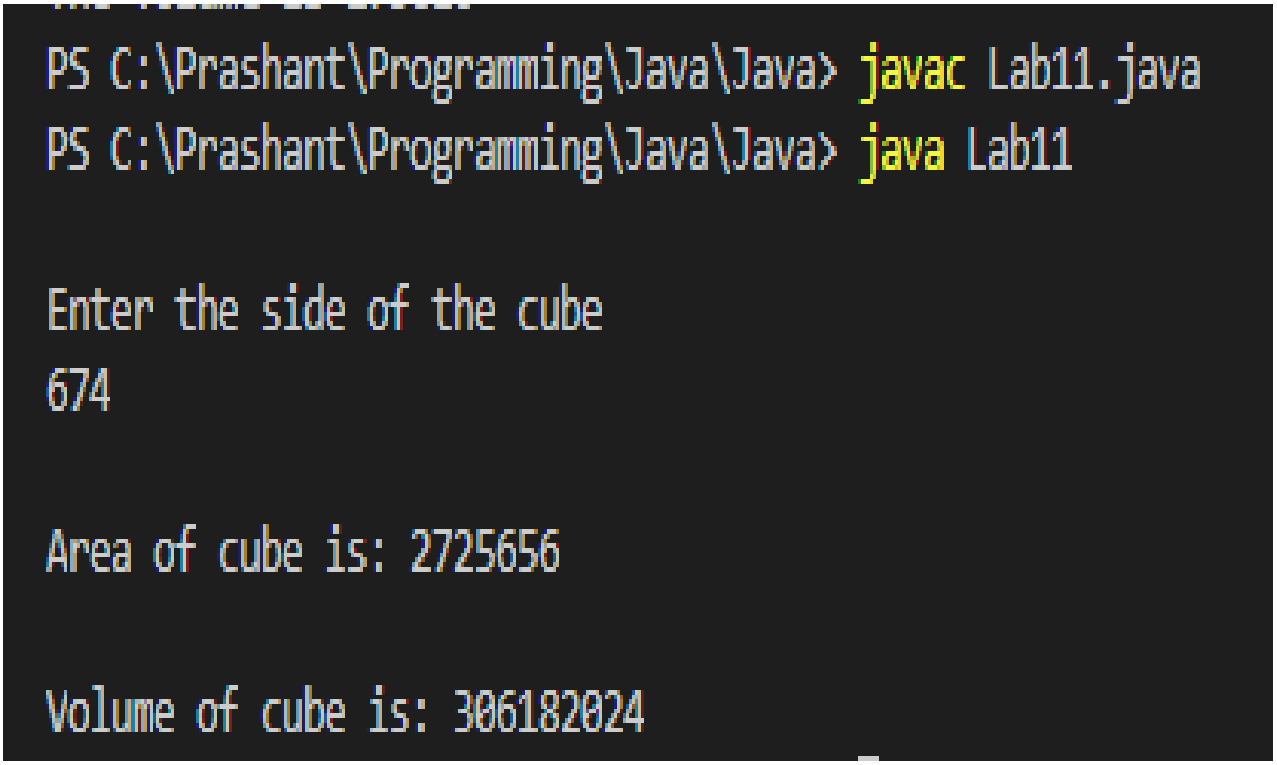
AIM : WAP in which a class inherits two interfaces

Program:

```
import java.util.Scanner;

interface area{
   public void ar(int s);
}
interface volume{
   public void vol(int s);
}



 class Shape implements area, volume {
   Scanner sc = new Scanner(System.in);
      public void ar(int s){

         System.out.println("\nArea of cube is: "+ 6*s*s);

      }
      public void vol(int s){
         System.out.println("\nVolume of cube is: "+ s*s*s);
      }
   }
```

```
        public class Lab11 {
            public static void main(String args[]){
                Scanner sc = new Scanner(System.in);
                System.out.println("\nEnter the side of the cube");
                int side= sc.nextInt();
                Shape sp = new Shape();
                sp.ar(side);
                sp.vol(side);
                sc.close();
            }

}
```

<u>Output:</u>



```
PS C:\Prashant\Programming\Java\Java> javac Lab11.java
PS C:\Prashant\Programming\Java\Java> java Lab11

Enter the side of the cube
674

Area of cube is: 2725656

Volume of cube is: 306182024
```