# Name: Shubhjyot Singh Chadha Sap id: 60009220197

**Branch: Computer Science and Engg(Data Science) Div: D2-3 Course**: **Object Oriented Programming using Java**

Experiment no. 6

**Aim:** To implement Constructors and constructor overloading

**Problem Statement 1:** WAOOP to count the no. of objects created of a class using constructors.

**Code**: shubhjyot@Shubhjyots-MacBook-Pro 60009220197\_D107 % cat Constructor.java

public class Constructor { static int count=0; public Constructor(){

count++;

}

static void display(){

System.out.println("The number of objects are: "+count);

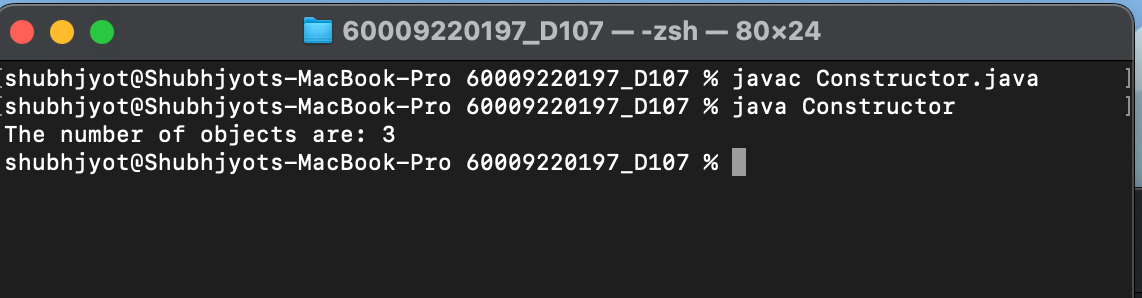
}

public static void main(String[] args) { Constructor obj1=new Constructor(); Constructor obj2=new Constructor(); Constructor obj3=new Constructor(); display();

}

}

# Output:



**Problem Statement 2:** WAP to display area of square and rectangle using the concept of overloaded constructor (use parameterized, non-parameterized and copy constructor).

**Code:** shubhjyot@Shubhjyots-MacBook-Pro 60009220197\_D107 % cat Shape.java

import java.util.\*; public class Shape {

int s; float l,b;

public static void main(String[] args) { Shape s=new Shape();

Shape s1=new Shape(6);

Shape s2=new Shape(2.1f,5.1f); Shape s3=new Shape(s2);

}

Shape(){

Scanner sc=new Scanner(System.in); System.out.println("Enter the side of the square: "); int s=sc.nextInt();

System.out.println("The area of square: "+(s\*s));

}

Shape(int a){ s=a;

System.out.println("The area of square= "+(s\*s));

}

Shape(float x,float y){ l=x;

b=y;

System.out.println("Area of the recttangle= "+(l\*b));

}

Shape(Shape s2){ l=s2.l; b=s2.b;

System.out.println("Area of rectangle= "+(l\*b));

}

}

# Output:

