# **Program -1**

# Write a program to show the concept of classes and objects.

**CODE-**

class Rectangle {

    int l, b;

    void getData(int x, int y) {

        l = x;

        b = y;

    }

    int getArea() {

        int a = l \* b;

        return a;

    }

}

public class MainProgram {

    public static void main(String[] args) {

        int a1, a2;

        Rectangle r1 = new Rectangle();

        Rectangle r2 = new Rectangle();

        r1.l = 20;

        r1.b = 10;

        a1 = r1.getArea();

        r2.getData(15, 5);

        a2 = r2.getArea();

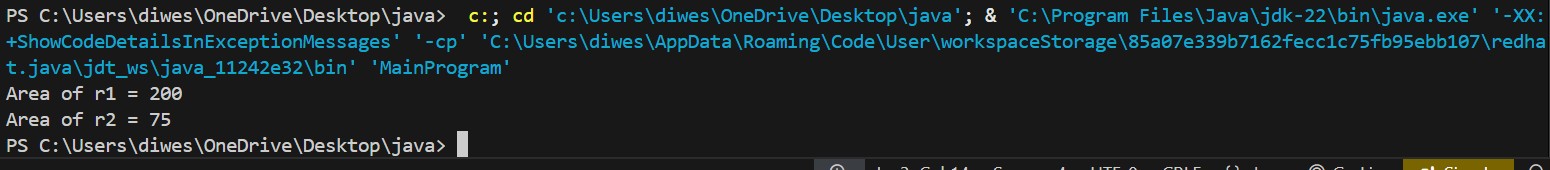
        System.out.println("Area of r1 = " + a1);

        System.out.println("Area of r2 = " + a2);

    }

**}**

**OUTPUT:-**



**PROGRAM-3**

**WAP to show the concept of inheritance and method overriding.**

**CODE-**

class Shape {

int area(int x, int y) {

return x \* y;

}

}

class R2 extends Shape {

int area(int x, int y) {

return x \* y;

}

int area(int x) {

return x \* x;

}

}

class Program4 {

public static void main(String[] args) {

R2 r = new R2();

int z1 = r.area(15, 10);

int z2 = r.area(6);

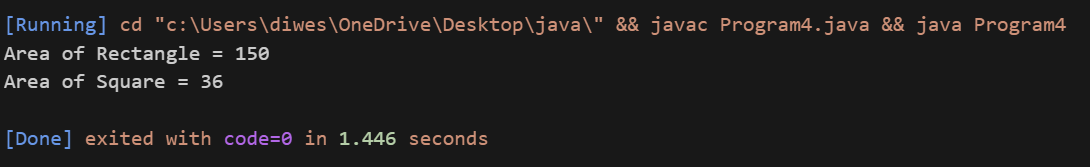
System.out.println("Area of Rectangle = " + z1);

System.out.println("Area of Square = " + z2);

}

}

**OUTPUT:-**

****

**PROGRAM-4**

WAP to implement and create an interface and also show multiple inheritance by interfaces.

CODES:-

class Students {

int rollNumber;

void getNumber(int n) {

rollNumber = n;

}

void putNumber() {

System.out.println("Roll No : " + rollNumber);

}

}

class Test extends Students {

float part1, part2;

void getMarks(float m1, float m2) {

part1 = m1;

part2 = m2;

}

void putMarks() {

System.out.println("Marks obtained:");

System.out.println("part1 = " + part1);

System.out.println("part2 = " + part2);

}

}

interface Sports {

float sm = 7.0F;

void put();

}

class Results extends Test implements Sports {

float total;

public void put() {

System.out.println("Sports Marks = " + sm);

}

void display() {

total = part1 + part2 + sm;

putNumber();

putMarks();

put();

System.out.println("Total score = " + total);

}

}

class Program12 {

public static void main(String[] args) {

Results student1 = new Results();

student1.getNumber(1234);

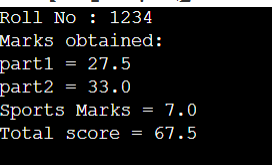
student1.getMarks(27.5F, 33.0F);

student1.display();

}

}

**OUTPUT:-**



**Program-2**

**WAP to create a default & parameterized constructor (constructor overloading)**

**CODE:-**

class R1 {

float l, b;

// Default constructor

R1() {

l = 0;

b = 0;

}

// Parameterized constructor

R1(float X, float Y) {

l = X;

b = Y;

}

R1(float x) {

l = x;

b = x;

}

int area() {

return (int)(l \* b);

}

}

public class Program3 {

public static void main(String[] args) {

int z1, z2;

R1 r = new R1(50, 20);

R1 c = new R1(3);

z1 = r.area();

z2 = c.area();

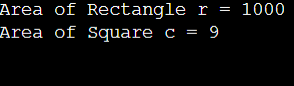
System.out.println("Area of Rectangle r = " + z1);

System.out.println("Area of Square c = " + z2);

}

}

**OUTPUT:-**

****

**PROGRAM-6(A)**

**WAP to get input from the user using Scanner class (util package.**

**CODES-**

 import java.util.Scanner;

 class Addition{

    public static void main(String args[])

    {

        int a,b,c;

        Scanner sc=new Scanner(System.in);

        System.out.println("Enter First Number:");

        a=sc.nextInt();

        System.out.println("Enter  Second Number:");

        b=sc.nextInt();

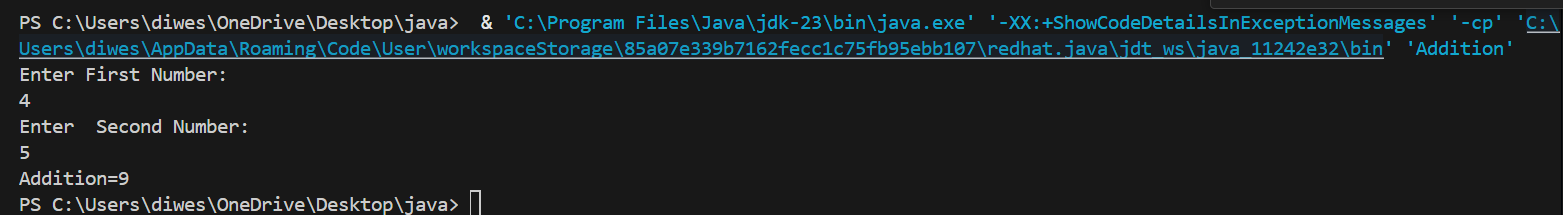
        c=a+b;

        System.out.println("Addition="+c);

    }

 }

**OUTPUT:-**

****