1. Write a program in java to overload add() method to add two integers, two strings and two complex numbers.

class complex

{

    int real;

    int imaginary;

    complex()

    {

        real = imaginary = 0;

    }

    complex(int a, int b)

    {

        real = a;

        imaginary = b;

    }

}

class addition

{

    void add(int a, int b)

    {

        System.out.println("Addition of " + a + " and " + b + " is " + (a + b));

    }

    void add(complex c, complex d)

    {

        complex res = new complex();

        res.real = c.real + d.real;

        res.imaginary = c.imaginary + d.imaginary;

        if(res.imaginary >= 0)

        {

            System.out.println("Result is: " + res.real + " + " + res.imaginary + "i");

        }

        else

        {

            System.out.println("Result is: " + res.real + " - " + Math.abs(res.imaginary) + "i");

        }

    }

    void add(String s1, String s2)

    {

        System.out.println("Concatenated string is: " + (s1 + s2));

    }

}

class testAddition

{

    public static void main(String[] args)

    {

        addition obj = new addition();

        obj.add(Integer.parseInt(args[0]), Integer.parseInt(args[1]));

        obj.add(args[2], args[3]);

        complex c1 = new complex(Integer.parseInt(args[4]), Integer.parseInt(args[5]));

        complex c2 = new complex(Integer.parseInt(args[6]), Integer.parseInt(args[7]));

        obj.add(c1, c2);

    }

}

OUTPUT:

Microsoft Windows [Version 10.0.19045.3324]

(c) Microsoft Corporation. All rights reserved.

C:\Users\Anurag Singh>d:

D:\>cd desktop

D:\Desktop>cd java

D:\Desktop\Java>javac ninth.java

D:\Desktop\Java>java testAddition -4 3 Code Together -7 4 3 -8

Addition of -4 and 3 is -1

Concatenated string is: CodeTogether

Result is: -4 - 4i

1. Write a program in java to overload area() method to compute the area of a circle, rectangle and triangle.

class computeArea

{

    void area(double radius)

    {

        System.out.println("Area of circle is: " + (3.14 \* radius \* radius));

    }

    void area(double length, double breadth)

    {

        System.out.println("Area of Rectangle is: " + (length \* breadth));

    }

    void area(double a, double b, double c)

    {

        double s = (a + b + c)/2.0;

        double areaOfTriangle = Math.sqrt(s \* (s - a) \* (s - b) \* (s - c));

        System.out.println("Area of triangle is: " + areaOfTriangle);

    }

}

class testArea

{

    public static void main(String[] args)

    {

        computeArea obj = new computeArea();

        obj.area(Double.parseDouble(args[0]));

        obj.area(Double.parseDouble(args[1]), Double.parseDouble(args[2]));

        obj.area(Double.parseDouble(args[3]), Double.parseDouble(args[4]), Double.parseDouble(args[5]));

    }

}

OUTPUT:

Microsoft Windows [Version 10.0.19045.3324]

(c) Microsoft Corporation. All rights reserved.

C:\Users\Anurag Singh>d:

D:\>cd desktop

D:\Desktop>cd java

D:\Desktop\Java>javac tenth.java

D:\Desktop\Java>java testArea 3.5 7 5.5 3 4 5

Area of circle is: 38.465

Area of Rectangle is: 38.5

Area of triangle is: 6.0

1. Write a program in java to implement a stack.

class stack

{

    int arr[] = new int[10];

    int top;

    int size;

    stack()

    {

        top = -1;

        size = 10;

    }

    void push(int ele)

    {

        top++;

        if(top == size)

        {

            System.out.println("Stack Overflow.");

            top--;

        }

        else

        {

            arr[top] = ele;

            System.out.println("Pushed element " + ele + " into the stack.");

        }

    }

    void pop()

    {

        if(top == -1)

            System.out.println("Stack Underflow.");

        else

        {

            int temp = arr[top];

            arr[top] = -1;

            top--;

            System.out.println(temp + " popped from the stack.");

        }

    }

    void top()

    {

        if(top == -1)

            System.out.println("Stack is empty.");

        else

            System.out.println(arr[top]);

    }

    void size()

    {

        System.out.println(top + 1);

    }

    void empty()

    {

        if(top == -1)

            System.out.println("true");

        else

            System.out.println("false");

    }

    void display()

    {

        if(top == -1)

            System.out.print("Stack is empty.");

        for(int i = top; i >= 0; i--)

            System.out.print(arr[i] + " ");

        System.out.println();

    }

}

class testStack

{

    public static void main(String[] args)

    {

        stack s = new stack();

        s.push(10);

        s.push(20);

        s.size();

        s.display();

        s.pop();

        s.pop();

        s.pop();

        s.empty();

        s.top();

        s.push(30);

        s.top();

        s.empty();

    }

}

OUTPUT:

Microsoft Windows [Version 10.0.19045.3324]

(c) Microsoft Corporation. All rights reserved.

C:\Users\Anurag Singh>d:

D:\>cd desktop

D:\Desktop>cd java

D:\Desktop\Java>javac eleven.java

D:\Desktop\Java>java testStack

Pushed element 10 into the stack.

Pushed element 20 into the stack.

2

20 10

20 popped from the stack.

10 popped from the stack.

Stack Underflow.

true

Stack is empty.

Pushed element 30 into the stack.

30

false