**Name : Rohan Kamble**
**STD : 2nd Year**
**PRN : 2020420004**
**Practical: Java Program.**

## Practical NO. 1

1) Accept integer values for a, b and c which are coefficients of quadratic equation. Find the solution of the quadratic equation.

```
public class QuadraticEquationSolver
{ public static void main(String[]
args)
{ int a = 2; int
      b = 6; int
      c = 4;
       //Finding out the roots double temp1 =
       Math.sqrt(b * b - 4 * a * c);

       double root1 = (-b +  temp1) / (2*a) ;
       double root2 = (-b -  temp1) / (2*a) ;

      System.out.println("The roots of the Quadratic Equation \"2x2 + 6x + 4 = 0\" are
"+root1+" and "+root2);

   }
}
```

**OUTPUT:**

```
E:\>java QuadraticEquationSolver
The roots of the Quadratic Equation "2x2 + 6x + 4 = 0" are -1.0 and -2.0
```

## Practical No. 2

2) Accept two nxn matrices. Write a Java program to find addition of these matrices.

```
import java.util.Scanner;
public class AddTwoMatrix
{ public static void main(String
args[])
{
int m, n, c, d;
Scanner in = new Scanner(System.in);
```

```
System.out.println("Enter the number of rows and columns of matrix");
m = in.nextInt(); n = in.nextInt();

int first[][] = new int[m][n]; int
second[][] = new int[m][n]; int
sum[][] = new int[m][n];

System.out.println("Enter the elements of first matrix");

for ( c = 0 ; c < m ; c++ ) for ( d = 0 ;
        d < n ; d++ ) first[c][d] =
        in.nextInt();
                System.out.println("Enter the elements of second matrix");
for ( c = 0 ; c < m ; c++ ) for ( d = 0 ; d <
        n ; d++ ) second[c][d] =
        in.nextInt();
for ( c = 0 ; c < m ; c++ ) for ( d = 0 ; d < n ; d++ ) sum[c][d] = first[c][d] +
        second[c][d]; //replace '+' with '-' to subtract
matrices
System.out.println("Sum of entered matrices:-");
for ( c = 0 ; c < m ; c++ )
{
for ( d = 0 ; d < n ; d++ )
System.out.print(sum[c][d]+"\t");
System.out.println();
}
}
}
```

**OUTPUT:**

```
C:\Users\suraj giri\Desktop\Java_practical>java AddTwoMatrix
Enter the number of rows and columns of matrix
3 3
Enter the elements of first matrix
1 2 3
4 3 2
7 6 5
Enter the elements of second matrix
4 3 2
4 5 7
5 6 7
Sum of entered matrices:-
5       5       5
8       8       9
12      12      12
```

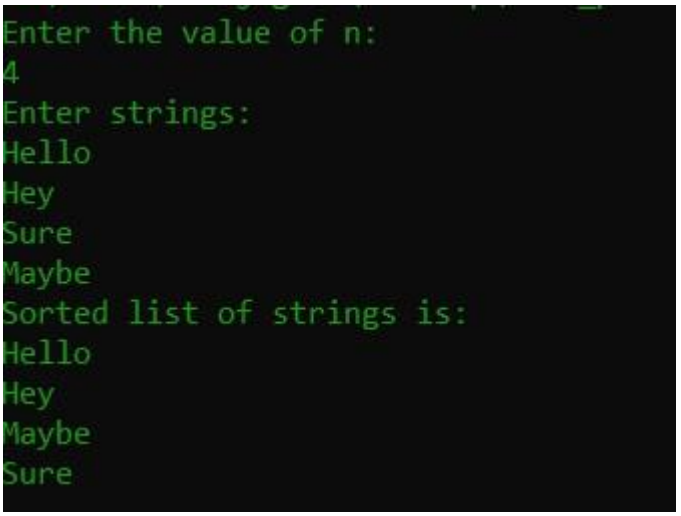
**Practical No. 3**

3) Accept n strings. Sort names in ascending order.

```
import java.util.*;
class Sorting
{ void
sortStrings()
{
Scanner s = new Scanner(System.in);
System.out.println("Enter the value of n: ");
int n = s.nextInt();
String[] str = new String[n];
System.out.println("Enter strings: ");
for(int i = 0; i < n; i++)
{ str[i] = new
String(s.next());
}
for(int i = 0; i < n; i++)
{ for(int j = i+1; j < n;
j++)
{
if(str[i].compareTo(str[j])>0)
{
String temp = str[i];
```

```
str[i] = str[j];
str[j] = temp;
}
}
}
System.out.println("Sorted list of strings is:");
for(int i = 0; i < n ; i++)
{
System.out.println(str[i]);
}
}
}
class Driver
{
public static void main(String args[])
{
Sorting obj = new Sorting();
obj.sortStrings();
}
}
```

**OUTPUT:**




## Practical No. 4

4) Demonstrate method overloading and method overriding in Java.

```
class DisplayOverloading
```

```
{
public void disp(char c)
{
System.out.println(c);
}
public void disp(char c, int num)
{
System.out.println(c + " "+num);
}
public void disp(int num)
{
System.out.println(num);
}
}
class Sample
{ public static void main(String
args[])
{
DisplayOverloading obj = new DisplayOverloading();
obj.disp('a'); obj.disp('a',10); obj.disp(10);
}
}
```

**OUTPUT:**


**Practical No. 5**

5) Demonstrate Java inheritance using extends keyword.

```
class Animal{
void eat(){
System.out.println("eating...");
}
}
class Dog extends Animal{
void bark(){
System.out.println("barking...");
}
}
```

```
class BabyDog extends Dog{
void weep(){
System.out.println("weeping...");
}
}
class TestInheritance2{
public static void main(String args[]){
BabyDog d=new BabyDog();
d.weep();
d.bark();
d.eat();
}}
```

**OUTPUT:**


**Practical No. 8**

6) Demonstrate creating your own exception in Java.

```
class InvalidAgeException extends Exception
{
InvalidAgeException(String s){ super(s);
}
}

class test
{
    static void validate(int age) throws InvalidAgeException
{
    if(age<18)
     throw new InvalidAgeException("not valid");
    else
     System.out.println("welcome to vote");
  } public static void main(String
args[])
{ try{
     vali
     dat
     e(3
     0);
     }
```

```
catch(Exception m)
{
System.out.println("Exception occured: "+m);
}
     System.out.println("rest of the code...");
  }
}
```

**OUTPUT:**

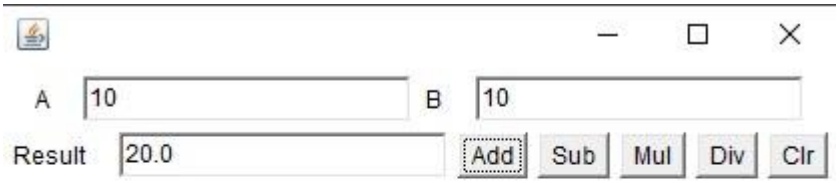

**Practical No. 7**

7) Design a simple calculator GUI application using AWT components.

```
import java.awt.*; import
java.awt.event.ActionEvent; import
java.awt.event.ActionListener; public
class Practical10 extends Frame{

    TextField txtA, txtB, txtResult;
    Button btnAdd, btnSub, btnMul, btnDiv, btnClr;

    Practical10(){ setLayout(new FlowLayout()); txtA
       = new TextField(20); txtB = new TextField(20);
       txtResult = new TextField(20); btnAdd = new
       Button("Add");
       btnAdd.addActionListener((ActionEvent e) -> {
       txtResult.setText("" + (get(txtA) + get(txtB)));
       });
       btnSub = new Button("Sub");
       btnSub.addActionListener((ActionEvent e) -> {
       txtResult.setText("" + (get(txtA) - get(txtB)));
       });
       btnMul = new Button("Mul");
       btnMul.addActionListener((ActionEvent e) -> {
            txtResult.setText("" + (get(txtA) * get(txtB)));
       });
```

```
            btnDiv = new Button("Div");
            btnDiv.addActionListener((ActionEvent e) -> {
            txtResult.setText("" + (get(txtA) / get(txtB)));
            });
            btnClr = new Button("Clr");
            btnClr.addActionListener((ActionEvent e) -> {
                txtA.setText("");
                txtB.setText("");
                txtResult.setText("");
            });
            add(new Label("A"));
            add(txtA); add(new
            Label("B")); add(txtB);
            add(new
            Label("Result"));
            add(txtResult);
            add(btnAdd);
            add(btnSub);
            add(btnMul);
            add(btnDiv); add(btnClr);
            setSize(600,1000);
            setVisible(true);

        } static float get(TextField
        txt){
            return Float.parseFloat(txt.getText());
        } public static void main(String[]
        args) {
            new Practical10();
        }

    }
```

**OUTPUT:**


A
10
B
10
Result
20.0
Add
Sub
Mul
Div
Clr