Index

S.no	Experiment	Date of Allotment	Date of Evaluation	Max Marks	Marks Obtained	Sign
1	Write a program to print a String in the stdout using Java.	11/01/23		1		
2	Write a program to input a String and Display It Using Java	11/01/23		1		
3	Write a program to Calculate Distance Traveled in a given time Using Java.	18/01/23		1		
4	Write a program to Print the given pattern Using Java.	18/01/23		1		
5	Write A Program to compare two given numbers	25/01/23		1		
6	Write A Program to Multiply two given Matrices	25/01/23		1		
7	Code to read an array of votes and classify them as valid or invalid and assign a ballot number	01/02/23		1		
8	Code to create a registry of Staff Members of Different Types	01/02/23		1		
9	Write A Program to input a string and output in alphabetical order	01/02/23		1		
10	Create a Result class that implements a sports interface	08/02/23		1		

S.no	Experiment	Date of Allotment	Date of Evaluation	Max Marks	Marks Obtained	Sign
	and extends a Test class which extends a student class					
11	Given the annual examination Results of 10 students. Write a program to read the data and determine total marks, highest marks in each subject and roll number of the student who scored it.	15/02/23		1		
12	Write a code to throw and catch ArrayIndexOutOfBoundsExcep tion	15/02/23		1		
13	Write A Program to check if two floating point numbers are the same upto 3 decimal places	22/02/23		1		
14	Write A Program to create a temperature catalog of 10 cities using two-dimensional arrays.	22/02/23		1		
15	Write A Program to implement MultiThreading in Java	01/03/23		1		
16	Write A Program to implement catch exceptions while calculating the root of a quadratic equation in Java	22/03/23		1		

S.no	Experiment	Date of Allotment	Date of Evaluation	Max Marks	Marks Obtained	Sign
17	Write A Program to catch multiple exceptions under one Try Block	22/03/23		1		
18	Write A Program to Use Interface to Calculate the Area/Volume of a Rectangle / Cuboid	22/03/23		1		
19	Write A Program to implement event handling using one label and one button	29/03/23		1		
20	Write a method called delete(String str, int m) that returns the input string with the mth element removed	05/04/23		1		
21	WAP to display the sum of factors of a number	05/04/23		1		

EXPERIMENT 1

OBJECTIVE : Print hello world in java.

SOFTWARE USED : IntelliJ IDEA

SOURCE CODE

```
public class hello_world{
        public static void main(String[] args)
        {
        System.out.println("Hello World!");
        }
}
```

OUTPUT :

EXPERIMENT 2

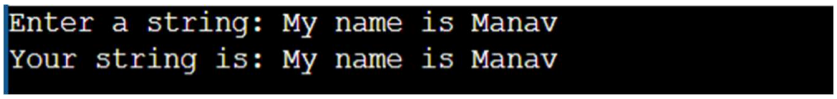
OBJECTIVE : Write a program to enter a string and print it.

SOFTWARE USED : IntelliJ IDEA

SOURCE CODE

```
Import java.util.Scanner;
class printing
{
        public static void main(String args[])
        {
        Scanner sc = new
        Scanner(System.in);
        System.out.print("Enter a
        string: "); String s =
        sc.nextLine();
        System.out.print("Your string
        is: " + s);
        }
}
```

OUTPUT :



EXPERIMENT 3

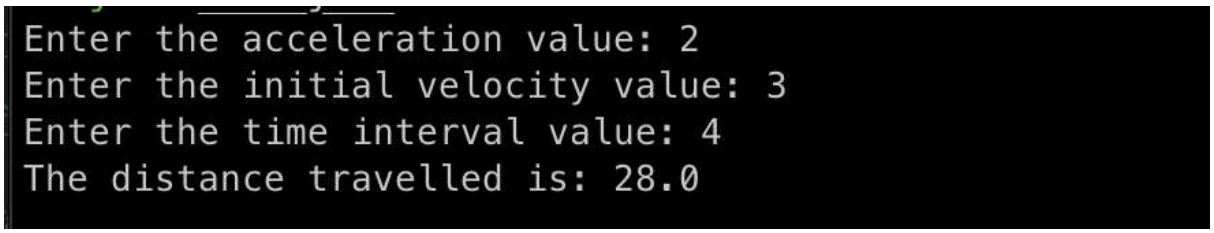
OBJECTIVE : Write a program to calculate the distance travelled by a vehicle at regular intervals of time given the values of acceleration and u. The program should provide the flexibility to the user to select their own time intervals and repeat calculations for different values of a and u.

SOFTWARE USED : IntelliJ IDEA

SOURCE CODE

```
import java.util.Scanner;

public class DistanceCalculator {
        public static void main(String[] args) {
                Scanner input = new Scanner(System.in);

                double acceleration, initialVelocity, timeInterval;
                double distance;

                System.out.print("Enter the acceleration value: ");
                acceleration = input.nextDouble();

                System.out.print("Enter the initial velocity value: ");
                initialVelocity = input.nextDouble();

                System.out.print("Enter the time interval value: ");
                timeInterval = input.nextDouble();

                distance = (initialVelocity * timeInterval) + (0.5 * acceleration
        Math.pow(timeInterval, 2));

                System.out.println("The distance travelled is: " + distance);
        }
}
```

OUTPUT :



EXPERIMENT 4

OBJECTIVE : WAP to print this pattern.

1
2 3
4 5 6
7 8 9 10
11 12 13 14 15

SOFTWARE USED : IntelliJ IDEA

SOURCE CODE

```
public class PatternPrinter {
        public static void main(String[] args) {
                int rows = 5;
                int count = 1;
                for (int i = 0; i < rows; i++) {
                        for (int j = 0; j <= i; j++) {
                                System.out.print(count + " ");
                                count++;
                        }
                        System.out.println();
                }
        }
}
```

OUTPUT :

EXPERIMENT 5

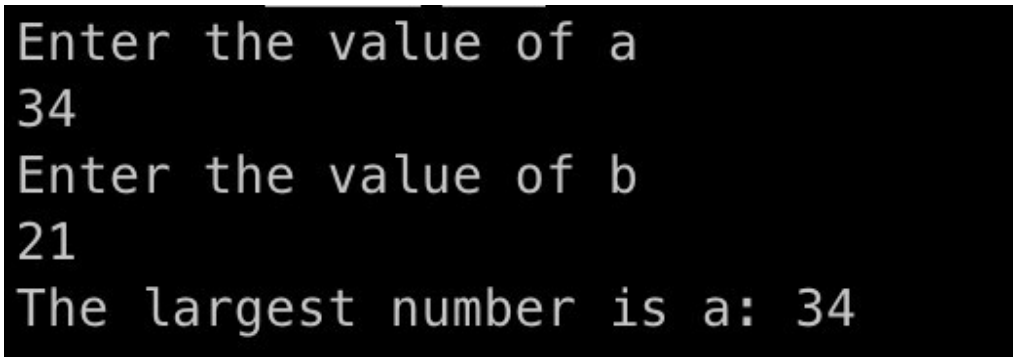
OBJECTIVE : WAP to find the larger number using constructors.

SOFTWARE USED : IntelliJ IDEA

SOURCE CODE

```
import java.util.Scanner;

class Main {
        public static void main(String[] args) {
                Scanner sc = new Scanner(System.in);
                System.out.println("Enter the value of a");
                int a = sc.nextInt();
                System.out.println("Enter the value of b");
                int b = sc.nextInt();

                if (a > b) {
                        System.out.println("The largest number is a: " + a);
                } else if (b > a) {
                        System.out.println("The largest number is b: " + b);
                } else {
                System.out.println("Both the numbers are equal");
                }
        }
}
```

OUTPUT :



EXPERIMENT 6

OBJECTIVE : WAP for matrix multiplication
SOFTWARE USED : IntelliJ IDEA

SOURCE CODE

```
import java.util.*;


class Main
{

    public static void main (String[]args)
     {
       int[][] matrixA = { {1, 2, 3}, {4, 5, 6} };

    int[][] matrixB = { {7, 8}, {9, 10}, {11, 12} };

    int[][] product = new int[2][2];


    for (int i = 0; i < 2; i++)
         {
           for (int j = 0; j < 2; j++)
             {
               product[i][j] = 0;

    for (int k = 0; k < 3; k++)
                 {

    product[i][j] += (matrixA[i][k] * matrixB[k][j]);

    }
    }
    }

    for (int i = 0; i < 2; i++)
         {
           for (int j = 0; j < 2; j++)
             {

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

    }
    System.out.println ();

    }
    }
}
```

EXPERIMENT 7

OBJECTIVE : An election is contested by 5 candidates the candidates are numbered 1-5 and voting is done by marking the candidate number on ballet paper. wap to read ballets and count the votes cast for each candidate using an array variable count. in case a number is outside the range 1-5 the ballet should be considered as spoilt ballet and program should also count the number of spoilt ballots.

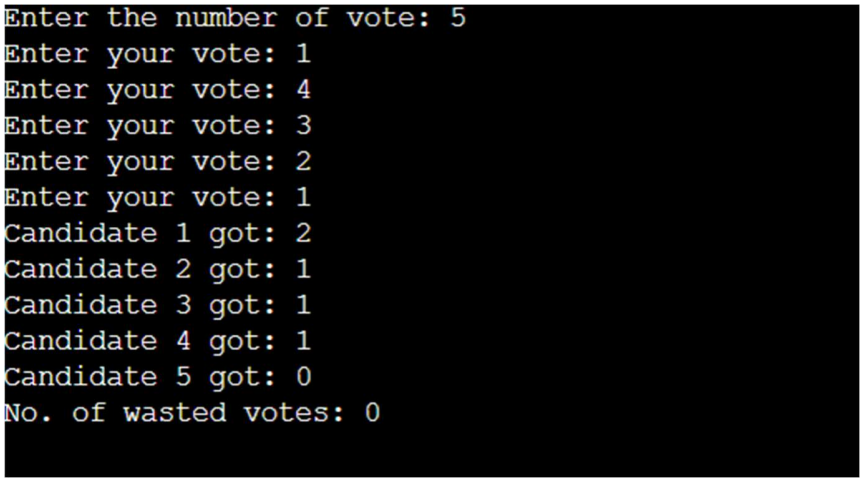
SOFTWARE USED : IntelliJ IDEA

SOURCE CODE

```
import java.util.Scanner;
   public class voters
   {
     public static void main (String[]args)
     {
       Scanner sc = new Scanner (System.in);
         System.out.print ("Enter the number of vote: ");
       int n = sc.nextInt ();
       int v[] = new int[n];
       for (int i = 0; i < n; i++)
         {
           System.out.print ("Enter your vote: ");
           v[i] = sc.nextInt ();
         }

       sc.close ();

       int c1 = 0;
       int c2 = 0;
       int c3 = 0;
       int c4 = 0;
       int c5 = 0;
       for (int i = 0; i < n; i++)
         {
           switch (v[i])
             {
             case 1:
               c1++;
               break;
             case 2:
               c2++;
               break;
             case 3:
               c3++;
               break;
             case 4:
               c4++;
               break;
             case 5:
               c5++;
               break;
             default:
               break;
```

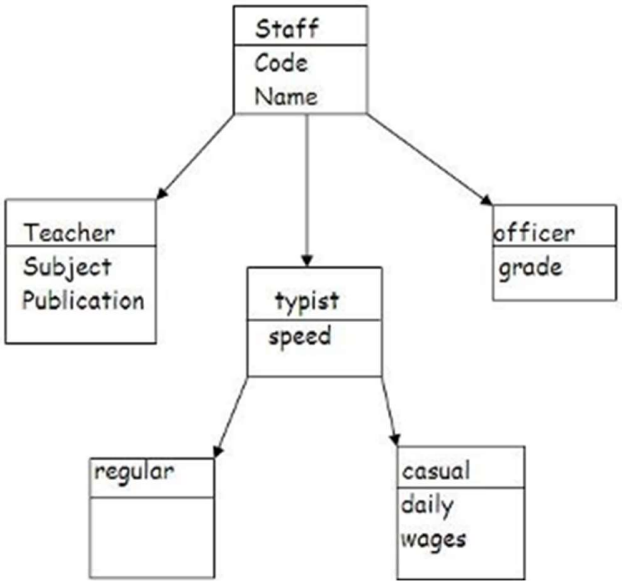
```
        }
      }

      System.out.println ("Candidate 1 got: " + c1);
      System.out.println ("Candidate 2 got: " + c2);
      System.out.println ("Candidate 3 got: " + c3);
      System.out.println ("Candidate 4 got: " + c4);
      System.out.println ("Candidate 5 got: " + c5);
      System.out.println ("No. of wasted votes: " +
                          (n - c1 - c2 - c3 - c4 - c5));
   }
}
```

OUTPUT :



EXPERIMENT 8

OBJECTIVE : WAP to display Staff details using inheritance of classes andinterfaces



SOFTWARE USED : IntelliJ IDEA

SOURCE CODE

```
import java.util.*;

abstract class Staff{
    String code,name;
    Staff(String Code , String Name){
        code = Code;
        name = Name;
    }
    abstract void display();
}

class Teacher extends Staff{
    String subject;
    int publication;
    Teacher(String Code, String Name, String Subject, int Publication){
```

```
        super(Code,Name);
        subject=Subject;
        publication=Publication;
    }
    void display(){
        System.out.println("Code: " + code + "\nName: " + name + "\nSubject: " + subject +
"\nPublication: " + publication);
    }
}

abstract class Typist extends Staff{
    int speed;
    Typist(String Code, String Name, int Speed){
        super(Code,Name);
        speed=Speed;
    }
}

class Regular extends Typist{
    int salary;
    Regular(String Code, String Name, int Speed, int Salary){
        super(Code,Name,Speed);
        salary=Salary;
    }
    void display(){
        System.out.println("Code: " + code + "\nName: " + name + "\nSpeed: " + speed + "\nSalary:
" + salary);
    }
}

class Casual extends Typist{
    int dailywage;
    Casual(String Code, String Name, int Speed, int Dailywage){
        super(Code,Name,Speed);
        dailywage=Dailywage;
    }
    void display(){
        System.out.println("Code: " + code + "\nName: " + name + "\nSpeed: " + speed +
"\nDailywage: " + dailywage);
    }
}

class Officer extends Staff{
    String grade;
```

```
    Officer(String Code, String Name, String Grade){
        super(Code,Name);
        grade=Grade;
    }
    void display(){
        System.out.println("Code: " + code + "\nName: " + name + "\nGrade: " + grade);
    }
}

class staffdb{
    public static void main(String args[]){
        int n;
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter the number of staffs");
        n = sc.nextInt();
        Staff s[] = new Staff[n];
        for(int i=0;i<n;i++){
            System.out.println("Enter the type of staff");
            String type = sc.next();
            if(type.equals("Teacher")){
                System.out.println("Enter the code");
                String code = sc.next();
                System.out.println("Enter the name");
                String name = sc.next();
                System.out.println("Enter the subject");
                String subject = sc.next();
                System.out.println("Enter the publication");
                int publication = sc.nextInt();
                s[i] = new Teacher(code,name,subject,publication);
            }
            else if(type.equals("Regular")){
                System.out.println("Enter the code");
                String code = sc.next();
                System.out.println("Enter the name");
                String name = sc.next();
                System.out.println("Enter the speed");
                int speed = sc.nextInt();
                System.out.println("Enter the salary");
                int salary = sc.nextInt();
                s[i] = new Regular(code,name,speed,salary);
            }
            else if(type.equals("Casual")){
                System.out.println("Enter the code");
                String code = sc.next();
```

```
            System.out.println("Enter the name");
            String name = sc.next();
            System.out.println("Enter the speed");
            int speed = sc.nextInt();
            System.out.println("Enter the dailywage");
            int dailywage = sc.nextInt();
            s[i] = new Casual(code,name,speed,dailywage);
        }
        else if(type.equals("Officer")){
            System.out.println("Enter the code");
            String code = sc.next();
            System.out.println("Enter the name");
            String name = sc.next();
            System.out.println("Enter the grade");
            String grade = sc.next();
            s[i] = new Officer(code,name,grade);
        }
    }
    for(int i=0;i<n;i++){
        s[i].display();
        System.out.println();
    }
  }
}
```

OUTPUT :

```
Enter the number of staffs
2
Enter the type of staff
Teacher
Enter the code
T21
Enter the name
Jmes
Enter the subject
Java
Enter the publication
2
Enter the type of staff
Regular
Enter the code
R123
Enter the name
Reg
Enter the speed
23
Enter the salary
123000
Code: T21
Name: Jmes
Subject: Java
Publication: 2
Code: R123
Name: Reg
Speed: 23
Salary: 123000
```

EXPERIMENT 9

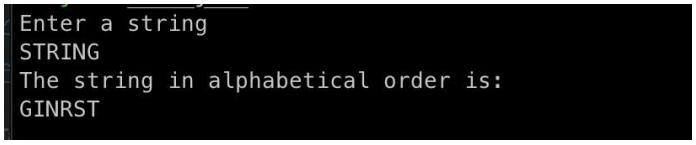
OBJECTIVE : WAP to input a string and output in alphabetical order

SOFTWARE USED : IntelliJ IDEA

SOURCE CODE

```
public class stringrewrite {
	public static void main(String args[]) {
		Scanner sc = new Scanner(System.in);
		System.out.println("Enter a string");
		String str = sc.nextLine();
		char ch[] = str.toCharArray();
		Arrays.sort(ch);
		System.out.println("The string in alphabetical order is: ");
		System.out.println(ch);
	}
}
```

OUTPUT :



EXPERIMENT 10

OBJECTIVE : WAP to display student details using inheritance

SOFTWARE USED : IntelliJ IDEA

SOURCE CODE

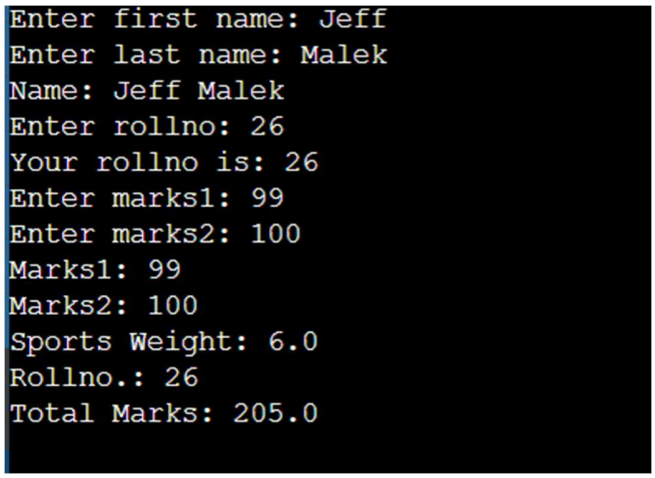
```
import java.util.*;
class Student
{

Scanner sc = new Scanner (System.in);
  int rollno;

void getrollno ()
  {

System.out.print ("Enter rollno: ");
    rollno = sc.nextInt ();

}
void putrollno ()
  {
System.out.println ("Your rollno is: " +
rollno);

}
}
class Details extends Student

{

String firstname;
  String lastname;
  void getname ()
  {

System.out.print ("Enter first name: ");
    firstname = sc.nextLine ();
    System.out.print ("Enter last name:
");
    lastname = sc.nextLine ();

}
```

```
void putname ()
  {

System.out.println ("Name: " +
firstname + " " + lastname);

}
}

class Test extends Student

{

int M1, M2;
  void getmarks ()
  {

System.out.print ("Enter marks1: ");
    M1 = sc.nextInt ();
    System.out.print ("Enter marks2: ");
    M2 = sc.nextInt ();

}
void putmarks ()
  {

System.out.println ("Marks1: " + M1);
    System.out.println ("Marks2: " + M2);

}
}

interface Sports



{

double sportwt = 6.0;
  void putwt ();

}
```

```
class Result extends Test implements
Sports

{

double total;
  public void putwt ()
  {

System.out.println ("Sports Weight: " +
sportwt);

}
void display ()
  {

total = M1 + M2 + sportwt;
    System.out.println ("Rollno.: " +
rollno);
    System.out.println ("Total Marks: " +
total);

}
}

class Exam

{

public static void main (String args[])
  {

Result R1 = new Result ();
    Details D1 = new Details ();
    D1.getname ();

D1.putname ();
    R1.getrollno ();
    R1.putrollno ();
    R1.getmarks ();
    R1.putmarks ();
```

```
    R1.putwt ();

R1.display ();

}
}
```

OUTPUT :



EXPERIMENT 11

OBJECTIVE : Given the annual examination Results of 10 students. Write a program to read the data and determine total marks, highest marks in each subject and roll number of the student who scored it.

SOFTWARE USED : IntelliJ IDEA

SOURCE CODE

```
import java.util.*;

class Results{
        int rollno;
        int marks1,marks2,marks3;
        int total;
        static int maxmarks1,maxmarks2,maxmarks3;
        static int totalmax;
        static int rollmax1,rollmax2,rollmax3,rollmax;
        Results(int rollno,int marks1,int marks2,int marks3){
        this.rollno = rollno;
        this.marks1 = marks1;
        this.marks2 = marks2;
        this.marks3 = marks3;
        total = marks1+marks2+marks3;
        if(marks1>maxmarks1){
        maxmarks1 = marks1;
        rollmax1 = rollno;
        }
        if(marks2>maxmarks2){
        maxmarks2 = marks2;
        rollmax2 = rollno;
        }
        if(marks3>maxmarks3){
        maxmarks3 = marks3;
        rollmax3 = rollno;
        }
        if(total>totalmax){
        totalmax = total;
        rollmax = rollno;
        }
        }
}
```

```
class ExamResults{
	public static void main(String args[]){
	Scanner sc = new Scanner(System.in);
	Results[] r = new Results[10];
	int rollno = 1;
	for(int i=0;i<10;i++){
	System.out.println("Enter the marks of the student "+rollno+" in 3 subjects separated by
spaces");
	int marks1 = sc.nextInt();
	int marks2 = sc.nextInt();
	int marks3 = sc.nextInt();
	r[i] = new Results(rollno,marks1,marks2,marks3);
	rollno++;
	}
	System.out.println("The total marks of each student are: ");
	for(int i=0;i<10;i++){
	System.out.println("The total marks of student "+r[i].rollno+" are "+r[i].total);
	}
	System.out.println("The highest marks in each subject are: ");
	System.out.println("The highest marks in subject 1 are "+Results.maxmarks1+" scored
by student "+Results.rollmax1);
	System.out.println("The highest marks in subject 2 are "+Results.maxmarks2+" scored
by student "+Results.rollmax2);
	System.out.println("The highest marks in subject 3 are "+Results.maxmarks3+" scored
by student "+Results.rollmax3);
	System.out.println("The highest total marks are "+Results.totalmax+" scored by student
"+Results.rollmax);
	}
}
```

OUTPUT :

```
Enter the marks of the student 1 in 3 subjects separated by spaces
1 1 1
Enter the marks of the student 2 in 3 subjects separated by spaces
1 2 2
Enter the marks of the student 3 in 3 subjects separated by spaces
2 3 3
Enter the marks of the student 4 in 3 subjects separated by spaces
3 4 4
Enter the marks of the student 5 in 3 subjects separated by spaces
5 5 5
Enter the marks of the student 6 in 3 subjects separated by spaces
5 6 6
Enter the marks of the student 7 in 3 subjects separated by spaces
6 7 7
Enter the marks of the student 8 in 3 subjects separated by spaces
7 8 8
Enter the marks of the student 9 in 3 subjects separated by spaces
0 0 10
Enter the marks of the student 10 in 3 subjects separated by spaces
2 3 4
The total marks of each student are:
The total marks of student 1 are 3
The total marks of student 2 are 5
The total marks of student 3 are 8
The total marks of student 4 are 11
The total marks of student 5 are 15
The total marks of student 6 are 17
The total marks of student 7 are 20
The total marks of student 8 are 23
The total marks of student 9 are 10
The total marks of student 10 are 9
The highest marks in each subject are:
The highest marks in subject 1 are 7 scored by student 8
The highest marks in subject 2 are 8 scored by student 8
The highest marks in subject 3 are 10 scored by student 9
The highest total marks are 23 scored by student 8
```

EXPERIMENT 12

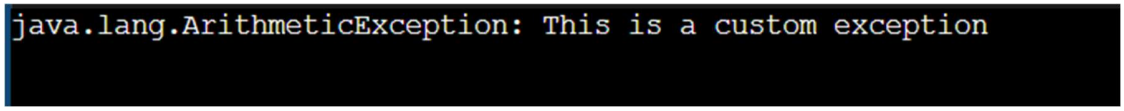
OBJECTIVE : WAP to print the maximum temperature and the city and day on which it was recorded, followed by the minimum temperature and the city and day on which it was recorded

SOFTWARE USED : IntelliJ IDEA

SOURCE CODE

```
import java.lang.*;
public class CustomException {
public static void main(String[] args) { int a=10;
int b=1; int c=10; try{
int d = b*b-4*a*c; if(d<0)
{
throw new ArithmeticException("This is a custom exception");
}
}
catch(ArithmeticException e){ System.out.println(e);
}
}
}
```

OUTPUT :



EXPERIMENT 13

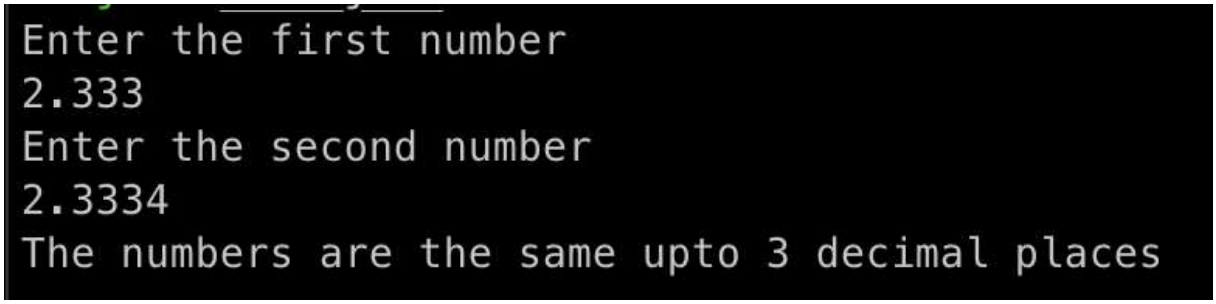
OBJECTIVE : WAP to check if two floating point numbers are the same upto 3 decimal places

SOFTWARE USED : IntelliJ IDEA

SOURCE CODE

```
import java.util.*;
import java.lang.*;


class threeDigitAccuracy
{

public static void main (String args[])
 {

Scanner sc = new Scanner (System.in);
   System.out.println ("Enter the first number");
   double a = sc.nextDouble ();
   System.out.println ("Enter the second number");
   double b = sc.nextDouble ();

if (Math.abs (a - b) < 0.001)
    {

System.out.
            println ("The numbers are the same upto 3 decimal places");


}
   else
    {


places ");
}

System.out.println(" The numbers are not the same upto 3 decimal

}

}
```

OUTPUT :



EXPERIMENT 14

OBJECTIVE : WAP to print the maximum temperature and the city and day on which it was recorded, followed by the minimum temperature and the city and day on which it was recorded

SOFTWARE USED : IntelliJ IDEA

SOURCE CODE

```
class temperatureCatalog{
	public static void main(String args[]){
	Scanner sc = new Scanner(System.in);
	System.out.println("Enter the number of cities");
	int n = 10;
	int Temperature[][] = new int[31][10];
	for(int i=0;i<31;i++){
	for(int j=0;j<n;j++){
		System.out.println("Enter the temperature of city "+(j+1)+" on day "+(i+1));
		Temperature[i][j] = sc.nextInt();
	}
	}
	int max = Temperature[0][0];
	int min = Temperature[0][0];
	int maxcity = 0;
	int mincity = 0;
	int maxday = 0;
	int minday = 0;
	for(int i=0;i<31;i++){
	for(int j=0;j<n;j++){
		if(Temperature[i][j]>max){
		max = Temperature[i][j];
		maxcity = j;
		maxday = i;
		}
		if(Temperature[i][j]<min){
		min = Temperature[i][j];
		mincity = j;
		minday = i;
		}
	}
	}
	System.out.println("The maximum temperature is "+max+" on day "+(maxday+1)+" in
city "+(maxcity+1));
```

```
        System.out.println("The minimum temperature is "+min+" on day "+(minday+1)+" in city "+(mincity+1));

        }
}
```

OUTPUT :

```
Enter the number of cities
3
Enter the temperature of city 1 on day 1
35
Enter the temperature of city 2 on day 1
32
Enter the temperature of city 3 on day 1
30
Enter the temperature of city 1 on day 2
33
Enter the temperature of city 2 on day 2
34
Enter the temperature of city 3 on day 2
31
Enter the temperature of city 1 on day 3
29
Enter the temperature of city 2 on day 3
30
Enter the temperature of city 3 on day 3
35
...
The maximum temperature is 37 on day 2 in city 2
The minimum temperature is 24 on day 4 in city 1
```

EXPERIMENT 15

OBJECTIVE : WAP to demonstrate multi-threading with sleep and interrupt methods.

SOFTWARE USED : IntelliJ IDEA

SOURCE CODE

```
import java.util.*;

class Alpha extends Thread{
    int Sec;
    Alpha(int sec){
        Sec = sec;
    }
    public void run(){
        try{
            Thread.sleep(Sec);
            System.out.println("Thread Type 1 finished after " + Sec + " milliseconds");
        }catch(InterruptedException e){
            System.out.println("Thread type 1 interrupted");
        }
    }
}

class Beta extends Thread{
    public void run(){
        try{
            Thread.sleep(2000);
            System.out.println("Thread Type 2 finished after 2000 milliseconds");
        }catch(InterruptedException e){
            System.out.println("Thread Type 2 interrupted");
        }
    }
}

class MultiThreading{
    public static void main(String args[]){
        Alpha t1 = new Alpha(5000);
        Beta t2 = new Beta();
        Alpha t3 = new Alpha(10000);
        t1.start();
        t2.start();
        t3.start();
```

EXPERIMENT 16

OBJECTIVE : WAP for Quadratic Exception.

SOFTWARE USED : IntelliJ IDEA

SOURCE CODE

```
import java.util.*;


class QuadraticEquationSolver
{

public static void main (String args[])
  {
    Scanner input = new Scanner (System.in);
    int coefficientA = input.nextInt ();

int coefficientB = input.nextInt ();
    int coefficientC = input.nextInt ();


double discriminantSquare =
      (coefficientB * coefficientB) - (4 * coefficientA * coefficientC);


try
    {

if (discriminantSquare < 0)
           {

throw new ArithmeticException ("Equation gives imaginary roots!");

}

}
    catch (ArithmeticException exception)
    {
      System.out.println (discriminantSquare);

}

input.close ();

}

}
```

EXPERIMENT 17

OBJECTIVE : WAP to throw and catch NullPointerException and ArrayIndexOutOfBoundsException.

SOFTWARE USED : IntelliJ IDEA

SOURCE CODE

```
import java.util.Scanner;

class ExceptionHandling{
        public static void main(String args[]){
                Scanner sc = new Scanner(System.in);
                int[] a = new int[5];
                for(int i=0;i<5;i++){
                try{
                        System.out.println("Enter the index of the array");
                        int index = sc.nextInt();
                        System.out.println("Enter the value to be stored at index "+index);
                        int value = sc.nextInt();
                        a[index] = value;
                }
                catch(ArrayIndexOutOfBoundsException e){
                        System.out.println("The index is out of bounds");
                        i--;
                }
                }
                for(int i=0;i<5;i++){
                System.out.print(a[i] + " ");
                }
        }
}
```

OUTPUT :

```
Enter the index of the array
2
Enter the value to be stored at index 2
12
Enter the index of the array
6
Enter the value to be stored at index 6
13
The index is out of bounds
```

EXPERIMENT 18

OBJECTIVE : Write a method to Use Interface to Calculate the Area/Volume of a Rectangle / Cuboid

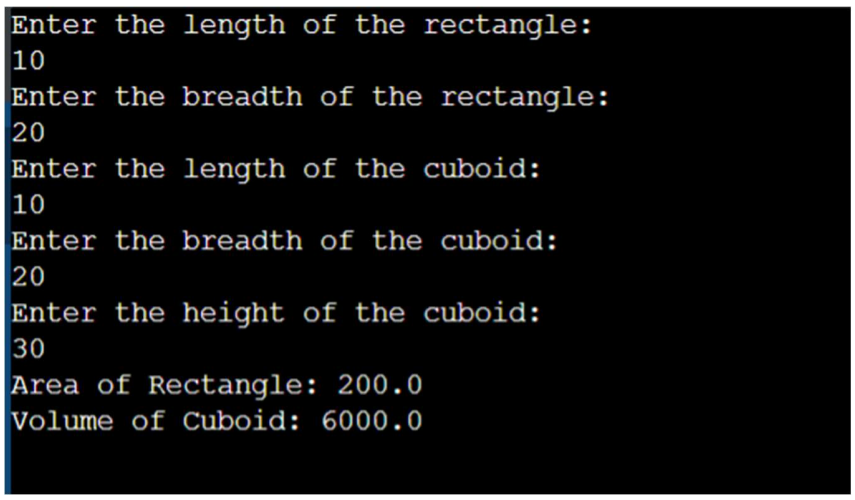
SOFTWARE USED : IntelliJ IDEA

SOURCE CODE

```
 import java.util.*;
interface Shape
{

void calculate ();

}

class Rectangle implements Shape
{
  double dim1, dim2;


Rectangle ()
  {

Scanner sc = new Scanner (System.in);
    System.out.println ("Enter the length of the rectangle: ");
    dim1 = sc.nextDouble ();

System.out.println ("Enter the breadth of the rectangle: ");
    dim2 = sc.nextDouble ();

}

public void calculate ()
  {

System.out.println ("Area of Rectangle: " + dim1 * dim2);

}
}

class Cuboid implements Shape

{
  double dim1, dim2, dim3;


Cuboid ()
  {

Scanner sc = new Scanner (System.in);
    System.out.println ("Enter the length of the cuboid:");
    dim1 = sc.nextDouble ();

System.out.println ("Enter the breadth of the cuboid: ");
```

```
    dim2 = sc.nextDouble ();

System.out.println ("Enter the height of the cuboid:");
    dim3 = sc.nextDouble ();

}

public void calculate ()
  {

System.out.println ("Volume of Cuboid: " + dim1 * dim2 * dim3);

}
}

class geometry

{

public static void main (String args[])
  {
    Rectangle r = new Rectangle ();
    Cuboid c = new Cuboid ();
    r.calculate ();

c.calculate ();

}
}
```

OUTPUT :



EXPERIMENT 19

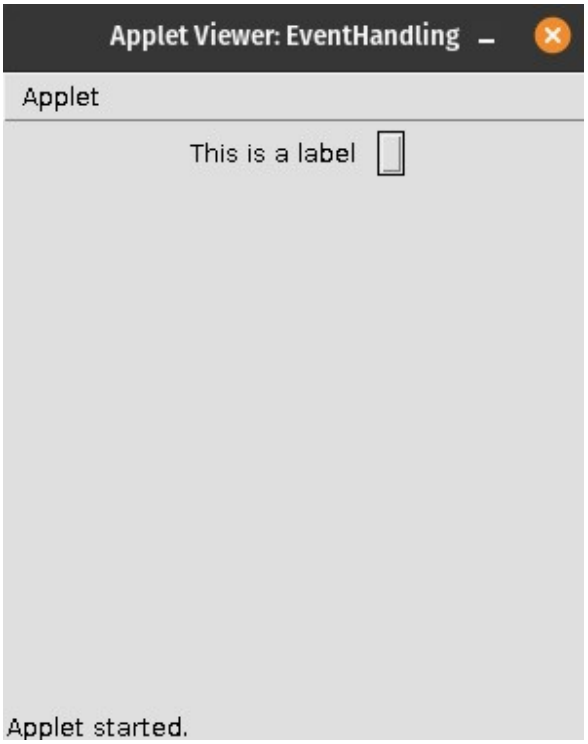
OBJECTIVE : Write A Program to implement event handling using one label and one button

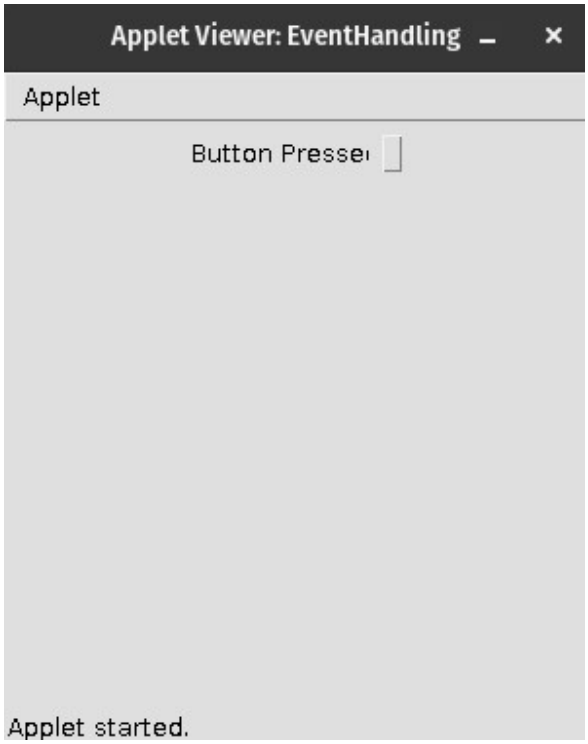
SOFTWARE USED : IntelliJ IDEA

SOURCE CODE

```
import java.awt.*;
import java.awt.event.*;
import java.applet.*;


public class EventHandling extends Applet implements ActionListener
{


  Label l;
  Button b;

public void init ()
  {

    l = new Label ("This is a label");
    b = new Button ();

    add (l);

    add (b);
    b.addActionListener (this);

}
public void actionPerformed (ActionEvent e)
  {
    l.setText ("Button Pressed");

}
}

/*
<applet code="EventHandling" width=300 height=300>
</applet>
*/
```

OUTPUT :


Applet Viewer: EventHandling
Applet
This is a label
Applet started.


Applet Viewer: EventHandling
Applet
Button Presse
Applet started.

EXPERIMENT 20

OBJECTIVE : Write a method called delete that returns the input string with the mth element removed.
SOFTWARE USED : IntelliJ IDEA

SOURCE CODE

```
import java.awt.*;

import java.util.*;

public class Main
{

public static String delete (String str, int m)
  {

String s2 = "";

for (int i = 0; i < str.length (); i++)

     {

if (i != m - 1)

            {

s2 = s2 + str.charAt (i);

}

}

return (s2);

}

public static void main (String[]args)
  {
    Scanner sc = new Scanner (System.in);
    System.out.println ("Enter the String: ");
    String s = sc.nextLine ();
    System.out.println ("Enter the index value");
    int m = sc.nextInt ();
    System.out.println (delete (s, m));

}
}
```