20MCA132 – OBJECT ORIENTED PROGRAMMING LAB

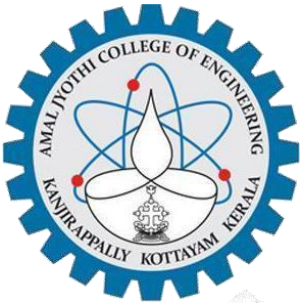
Lab Report Submitted By

SREELAKSHMI R

Reg. No.: AJC21MCA-2101

In Partial fulfillment for the Award of the Degree Of

MASTER OF COMPUTER APPLICATIONS (2 Year)
(MCA) APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY



AMAL JYOTHI COLLEGE OF ENGINEERING KANJIRAPPALLY

[Affiliated to APJ Abdul Kalam Technological University, Kerala. Approved by AICTE, Accredited by NAAC with 'A' grade. Koovappally, Kanjirappally, Kottayam, Kerala – 686518]

2021-2022

DEPARTMENT OF COMPUTER APPLICATIONS

AMAL JYOTHI COLLEGE OF ENGINEERING

KANJIRAPPALLY

CERTIFICATE

This is to certify that the Lab report, **"20MCA132 OBJECT ORIENTED PROGRAMMING LAB"** is the bonafide work of **SREELAKSHMI R (Reg.No: AJC21MCA-2101)** in partial fulfillment of the requirements for the award of the Degree of Master of Computer Applications under APJ Abdul Kalam Technological University during the year 2021-22.

Sr. Elsin Chakkalackal S.H — **Lab In-Charge**

Rev.Fr.Dr.Rubin Thottupurathu Jose — **Head of the Department**

Internal Examiner

External Examiner

CONTENT

Object Oriented Programming LAB

Name: Sreelakshmi R

Roll No:41

Batch:RMCA S2B

Date:26-03-2022

Experiment No.: 1

Aim

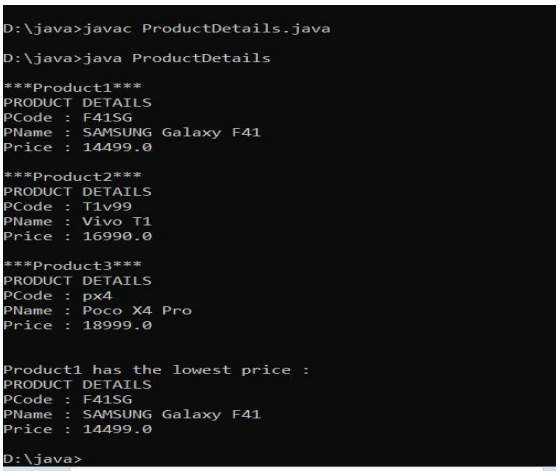
Define a class product with data member pcode,pname and price .Create three objects usin the class and find the product having lowest price.

Sourcecode

```
class Product{

        String pcode,
        pname;
        double price;
        void details(){
                System.out.println("PRODUCT DETAILS");

                System.out.println("PCode : "+pcode);

                System.out.println("PName : "+pname);

                System.out.println("Price : "+price);

        }

}

public class ProductDetails{

        public static void main(String args[]){

                Product p1 = new Product();
                p1.pcode = "F41SG";
                p1.pname = "SAMSUNG GalaxyF41";
                p1.price = 14499;
                System.out.println("\n***Product1***");
                p1.details();



                Product p2 = new Product();

                p2.pcode = "T1v99";
                p2.pname = "Vivo T1";
                p2.price = 16990;
                System.out.println("\n***Product2***");
                p2.details();
```

```
        Product p3 = new Product();
        p3.pcode =  "px4";
        p3.pname = "Poco X4 Pro";
        p3.price = 18999;
        System.out.println("\n***Product3***");
        p3.details();


        if(p1.price<p2.price && p1.price<p3.price){

            System.out.println("\n\nProduct1 has the lowest price :");

            p1.details();

        }

        else if(p2.price < p3.price){

            System.out.println("\nProduct2 has the lowest price:\n");
            p2.details();
        }

        else

        {

            System.out.println("\nProduct3 has the lowest price:\n");

            p3.details();

        }

    }

}
```

OutputScreenshot



Object Oriented Programming LAB

Name: SREELAKSHMI R

Roll No:41

Batch:RMCA S2B

Date:06-04-2022

Experiment No.: 2

Aim

Matrix Addition

Sourcecode

```
import  java.util.*;
 class MatrixAddition{
          public static void main(String[] args){
                    int row, col;
                    Scanner sc= new Scanner(System.in);
                    System.out.print("Enter the number of rows for the Matrices : ");
                   row= sc.nextInt();
                   System.out.print("Enter the number of columns for the Matrices :");
                   col= sc.nextInt();
                    int[][] matrixA= new int[row][col];
                   int[][] matrixB= new int[row][col];
                   int[][] matrixSum= new int[row][col];
                    System.out.println("Enter the elements for the Matrix A : ");
         for(int i=0;i<row;i++){
                    for(int j=0;j<col;j++){
                                        matrixA[i][j]= sc.nextInt();
                              }
                   }
                   System.out.println("\n");

                   System.out.println("Enter the elements for the Matrix B : ");
for(int i=0;i<row;i++){
                              for(int j=0;j<col;j++){
         matrixB[i][j]= sc.nextInt();
                              }
                   }
```

```
        System.out.println("\n");

        System.out.println("Matrix A is : ");
for(int i=0;i<row;i++){
        for(int j=0;j<col;j++){
                System.out.print(matrixA[i][j]+"  ");
        }
        System.out.println("\n");
        }

        System.out.println("Matrix B is : ");
        for(int i=0;i<row;i++){
        for(int j=0;j<col;j++){
                System.out.print(matrixB[i][j]+"  ");
        }
        System.out.println("\n");
        }

        for(int i=0;i<row;i++){
                for(int j=0;j<col;j++){
                matrixSum[i][j]= matrixA[i][j] + matrixB[i][j];
        }
        }

        System.out.println("Resultant of the Matrix Addition is :");
        for(int i=0;i<row;i++){

        for(int j=0;j<col;j++){
                System.out.print(matrixSum[i][j]+"  ");

        }
        System.out.println("\n");
```

```
                }
        }
}
```

OutputScreenshot

Object Oriented Programming LAB

Name: SREELAKSHMI R

Roll No:41

Batch:RMCA S2B

Date:06-04-2022

Experiment No.: 3

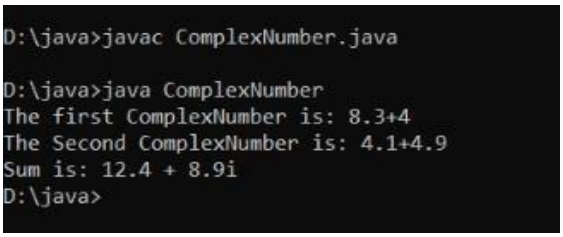
Aim

Complex Number

Sourcecode

```
public class ComplexNumber{
        double real, img;
        ComplexNumber(double r, double i){
        this.real = r;
        this.img = i;
}
 public static ComplexNumber sum(ComplexNumber c1, ComplexNumber c2){
 ComplexNumber temp = new ComplexNumber(0, 0);
                  temp.real = c1.real + c2.real;
                 temp.img = c1.img + c2.img;
                 return temp;
                }
public static void main(String args[]) {
                 System.out.println("The first ComplexNumber is: 8.3+4");
                 ComplexNumber c1 = new ComplexNumber(8.3, 4);
                 System.out.println("The Second ComplexNumber is: 4.1+4.9");
                 ComplexNumber c2 = new ComplexNumber(4.1,4.9);
                 ComplexNumber temp = sum(c1, c2);
                 System.out.printf("Sum is: "+ temp.real+" + "+ temp.img +"i");
        }
}
```

OutputScreenshot



OBJECT ORIENTED PROGRAMMING LAB

Name: SREELAKSHMI R

Roll No:41

Batch:MCA

Date:06-04-22

Experiment No.: 4

Aim

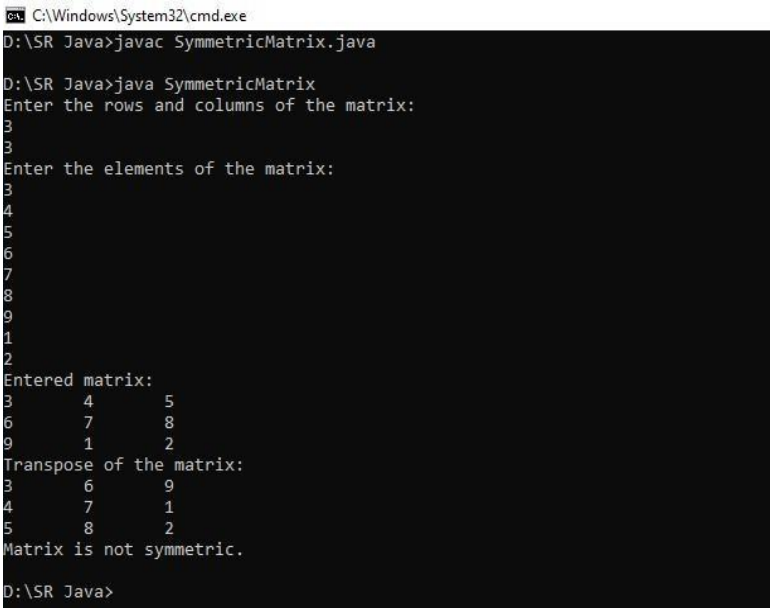
Read a matrix from the console and check whether it is symmetric or not.

Sourcecode

```
import  java.util.Scanner;
public class SymmetricMatrix{
public void Display(int [][]
arr,int row,int col){
        for(int i=0;i<row;i++){
          for(int j=0;j<col;j++){
                        System.out.print(arr[i][j]+"\t");
        }
        System.out.println();
      }
}
   public static void main(String[] args){
            int [][] mat = new int[3][3];
            int [][] trans=new int[3][3];
            int row,col;

   SymmetricMatrix obj=new SymmetricMatrix();
   Scanner s=new Scanner(System.in);
   System.out.println("Enter the rows and columns of the matrix:");
                row=s.nextInt();
                col=s.nextInt();
    System.out.println("Enter the elements of the matrix:");
        for(int i=0;i<row;i++) {
                for(int j=0;j<col;j++) {
                        mat[i][j]=s.nextInt();
```

```
                    }
        }
        for(int i=0;i<row;i++) {
                for(int j=0;j<col;j++)
           {
               trans[j][i]=mat[i][j];
            }
         }
      System.out.println("Entered matrix:");
      obj.Display(mat,row,col);
      System.out.println("Transpose of the matrix:");
      obj.Display(trans,row,col);
          for(int i=0;i<row;i++){
                      for(int j=0;j<col;j++){
                            if(mat[i][j]!=trans[i][j]){
                System.out.println("Matrix is not symmetric.");
                System.exit(0);
              }
           }
        }
        System.out.println("The given matrix is symmetric.");
    }
}
```

OutputScreenshot



OBJECT ORIENTED PROGRAMMING LAB

Name: SREELAKSHMI R

Roll No: 41

Batch: MCAS2B

Date: 31/05/22

Experiment No.: 5

Aim

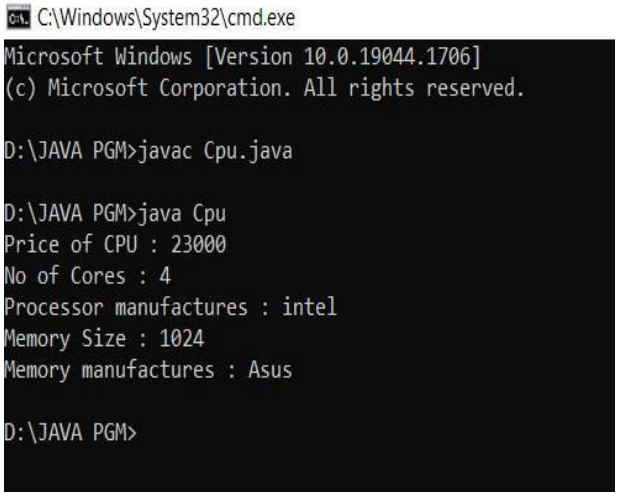
Create CPU with attribute price. Create inner class Processor (no. of cores, manufacturer) and static nested class RAM (memory, manufacturer). Create an object of CPU and print information of Processor and RAM.

Sourcecode

```
class Cpu {

       int price;

 Cpu(int p) {

    this.price = p;

}

   class Processor {

       int cores;
       String manufacture;

  Processor(int n, String m) {

       this.cores = n;
        this.manufacture = m;

       }

       void display() {
          System.out.println("No of Cores : " + this.cores);
          System.out.println("Processor manufactures : " + this.manufacture);
       }
   }

          static class Ram
{
```

```
int memory;
String manufacture;

Ram(int n, String m) {

        this.memory = n;
         this.manufacture = m;

      }

      void display() {

         System.out.println("Memory Size : " + this.memory);
         System.out.println("Memory manufactures : " + this.manufacture);

      }
   }

 void display() {
      System.out.println("Price of CPU : " + this.price);
   }

   public static void main(String[] args) {
      Cpu intel = new Cpu(23000);
      Cpu.Processor i_processor = intel.new Processor(4, "intel");
      Cpu.Ram i_ram = new Ram(1024, "Asus");
      intel.display();
      i_processor.display();
      i_ram.display();
   }
}
```

OutputScreenshot



OBJECT ORIENTED PROGRAMMING LAB

Experiment No.: 6

Name: SREELAKSHMI R

Roll No:41

Batch:RMCA S2

Date:22-04-2022

Aim

Program to Sort strings

Sourcecode

```
import java.util.*;
public class SortString{
        static Scanner s=new
        Scanner(System.in);
        public static void main(String
args[]){
                String temp;
                String[] A=new String[8];
                int a;
 System.out.println("Enter the size of the array : ");
  a=s.nextInt();
System.out.println("Enter the strings into the array : ");
                for(int i=0;i<=a;i++){
                        A[i]=s.nextLine();

}

System.out.println("Sorted  array elements : ");
                for(int i=0;i<=a;i++){
                for(int j=i+1;j<=a;j++)
           if(A[i].compareTo(A[j])>0){
                temp=A[i];
                A[i]=A[j];
                  A[j]=temp;
                                }
                        }
```

```
  for(int i=0;i<=a;i++){
          System.out.println(A[i]);
                  }
          }
}
```

OutputScreenshot

OBJECT ORIENTED PROGRAMMING LAB

Name: SREELAKSHMI R

Roll No:41

Batch: RMCA S2

Date:22-04-2022

Experiment No.: 7

Aim

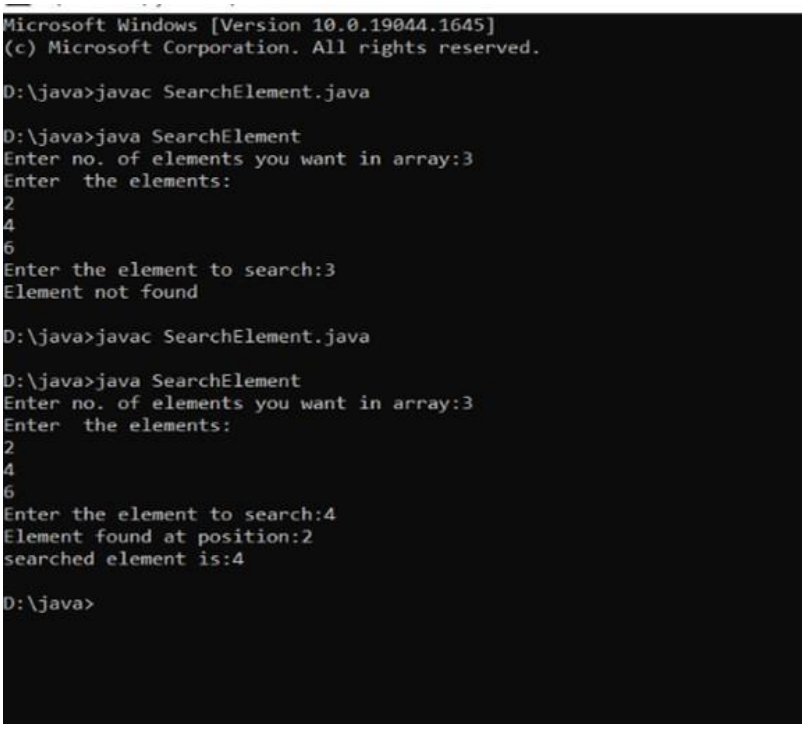
Search an element in an array.

Sourcecode

```
 import java.util.Scanner;
public class SearchElement {
public static void main(String[] args)
   {
      int n, element, flag = 0, i = 0;
      Scanner s = new Scanner(System.in);
      System.out.print("Enter no. of elements you want in array:");
       n = s.nextInt();
          int a[] = new int[n];
      System.out.println("Enter  the elements:");
   for(i = 0; i < n; i++) {
           a[i] = s.nextInt();
      }
      System.out.print("Enter the element to search:");
     element = s.nextInt();
           for(i = 0; i < n; i++) {
                    if(a[i] == element) {
                          flag = 1;
                          break;
          }
           else {

          flag = 0;
          }
    }
        if(flag ==1)
```

```
        {
            System.out.println("Element found at position:"+(i + 1)+""+"\n"+"searched
            element is:"+element);
        }
else
        {
            System.out.println("Element not found");
        }
    }
}
```

OutputScreenshot



OBJECT ORIENTED PROGRAMMING LAB

Name: Sreelekshmi R

Roll No: 41

Batch: S2 MCA

Date:22-04-2022

Experiment No.: 8

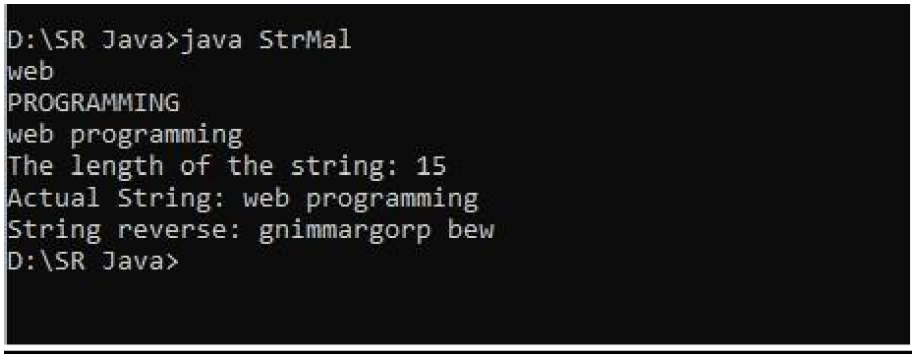
Aim

Perform string manipulations

Sourcecode

```
public class StrMal {
        public static void main(String[] args) {
            String str1 = "web";
             String str2 = "programming";
             String str3 = str1.toLowerCase();
             String str4 = str2.toUpperCase();

             System.out.println(str3);
             System.out.println(str4);

             String result = str1+" "+str2;
             System.out.println(result);

                System.out.println("The length of the string: "+ result.length());
            System.out.println("Actual String: "+result);
            System.out.print("String reverse: ");
            for(int i = result.length()-1; i>=0; i--){
                                System.out.print(result.charAt(i));
                        }
            }
    }
```

OutputScreenshot



OBJECT ORIENTED PROGRAMMING LAB

Name:Sreelakshmi R

Roll No:41

Batch:S2 RMCA B

Date:23/05/2022

Experiment No.: 9

Aim

Program to create a class for Employee having attributes eNo, eName eSalary. Read n employ information and Search for an employee given eNo, using the concept of Array of Objects

Sourcecode

```
import java.util.*;
public class Employee1
{
 Scanner sc=new
Scanner(System.in);
int eNo;
String eName;
 int eSal;
 void read_emp()
 {
 System.out.println("\n Enter employee number:");
 eNo=sc.nextInt();
  sc.nextLine();
 System.out.println("\n Enter employee name: ");
 eName=new String(sc.nextLine());
 System.out.println("\n Enter salary of employee: ");
 eSal=sc.nextInt();
 }
 void print_emp()
 {
 System.out.println("\n Employee Information\n");
 System.out.println("\n EMPLOYEE NO. : "+eNo);
 System.out.println("\n EMPLOYEE NAME: "+eName);
```

```
 System.out.println("\n SALARY : "+eSal);
 }
 public static void main(String[] args) {
                    int n,i,key;
                    int opt;
                    int flag=0;
 Scanner sc=new Scanner(System.in);
 System.out.println("\n How many records you have to save\n");
  n=sc.nextInt();
 Employee1 ob[]=new Employee1[n];
for(i=0;i<n;i++)  {
 ob[i]=new Employee1();
 System.out.println("\n ENTER DETAILS OF EMPLOYEE "+(i+1));
ob[i].read_emp();
 }
do
 {
 System.out.println("\n Enter an employee number to be searched:");
 key=sc.nextInt();
for(i=0;i<n;i++) {
if(ob[i].eNo==key) {
flag=1;
  break;
 }
else continue;
 }
if(flag==1) {
System.out.println("\n INFORMATION OF EMPLOYEE HAVING EMPLOYEE NUMBER"
+ob[i].eNo);
ob[i].print_emp();
 }
```

```
else {
 System.out.println("\n employee not found");
flag=0;
 }
 System.out.println("\n Do you want to visit anymore records ? (1 for yes , 0 for No) ");
opt=sc.nextInt();
 }
while(opt!=0);
        }
}
```

OutputScreenshot

```
D:\JAVA PGM>javac Employee1.java

D:\JAVA PGM>java Employee1

 How many records you have to save

1

 ENTER DETAILS OF EMPLOYEE 1

 Enter employee number:
1335

 Enter employee name:
Zayn

 Enter salary of employee:
65000

 Enter an employee number to be searched:
1001

 employee not found

 Do you want to visit anymore records ? (1 for yes , 0 for No)
1

 Enter an employee number to be searched:
1335

 INFORMATION OF EMPLOYEE HAVING EMPLOYEE NUMBER1335

 Employee Information


 EMPLOYEE NO. : 1335

 EMPLOYEE NAME: Zayn

 SALARY : 65000

 Do you want to visit anymore records ? (1 for yes , 0 for No)
0
```

OBJECT ORIENTED PROGRAMMING LAB

Name: Sreelakshmi R

Roll No:41

Batch:S2 RMCA B

Date:17/05/2022

Experiment No.: 10

Aim

Area of different shapes using overloaded functions

Sourcecode

```
import java.util.*;

class areaOverLoading{

    private void area(int side){
        int area= side*side;
        System.out.println("The area of square is "+area+"sqcm");
    }

    private void area(int length, int breadth){
        int area= length*breadth;
        System.out.println("The area of rectangle is "+area+"sqcm");
    }

  private void area(double length, double breadth){
  double area= (length*breadth)/2;
        System.out.println("The area of triangle is "+area+"sqcm");
    }

  private void area(double radius){
  double area= 3.14*radius*radius;
        System.out.println("The area of circle is "+area+"sqcm");
    }

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

      int length1,breadth1,side;
  double radius,length2,breadth2;
        areaOverLoading area= new areaOverLoading();

        System.out.println("\nChoose the Operations to perform:\n1. Area of square.\n2.
Area of rectangle.\n3. Area of triangle.\n4. Area of circle.\n");
        int ch= sc.nextInt();
```

```
    switch(ch){
                        case 1: {
                                System.out.println("\nEnter the value of side of the square: ");
                                side= sc.nextInt();

                                area.area(side);
                                break;
                        }

                        case 2: {
                                System.out.println("\nEnter the value of length of the rectangle: ");
                                length1= sc.nextInt();
                                System.out.println("\nEnter the value of breadth of the rectangle: ");
                                breadth1= sc.nextInt();
                                area.area(length1, breadth1);
                                break;
                        }

                        case 3:{
                                System.out.println("\nEnter the value of base of the triangle: ");
                                length2= sc.nextDouble();
                                System.out.println("\nEnter the value of height of the rectangle: ");
                                breadth2= sc.nextDouble();
                                area.area(length2, breadth2);
                                break;
                        }

                        case 4:{
                                System.out.println("\nEnter the value of radius of the circle: ");
                                radius= sc.nextDouble();
                                area.area(radius);
                                break;
                        }
                }

        }
}
```

OutputScreenshot

```
D:\JAVA PGM>java areaOverLoading

Choose the Operations to perform:
1. Area of square.
2. Area of rectangle.
3. Area of triangle.
4. Area of circle.

2

Enter the value of length of the rectangle:
4

Enter the value of breadth of the rectangle:
2
The area of rectangle is 8sqcm

D:\JAVA PGM>java areaOverLoading

Choose the Operations to perform:
1. Area of square.
2. Area of rectangle.
3. Area of triangle.
4. Area of circle.

3

Enter the value of base of the triangle:
3

Enter the value of height of the rectangle:
4
The area of triangle is 6.0sqcm

D:\JAVA PGM>java areaOverLoading

Choose the Operations to perform:
1. Area of square.
2. Area of rectangle.
3. Area of triangle.
4. Area of circle.

4

Enter the value of radius of the circle:
8
The area of circle is 200.96sqcm
```

```
D:\JAVA PGM>javac areaOverLoading.java

D:\JAVA PGM>java areaOverLoading

Choose the Operations to perform:
1. Area of square.
2. Area of rectangle.
3. Area of triangle.
4. Area of circle.

1

Enter the value of side of the square:
3
The area of square is 9sqcm

D:\JAVA PGM>java areaOverLoading

Choose the Operations to perform:
1. Area of square.
2. Area of rectangle.
3. Area of triangle.
4. Area of circle.

2

Enter the value of length of the rectangle:
4

Enter the value of breadth of the rectangle:
2
The area of rectangle is 8sqcm

D:\JAVA PGM>java areaOverLoading

Choose the Operations to perform:
1. Area of square.
2. Area of rectangle.
3. Area of triangle.
4. Area of circle.

3

Enter the value of base of the triangle:
3

Enter the value of height of the rectangle:
4
The area of triangle is 6.0sqcm
```

OBJECT ORIENTED PROGRAMMING LAB

Name:Sreelakshmi R

Roll No:41

Batch:S2 RMCA B

Date:23/05/2022

Experiment No.: 11

Aim

Create a class 'Employee' with data members Empid, Name, Salary, Address and constructors to initialize the data members. Create another class 'Teacher' that inherit the properties of class employee and contain its own data members department, Subjects taught and constructors to initialize these data members and also include display function to display all the data members. Use array of objects to display details of N teachers.

Sourcecode

```
class Main1{

   public static void main(String[] args) {

Teacher teacObj[] = new Teacher[2];
teacObj[0] = new Teacher("1","Sudha","A41",10000,"IntMCA","Java");
teacObj[1] = new Teacher("2","Deepa","A42",75000,"RMCA","Python");
teacObj[0].display();
 teacObj[1].display();
   }
}

class Employees {
          String  Empid;
        String    Name;
        String Address;
        int Salary;

 Employees(String id,String name,String addr,int salary
            this.Empid   = id;        this.Name    = name;
            this.Address = addr;        this.Salary  = salary;
   }
   void display(){
      System.out.println("EmpID : " + this.Empid);
      System.out.println("Name    : " + this.Name);
      System.out.println("Address : " + this.Address);
      System.out.println("Salary  : " + this.Salary);
   }
}
```

```
class Teacher extends Employees{

    String Department;
    String Subject;

   Teacher(String id,String name,String addr,int salary,String dept,String subj){
          super(id,name,addr,salary);
           this.Department=dept;
          this.Subject=subj;
    }
    void display(){
       System.out.println("----------------------------------------");
         super.display();
       System.out.println("Dept Name : " + this.Department);
       System.out.println("Subject Name : " + this.Subject);
       System.out.println("----------------------------------------");
          }
}
```

OutputScreenshot

```
D:\JAVA>javac Main1.java

D:\JAVA>java Main1
----------------------------------------
EmpID   : 1
Name    : Sudha
Address : A41
Salary  : 100000
Dept Name : Int MCA
Subject Name : Java
----------------------------------------
----------------------------------------
EmpID   : 2
Name    : Deepa
Address : A42
Salary  : 75000
Dept Name : RMCA
Subject Name : Python
----------------------------------------
```

OBJECT ORIENTED PROGRAMMING LAB

Name: Sreelakshmi R

Roll No:41

Batch:S2 RMCA B

Date:18/05/2022

Experiment No.: 12

Aim

Create a class ‘Person’ with data members Name, Gender, Address, Age and a constructor to initialize the data members and another class ‘Employee’ that inherits the properties of class Person and also contains its own data members like Empid, Company_name, Qualification, Salary and its own constructor. Create another class ‘Teacher’ that inherits the properties of class Employee and contains its own data members like Subject, Department, Teacherid and also contain constructors and methods to display the data members. Use array of objects to display details of N teachers.

Sourcecode

```
import java.util.*;
class Person{
    String Name;
    String Gender;
    String Address;
    int Age;
Person(String name,String gender,String address,int age) {
          this.Name=name;
          this.Gender=gender;
          this.Address=address;
          this.Age=age;
         }
}class Employe extends Person{
int Empid;
```

```
String Company_name;
String Qualificatiom; long
Salary;
Employe (String name,String gender,String address,int age,int empid,String
company_name,String qualification,long salary) {
 super(name,gender,address,age);
  this.Empid=empid;
  this.Company_name=company_name;
  this.Qualificatiom=qualification;
  this.Salary=salary;
  }
}
public class Teacher extends Employe{
String Teacher_id;
String Department,Subject;
Teacher(String name,String gender,String address,int age,int empid,String
company_name,String qualification,long salary,String teacher_id,String
department,String subject) {
super(name,gender,address,age,empid,company_name,qualification,salary;
 this.Teacher_id=teacher_id;
this.Department=department;
 this.Subject=subject;
}
void display()
{
System.out.println("Name: "+Name);
System.out.println("Gender: "+Gender);
System.out.println("Address: "+Address);
System.out.println("Age: "+Age);
System.out.println("Employee Id: "+Empid);
System.out.println("Company Name: "+Company_name);
System.out.println("Qualificatiom: "+Qualificatiom);
System.out.println("Salary: "+Salary);
System.out.println("Teacher Id: "+Teacher_id);
```

```
System.out.println("Department: "+Department);
System.out.println("Subject: "+Subject);
}
public static void main(String args[]) {
Scanner sc=new Scanner(System.in);
int n;
System.out.println("Enter number of teachers:");
n=sc.nextInt();
Teacher ob[]=new Teacher[n];
System.out.println("Enter The Teacher Details");
int x=0,j=0;
Scanner s=new Scanner(System.in);
 for(int i=0;i<n;i++) {
 x=i+1;
System.out.println("\n"+x+").");
System.out.println("Name:");
String a=s.next();
System.out.println("Gender:");
String b=s.next();
System.out.println("Address:");
String c=s.next();
System.out.println("Age:");
int d=s.nextInt();
System.out.println("Employee  Id:");
int e=s.nextInt();
System.out.println("Company name:");
String f=s.next();
System.out.println("Qualificatiom:");
Stringg=s.next();
System.out.println("Salary :");
int h=s.nextInt();
System.out.println("Teacher Id:");
String l=s.next();
```

```
System.out.println("Department:");
String m=s.next();
System.out.println("Subject:");
String k=s.next();
ob[i]=new    Teacher(a,b,c,d,e,f,g,h,l,m,k);
 }
s.close();
System.out.println(".....Teacher Details..... ");
for(int i=0;i<n;i++) {
 j=i+1;
System.out.println("\n"+j+"”);
ob[i].display();
}
 sc.close();
   }
}
```

OutputScreenshot

```
D:\JAVA>java Teacher
Enter number of teachers:
2
Enter The Teacher  Details

1).
Name:
Jyothi
Gender:
Female
Address:
Puthuparambil(H)
Age:
33
Employee Id:
13335
Company name:
TATA
Qualificatiom:
MCA
Salary :
40000
Teacher Id:
101
Department:
RMCA
Subject:
Java

2).
Name:
Jin
Gender:
Male
Address:
Nilayam(H)
Age:
36
```

```
Company name:
TeDex
Qualificatiom:
MCA
Salary :
45000
Teacher Id:
100
Department:
INTMCA
Subject:
Python
.........Teacher Details.........

1).
Name: Jyothi
Gender: Female
Address: Puthuparambil(H)
Age: 33
Employee Id: 13335
Company Name: TATA
Qualificatiom: MCA
Salary: 40000
Teacher Id: 101
Department: RMCA
Subject: Java

2).
Name: Jin
Gender: Male
Address: Nilayam(H)
Age: 36
Employee Id: 13321
Company Name: TeDex
Qualificatiom: MCA
Salary: 45000
Teacher Id: 100
Department: INTMCA
Subject: Python

D:\JAVA>
```

OBJECT ORIENTED PROGRAMMING LAB

Name: Sreelakshmi R

Roll No:41

Batch:RMCA S2B

Date:31-05-2022

Experiment No.:13

Aim

Write a program has class Publisher, Book, Literature and Fiction. Read the information and print the details of books from either the category, using inheritance.

Sourcecode

```
class Book {
    int ISBN;
    String title;
    String author
    int price;
    String Publisher;

    Book(){
   Book(int isbn, String title, String author, int price, String publisher) {
        this.ISBN = isbn;
        this.title = title;
        this.author = author;
        this.price = price;
        this.Publisher = publisher;
    }

    public void display() {
        System.out.print(this.ISBN + "\t");
        System.out.print(this.title + "\t");
        System.out.print(this.author + "\t");
        System.out.print(this.price + "\t");
        System.out.print(this.Publisher + "\t");
        System.out.println();
    }

public static void main(String []args){
 Book books[] = new Book[5];
books[0] =new Book(1,"A","S",11,"AD");
books[1] =new Book(2,"B","D",12,"AA");
books[2] =new Book(3,"AC","F",13,"AA");
books[3] =new Book(4,"E","A",14,"AA");
books[4] =new Book(5,"DA","G",15,"AA");

 int size = 5;
   for(int i = 0; i<size-1; i++) {
            for (int j = i+1; j<size; j{

if(books[i].title.compareTo(books[j].title)>0) {
```

```
books[i] = books[j];
                    books[j] = temp;
                }
            }
        }

        for(int i=0;i<5;i++){
            books[i].display();
        }
    }
}
```

OutputScreenshot

```
D:\JAVA PGM>javac Book.java

D:\JAVA PGM>java Book
1       A       S       11      AD
3       AC      F       13      AA
2       B       D       12      AA
5       DA      G       15      AA
4       E       A       14      AA

D:\JAVA PGM>
```

OBJECT ORIENTED PROGRAMMING LAB

Name: Sreelakshmi R

Roll No:41

Batch:S2 RMCA B

Date:18/05/2022

Experiment No.: 14

Aim

Create classes Student and Sports. Create another class Result inherited from Student and Sports. Display the academic and sports score of a student..

Sourcecode

```
import java.util.Scanner;
 class Sports{
   String sport;
    int Rating;
    Sports(String spo, int ra){
        sport = spo;
        Rating = ra;
    }
 }
class Student extends Sports{
 String Grade;
 double Overall_per;
    Student(String spo, int ra,String gd, double per ){
         super(spo, ra);
        Grade = gd;
        Overall_per = per;
    }
}
```

```
public class Result extends Student {
    Result(String spo, int ra,String gd, double per ){
 super(spo, ra, gd, per);
    }
    void display(){
        System.out.println("\nSports Details of Student");
        System.out.println("Sport :"+sport);
        System.out.println("Rating :"+Rating);
        System.out.println("\nAcademic Details of Student");
        System.out.println("Academic Grade :"+Grade);
        System.out.println("Overall percentage :"+Overall_per);
    }
public static void main(String[] args) {
      Scanner sc =new Scanner(System.in);
        System.out.println("\nEnter the Sports Details of Student");
        System.out.println("\n Sport: ");
        String a =sc.next();
        System.out.println("\n Sport Rating out of 10: ");
         int b =sc.nextInt();
        System.out.println("\nEnter the Academic Details of Student");
        System.out.println("\n Academic Grade: ");
        String c =sc.next();
        System.out.println("\n Overall percentage:");
         double d =sc.nextDouble();
       sc.close();
      Result obj= new Result(a,b,c,d);
      obj.display();
    }
}
```

OutputScreenshot

```
D:\JAVA>javac Result.java

D:\JAVA>java Result

Enter the Sports Details of Student

Sport:
FootBall

Sport Rating  out of 10:
9

Enter the Sports Details of Student

Academic Grade:
A

Overall percentage:
93

**********
Sports Details of Student
Sport :FootBall
Rating :9
Academic Details of Student
Academic Grade :A
Overall percentage :93.0

**********
```

OBJECT ORIENTED PROGRAMMING LAB

Name: Sreelakshmi R

Roll No:41

Batch:S2 RMCA B

Date:24/05/2022

Experiment No.: 15

Aim

Create an interface having prototypes of functions area() and perimeter(). Create two classes Circle and Rectangle which implements the above interface. Create a menu driven program to find area and perimeter of objects.

Sourcecode

```
import java.util.*;
interface Shape{
void getdata(); void
area();
  void perimeter();
  }

  class Circle implements
  Shape{

  double pi = 3.14;
     double r;
     Scanner sc = new Scanner(System.in);

     public void getdata(){
        System.out.println("Enter the radius of the circle:");
         r = sc.nextDouble();
     }
```

```
    public void perimeter(){
        System.out.println("Perimeter of the circle: "+(2 * pi * r));
    }

    public void area(){
        System.out.println("Area of the circle: "+ (pi * r * r) );
    }
}

class Rectangle implements Shape{
double l,b;
    Scanner sc = new Scanner(System.in);

    public void getdata(){
        System.out.println("Enter the length of the rectangle:");
        l = sc.nextDouble();
        System.out.println("Enter the breadth of the rectangle:");
        b = sc.nextDouble();
    }

    public void area(){
        System.out.println("Area of a rectangle: "+(l*b));
    }

    public void perimeter()
    {
        System.out.println("Perimeter of a rectangle: "+(2*(l+b)));
    }
}
```

```
public class CO3_Q6 {
   public static void main(String[] args) {
     int ch;
        Scanner sc = new Scanner(System.in);
        Circle ob = new Circle();
Rectangle obj = new Rectangle();
do{
           System.out.println("\n1.Circle\n2.Rectangle\n3.exit");
            System.out.println("Enter your choice:");
           ch = sc.nextInt();
 switch(ch){
        case 1 :ob.getdata();
              ob.area();
              ob.perimeter();
              break;
         case 2 :obj.getdata();
                    obj.area();
                    obj.perimeter();
                    break;
         case 3 :System.out.println("Exited...");
                     System.exit(0);
           }
        }while(true);
    }
}
```

OutputScreenshot

```
Microsoft Windows [Version 10.0.19044.1645]
(c) Microsoft Corporation. All rights reserved.

D:\JAVA>javac CO3_Q6.java

D:\JAVA>java CO3_Q6

1.Circle
2.Rectangle
3.exit
Enter your choice:
1
Enter the radius of the circle:
3
Area of the circle: 28.259999999999998
Perimeter of the circle: 18.84

1.Circle
2.Rectangle
3.exit
Enter your choice:
2
Enter the length of the rectangle:
5
Enter the breadth of the rectangle:
8
Area of a rectangle: 40.0
Perimeter of a rectangle: 26.0

1.Circle
2.Rectangle
3.exit
Enter your choice:
3
Exited...

D:\JAVA>
```

OBJECT ORIENTED PROGRAMMING LAB

Name: Sreelakshmi R

Roll No:41

Batch:RMCA S2B

Date:30-05-2022

Experiment No.: 16

Aim

Prepare bill with the given format using calculate method from interface.

Order No.

Date :

Product Id Name Quantity unit price Total

101 A 2 25 50

102 B 1 100 100

Net. Amount 150

Sourcecode

```
import java.util.Scanner;

interface calc
{
   void calculate();
}

class bill implements calc
{
   String date,name,p_id;
int quantity;
   double unit_price,total,namount=0;
Scanner sc = new Scanner(System.in);
public void getdata()
   {
```

```
        System.out.println("\nEnter product id:");
         p_id = sc.nextLine();
System.out.println("Enter product name:");
      name = sc.nextLine();
      System.out.println("Enter the Quantity:");
      quantity = sc.nextInt();
      System.out.println("Enter the unit price:");
      unit_price = sc.nextDouble();
    }
   public void calculate()
   {
       total = quantity * unit_price;
   }
   public void display()
   {

System.out.println(p_id+"\t\t"+name+"\t\t"+quantity+"\t\t"+unit_price+"\t"+total);
   }
}
public class CO3_Q7
{
   public static void main(String[] args) {
       int n,i;
   double namount=0,t;
      int ran;
      String date;
       t = Math.random() *1000000;
       ran = (int) t;
       Scanner sc = new Scanner(System.in);
       System.out.println("Order no. #"+ran);
       System.out.println("Enter the date:");
       date = sc.nextLine();
```

```
   System.out.println("Enter how many products are there:");
       n = sc.nextInt();
       bill ob[] = new bill[n];
 for(i=0;i<n;i++)
                    ob[i] = new bill();
   for(i=0;i<n;i++){
            ob[i].getdata();
            ob[i].calculate();
  }
        System.out.println("Date:"+date);
        System.out.println("Product Id \tName\t Quantity\t unit price\t Total ");
         System.out.println("----------------------------------------------------------------");
               for(i=0;i<n;i++){
              ob[i].display();
              namount += ob[i].total;
        }
        System.out.println("----------------------------------------------------------------");
        System.out.println("\t\t\tNet.Amount\t"+ namount);


    }
}
```

OutputScreenshot

```
14-05-2022
Enter how many products are there:
2

Enter product id:
KISTF
Enter product name:
Maybelline
Enter the Quantity:
3
Enter the unit price:
180

Enter product id:
QIXTS
Enter product name:
Labelllo
Enter the Quantity:
2
Enter the unit price:
90
Date:14-05-2022
Product Id      Name       Quantity        unit price    Total
----------------------------------------------------------------
KISTF           Maybelline              3               180.0   540.0
QIXTS           Labelllo                2               90.0    180.0
----------------------------------------------------------------
                        Net.Amount      720.0

D:\java>
```

OBJECT ORIENTED PROGRAMMING LAB

Name: SREELAKSHMI R

Roll No: 41

Batch: MCAS2B

Date: 31/05/22

Experiment No.: 17

Aim

Create a Graphics package that has classes and interfaces for figures Rectangle, Triangle, Square and Circle. Test the package by finding the area of these figures.

Sourcecode

File 1: graphics circle.java

```
package graphics; interface AreaInterface {
void Area();
}
public class circle implements AreaInterface{
   double radius;
public circle(double radius){
this.radius= radius;
   }
   public void Area() {
      double area= 3.14 * this.radius * this.radius;
      System.out.println("The area of the given circle is : "+area);

   }
}
```

rectangle.java

```
package graphics;
interface AreaInterface {
   void Area();
}
public class rectangle implements AreaInterface{
double length,breadth;
```

```
 public rectangle(double length,double breadth){
    this.length=length;
    this.breadth=breadth;
    }
    public void Area() {
        double area= this.length*this.breadth;
        System.out.println("The area of the given circle is : "+area);
    }
}
```

square.java

```
 package graphics;
 interface AreaInterface {
 void Area();
}
 public class square implements AreaInterface{
  double side;
  public square(double side){
     this.side= side;
    }
    public void Area() {
      double area= this.side * this.side;
        System.out.println("The area of the given square is : "+area);
    }
}
```

triangle.java

```
 package graphics;
 interface AreaInterface {
 void Area();
}
 public class Triangle implements AreaInterface{
    double length, breadth;
```

```
public Triangle(double length, double breadth){
      this.length= length;
    this.breadth= breadth;
   }
   public void Area() {
      double area= 0.5 * this.length * this.breadth;
      System.out.println("The area of the given triangle is : "+area);
   }
}
```

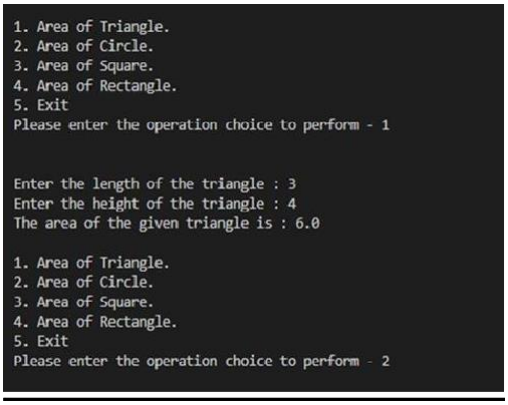
File 2: result.java

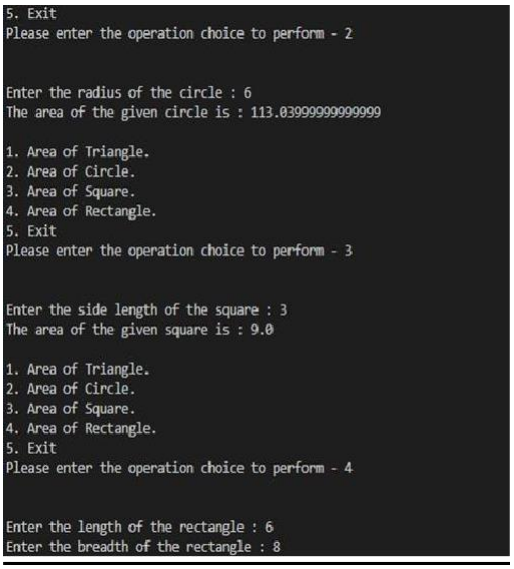
```
import  java.util.*;
import graphics.*;

public class result {
   public static void main(String[] args) {
Scanner sc= new Scanner(System.in);
int choice,isexit=0;
          while(isexit==0){
         double length, breadth, side, radius;
         System.out.println("\n1. Area of Triangle.\n2. Area of Circle.\n3. Area of Square.\n4. Area
of Rectangle.\n5. Exit");
         System.out.print("Please enter the operation choice to perform - ");
choice= sc.nextInt();
         System.out.println("\n");

         switch(choice){
case 1:{
               System.out.print("Enter the length of the triangle : ");
            length= sc.nextDouble();
               System.out.print("Enter the height of the triangle : ");
             breadth= sc.nextDouble();
               Triangle triangle = new Triangle(length,breadth);
```

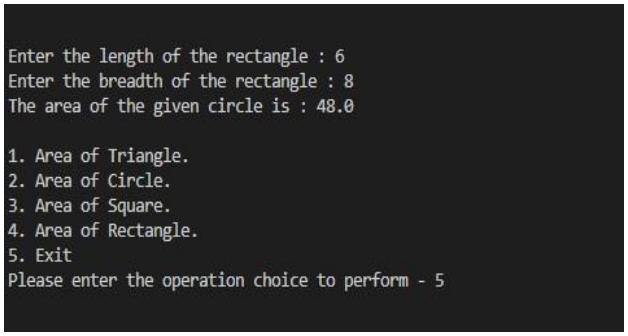
```
            triangle.Area();
             break;
            }

 case 2:{
              System.out.print("Enter the radius of the circle :");
                 radius= sc.nextDouble();
                  circle cir= new circle(radius);
                  cir.Area();
                  break;
            }

case 3:{
              System.out.print("Enter the side length of the square :");
               side= sc.nextDouble();
             square square= new square(side);
             square.Area();
            break;
   }
 case 4:{
              System.out.print("Enter the length of the rectangle : ");
            length= sc.nextDouble();
              System.out.print("Enter the breadth of the rectangle : ");
            breadth= sc.nextDouble();
            rectangle rec= new rectangle(length,breadth);
               rec.Area();
                break;
            }

case 5:{
               isexit=1;
                 break;
```

```
            }


            default:{
break;
            }
        }
}
sc.close();
   }
}
```

OutputScreenshot







OBJECT ORIENTED PROGRAMMING LAB

Name: Sreelakshmi R

Roll No:41

Batch: RMCA S2B

Date:1-06-2022

Experiment No.: 18

Aim

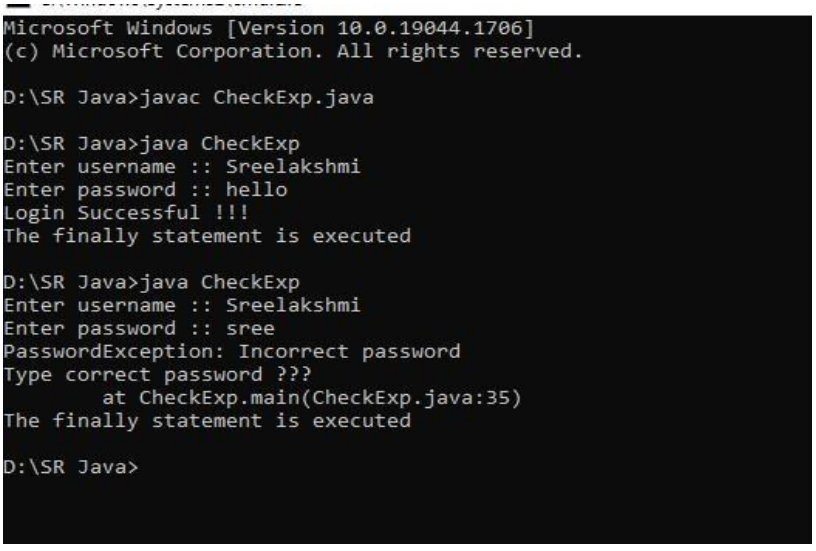
Write a user defined exception class to authenticate the user name and password

Sourcecode

```
import java.util.Scanner;

class UsernameException extends Exception {


 public UsernameException(String msg) {
   super(msg);
 }

}


class PasswordException extends Exception {


 public PasswordException(String msg) {
 super(msg);
 }

}

 public class CheckExp {


 public static void main(String[] args) {

Scanner s = new Scanner(System.in);

String username, password;


  System.out.print("Enter username :: ");
   username = s.nextLine();
```

```
 System.out.print("Enter password :: ");
  password = s.nextLine();


   int length = username.length();


 try {

  if(length < 6)     throw new UsernameException("Username must be greater than 6 characters
???");
  else if(!password.equals("hello")) throw new PasswordException("Incorrect password\nType
correct password ???");

 else
   System.out.println("Login Successful !!!");

 }

 catch (UsernameException u) {

 u.printStackTrace();
 }

 catch (PasswordException p) {
 p.printStackTrace();
 }  finally
{
  System.out.println("The finally statement is executed");

 }

}

}
```

Output Screenshot



OBJECT ORIENTED PROGRAMMING LAB

Name: Sreelakshmi R

Roll No:41

Batch:RMCA S2

Date:1-06-2022

Experiment No.:19

AIM

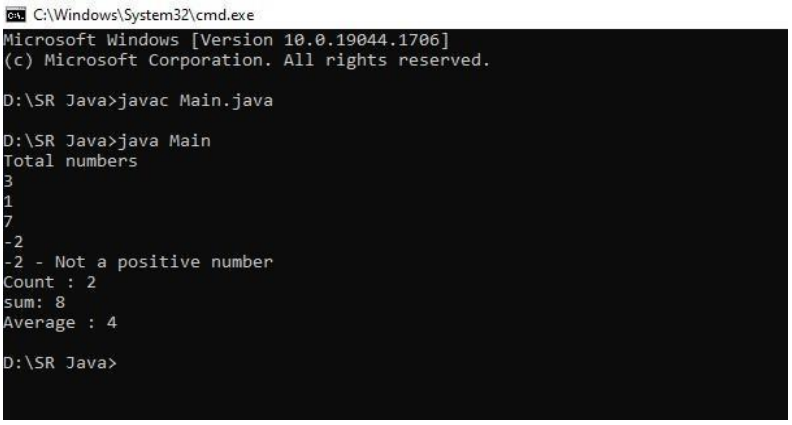
Find the average of N positive integers, raising a user defined exception for each negative input**.**

Sourcecode

```
import java.util.*;


class MyException extends Exception {
    public MyException(String value) {
    super(value);
    }
}


public class Main {
    public static void main(String args[]) {
int totalNums;
        int i;
        int temp, count = 0;
        int sum = 0;


        Scanner sc = new Scanner(System.in);


       System.out.println("Total numbers");
     totalNums = Integer.parseInt(sc.nextLine());
     for (i = 0; i < totalNums; i++) {
 try {
```

```
            temp = Integer.parseInt(sc.nextLine());
if (temp > 0) {
                sum += temp;
              count += 1;
            }
else {
                throw new MyException(Integer.toString(temp));
            }
          }
catch (MyException ex) {
            System.out.print(ex.getMessage());
            System.out.println(" - Not a positive number");
          }
      }
      System.out.print("Count : ");
      System.out.println(count);
      System.out.print("sum: ");
      System.out.println(sum);
      System.out.print("Average : ");
      System.out.println(sum / count);
  }
}
```

OutputScreenshot



OBJECT ORIENTED PROGRAMMING LAB

Name: Sreelakshmi R

Roll No: 41

Batch: RMCA B

Date: 01/06/2022

Experiment No.: 20

Aim

Define 2 classes; one for generating Fibonacci numbers and other for displaying even numbers in a given range. Implement using threads. (Runnable Interface).

Source Code

```
import java.util.Scanner;

class Fib implements Runnable{
 public void run(){
    int a=0,b=1,c=0,n=20;
      System.out.println("Fibonacci Series upto "+n+":\n");
while (n>0)
      {
          System.out.print(c+"");
           a=b;
           b=c;
           c=a+b;
           n=n-1;
      }
      System.out.println("\n\n**************************\n");
   }
}

class EvenNo implements Runnable{
  public void run(){
      int n;
      Scanner sc=new Scanner(System.in);

      System.out.println("Enter the Value of N:");
```

```
     n=sc.nextInt();

    System.out.println("Even Numbers from 1 to "+n+":");
      for(int i=1;i<=n;i++) {
           if(i%2==0) {
              System.out.println(i);
              }
       }
   }
}
 public class
EvenOdd{
    public static void main(String[] args) {
        Fib obj=new Fib();
        Thread t=new Thread(obj);
        t.start();
        EvenNo obj1=new EvenNo();
Thread t1=new Thread(obj1);
        t1.start();
    }
}
```

OUTPUT

```
D:\SR Java>javac EvenOdd.java

D:\SR Java>java EvenOdd
Fibonacci Series upto 20:

0 1 1 2 3 5 8 13 21 34 55 89 144 233 377 610 987 1597 2584 4181

************************

Enter the Value of N:
8
Even Numbers from 1 to 8:
2
4
6
8

D:\SR Java>
```

OBJECT ORIENTED PROGRAMMING LAB

Name: Sreelakshmi R

Roll No:41

Batch:RMCA S2B

Date:1-06-2022

Experiment No.: 21

AIM

Program to create a generic stack and do the Push and Pop operations.

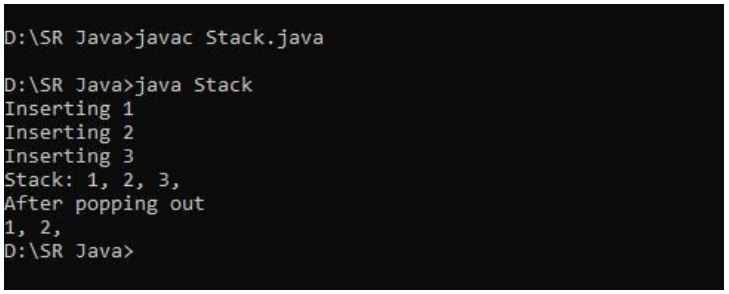
Source Code

```
public class Stack{
private int arr[];
private int top;
private int
capacity;
Stack(int size) {
arr = new int[size];
    capacity = size;
    top = -1;
  }
public void push(int x) {
if (isFull()) {
     System.out.println("Stack OverFlow");
     System.exit(1);
   }
 System.out.println("Inserting " + x);
 arr[++top] = x;
  }
  public int pop() {
```

```
    if (isEmpty()) {
      System.out.println("STACK EMPTY");

      System.exit(1);
    }

    return arr[top--];
  }

  public int getSize() {
return top + 1;
  }

  public Boolean isEmpty() {
return top == -1;
  }

  public Boolean isFull() {
return top == capacity - 1;
  }

  public void printStack() {
for (int i = 0; i <= top; i++) {
      System.out.print(arr[i] + ", ");
    }
  }
  public static void main(String[] args){
    Stack stack = new Stack(5);
    stack.push(1);
  stack.push(2);
```

```
    stack.push(3);
System.out.print("Stack: ");
stack.printStack();
          stack.pop();
   System.out.println("\nAfter popping out");
   stack.printStack();
  }
}
```

Output Screenshot



OBJECT ORIENTED PROGRAMING LAB

Name : Sreelakshmi R

Roll No : 41

Batch :S2B RMCA

Date : 07-06-22

Experiment No.: 22

Aim

Maintain a list of Strings using ArrayList from collection framework, perform built-in operations.

Source Code

```
import java.util.*;
 public class arraylist{
   public static void main(String[] args){
      ArrayList<String> arrayList= new ArrayList<>();
      arrayList.add("Thor");
        arrayList.add("Loki");
        arrayList.add("Wanda");
        arrayList.add("Nebula");
      System.out.println("The elements of the arraylist is - "+arrayList);
      Collections.sort(arrayList);
      System.out.println("\nThe ArrayList Sort : "+arrayList);
      Collections.addAll(arrayList,"Steve","Tony","Thanos","Banners","Natasha");
      System.out.println("\nAdding new items in the arraylist is : "+arrayList);
      Collections.sort(arrayList, Collections.reverseOrder());
      System.out.println("\nThe reverse  order of  the arraylist : "+arrayList);
      System.out.println("\nThe maximum element of the arraylist :  "+Collections.max(arrayList));
   }


}
```

Output Screenshot

```
C:\Users\Student\Documents\Java>java arraylist
The elements of the arraylist is - [Thor, Loki, Wanda, Nebula]

The ArrayList Sort : [Loki, Nebula, Thor, Wanda]

Adding new items in the arraylist is : [Loki, Nebula, Thor, Wanda, Steve, Tony, Thanos, Banners, Natasha]

The reverse  order of  the arraylist : [Wanda, Tony, Thor, Thanos, Steve, Nebula, Natasha, Loki, Banners]

The maximum element of the arraylist : Wanda
```

OBJECT ORIENTED PROGRAMMING LAB

Name: Sreelakshmi R

Roll No:41

Batch: RMCA S2B

Date:1-06-2022

Experiment No.: 23

Aim

Program to demonstrate the creation of queue object using the Priority Queue class.

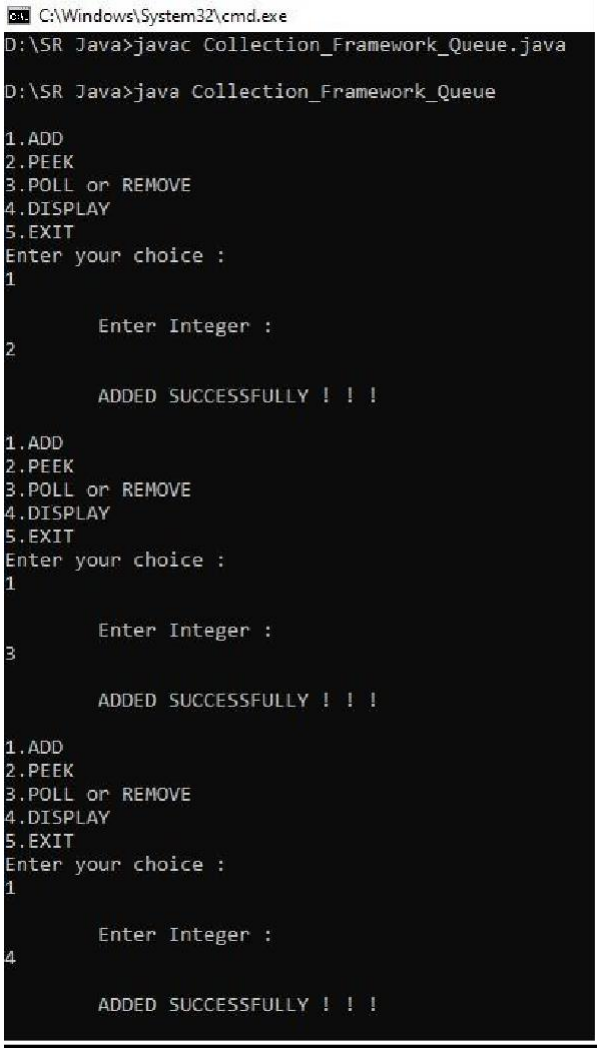
Source Code

```
import java.util.*;
 class Collection_Framework_Queue {
    public static void main(String args[]) {
       Queue<Integer> q = new PriorityQueue<Integer>(new Comp());
         int ch;
       Scanner sc = new Scanner(System.in);
do {
          System.out.println("\n1.ADD\n2.PEEK\n3.POLL or REMOVE\n4.DISPLAY\n5.EXIT");
        System.out.println("Enter your choice : ");
            ch = sc.nextInt();
switch (ch) {
     case 1:
                System.out.println("\n\tEnter Integer : ");
               int n1 = sc.nextInt();
                q.add(n1);
                System.out.println("\n\tADDED SUCCESSFULLY ! ! ! ");
                break;
   case 2:
                if (q.isEmpty()) {
                   System.out.print("\n\tQueue Empty ! ! !");
                }
```

```
else {
                    System.out.print("\n\tPeeked element is " + q.peek());
                }
            break;
case 3:
                if (!q.isEmpty()) {
                    System.out.print("\n\tRemoved element is " +q.poll());
                }
 else {
                    System.out.print("\n\tQueue Empty ! ! !");
                }
                 break;
case 4:
                if (!q.isEmpty()) {
                    System.out.print("\nSize of queue : " + q.size());
                    System.out.print("\nQueue elements  : " + q);
System.out.println("\nQueue elements are");
for (int i : q) {
                        System.out.println(i);
                    }
                }
 else {
                    System.out.print("\n\tQueue Empty ! ! !");
                }
              break;
case 5:
                break;
default:
                 System.out.println("\n\tPlease enter valid choice ! ! ! ");
            }
        } while (ch != 5);
```

```
    }
}
class Comp implements Comparator<Integer> {
   public int compare(Integer a, Integer b) {
   return a % 10 > b % 10 ? 1 : -1;
    }
}
```

Output Screenshot



C:\Windows\System32\cmd.exe

```
1.ADD
2.PEEK
3.POLL or REMOVE
4.DISPLAY
5.EXIT
Enter your choice :
1

        Enter Integer :
8

        ADDED SUCCESSFULLY ! ! !

1.ADD
2.PEEK
3.POLL or REMOVE
4.DISPLAY
5.EXIT
Enter your choice :
2

        Peeked element is 2
1.ADD
2.PEEK
3.POLL or REMOVE
4.DISPLAY
5.EXIT
Enter your choice :
3

        Removed element is 2
1.ADD
2.PEEK
3.POLL or REMOVE
4.DISPLAY
5.EXIT
Enter your choice :
4
```

```
Size of queue : 3
Queue elements  : [3, 8, 4]
Queue elements are
3
8
4

1.ADD
2.PEEK
3.POLL or REMOVE
4.DISPLAY
5.EXIT
Enter your choice :
5

D:\SR Java>
```

OBJECT ORIENTED PROGRAMING LAB

Name : Sreelakshmi R

Roll No : 41

Batch :S2B RMCA

Date : 07-06-22

Experiment No.: 24

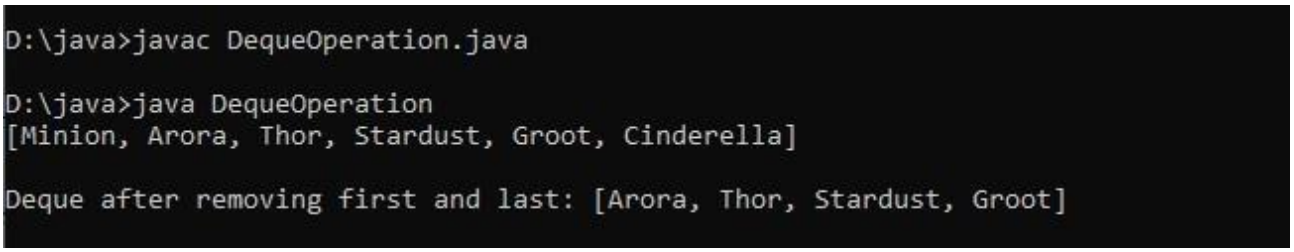
Aim

Program to demonstrate the addition and deletion of elements in deque.

Source Code

```
import java.util.*;

 class DequeOperation{

public static void main(String[] args){

  Deque<String> deque = new LinkedList<String>();

            deque.add("Stardust");

        deque.addFirst("Thor");

        deque.addLast("Groot");

        deque.push("Arora");

        deque.offer("Cinderella");

        deque.offerFirst("Minion");

        System.out.println(deque + "\n");

        deque.removeFirst();

        deque.removeLast();

System.out.println("Deque after removing " + "first and last: " + deque);

      }

}
```

Output Screenshot



OBJECT ORIENTED PROGRAMMING LAB

Name:Sreelakshmi R

Roll No: 41

Batch:RMCA S2B

Date: 2-06-2022

Experiment No.: 25

Aim

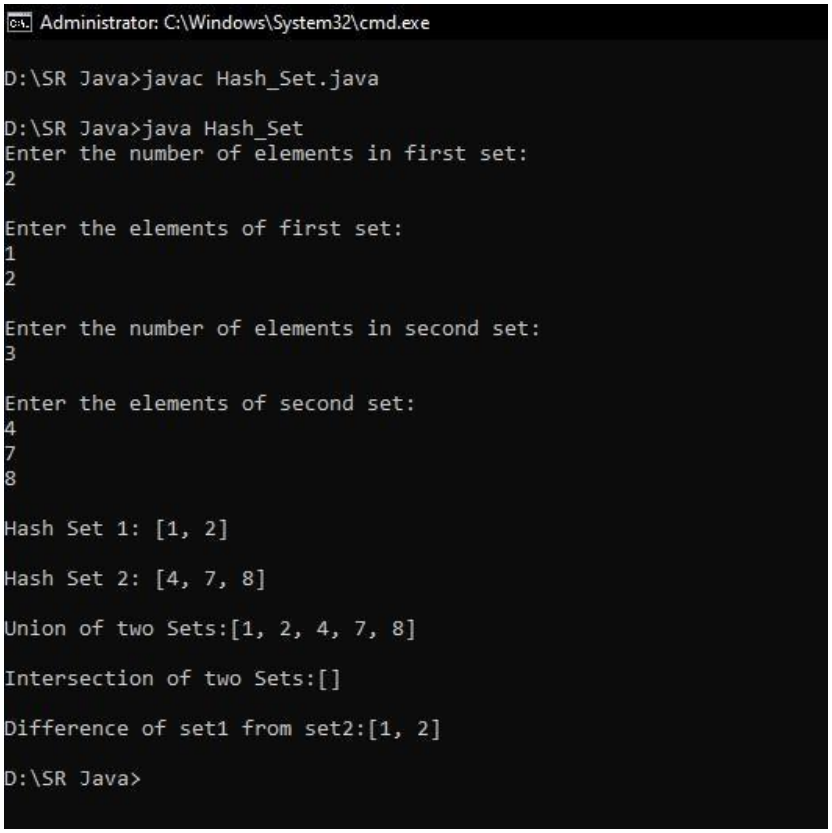
Write a Java program to compare two hash set.

Source Code

```
import java.util.HashSet;
import java.util.Scanner;
import java.util.Set;
public class Hash_Set {

	public static void main(String[] args) {
		Set<String> set1 = new HashSet<String>();
		Set<String> set2 = new HashSet<String>();
		Scanner sc=new Scanner(System.in);
		System.out.println("Enter the number of elements in first set: ");
		int n=sc.nextInt();
		System.out.println("\nEnter the elements of first set: ");
		for(int i =0;i<n;i++)
		{
			String st=sc.next();
			set1.add(st);

		}

		System.out.println("\nEnter the number of elements in second set:");
		int n1=sc.nextInt();
```

```
            System.out.println("\nEnter the elements of second set: ");
            for(int i =0;i<n1;i++)
            {
                  String st=sc.next();
      set2.add(st);
       }
            System.out.println("\nHash Set 1: " + set1);
            System.out.println("\nHash Set 2: " + set2);
            Set<String> union = new HashSet<String>(set1);
          union.addAll(set2);
          System.out.print("\nUnion of two Sets:");
          System.out.println(union);
          Set<String> intersection = new HashSet<String>(set1);
          intersection.retainAll(set2);
          System.out.print("\nIntersection of two Sets:");
          System.out.println(intersection);
          Set<String> difference = new HashSet<String>(set1);
          difference.removeAll(set2);
          System.out.print("\nDifference of set1 from set2:");
          System.out.println(difference);

      }
}
```

Output Screenshot



OBJECT ORIENTED PROGRAMING LAB

Name : Sreelakshmi R

Roll No : 41

Batch :S2 RMCA B

Date : 07-06-22

Experiment No.: 26

Aim

Program to demonstrate the working of Map interface by adding, changing and removing elements.

Source Code

```
import java.util.*;
class MapElements{
        public static void main(String args[]){
         Map<String,Integer> hm = new HashMap<String,Integer>();
        hm.put("Amit", new Integer(8));
        hm.put("Rayan", new Integer(3));
        hm.put("Rohan", new Integer(1));
                for (Map.Entry<String, Integer> me : hm.entrySet()){
                                System.out.print(me.getKey() + ":");
                                System.out.println(me.getValue());
                        }
        }
}
```

Output Screenshot

OBJECT ORIENTED PROGRAMING LAB

Name : Sreelakshmi R

Roll No : 41

Batch : S2 RMCAB

Date : 07-06-22

Experiment No.: 27

Aim

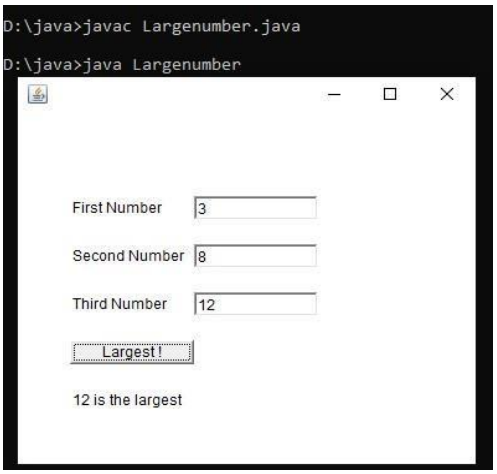
Program to find maximum of three numbers using AWT.

Source Code

```
importjava.awt.*;

 import java.awt.event.*;
public class Largenumber implements ActionListener{
        Frame f=new Frame();
        Label l1=new Label("First Number");
        Label l2=new Label("Second Number");
        Label l3=new Label("Third Number");
        Label res=new Label("Result");
        TextField t1=new  TextField();
        TextField t2=new  TextField();
        TextField t3=new TextField();
        Button b1=new Button("Largest !");
        Largenumber(){
        l1.setBounds(50,100,100,20);
        l2.setBounds(50,140,100,20);
        l3.setBounds(50,180,100,20);
        t1.setBounds(150,100,100,20);
        t2.setBounds(150,140,100,20);
        t3.setBounds(150,180,100,20);
```

```
        b1.setBounds(50,220,100,20); res.setBounds(50,260,100,20);
                f.add(l1);
                f.add(l2);
                f.add(l3);
                f.add(t1);
                f.add(t2);
                f.add(t3);
                f.add(res);
                f.add(b1);
b1.addActionListener(this); f.setLayout(null);
f.setVisible(true);
f.setSize(400,400); }
public static void main(String[] args){
new Largenumber();
}
 public void actionPerformed(ActionEvent e){ if(e.getSource()==b1){ int
n1=Integer.parseInt(t1.getText());
int n2=Integer.parseInt(t2.getText());
 int n3=Integer.parseInt(t3.getText());
int largeres= (n1 > n2) ? (n1 > n3 ? n1 : n3) : (n2 > n3 ? n2 : n3);
 res.setText(String.valueOf(largeres)+" is the largest");
          }
        }
}
```

Output Screenshot



OBJECT ORIENTED PROGRAMING LAB

Name : Sreelakshmi R

Roll No : 41

Batch : S2B RMCA

Date : 07-06-22

Experiment No.: 28

Aim

Implement a simple calculator using AWT components.

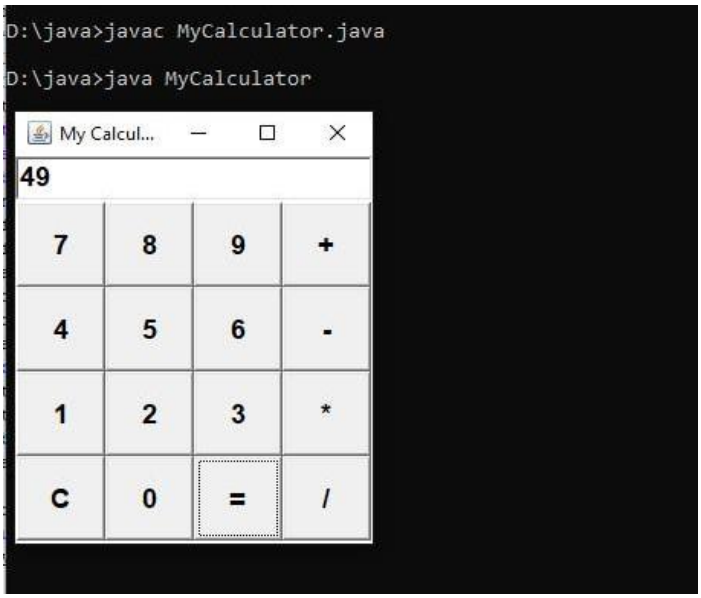
Source Code

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

class MyCalculator extends Frame implements ActionListener {
TextField tfInput;
Panel panel;
String btnString[] = {"7", "8", "9", "+",
                      "4", "5", "6", "-",
                      "1", "2", "3", "*",
                      "C", "0", "=", "/"};

Button btn[] = new Button[16];
int num1 = 0, num2 = 0, result =0;
char op;

public MyCalculator() {
Font f = new Font("Cambria", Font.BOLD, 18);
 tfInput = new TextField(10);
 tfInput.setFont(f);
panel = new Panel();
add(tfInput, "North");
add(panel, "Center");
 panel.setLayout(new GridLayout(4,4));

 for(int i=0; i < 16; i++) {
btn[i] = new Button(btnString[i]);
btn[i].setFont(f);
btn[i].addActionListener(this);
panel.add(btn[i]);
}
```

```
 addWindowListener(new WindowAdapter(){
public void windowClosing(WindowEvent we) {
System.exit(0);
          }
    }
}
public void actionPerformed(ActionEvent ae) {
String str = ae.getActionCommand();
if(str.equals("+")) {
op = '+';
 num1 = Integer.parseInt(tfInput.getText());
 tfInput.setText("");
 }
else if(str.equals("-")) {
 op = '-';
 num1 = Integer.parseInt(tfInput.getText());
 tfInput.setText("");
 }
 else if(str.equals("*"))
{
 op = '*';
 num1 = Integer.parseInt(tfInput.getText());
 tfInput.setText("");
 }
  else if(str.equals("/"))
{
 op = '/';
 num1 = Integer.parseInt(tfInput.getText());
 tfInput.setText("");
 }
else if(str.equals("=")) {
 num2 =Integer.parseInt(tfInput.getText());
switch(op) {
 case '+' : result = num1 + num2;
  break;
 case '-' : result = num1 - num2;
  break;
 case '*' : result = num1 *num2;
break;
 case '/' : result = num1 / num2;
break;
}
 tfInput.setText(result +"");
result = 0;
}
```

```
  else if(str.equals("C")) {
tfInput.setText("");
 num1 = num2 = result = 0;
 }
else {
tfInp
ut.set
Text(
fInpu
t.getT
ext()
+ str);

          }
 }
 public static void main(String args[]) {
MyCalculator m = new MyCalculator();
m.setTitle("My Calculator");
m.setSize(250,300);
 m.setVisible(true);
}
```

Output Screenshot



OBJECT ORIENTED PROGRAMING LAB

Name : Sreelakshmi R

Roll No : 41

Batch : S2B RMCA

Date : 07-06-22

Experiment No.: 29

Aim

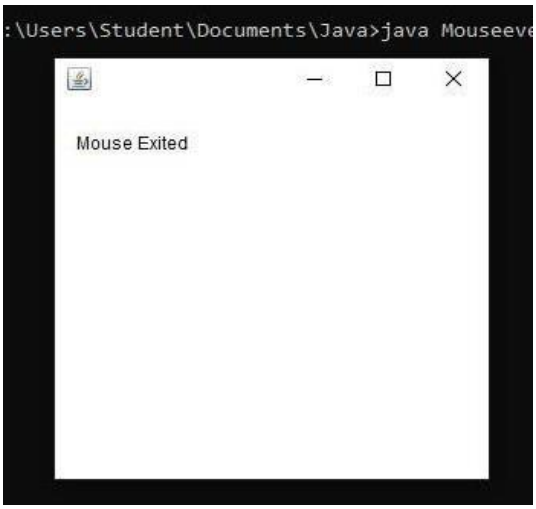
Develop a program to handle all mouse events and windowevents.

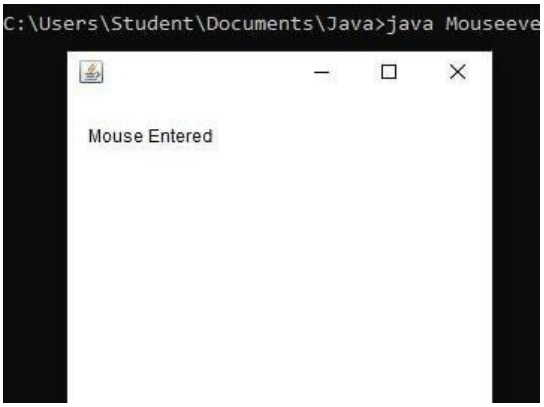
Source Code

```
 import java.awt.*;
 import java.awt.event.*;
public class Mouseevents extends Frame implements MouseListener{
    Label l;
    Mouseevents(){
addMouseListener(this);
       l=new Label();
        l.setBounds(20,50,100,20);
    add(l);
setSize(300,300);
setLayout(null);
setVisible(true);
    }
    public void mouseClicked(MouseEvent e) {
      l.setText("Mouse Clicked");
    }
    public void mouseEntered(MouseEvent e) {
       l.setText("Mouse Entered");
    }
    public void mouseExited(MouseEvent e) {
            l.setText("Mouse Exited");
    }
```

```
    public void mousePressed(MouseEvent e) {
            l.setText("Mouse Pressed");
    }
    public void mouseReleased(MouseEvent e) {
             l.setText("Mouse Released");
    }
public static void main(String[] args) {
            new Mouseevents();
 }
}
```

Output Screenshot





OBJECT ORIENTED PROGRAMING LAB

Name : Sreelakshmi R

Roll No : 41

Batch : S2B RMCA

Date : 07-06-22

Experiment No.: 30

Aim

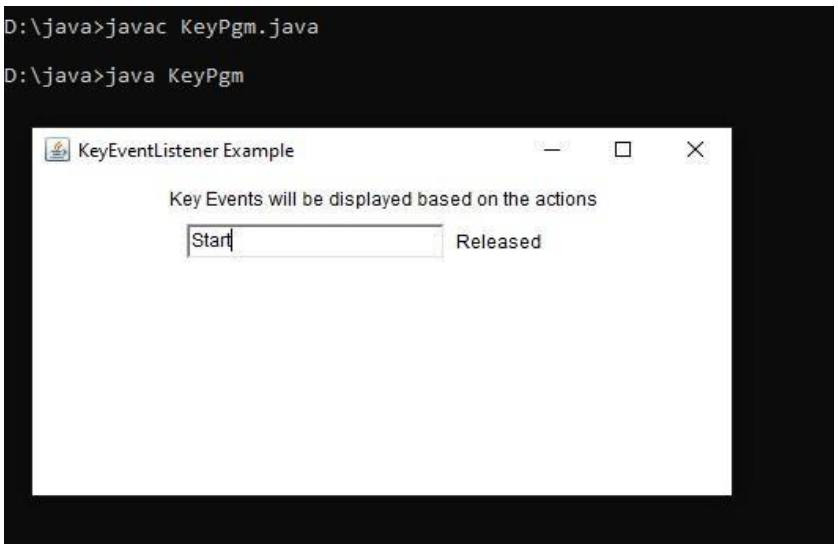
Develop a program to handle Key events.

Source Code

```
import java.awt.FlowLayout;
import java.awt.Frame;
import java.awt.Label;
import java.awt.TextField;
 import java.awt.event.KeyEvent;
import java.awt.event.KeyListener;
public class KeyPgm implements
KeyListener{
   Label lb1, lbl2, lb;
   TextField tf1;
   Frame fr;
   String s;
   KeyPgm(){
fr = new Frame("KeyEventListener Example");
   lb1= new Label(" Key Events will be displayed based on the actions",
Label.CENTER);
          lbl2= new Label();
           lb= new Label();
                  tf1 = new TextField(20);
fr.setLayout(new FlowLayout());
                  fr.add(lb1);
```

```
                fr.add(tf1);
                fr.add(lbl2);
                tf1.addKeyListener(this);
                fr.setSize(460,250);
                fr.setVisible(true);
        }
public void keyPressed(KeyEvent ev){
lbl2.setText(" Key pressed");
   }
public void keyReleased(KeyEvent ev){
lbl2.setText("Released");
   }
 public void keyTyped(KeyEvent ev){
lbl2.setText("Key is typed");
                fr.setVisible(true);
   }
 public static void main(String[] args){
new KeyPgm();
   }
}
```

Output Screenshot



OBJECT ORIENTED PROGRAMMING LAB

Name: Sreelakshmi R

Roll No:41

Batch:RMCA S2B

Date:31-05-2022

Experiment No.: 31

Aim

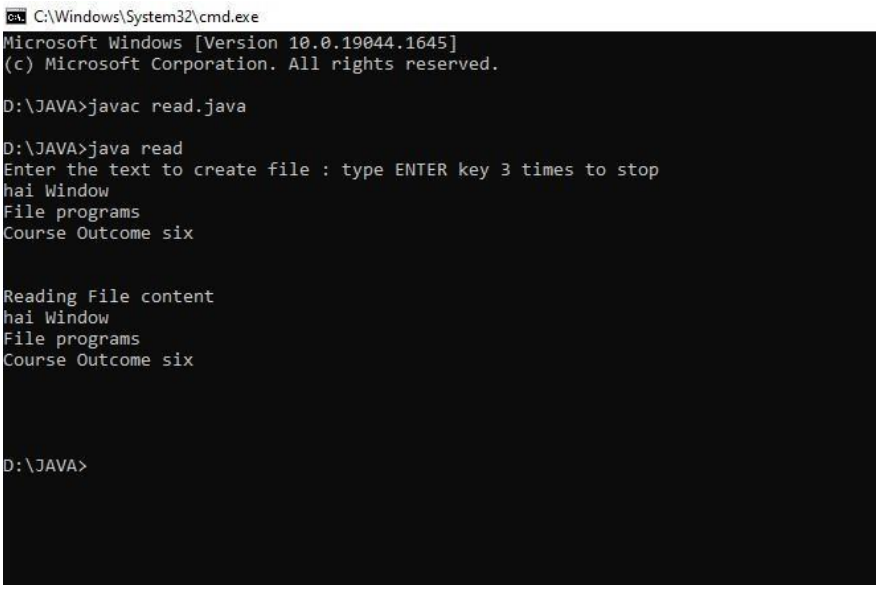
Write a program to write to a file, then read from the file and display the contents on the console.

Source Code

```
import java.io.FileReader;
import java.io.FileWriter;
import
java.io.IOException;
import java.io.*; import
java.util.*; import
java.io.File; class read {
public static void main(String[] args) {

String var = "";
Scanner scan = new Scanner(System.in);
System.out.println("Enter the text to create file : type ENTER key 3 times to stop");
while (!var.endsWith("\n\n\n")) var = var + scan.nextLine() + "\n"; try {

File file = new File("output.txt");

FileWriter fw = new
FileWriter(file); fw.write(var);
fw.close();
System.out.println("Reading File content");
FileReader fr = new FileReader("output.txt");
String str = "";
int i;
while ((i = fr.read()) != -1) {
```

```
str += (char) i;
}
System.out.println(str);
fr.close();
} catch (IOException e) {
System.out.println("There are some exception");
}
}
}
```

Output Screenshot



OBJECT ORIENTED PROGRAMMING LAB

Name: Sreelakshmi R

Roll No:41

Batch:RMCA S2B

Date:31-05-2022

Experiment No.:32

Aim

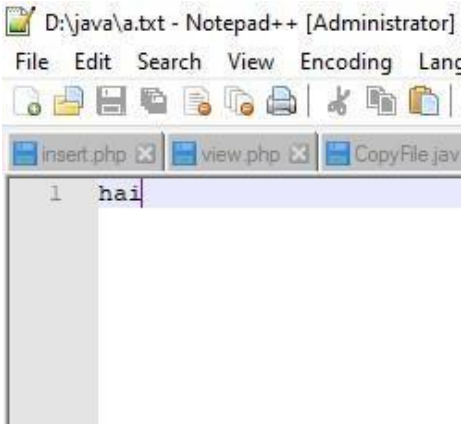
Write a program to copy one file to another

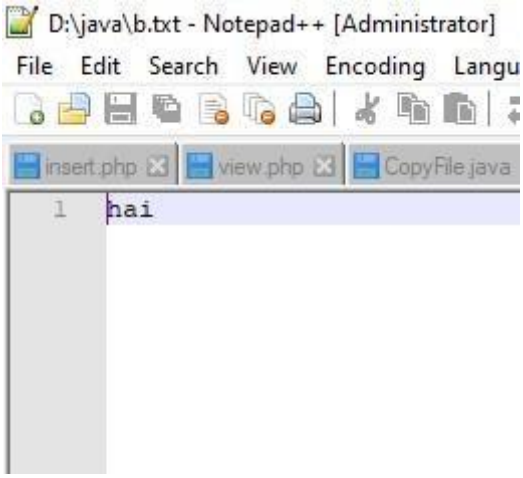
Source Code

```
import java.io.FileInputStream;
import java.io.FileOutputStream;
import java.io.IOException;

publicclass CopyFile{
        public static void main(String[] args)throws IOException{
                FileInputStream fileinput = new FileInputStream("1.txt");
FileOutputStream fileoutput = new FileOutputStream("2.txt");
                int i;
                while((i=fileinput.read())!=-1){
                fileoutput.write(i);
                }
                System.out.println("Successfully Copied");
                fileinput.close();fileoutput.close();
        }
}
```

Output Screenshot

```
D:\java>javac CopyFile.java

D:\java>java CopyFile
Successfully Copied

D:\java>
```





OBJECT ORIENTED PROGRAMMING LAB

Name: Sreelakshmi R

Roll No:41

Batch:RMCA S2B

Date:31-05-2022

Experiment No.: 33

Aim

Write a program that reads from a file having integers. Copy even numbers and odd numbers to separate files.

Source Code

```
import java.io.FileReader;
import java.io.FileWriter;
import java.io.IOException;
import java.io.*;
 import java.util.*;
import java.io.File;
public class oddeven {
public static void main(String[] args)
{ try
{
FileReader fr = new FileReader("numbers.txt");

BufferedReader br = new BufferedReader(fr);

File file1 = new File("oddnumbers.txt");

FileWriter fw1 = new FileWriter(file1);

File file2 = new File("evennumbers.txt");

FileWriter fw2 = new FileWriter(file2);
String num;
while ((num = br.readLine()) != null) {
if (Integer.parseInt(num) % 2 == 0) {
fw2.write(num + "\n");
}
 else {
fw1.write(num +"\n");
          }
 }
fw1.close()
;
fw2.close()
;
}
catch(Exce
ption e) {
 System.out.println("copied successfully");

                    }
```

```
        }
}
```

Output Screenshot