20MCA132 – OBJECT ORIENTED PROGRAMMING LAB

Lab Report Submitted By

NEHA ANTONY

Reg. No.:AJC21MCA-2081

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

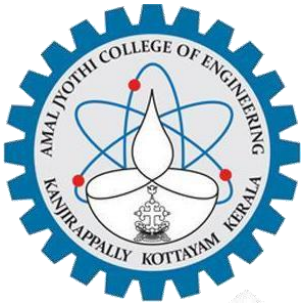
[Affiliated to APJ Abdul Kalam Technological University, Kerala. Approved by AICTE, Accredited by NAAC with 'A' grade. Koovappally, Kanjirappally, Kottayam, Kerala – 686518]

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 **Neha Antony (Reg.No:AJC21MCA-2081)** 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.

Mrs. Gloriya Mathew
Lab In-Charge

Rev.Fr.Dr.Rubin Thottupuram Jose
Head of the Department

Internal Examiner

External Examiner

CONTENT

	teachers.		
12	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.	**18-05-2022**	**31**
13	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	**18-05-2022**	**36**
14	Create classes Student and Sports. Create another class Result inherited from Student and Sports. Display the academic and sports score of a student.	**18-05-2022**	**39**
15	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.	**24-05-2022**	**42**
16	Prepare bill with the given format using calculate method from interface.	**24-05-2022**	**46**
17	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	**24-05-2022**	**50**
18	Write a user defined exception class to authenticate the user name and password.	**31-05-2022**	**54**
19	Find the average of N positive integers, raising a user defined exception for each negative input.	**31-05-2022**	**56**
20	Define 2 classes; one for generating Fibonacci numbers and other for displaying even numbers in a given range. Implement using threads. (Runnable Interface).	**01-06-2022**	**58**
21	Program to create a generic stack and do the Push and Pop operations	**01-06-2022**	**61**
22	Maintain a list of Strings using ArrayList from collection framework, perform built-in operations.	**07-06-2022**	**64**

OBJECT ORIENTED PROGRAMMING LAB

Name: NEHA ANTONY
Roll No:23
Batch:MCA-B
Date:29-03-2022

Experiment No.: 1

Aim

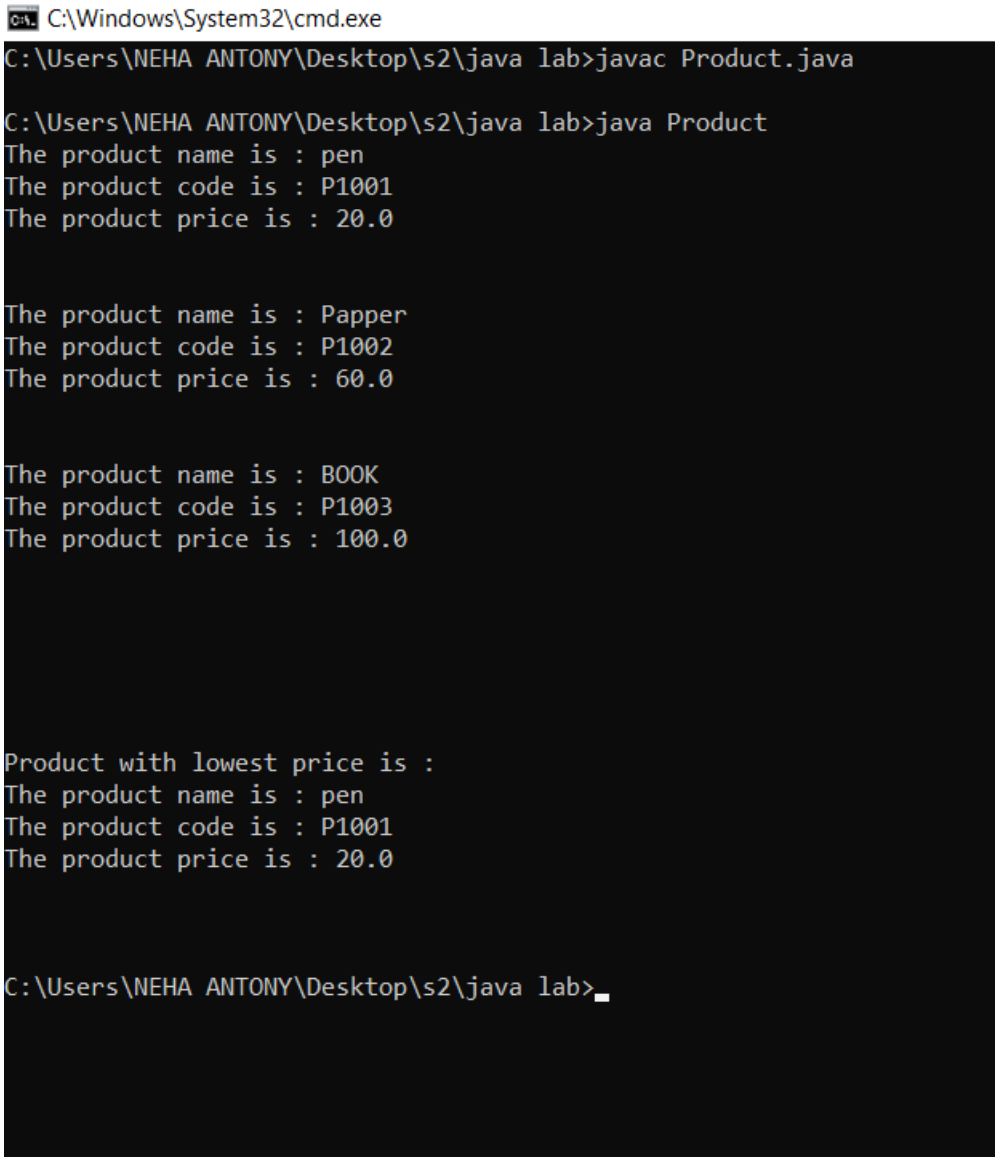
Define a class product with data members pcode,pname and and price .Create 3 objects of the class and find the product having the lowest price

Procedure

```
public class Product{

    String pcode, pname;
    double price;

    public void details(){
        System.out.println("The product name is : "+pname);
        System.out.println("The product code is : "+pcode);
        System.out.println("The product price is : "+price);
        System.out.println("\n");
    }

    public static void main(String[] args){

        Product p1= new Product();
        p1.pcode= "P1001";
        p1.pname= "pen";
        p1.price= 20;
        p1.details();

        Product p2= new Product();
        p2.pcode= "P1002";
```

```
        p2.pname= "Papper";
        p2.price= 60;
        p2.details();

        Product p3= new Product();
        p3.pcode= "P1003";
        p3.pname= "BOOK";
        p3.price= 100;
        p3.details();



        System.out.println("\n");
        if(p1.price<p2.price && p1.price<p3.price){
            System.out.println("\n\nProduct with lowest price is :");
            p1.details();
        }
        else if(p2.price < p3.price){
            System.out.println("\n\nProduct with lowest price is :");
            p2.details();
        }
        else{
            System.out.println("\n\nProduct with lowest price is :");
            p3.details();
        }
    }
}
```

Output Screenshot



OBJECT ORIENTED PROGRAMMING LAB

Name: NEHA ANTONY
Roll No:23
Batch:MCA-B
Date:06-04-2022

Experiment No.: 2

Aim

Read to matrix from the console and perform matrix addition

Procedure

```
import java.util.*;
import java.io.*;
 class mat
{
        int p,q ,m,n,i,j;
        public void readmat()
        {
                Scanner in =new Scanner(System.in);
                System.out.println("enter the row of first matrix");
                m=in.nextInt();
                System.out.println("enter the column of matrix");
                n=in.nextInt();
                int a[][]=new int[m][n];
                System.out.println("enter the elements of first matrix");
                for(i=0;i<m;i++)
                {
                        for(j=0;j<n;j++)
                        {
                                a[i][j]=in.nextInt();
                        }
                }
```

```
System.out.println("enter the row of second matrix");
p=in.nextInt();
System.out.println("enter the column of matrix");
q=in.nextInt();
System.out.println("enter the elements of second matrix");
int b[][]= new int[p][q];
for(i=0;i<p;i++)
{for(j=0;j<q;j++)
        {
                b[i][j]=in.nextInt();
        }
}

int sum[][]=new int[m][n];
for(i=0;i<m;i++)
{
        for(j=0;j<n;j++)
        {
                sum[i][j]=a[i][j]+b[i][j];
        }
}
System.out.println("Sum of the matrices are=");
for(i=0;i<m;i++)
{
        for(j=0;j<n;j++)
        {
                System.out.println(sum[i][j]);
        }
}
}
}
 public class Matrix
```

```
    {
        public static void main(String[] args)
        {
            mat m=new mat();
            m.readmat();


        }
    }
```

Output Screenshot

```
D:\lab>javac Matrix.java

D:\lab>java Matrix
enter the row of first matrix
2
enter the column of matrix
2
enter the elements of first matrix
1
1
1
1
enter the row of second matrix
2
enter the column of matrix
2
enter the elements of second matrix
1
2
3
4
Sum of the matrices are=
2
3
4
5

D:\lab>
```

OBJECT ORIENTED PROGRAMMING LAB

Name: NEHA ANTONY
Roll No:23
Batch:MCA-B
Date:06-04-2022

Experiment No.: 3

Aim

Add Complex Numbers

Procedure

```
import java.util.*;
class ComplexNumbers{
        int real, imaginary;
        ComplexNumbers(){  }
        ComplexNumbers(int real, int imaginary){
                this.real= real;
                this.imaginary= imaginary;
        }
        void complexAdd(ComplexNumbers compNum){
                int real_sum, imaginary_sum;
                real_sum= this.real+compNum.real;
                imaginary_sum= this.imaginary+compNum.imaginary;
                System.out.println("The sum of the mentioned complex numbers is : "+real_sum+" +
"+imaginary_sum+"i");
        }
        void display(){
                System.out.println("The entered complex number is : "+real+" + "+imaginary+"i");
                System.out.println("\n");
        }
        public static void main(String[] args){
                int real_num, imaginary_num;
                Scanner sc= new Scanner(System.in);
                System.out.print("Enter the real value of the 1st complex number : ");
```

```
            real_num= sc.nextInt();
            System.out.print("Enter the imaginary value of the 1st complex number : ");
            imaginary_num= sc.nextInt();
            ComplexNumbers com1= new ComplexNumbers(real_num, imaginary_num);
            com1.display();
            System.out.print("Enter the real value of the 2nd complex number : ");
            real_num= sc.nextInt();
            System.out.print("Enter the imaginary value of the 2nd complex number : ");
            imaginary_num= sc.nextInt();
            ComplexNumbers com2= new ComplexNumbers(real_num, imaginary_num);
            com2.display();
        com1.complexAdd(com2);
    }
}
```

Output Screenshot

```
Microsoft Windows [Version 10.0.22000.556]
(c) Microsoft Corporation. All rights reserved.

C:\Users\NEHA ANTONY\Desktop\s2\java lab>javac ComplexNumbers.java

C:\Users\NEHA ANTONY\Desktop\s2\java lab>java ComplexNumbers
Enter the real value of the 1st complex number : 3
Enter the imaginary value of the 1st complex number : 4
The entered complex number is : 3 + 4i


Enter the real value of the 2nd complex number : 6
Enter the imaginary value of the 2nd complex number : 8
The entered complex number is : 6 + 8i


The sum of the mentioned complex numbers is : 9 + 12i

C:\Users\NEHA ANTONY\Desktop\s2\java lab>
```

OBJECT ORIENTED PROGRAMMING LAB

Name: NEHA ANTONY
Roll No:23
Batch:MCA-B
Date:06-04-2022

Experiment No.: 4

Aim

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

Procedure

```
import java.util.Scanner;
public class symMatrix{
    public static void main(String args[]){
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter the size of Row \t: ");
        int row = sc.nextInt();
        System.out.print("Enter the size of Cols \t: ");
        int col = sc.nextInt();
        int[][] a = new int[row][col];
        boolean yes = true;
        int i, j;
        if(row == col){
            for(i=0; i<row; i++){
                for(j=0; j<col; j++){
                    System.out.print("Enter ("+i+","+j+")th Value \t: ");
                    a[i][j] = sc.nextInt();
                }
            }
            System.out.println("\nMatrix A :");
            for(i=0; i<row; i++){
                for(j=0; j<col; j++){
                    System.out.print(a[i][j]+"\t");
```

```
                    }System.out.println("\n");
                }
                for(i=0; i<row; i++){
                    for(j=0; j<col; j++){
                        if(a[i][j] != a[j][i]){
                            yes = false;
                        }
                    }
                }
                if(yes){
                    System.out.println("The Matrix is Symmetric\n");
                }
                else
                    System.out.println("The Matrix is NOT Symmetric\n");
            }
            else
                System.out.println("The Rows and Columns are NOT equal.");
}
}
```

Output Screenshot

```
D:\neha>javac symMatrix.java

D:\neha>java symMatrix
Enter the size of Row   : 2
Enter the size of Cols  : 2
Enter (0,0)th Value     : 1
Enter (0,1)th Value     : 0
Enter (1,0)th Value     : 0
Enter (1,1)th Value     : 1

Matrix A :
1       0

0       1

The Matrix is Symmetric


D:\neha>
```

```
D:\lab>javac Matrix.java

D:\lab>java Matrix
enter the row of first matrix
2
enter the column of matrix
2
enter the elements of first matrix
1
1
1
1
enter the row of second matrix
2
enter the column of matrix
2
enter the elements of second matrix
1
2
3
4
Sum of the matrices are=
2
3
4
5

D:\lab>
```

OBJECT ORIENTED PROGRAMMING LAB

Name: Neha Antony
Roll No:23
Batch: MCA B
Date:17/05/2022

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.

Procedure

```
class CPU
{
float price=20000;
class Processor
{
int NoOfCores=4;
String  Manufacturers="Rayzen s5";
}
static class RAM
{
static int Memory=8;
String M_Manufacturer="HP";
}
}
public class List
{
public static void main(String args[])
{
CPU C1 =new CPU();
CPU.Processor Pro =C1.new Processor();
CPU.RAM R=new CPU.RAM();
System.out.println("Price=" + C1.price);
System.out.println("Number of cores" + Pro.NoOfCores);
```

```
System.out.println("Manufacturers=" + Pro.Manufacturers);

System.out.println("Memory=" +CPU.RAM.Memory);

System.out.println("Memory Manufacturers=" + R.M_Manufacturer);

}

}
```

Output Screenshot

```
C:\Users\Student\Documents\Neha Antony>javac List.java

C:\Users\Student\Documents\Neha Antony>java List
Price=20000.0
Number of cores4
Manufacturers=Rayzen s5
Memory=8
Memory Manufacturers=HP

C:\Users\Student\Documents\Neha Antony>
```

OBJECT ORIENTED PROGRAMMING LAB

Name: Neha Antony
Roll No:23
Batch: MCA B
Date:24/04/2022

Experiment No.: 6

Aim

Program to sort strings.

Procedure

```
import java.util.*;
public class Stringsort
{
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]);

}

}

}
```

Output Screenshot

```
C:\Users\NEHA ANTONY\Desktop\s2\java lab>javac Stringsort.java

C:\Users\NEHA ANTONY\Desktop\s2\java lab>java Stringsort
enter the size of the array
4
enter the Strings into the array
manu
james
willam
rahu
Sorted array elements:

james
manu
rahu
willam

C:\Users\NEHA ANTONY\Desktop\s2\java lab>
```

OBJECT ORIENTED PROGRAMMING LAB

Name: Neha Antony
Roll No:23
Batch:MCA-B
Date:24/04/2022

Experiment No.: 7

Aim

Program to search an element in array

Procedure

```
import java.util.*;
import java.lang.*;
public class ArraySearch
{
static Scanner s=new Scanner(System.in);
 public static void main(String args[])
  {
            int a,b;
            int [] A =new int[10];
            Scanner s=new Scanner(System.in);
            System.out.println("enter the number of elements");
            a=s.nextInt();
        System.out.println("enter the data into matrix");
        inputdata(a,A);
        System.out.println("Array Search");
        System.out.println("Enter the element you want to search");
        b=s.nextInt();
        search(a,b,A);

  }

 public static void inputdata(int a,int [] A)
  {
```

```
        for(int i=0;i<a;i++)
        {

          A[i]=s.nextInt();
        }

 }

public static void search(int a, int b,int [] A)
{
      int f=0;
      for(int i=0;i<a;i++)
      {
       if(A[i]==b)
        {
         f++;
        }
        }
       if(f==0)
      System.out.print("Element is not found in the array");
       else
      System.out.print("Element is found in the array");
}

}
```

Output Screenshot

```
C:\Users\NEHA ANTONY\Desktop\s2\java lab>javac ArraySearch.java

C:\Users\NEHA ANTONY\Desktop\s2\java lab>java ArraySearch
enter the number of elements
5
enter the data into matrix
5
4
7
3
2
Array Search
Enter the element you want to search
3
Element is found in the array
```

OBJECT ORIENTED PROGRAMMING LAB

Name: NEHA ANTONY
Roll No:23
Batch:MCA-B
Date:30-05-2022

Experiment No.: 8

Aim

Perform string manipulations

Procedure

```
import java.util.*;
class StringManip
{
String s1;
String s2;
int len;
Scanner sc=new Scanner(System.in);
String concat_string(String str1,String str2)
{
return str1.concat(str2);
}
int countLength(String str1)
{
return str1.length();
}
String caseConvert(String str1)
{
if(str1.equals(str1.toUpperCase()))

return str1.toLowerCase();
else
return str1.toUpperCase();
}
String replaceSubstring(String str1,String str2,String str3)
{
return str1.replace(str3,str2);
}
String sortString(String str1)
{
char[] a=new char[str1.length()];
a=str1.toCharArray();
Arrays.sort(a);
str1=new String(a);
return str1;}
int returnCharPos(String str1,char ch)
{return str1.indexOf(ch);
```

```
}}
public class StringManipulation
{
public static void main(String args[])
{
StringManip ob=new StringManip();
int opt;
String str1,str2,str3;
char ch;
Scanner sc=new Scanner(System.in);
do
{
System.out.println("__________Menu_____________________");
System.out.println(" 1. FIND AN INDEX OF A CHARACTER IN A STRING");
System.out.println(" 2. CONCATENATE TWO STRINGS");
System.out.println(" 3. REPLACE A SUBSTRING");
System.out.println(" 4. SEE THE LENGTH OF A STRING");
System.out.println(" 5. CONVERT THE CASE OF STRING");
System.out.println(" 6. EXIT");
System.out.println("_____________________________________________");
System.out.println("Enter your option=");
opt=sc.nextInt();
switch(opt)
{
case 1:  sc.nextLine();
System.out.println("Enter a string : ");
str1=sc.nextLine();
System.out.println(" Enter a character to be searched: ");
ch=sc.next().charAt(0);
System.out.println(" The character "+ch+" found at "+ob.returnCharPos(str1,ch)+" in the string "+str1);
break;
case 2:  sc.nextLine();
System.out.println(" Enter string 1: ");
str1=sc.nextLine();
System.out.println(" Enter string 2: ");
str2=sc.nextLine();
System.out.println("After concatenating the above string, we get "+ob.concat_string(str1,str2));
break;
case 3: sc.nextLine();
System.out.println(" Enter a string : ");
str1=sc.nextLine();
System.out.println(" Enter a word:  ");
str2=sc.nextLine();
System.out.println(" Enter a substring : ");
str3=sc.nextLine();
if(str1.contains(str3))
```

```
System.out.println("Replacing "+str3+" with the word "+str2+" and the result is : "+ob.replaceSubstring(str1,str2,str3));
else
System.out.println(" Substring do not match !!!");
break;
case 4: sc.nextLine();
System.out.println(" Enter a string : ");
str1=sc.nextLine();
System.out.println(" The length of the string is : "+ob.countLength(str1));
break;
case 5: sc.nextLine();
System.out.println(" Enter a string to be converted: (Enter either in capital or not)");
str2=sc.nextLine();
if(str2.equals(str2.toUpperCase())==false && str2.equals(str2.toLowerCase())==false)
System.out.println(" Enter in correct format\n");
System.out.println(" The converted string is : "+ob.caseConvert(str2));
break;
case 6: System.exit(0);
default: System.out.println(" INVALID CHOICE !!!");
}}
while(opt!=6);
}}
```

Output Screenshot

```
C:\Users\Student\Documents\Neha Antony>java StringManipulation
_________Menu_____________________
 1. FIND AN INDEX OF A CHARACTER IN A STRING
 2. CONCATENATE TWO STRINGS
 3. REPLACE A SUBSTRING
 4. SEE THE LENGTH OF A STRING
 5. CONVERT THE CASE OF STRING
 6. EXIT
_____________________________________________
Enter your option=
1
Enter a string :
Hi how are u
 Enter a character to be searched:
u
 The character u found at 11 in the string Hi how are u
_________Menu_____________________
 1. FIND AN INDEX OF A CHARACTER IN A STRING
 2. CONCATENATE TWO STRINGS
 3. REPLACE A SUBSTRING
 4. SEE THE LENGTH OF A STRING
 5. CONVERT THE CASE OF STRING
 6. EXIT
_____________________________________________
Enter your option=
2
 Enter string 1:
hello guys
 Enter string 2:
welcome to kerala
After concatenating the above string, we get hello guyswelcome to kerala
_________Menu_____________________
 1. FIND AN INDEX OF A CHARACTER IN A STRING
 2. CONCATENATE TWO STRINGS
 3. REPLACE A SUBSTRING
 4. SEE THE LENGTH OF A STRING
 5. CONVERT THE CASE OF STRING
 6. EXIT
_____________________________________________
Enter your option=
6

C:\Users\Student\Documents\Neha Antony>
```

OBJECT ORIENTED PROGRAMMING LAB

Name: Neha Antony

Roll No:23

Batch: MCA B

Date:17/05/2022

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

Procedure

```
import java.util.*;
public class Employee
{
int eNo;
String eName;
Float eSalary;
public void read()
{
Scanner in =new Scanner(System.in);
System.out.println("Enter the Employee Number=");
eNo=in.nextInt();
System.out.println("Enter the Employee Name=");
eName=in.next();
System.out.println("Enter the Employee Salary=");
eSalary=in.nextFloat();
}
public void display()
{
System.out.println("_____Employee Details_____");
System.out.println(" Number=" + eNo);
System.out.println(" Name=" + eName);
System.out.println(" Salary=" + eSalary);
                System.out.println("-----------------------------------------------------------");
}
```

```
public static void main(String Args[])
{
                    Scanner in =new Scanner(System.in);
                    int n,i,Item;
                System.out.println("enter the number of employees=");
  n=in.nextInt();
                    Employee E[] =new Employee[n];
                    for(i=0;i<n;i++)
                    {
                    E[i] =new Employee();
                    E[i].read();
                    }
                    for(i=0;i<n;i++)
                    {

                    E[i].display();
                    }
                    int item,flag=0 ;
                System.out.println("Enter the EMployee number to be searched");
                    Item=in.nextInt();
  for(i=0;i<n;i++)
                    {
                    if(E[i].eNo==Item)
                     {
                    flag=1;
                    break;
                    }
                    }
                    if(flag==1)

                    E[i].display();
```

```
else
System.out.println("Employee doesn't Exist");
}
}
```

Output Screenshot

```
C:\Users\Student\Documents\Neha Antony>javac Employee.java

C:\Users\Student\Documents\Neha Antony>java Employee
enter the number of employees=
2
Enter the Employee Number=
1
Enter the Employee Name=
Manu
Enter the Employee Salary=
34000
Enter the Employee Number=
2
Enter the Employee Name=
Rahul
Enter the Employee Salary=
45000
____________________Employee Details____________________
 Number=1
 Name=Manu
 Salary=34000.0
------------------------------------------------------------
____________________Employee Details____________________
 Number=2
 Name=Rahul
 Salary=45000.0
------------------------------------------------------------
Enter the EMployee number to be searched
1
____________________Employee Details____________________
 Number=1
 Name=Manu
 Salary=34000.0
------------------------------------------------------------

C:\Users\Student\Documents\Neha Antony>
```

OBJECT ORIENTED PROGRAMMING LAB

Name: Neha Antony

Roll No:23

Batch: MCA B

Date:17/05/2022

Experiment No.: 10

Aim

Area of different shapes using overloaded functions

Procedure

```
class Area
{
int shape(int l, int b)
{
return l*b;
}

double shape(double l,double b)
{
return (0.5*l*b);
}
double shape(double  l)
{
return (3.14*l*l);
}
}
public class MainArea
{
public static void main(String Args[])
{
Area A =new Area();
System.out.println("Area of rectangle=" + A.shape(4,5));
System.out.println("Area of  Triangle=" + A.shape(6,5));
System.out.println("Area of circle=" + A.shape(4));
}
```

Output Screenshot

```
C:\Users\Student\Documents\Neha Antony>javac MainArea.java

C:\Users\Student\Documents\Neha Antony>java MainArea
Area of rectangle=20
Area of  Triangle=30
Area of circle=50.24

C:\Users\Student\Documents\Neha Antony>
```

OBJECT ORIENTED PROGRAMMING LAB

Name: Neha Antony
Roll No:23
Batch: MCA B
Date:18/05/2022

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.

Procedure

```
import java.util.*;
 class  Employee
{
String Empid;
String Name;
String Salary;
String Address;
public Employee(String Empid,String Name,String Salary,String Address)
{
this.Empid= Empid;
this.Name=Name;
this.Salary=Salary;
this.Address=Address;
}
}
 class Teacher extends Employee
{
String Department;
String Subject;
public Teacher(String Empid,String Name,String Salary,String Address,String Department,String Subject)
```

```
{
super(Empid,Name,Salary,Address);
this.Department=Department;
this.Subject=Subject;
}
public void read()
{   Scanner in =new Scanner(System.in);
System.out.println("enter the Employ id=");
      Empid=in.nextLine();
      System.out.println("enter the Name=");
      Name=in.nextLine();
      System.out.println("enter the Salary=");
      Salary=in.nextLine();
      System.out.println("enter the Address=");
      Address=in.nextLine();
      System.out.println("enter the Department=");
      Department=in.nextLine();
      System.out.println("Enter the Subject=");
      Subject=in.nextLine();
}
public void display()
{   System.out.println("_____Employee Details_____");
   System.out.println("Empid=" + Empid);
   System.out.println("Name="+ Name);
   System.out.println("Salary=" + Salary);
   System.out.println("Address=" + Address);
   System.out.println("Department=" + Department);
   System.out.println("Subject=" + Subject);

System.out.println("++++++++++++++++++++++++++++++++++++++++++++++++++++++++++");
}
}
```

```
public class Inherit
{
   public static void main(String Args[])
   { int i,n;
    Scanner in =new Scanner(System.in);
      System.out.println("Enter the Number of employee=");
      n=in.nextInt();
          Teacher T[] = new Teacher[n];
      for(i=0;i<n;i++)
      {
      T[i]=new Teacher("Empid","Name","Salary","Address","Department","Subject");
      T[i].read();
      }

   for(i=0;i<n;i++)
   {
       T[i].display();
   }
   }
   }
```

Output Screenshot

```
C:\Users\NEHA ANTONY\Desktop\s2\java lab>javac Inherit.java

C:\Users\NEHA ANTONY\Desktop\s2\java lab>java Inherit
Enter the Number of employee=
2
enter the Employ id=
123
enter the Name=
Rahul
enter the Salary=
34000
enter the Address=
Sorgam
enter the Department=
MCA
Enter the Subject=
acn
enter the Employ id=
129
enter the Name=
Manu
enter the Salary=
230000
enter the Address=
Pavana
enter the Department=
MCA
Enter the Subject=
Java
______Employee Details__________
Empid=123
Name=Rahul
Salary=34000
Address=Sorgam
Department=MCA
Subject=acn
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
______Employee Details__________
Empid=129
Name=Manu
Salary=230000
Address=Pavana
Department=MCA
Subject=Java
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

C:\Users\NEHA ANTONY\Desktop\s2\java lab>
```

OBJECT ORIENTED PROGRAMMING LAB

Name: Neha Antony
Roll No:23
Batch: MCA 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.

Procedure

```
import java.util.*;
class Person{
    String Name;
    String Gender;
    String Address;
    String Age;
    public Person(String Name,String Gender,String Address,String Age)
{
this.Name=Name;
this.Gender=Gender;
this.Address=Address;
this.Age=Age;
}
}
 class  Employee extends Person
{
String Empid;
String Company_Name;
```

```
String Qualification;
String Salary;


public Employee(String Name,String Gender,String Address,String Age ,String Empid,String Company_Name, String Qualification,String Salary)
{
super(Name,Gender,Address,Age);
this.Empid= Empid;
this.Company_Name=Company_Name;
this.Qualification=Qualification;
this.Salary=Salary;


}
}
 class Teacher extends Employee
{
    String Teacherid;
String Department;
String Subject;
public Teacher(String Name,String Gender,String Address,String Age,String Empid,String Company_Name,String Qualification,String Salary,String Teacherid, String Department,String Subject)
{
super(Name,Gender,Address,Age,Empid,Name,Qualification, Salary);
this.Teacherid=Teacherid;
this.Department=Department;
this.Subject=Subject;
}
public void read()
{   Scanner in =new Scanner(System.in);
System.out.println("enter the Name=");
      Name=in.nextLine();
      System.out.println("enter the Gender=");
      Gender=in.nextLine();
```

```
        System.out.println("enter the Address=");
        Address=in.nextLine();
        System.out.println("enter the Age=");
        Age=in.nextLine();
        System.out.println("enter the Employ id=");
        Empid=in.nextLine();
        System.out.println("enter the Company Name=");
        Company_Name=in.nextLine();
        System.out.println("enter the Qualification=");
        Qualification=in.nextLine();
        System.out.println("enter the Salary=");
        Salary=in.nextLine();
        System.out.println("enter the Teacher id=");
        Teacherid=in.nextLine();
        System.out.println("enter the Department=");
        Department=in.nextLine();
        System.out.println("Enter the Subject=");
        Subject=in.nextLine();
}
public void display()
{   System.out.println("_____Employee Details_____");

    System.out.println("Name="+ Name);
    System.out.println("Gender=" + Gender);
    System.out.println("Address=" + Address);
    System.out.println("Age=" + Age);
    System.out.println("Empid=" + Empid);
    System.out.println("Company Name=" + Company_Name);
    System.out.println("Qualification=" + Qualification);
    System.out.println("Salary=" + Salary);
    System.out.println("Teacher id=" + Teacherid);
    System.out.println("Department=" + Department);
```

```
   System.out.println("Subject=" + Subject);


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

}

}

public class Inheritance

{

   public static void main(String Args[])

   { int i,n;

    Scanner in =new Scanner(System.in);

      System.out.println("Enter the Number of employee=");

      n=in.nextInt();

          Teacher T[] = new Teacher[n];

      for(i=0;i<n;i++)

      {

      T[i]=new
Teacher("Name","Gender","Address","Age","Empid","Name","Qualification","Salary","Teacherid","De
partment","Subject");

      T[i].read();

      }


   for(i=0;i<n;i++)

   {

      T[i].display();

   }

   }

   }
```

Output Screenshot

```
C:\Users\NEHA ANTONY\Desktop\s2\java lab>javac Inheritance.java

C:\Users\NEHA ANTONY\Desktop\s2\java lab>java Inheritance
Enter the Number of employee=
1
enter the Name=
Manoj
enter the Gender=
male
enter the Address=
Anthkattad
enter the Age=
29
enter the Employ id=
123
enter the Company Name=
TCS
enter the Qualification=
MCA
enter the Salary=
359000
enter the Teacher id=
T45
enter the Department=
MCA
Enter the Subject=
Java
______Employee Details__________
Name=Manoj
Gender=male
Address=Anthkattad
Age=29
Empid=123
Company Name=TCS
Qualification=MCA
Salary=359000
Teacher id=T45
Department=MCA
Subject=Java
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

C:\Users\NEHA ANTONY\Desktop\s2\java lab>
```

OBJECT ORIENTED PROGRAMMING LAB

Name: NEHA ANTONY
Roll No:23
Batch:MCA-B
Date:18-05-2022

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.

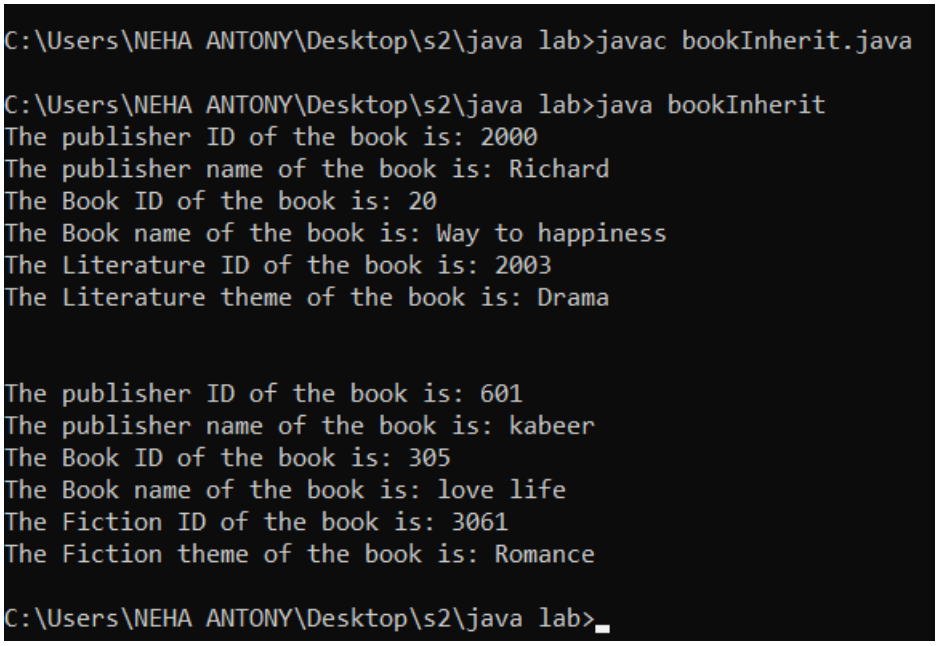
Procedure

```
import java.util.Scanner;


class Publisher{
   int publisher_id;
   String publisher_name;


   Publisher(int publisher_id, String publisher_name){
      this.publisher_id= publisher_id;
      this.publisher_name= publisher_name;
   }
}
class Book extends Publisher{


   int book_id;
   String book_name;
   Book(int publisher_id, String publisher_name, int book_id, String book_name) {
      super(publisher_id, publisher_name);
      this.book_id= book_id;
      this.book_name= book_name;
   }
}
class Literature extends Book{
   int literature_id;
```

```
    String literature_theme;


    Literature(int publisher_id, String publisher_name, int book_id, String book_name, int literature_id, String literature_theme) {
        super(publisher_id, publisher_name, book_id, book_name);
        this.literature_id= literature_id;
        this.literature_theme= literature_theme;
    }
    void displayDetails() {
        System.out.println("The publisher ID of the book is: " + this.publisher_id);
        System.out.println("The publisher name of the book is: " + this.publisher_name);
        System.out.println("The Book ID of the book is: " + this.book_id);
        System.out.println("The Book name of the book is: " + this.book_name);
        System.out.println("The Literature ID of the book is: " + this.literature_id);
        System.out.println("The Literature theme of the book is: " + this.literature_theme);
    }
}
class Fiction extends Book{
    int fiction_id;
    String fiction_theme;
    Fiction(int publisher_id, String publisher_name, int book_id, String book_name, int fiction_id, String fiction_theme) {
        super(publisher_id, publisher_name, book_id, book_name);
        this.fiction_id= fiction_id;
        this.fiction_theme= fiction_theme;
    }
    void displayDetails() {
        System.out.println("The publisher ID of the book is: " + this.publisher_id);
        System.out.println("The publisher name of the book is: " + this.publisher_name);
        System.out.println("The Book ID of the book is: " + this.book_id);
        System.out.println("The Book name of the book is: " + this.book_name);
        System.out.println("The Fiction ID of the book is: " + this.fiction_id);
        System.out.println("The Fiction theme of the book is: " + this.fiction_theme);
```

```
    }

}

public class bookInherit{

    public static void main(String[] args) {

        Literature literature= new Literature(2000,"Richard",20,"Way to happiness",2003,"Drama");

        Fiction fiction= new Fiction(601, "kabeer", 305, "love life", 3061, "Romance");

        literature.displayDetails();

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

        fiction.displayDetails();

    }

}
```

Output Screenshot



OBJECT ORIENTED PROGRAMMING LAB

Name: NEHA ANTONY
Roll No:23
Batch:MCA-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

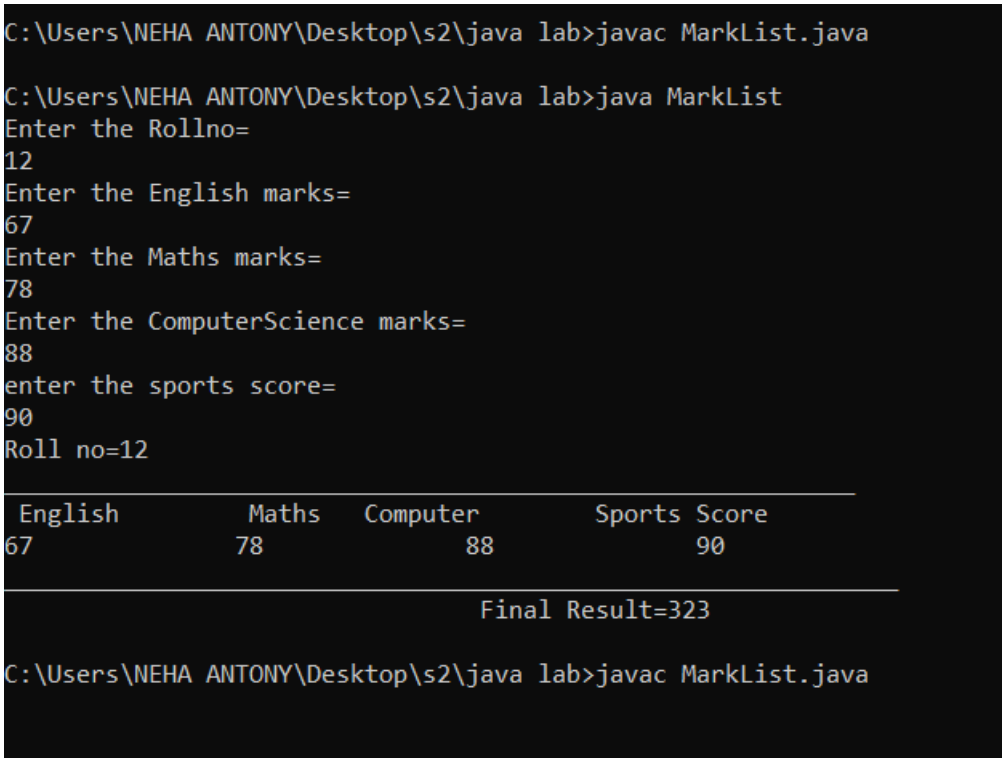
Procedure

```
import java.util.*;
class Student
{
   int rno,m1,m2,m3;
   public Student(int rno,int m1,int m2,int m3)
   {
      this.rno=rno;
      this.m1=m1;
      this.m2=m2;
      this.m3=m3;
   }
}
class Sports extends Student
{
   int score;
   public  Sports(int rno,int m1,int m2,int m3,int score)
   {
      super(rno,m1,m2,m3);
      this.score=score;
   }
}
class Result extends Sports
```

```
{
     public Result(int rno,int m1,int m2,int m3,int score)
    {
        super(rno,m1,m2,m3,score);
    }
   public int calc(int m1,int m2,int m3,int score)
    {
        int total=0;
        total=total+m1+m2+m3+score;
        return total;
    }
    void display()
    {
        System.out.println("Roll no=" +rno);

        System.out.println("_____");
        System.out.println(" English \t Maths \t Computer \t Sports Score");
        System.out.println(m1 +"\t\t"+ m2 +"\t\t" +m3 +"\t\t"+ score);

System.out.println("_____");
    }
}
public class MarkList
{
    public static void main(String args[])
    {   int Total,rno,m1,m2,m3,score;
        Scanner in=new Scanner(System.in);
        System.out.println("Enter the Rollno=");
        rno=in.nextInt();
        System.out.println("Enter the English marks=");
        m1=in.nextInt();
        System.out.println("Enter the Maths marks=");
```

```
        m2=in.nextInt();
        System.out.println("Enter the ComputerScience marks=");
        m3=in.nextInt();
        System.out.println("enter the sports score=");
        score=in.nextInt();
        Result r=new Result(rno,m1,m2,m3,score);
        Total=r.calc(m1,m2,m3,score);
        r.display();
        System.out.println("\t\t\t\t Final Result=" +Total);
    }
}
```

Output Screenshot



OBJECT ORIENTED PROGRAMMING LAB

Name: NEHA ANTONY
Roll No:23
Batch:MCA-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

Procedure

```
import java.io.*;
import java.util.*;
interface Shape
{
final double  pi=3.14;
void Area(float l ,float b);
void Perimeter(float l , float b);
}
class Circle implements Shape
{
public void Area(float r,float x)
{ double area=pi*r*r;
System.out.println("Area Of circle=" + area);
}
public void Perimeter (float r,float x)
{
        double Peri=2*pi*r;
System.out.println("Perimeter of the circle=" + Peri);
}
}
class Rectangle implements Shape
```

```
{
 public void Area(float l,float b)
{
        float area=l*b;
System.out.println("Area Of Rectangle=" + area);
}
public void Perimeter (float l,float b)
{ float Peri=2*(l+b);
System.out.println("Perimeter of the Rectangle=" + Peri );
}
}
public class MainClass2
{
public static void main(String Args[])
{int ch;
float r,l,b;
Circle C=new Circle();
Rectangle R=new Rectangle();
Scanner in=new Scanner(System.in);
do
{
System.out.println("_________Menu__________");
System.out.println("1.Area  of Circle");
System.out.println("2. Perimeter  of Circle");
System.out.println("3. Area of Rectangle");
System.out.println("4. Perimeter of Rectangle");
System.out.println("enter the choice=");
ch=in.nextInt();
switch(ch)
{
case 1:
      C.Area(5,5);
```

```
        break;
case 2:
        C.Perimeter(5,5);
        break;
case 3:
       R.Area(5,7);
       break;
case 4:
        R.Perimeter(5,7);
         break;
case 5: System.exit(0);
        break;
}
}while(ch!=5);
}
}
```

Output Screenshot

```
C:\Users\NEHA ANTONY\Desktop\s2\java lab>javac MainClass2.java

C:\Users\NEHA ANTONY\Desktop\s2\java lab>java MainClass2
________Menu__________
1.Area  of Circle
2. Perimeter  of Circle
3. Area of Rectangle
4. Perimeter of Rectangle
enter the choice=
1
Area Of circle=78.5
________Menu__________
1.Area  of Circle
2. Perimeter  of Circle
3. Area of Rectangle
4. Perimeter of Rectangle
enter the choice=
2
Perimeter of the circle=31.400000000000002
________Menu__________
1.Area  of Circle
2. Perimeter  of Circle
3. Area of Rectangle
4. Perimeter of Rectangle
enter the choice=
3
Area Of Rectangle=35.0
________Menu__________
1.Area  of Circle
2. Perimeter  of Circle
3. Area of Rectangle
4. Perimeter of Rectangle
enter the choice=
4
Perimeter of the Rectangle=24.0
________Menu__________
1.Area  of Circle
2. Perimeter  of Circle
3. Area of Rectangle
4. Perimeter of Rectangle
enter the choice=
5

C:\Users\NEHA ANTONY\Desktop\s2\java lab>
```

OBJECT ORIENTED PROGRAMMING LAB

Name: NEHA ANTONY
Roll No:23
Batch:MCA-B
Date:24-05-2022

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

Procedure

```
import java.util.*;
import java.time.format.DateTimeFormatter;
import java.time.LocalDateTime;
interface calculate
{
   int calc(int Quanlity,int unit_Price);
}
class Order implements calculate
{
   int Productid,Quanlity,unit_Price;
   String Name;
   public Order(int Productid,String Name,int Quanlity,int unit_Price)
   {
```

```
    this.Productid=Productid;
    this.Name=Name;
    this.Quanlity=Quanlity;
    this.unit_Price=unit_Price;
  }
  public int calc(int Quanlity,int unit_Price)
  {
     int Total=unit_Price*Quanlity;
     return Total;
  }
   public void display()
  {
  int Total;
  int Net_Amount=0;
  Total=calc(Quanlity,unit_Price);
  System.out.println("\t" + Productid + "\t\t" + Name + "\t\t" + Quanlity + "\t\t" + unit_Price + "\t\t" +
Total);
}
}
public class Bill
{
  public static void main(String[] args)
  {   int i,n;
     int Total;
     int Net_Amount=0;
      int Productid,Quanlity,unit_Price;
      String Name,Ono;
      Scanner in =new Scanner(System.in);
      DateTimeFormatter dtf = DateTimeFormatter.ofPattern("yyyy/MM/dd HH:mm:ss");
       LocalDateTime now = LocalDateTime.now();
      System.out.println("Enter the order no");
      Ono=in.nextLine();
```

```
        System.out.println("enter the number of product");
        n=in.nextInt();
         Order Obj[]=new Order[n];
        for(i=0;i<n;i++)
        {
           System.out.println("Enter the Product id");
           Productid=in.nextInt();
           System.out.println("Enter the Name");
           Name=in.nextLine() + in.nextLine();
           System.out.println("Enter the  Quanlity ");
           Quanlity=in.nextInt();
           System.out.println("Enter the unit price");
           unit_Price=in.nextInt();
           Obj[i] =new Order(Productid,Name,Quanlity,unit_Price);
           Total= Obj[i].calc(Quanlity,unit_Price);
           Net_Amount=Net_Amount+Total;
        }
          System.out.println("Order no: " +Ono);
          System.out.println( "Date : " +dtf.format(now));
         System.out.println(" \t Productid \t Name \t Quanlity \t unit_Price \t Total \t ");

System.out.println("_____
_____");
        for(i=0;i<n;i++)
        {
             Obj[i].display();
        }

System.out.println("_____
_____");
       System.out.println("\t\t\t\t\t    Net Amount="+  Net_Amount);


    }
}
```

Output Screenshot

```
C:\Users\NEHA ANTONY\Desktop\s2\java lab>javac Bill.java

C:\Users\NEHA ANTONY\Desktop\s2\java lab>java Bill
Enter the order no
T11
enter the number of product
2
Enter the Product id
101
Enter the Name
Soap
Enter the  Quanlity
5
Enter the unit price
45
Enter the Product id
105
Enter the Name
Pen
Enter the  Quanlity
10
Enter the unit price
10
Order no: T11
Date : 2022/05/29 23:10:46
         Productid       Name    Quanlity        unit_Price      Total

-------------------------------------------------------------------------
        101             Soap            5               45              225
        105             Pen             10              10              100

-------------------------------------------------------------------------
                                        Net Amount=325

C:\Users\NEHA ANTONY\Desktop\s2\java lab>S
```

OBJECT ORIENTED PROGRAMMING LAB

Name: NEHA ANTONY
Roll No:23
Batch:MCA-B
Date:31-05-2022

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.

Procedure

Package Graphics

Area.java

```
package g.Graphics;

public interface Area

{

    void area();

}
```

Rectangle.java

```
package g.Graphics;

public class Rectangle implements Area

{

    int l,b;

   public Rectangle(int l,int b)

   {

       this.l=l;

       this.b=b;


   }

    public void area()

    {

        int area;

        area=l*b;
```

```
        System.out.println("AREA of Rectangle="+ area);


    }
}
```

Circle.java

```
package g.Graphics;
public class Circle implements Area
    {
        int r;
        public Circle(int r)
        {
            this.r=r;
        }
     public void area()
    {
        double area;
        area=3.14*r*r;
        System.out.println("AREA of Circle="+ area);
    }
    }
```

Square.java

```
package g.Graphics;
public class Square implements Area
    {
        int a;
     public Square(int a)
     {
         this.a=a;
    }
     public void area()
    {
        float area;
```

```
        area=a;
        System.out.println("AREA of Square="+ area);
    }
}
```

Triangle.java

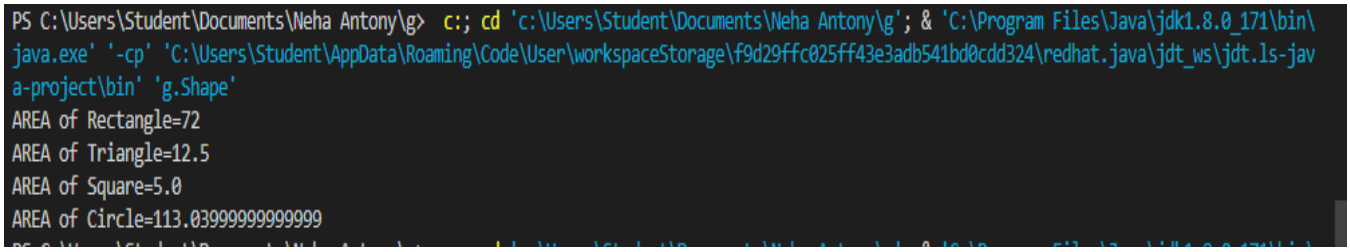
```
package g.Graphics;
 public class Triangle implements Area
    {
        int b,h;
        public Triangle(int b,int h)
        {
            this.b=b;
            this.h=h;
        }
     public void area()
    {
        float area;
        area=(float) (0.5*b*h);
        System.out.println("AREA of Triangle="+ area);
    }
    }
```

Shape.java

```
package g;
import g.Graphics.Circle;
import g.Graphics.Rectangle;
import g.Graphics.Square;
import g.Graphics.Triangle;
public class Shape{
    public static void main(String[] args){
        g.Graphics.Area r=new Rectangle(8,9);
        g.Graphics.Area T=new Triangle(5,5);
        g.Graphics.Area S=new Square(5);
```

```
        g.Graphics.Area C=new Circle(6);

        r.area();

        T.area();

        S.area();

        C.area();

    }

}
```

Output Screenshot



OBJECT ORIENTED PROGRAMMING LAB

Name: NEHA ANTONY
Roll No:23
Batch:MCA-B
Date:31-05-2022

Experiment No.: 18

Aim

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

Procedure

```
import java.util.*;
public class CustomerException {
   // A custom exception class
   public static class InvalidUserException extends Exception {
      public InvalidUserException() {
         super("Invalid username / password provided!");
      }
   }
   public static void main(String[] args) {
      String username =" John";
      String password ="john";
      try {
         if (username.equals("user") && password.equals("pass")) {
            System.out.println("Authenticated successfully!");
         } else {
            throw new InvalidUserException();
         }
      } catch (InvalidUserException e) {
         System.out.println(e);
      }
   }
}
```

Output Screenshot

```
C:\Users\Student\Documents\Neha Antony>javac CustomerException.java

C:\Users\Student\Documents\Neha Antony>java CustomerException
CustomerException$InvalidUserException: Invalid username / password provided!

C:\Users\Student\Documents\Neha Antony>
```

OBJECT ORIENTED PROGRAMMING LAB

Name: NEHA ANTONY
Roll No:23
Batch:MCA-B
Date:01-06-2022

Experiment No.: 19

Aim

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

Procedure

```
import java.util.Scanner;
public class AverageException{
   public static class InvalidNumberException extends Exception {

      public InvalidNumberException() {
         super("Please provide a valid number!");
      }
   }
   public static void main(String [] args){
      Scanner sc=new Scanner(System.in);
      int c,num,sum=0;
      double avg;
      System.out.println("enter the count");
      c=sc.nextInt();
      for(int i=0;i<c;i++){
         try{
            num=sc.nextInt();
            if(num>0){
               sum+=num;
            }else{
               i--;
```

```
                throw new InvalidNumberException();
            }
        }
        catch(InvalidNumberException e){
            System.out.println(e.getMessage());
        }
    }
    System.out.println("sum is"+sum);
    System.out.println("avg is"+sum/c);
  }
}
```

Output Screenshot

```
C:\Users\NEHA ANTONY\Desktop\s2\java lab\co4>javac AverageException.java

C:\Users\NEHA ANTONY\Desktop\s2\java lab\co4>java AverageException
enter the count
5
1
2
3
4
5
sum is15
avg is3

C:\Users\NEHA ANTONY\Desktop\s2\java lab\co4>
```

OBJECT ORIENTED PROGRAMMING LAB

Name: NEHA ANTONY
Roll No:23
Batch:MCA-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)

Procedure

```
import java.util.Scanner;


class Fib extends Thread{
    int f,n1=0,n2=1,n3;
    Fib(int c){
        this.f=c;
    }
    public void run(){
        System.out.println("fib is "+n1);
        System.out.println("fib is "+n2);
        for(int i=2;i<this.f;++i) {
        n3=n1+n2;
        System.out.println("fib is "+n3);
        n1=n2;
        n2=n3;
    }
    }
}


class even extends Thread{
    int range;
    even(int range){
```

```
        this.range=range;
    }
    public void run(){
        for(int i=0;i<this.range;i++){
            if(i%2==0){
                System.out.println("even num is "+i);
            }
        }
    }
}


public class MulThread {
    public static void main(String [] args){
        int c,range;
        Scanner sc=new Scanner(System.in);
        System.out.println("enter the count of Fibinooci");
        c=sc.nextInt();
        Fib fi=new Fib(c);
        System.out.println("enter the range of even number");
        range=sc.nextInt();
        even ev = new even(range);
        fi.start();
        ev.start();

    }

}
```

Output Screenshot

```
C:\Users\NEHA ANTONY\Desktop\s2\java lab\co4>javac MulThread.java

C:\Users\NEHA ANTONY\Desktop\s2\java lab\co4>java MulThread
enter the count of Fibinooci
5
enter the range of even number
10
even num is 0
even num is 2
fib is 0
even num is 4
fib is 1
even num is 6
even num is 8
fib is 1
fib is 2
fib is 3

C:\Users\NEHA ANTONY\Desktop\s2\java lab\co4>
```

OBJECT ORIENTED PROGRAMMING LAB

Name: NEHA ANTONY
Roll No:23
Batch:MCA-B
Date:01-06-2022

Experiment No.: 21

Aim

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

Procedure

```
class Stack {
  private int arr[];
  private int top;
  private int N;
  Stack(int size) {
    arr = new int[size];
    N= 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 == N- 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

```
C:\Users\Student\Documents\Neha Antony\1-06-2022>javac Stack.java

C:\Users\Student\Documents\Neha Antony\1-06-2022>java Stack
Inserting 1
Inserting 2
Inserting 3
Stack: 1, 2, 3,
After popping out
1, 2,
C:\Users\Student\Documents\Neha Antony\1-06-2022>
```

OBJECT ORIENTED PROGRAMMING LAB

Name: NEHA ANTONY

Roll No:23

Batch:MCA-B

Date:07-06-2022

Experiment No.: 22

Aim

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

Procedure

```
import java.util.*;

public class arraylist{

   public static void main(String[] args) {

      ArrayList<String> arrayList= new ArrayList<>();

      arrayList.add("Anakha");

      arrayList.add("Aneena");

      arrayList.add("Mahima");

      arrayList.add("Shone");

      System.out.println("The elements of the arraylist is - "+arrayList);

      Collections.sort(arrayList);

      System.out.println("\nThe ArrayList Sort : "+arrayList); // ArrayList Sort

      Collections.addAll(arrayList,"Sreeraj","Hemanth","Amal","Shalvin","Athin");

      System.out.println("\nAdding new items in the arraylist is : "+arrayList); // ArrayList AddAll

      Collections.sort(arrayList, Collections.reverseOrder()); //Arraylist in reverse order

      System.out.println("\nThe reverse order of the arraylist : "+arrayList);

      System.out.println("\nThe maximum element of the arraylist : "+Collections.max(arrayList));
//Max elements in the arraylist

   }

}
```

Output Screenshot

```
C:\Users\Student\Documents\Neha Antony\07-06-2022>javac arraylist.java

C:\Users\Student\Documents\Neha Antony\07-06-2022>java arraylist
The elements of the arraylist is - [Anakha, Aneena, Mahima, Shone]

The ArrayList Sort : [Anakha, Aneena, Mahima, Shone]

Adding new items in the arraylist is : [Anakha, Aneena, Mahima, Shone, Sreeraj, Hemanth, Amal, Shalvin, Athin]

The reverse order of the arraylist : [Sreeraj, Shone, Shalvin, Mahima, Hemanth, Athin, Aneena, Anakha, Amal]

The maximum element of the arraylist : Sreeraj

C:\Users\Student\Documents\Neha Antony\07-06-2022>
```

OBJECT ORIENTED PROGRAMMING LAB

Name: NEHA ANTONY
Roll No:23
Batch:MCA-B
Date:07-06-2022

Experiment No.: 23

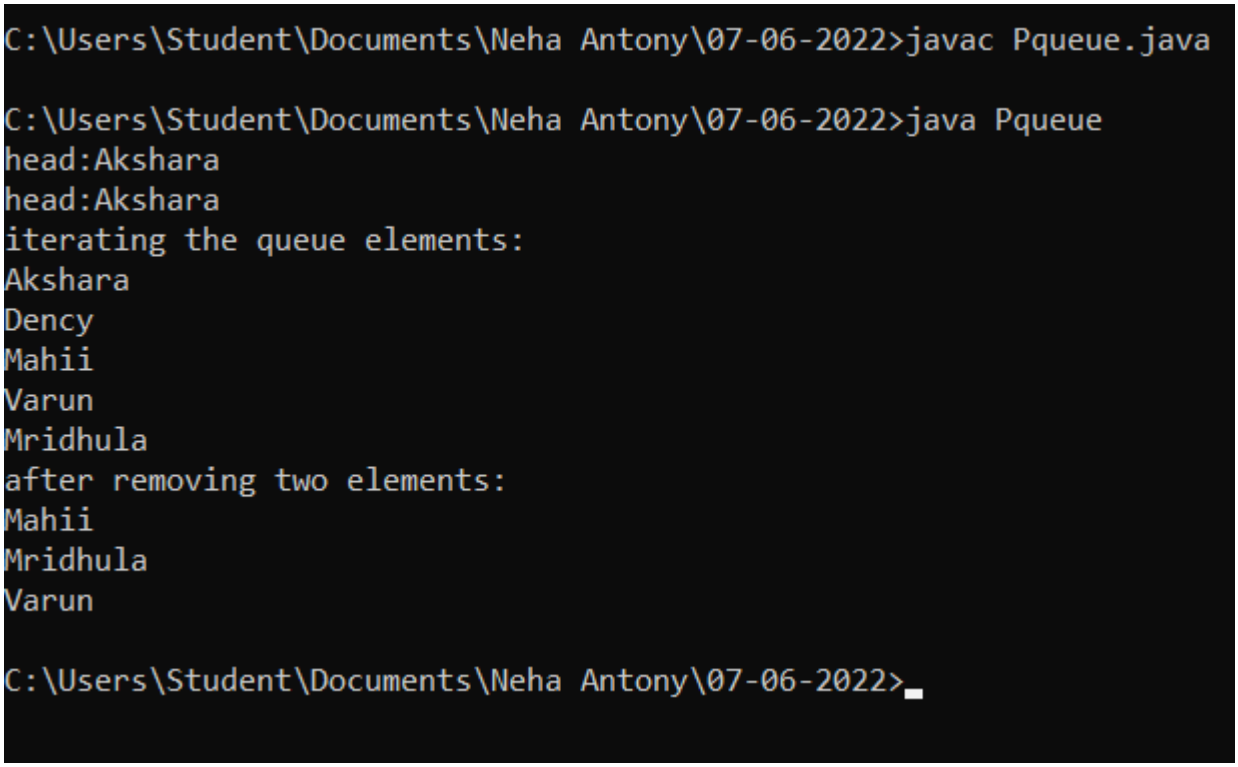
Aim

Program to demonstrate the creation of queue object using the PriorityQueue class

Procedure

```
import java.util.*;
class Pqueue{
public static void main(String args[]){
PriorityQueue<String> queue=new PriorityQueue<String>();
queue.add("Mahii");
queue.add("Varun");
queue.add("Akshara");
queue.add("Dency");
queue.add("Mridhula");
System.out.println("head:"+queue.element());
System.out.println("head:"+queue.peek());
System.out.println("iterating the queue elements:");
Iterator itr=queue.iterator();
while(itr.hasNext()){
System.out.println(itr.next());
}
queue.remove();
queue.poll();
System.out.println("after removing two elements:");
Iterator<String> itr2=queue.iterator();
while(itr2.hasNext()){
System.out.println(itr2.next());
}
}
}
```

Output Screenshot



OBJECT ORIENTED PROGRAMMING LAB

Name: NEHA ANTONY
Roll No:23
Batch:MCA-B
Date:07-06-2022

Experiment No.: 24

Aim

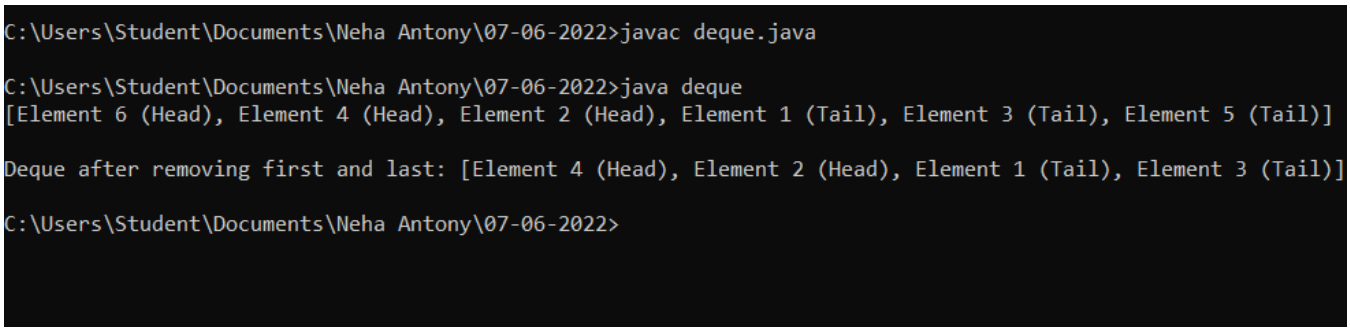
Program to demonstrate the addition and deletion of elements in deque.

Procedure

```
import java.util.*;
 class deque
{
 public static void main(String[] args)
{
Deque<String> deque = new LinkedList<String>();
// We can add elements to the queue
// in various ways
// Add at the last
deque.add("Element 1 (Tail)");
// Add at the first
deque.addFirst("Element 2 (Head)");
// Add at the last
deque.addLast("Element 3 (Tail)");
// Add at the first
deque.push("Element 4 (Head)");
// Add at the last
deque.offer("Element 5 (Tail)");
// Add at the first
deque.offerFirst("Element 6 (Head)");
System.out.println(deque + "\n");
// We can remove the first element
// or the lastelement.
deque.removeFirst();
deque.removeLast();
```

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

}

}
```

Output Screenshot



OBJECT ORIENTED PROGRAMMING LAB

Name: NEHA ANTONY
Roll No:23
Batch:MCA-B
Date:07-06-2022

Experiment No.: 25

Aim

Write a Java program to compare two hash set

Procedure

```
import java.util.*;
 public class  Hash {
 public static void main(String[] args) {
   // Create a empty hash set
      HashSet<String> h_set = new HashSet<String>();
   // use add() method to add values in the hash set
       h_set.add("Red");
       h_set.add("Green");
       h_set.add("Black");
       h_set.add("White");

       HashSet<String>h_set2 = new HashSet<String>();
       h_set2.add("Red");
       h_set2.add("Pink");
       h_set2.add("Black");
       h_set2.add("Orange");
       //comparison output in hash set
       HashSet<String>result_set = new HashSet<String>();
      for (String element : h_set){
         System.out.println(h_set2.contains(element) ? "Yes" : "No");
       }
   }
}
```

Output Screenshot

```
C:\Users\Student\Documents\Neha Antony\07-06-2022>javac Hash.java

C:\Users\Student\Documents\Neha Antony\07-06-2022>java Hash
Yes
No
Yes
No

C:\Users\Student\Documents\Neha Antony\07-06-2022>^S
```

OBJECT ORIENTED PROGRAMMING LAB

Name: NEHA ANTONY
Roll No:23
Batch:MCA-B
Date:07-06-2022

Experiment No.: 26

Aim

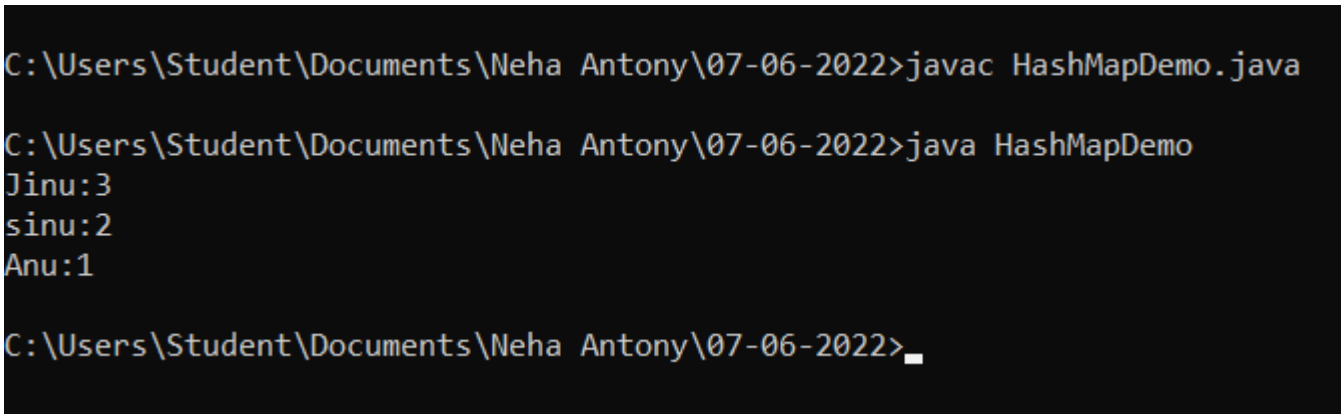
Program to demonstrate the working of Map interface by adding, changing and removing

elements

Procedure

```
import java.util.*;

class HashMapDemo {

public static void main(String args[]) {

Map<String, Integer> hm = new HashMap<String, Integer>();

hm.put("Anu", new Integer(1));

hm.put("sinu", new Integer(2));

hm.put("Jinu", new Integer(3));

// Traversing through the map

for (Map.Entry<String, Integer> me : hm.entrySet()) {

System.out.print(me.getKey() + ":");

System.out.println(me.getValue()); }

} }
```

Output Screenshot



OBJECT ORIENTED PROGRAMMING LAB

Experiment No.: 27

Name: Neha Antony
Roll No:23
Batch: MCA-B
Date: 7-06-2022

Aim

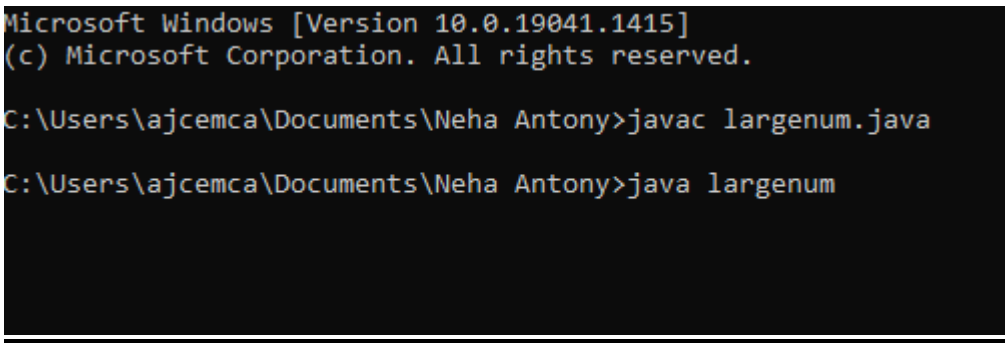
Program to find maximum of three numbers using AWT.

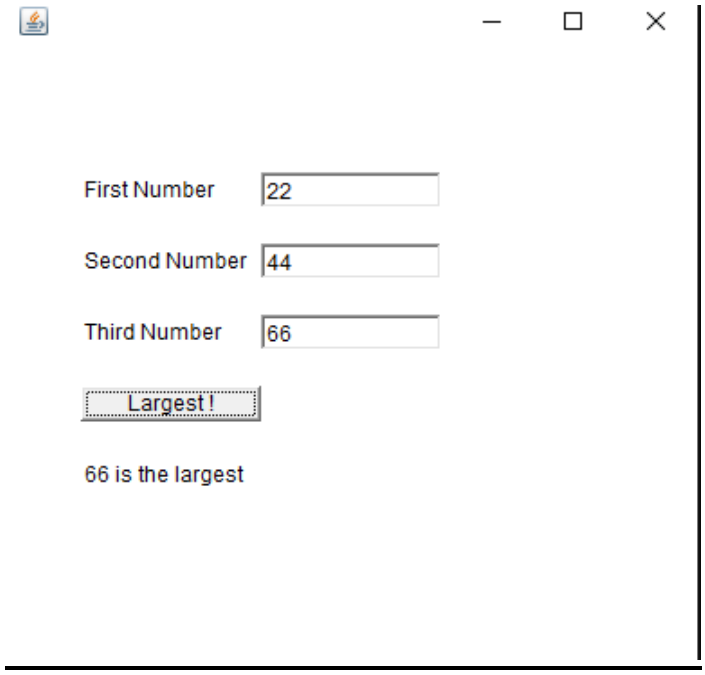
Procedure

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

public class largenum 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 !");

        largenum(){
                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 largenum();
    }

    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




First Number
22
Second Number
44
Third Number
66
Largest !
66 is the largest

OBJECT ORIENTED PROGRAMMING LAB

Name: Neha Antony
Roll No:23
Batch: MCA-B
Date:11-06-2022

Experiment No.: 28

Aim

Implement a simple calculator using AWT components.

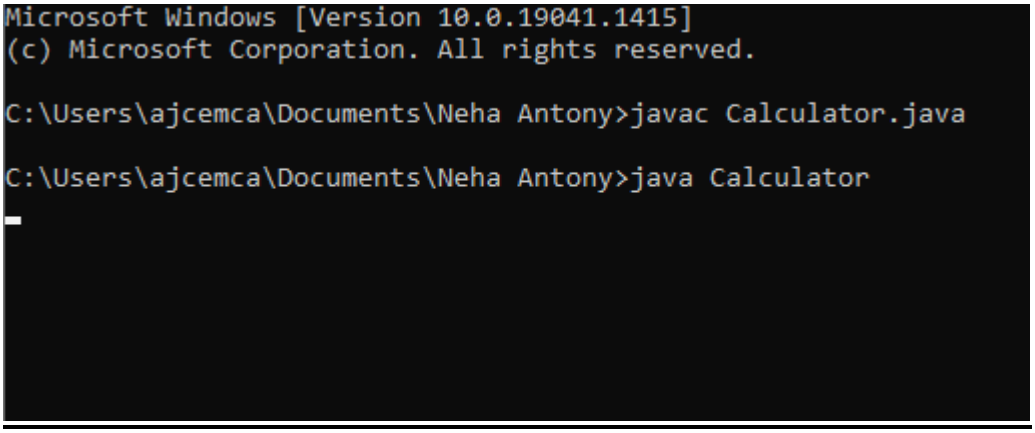
Procedure

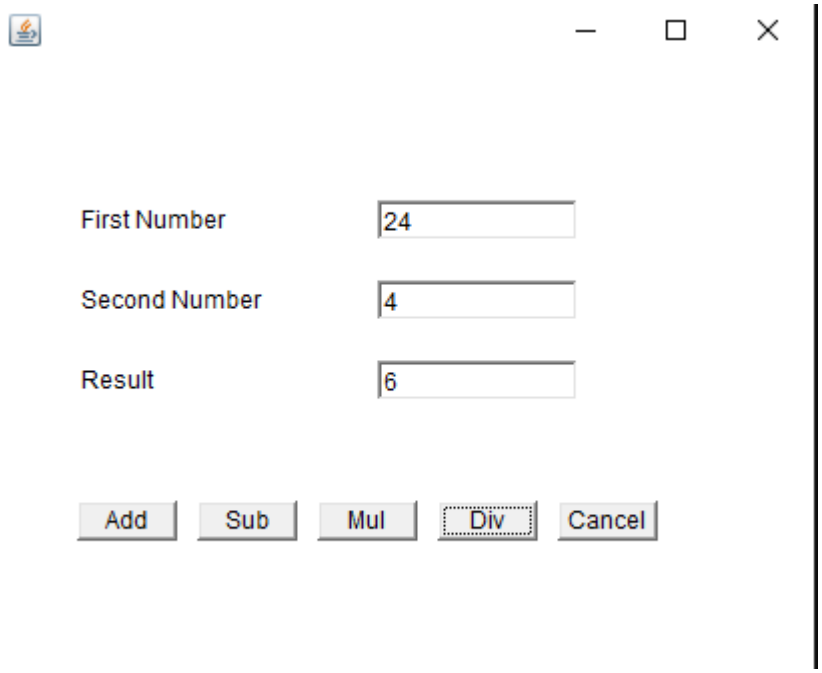
```
import java.awt.*;
import java.awt.event.*;

public class Calculator implements ActionListener
{

Frame f=new Frame();
Label l1=new Label("First Number");
Label l2=new Label("Second Number");
Label l3=new Label("Result");
TextField t1=new TextField();
TextField t2=new TextField();
TextField t3=new TextField();
Button b1=new Button("Add");
Button b2=new Button("Sub");
Button b3=new Button("Mul");
Button b4=new Button("Div");
Button b5=new Button("Cancel");
Calculator()
{

l1.setBounds(50,100,100,20);
l2.setBounds(50,140,100,20);
l3.setBounds(50,180,100,20);
t1.setBounds(200,100,100,20);
t2.setBounds(200,140,100,20);
t3.setBounds(200,180,100,20);
b1.setBounds(50,250,50,20);
b2.setBounds(110,250,50,20);
b3.setBounds(170,250,50,20);
b4.setBounds(230,250,50,20);
b5.setBounds(290,250,50,20);
```

```
f.add(l1);
f.add(l2);
f.add(l3);
f.add(t1);
f.add(t2);
f.add(t3);
f.add(b1);
f.add(b2);
f.add(b3);
f.add(b4);
f.add(b5);
b1.addActionListener(this);
b2.addActionListener(this);
b3.addActionListener(this);
b4.addActionListener(this);
b5.addActionListener(this);
f.setLayout(null);
f.setVisible(true);
f.setSize(400,350);
}
public void actionPerformed(ActionEvent e)
{
int n1=Integer.parseInt(t1.getText());
int n2=Integer.parseInt(t2.getText());
if(e.getSource()==b1)
{
t3.setText(String.valueOf(n1+n2));
}
if(e.getSource()==b2)
{
t3.setText(String.valueOf(n1-n2));
}
if(e.getSource()==b3)
{
t3.setText(String.valueOf(n1*n2));
}
if(e.getSource()==b4)
{
t3.setText(String.valueOf(n1/n2));
}
if(e.getSource()==b5)
{
System.exit(0);
}
```

```
}
public static void main(String...s)
{
new Calculator();
}
}
```

Output Screenshot





OBJECT ORIENTED PROGRAMMING LAB

Experiment No.: 29

Name: Neha Antony
Roll No:23
Batch:MCA-B
Date:11-06-2022

Aim

Develop a program to handle all mouse events and window events

Procedure

```
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

```
Microsoft Windows [Version 10.0.19041.1415]
(c) Microsoft Corporation. All rights reserved.

C:\Users\ajcemca\Documents\Neha Antony>javac Mouseevents.java

C:\Users\ajcemca\Documents\Neha Antony>java Mouseevents
```

OBJECT ORIENTED PROGRAMMING LAB

Experiment No.: 30

Name: Neha Antony
Roll No:23
Batch: MCA-B
Date: 11-06-2022

Aim

Develop a program to handle Key events.

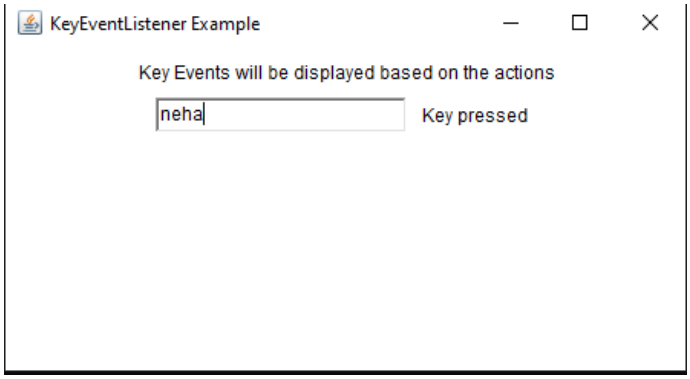
Procedure

```
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 KE implements KeyListener
{
   Label lb1, lbl2, lb;
   TextField tf1;
   Frame fr;
   String s;
   KE()
   {

      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 KE();
    }
}
```

Output Screenshot

```
Microsoft Windows [Version 10.0.19041.1415]
(c) Microsoft Corporation. All rights reserved.

C:\Users\ajcemca\Documents\Neha Antony>javac KE.java

C:\Users\ajcemca\Documents\Neha Antony>java KE
```



OBJECT ORIENTED PROGRAMMING LAB

Name: NEHA ANTONY
Roll No:23
Batch:MCA-B
Date:30-05-2022

Experiment No.: 31

Aim

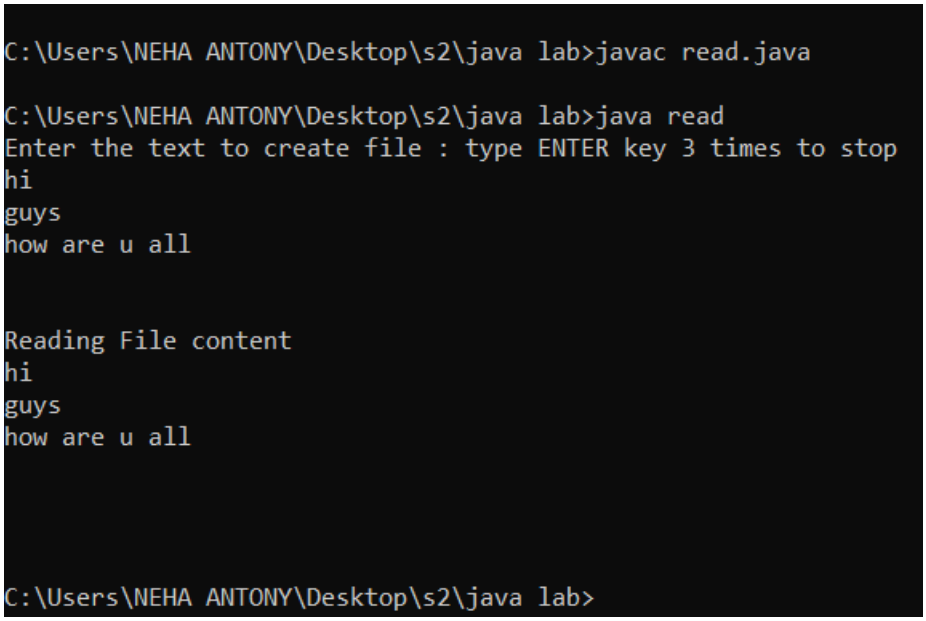
Write a program to write to a file, then read from the file and display the contents on the console.

Procedure

```
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) {
// initialize String
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 {
// create file object
File file = new File("output.txt");
// create filewriter object
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) {

// Storing every character in the string

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: NEHA ANTONY
Roll No:23
Batch:MCA-B
Date:30-05-2022

Experiment No.: 32

Aim

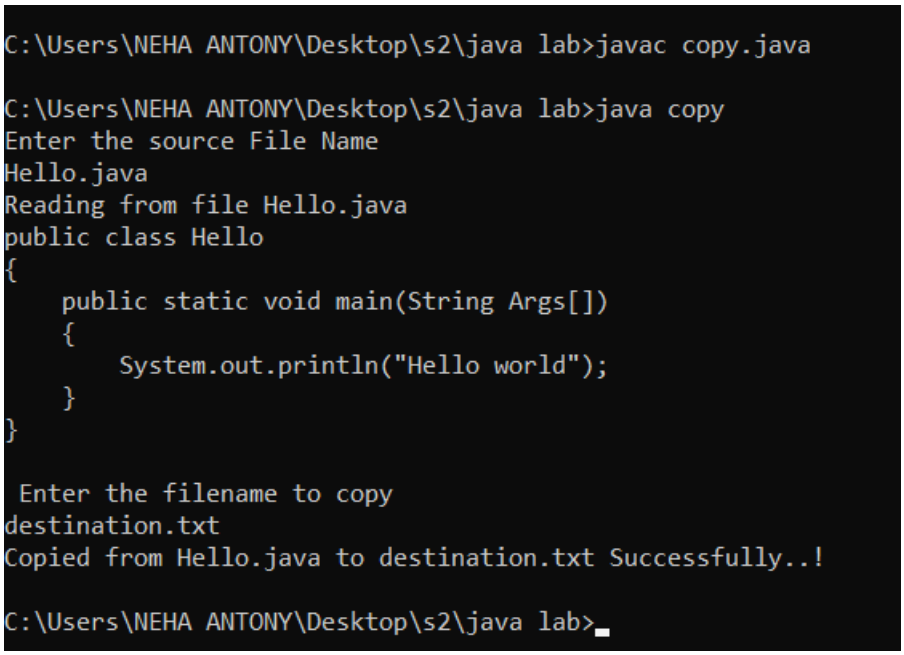
Write a program to copy one file to another.

Procedure

```
import java.io.FileReader;
import java.io.FileWriter;
import java.io.IOException;
import java.io.*;
import java.util.*;
import java.io.File;
public class copy {
public static void main(String[] args) {
Scanner scan=new Scanner(System.in);
System.out.println("Enter the source File Name");
String source=scan.nextLine();
try {
FileReader fr=new FileReader(source);
String str = "";
int i;
System.out.println("Reading from file "+source);
while ((i = fr.read()) != -1) {
// Storing every character in the string
str += (char) i;
}
System.out.println(str);
System.out.println("\n Enter the filename to copy");
String destination=scan.nextLine();
File file=new File(destination);
FileWriter fw = new FileWriter(file);
fw.write(str);
```

```
fr.close();

fw.close();

System.out.println("Copied from "+source+" to "+destination+ " Successfully..!");

} catch (Exception e) {

//TODO: handle exception

System.out.println("Exception Occured");

}

}

}
```

Output Screenshot



OBJECT ORIENTED PROGRAMMING LAB

Name: NEHA ANTONY

Roll No:23

Batch:MCA-B

Date:30-05-2022

Experiment No.:33

Aim

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

Procedure

```
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 (Exception e) {

// TODO: handle exception

System.out.println("Error");

}

}}
```

Output Screenshot

```
C:\Users\Student\Documents\Neha Antony>javac oddeven.java

C:\Users\Student\Documents\Neha Antony>java oddeven

C:\Users\Student\Documents\Neha Antony>numbers.txt

C:\Users\Student\Documents\Neha Antony>evennumbers.txt

C:\Users\Student\Documents\Neha Antony>oddnumbers.txt

C:\Users\Student\Documents\Neha Antony>
```