

OBJECT ORIENTED PROGRAMMING LAB
RECORD

HARITHA KRISHNAN

ROLL NO: 40

MCA A

PROGRAM NO: 1

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

```
public class Product {
int pcode;
String pname;
int price;
public static void main(String[] args) {
int smallest;
    Product p1 = new Product();
    Product p2 = new Product();
    Product p3 = new Product();
    p1.pcode=101;
    p1.pname="Fan";
    p1.price=7000;
    p2.pcode=102;
    p2.pname="AC";
    p2.price=25000;
    p3.pcode=103;
    p3.pname="TV";
    p3.price=15000;
    if(p1.price<p2.price)      {
    if(p3.price<p1.price) {
        smallest = p3.price;
    } else {
        smallest = p1.price;
    }
    } else {
    if(p2.price<p3.price) {
        smallest = p2.price;
    } else {
        smallest = p3.price;
    }
    }

    System.out.println(smallest + " is the cheapest.");
}
}
```

Output:



```
C:\java\abc01>java Product.java
C:\java\abc01>java Product
7000 is the cheapest.
C:\java\abc01>
```

PROGRAM NO : 2

A/M: Add complex numbers

```
public class Complex{  
    double a, b;  
    Complex(double r, double i){  
        this.a = r;  
        this.b = i;  
    }  
    public static Complex sum(Complex c1, Complex c2)  
    {  
        Complex temp = new Complex(0, 0);  
  
        temp.a = c1.a + c2.a;  
        temp.b = c1.b + c2.b;  
        return temp;  
    }  
    public static void main(String args[]) {  
        Complex c1 = new Complex(5, 4);  
        Complex c2 = new Complex(6, 3.5);  
        Complex temp = sum(c1, c2);  
        System.out.printf("Sum is: "+ temp.a+" "+ temp.b+"i");  
    }  
}
```

Output:



```
C:\javatrab\c01>java Complex.java  
C:\javatrab\c01>java Complex  
Sum is: 11.0 + 7.5i  
C:\javatrab\c01>
```

PROGRAM NO:3

A/M: Read 2 matrices from the console and perform matrix addition.

```
import
java.util.Scanner;
class matrixadd
{
public static void main(String[]args){
int row,col,i,j;
Scanner sc= new Scanner(System.in); System.out.println("Enter the number of rows");
row = sc.nextInt();
System.out.println("Enter the no.of columns");
col = sc.nextInt();

int mat1[][]=new int [row][col];
int mat2[][]=new int [row][col];
int mat3[][]=new int [row][col];

System.out.println("Enter the matrix);
for(i=0;i<row;i++)
{
for(j=0;j<col;j++)
{
mat1[i][j]=sc.nextInt();
}
System.out.println();
}
System.out.println("Enter the matrix");

for(i=0;i<row;i++)
{
for(j=0;j<col;j++)
{
```

```

mat2[i][j]=sc.nextInt();
}
System.out.println();
}
for(i=0;i<row;i++)
{
for(j=0;j<col;j++)
{
mat3[i][j]=mat1[i][j]+mat2[i][j];
}
}
System.out.println("The sum of the matrix is");

for(i=0;i<row;i++)
{
for(j=0;j<col;j++)
{
System.out.println(mat3[i][j]);
}
}
System.out.println();
}
}}

```

Output

```

E:\javalab>java matrixadd
Enter the number of rows
2
Enter the no. of columns
2
Enter the matrix
2
4
7
4
Enter the matrix
3
4
6
1
The sum of the matrix is
5
8
13
5

```

PROGRAM NO: 4

A/M: Read a matrix from the console and check whether it is symmetric or not.

```
import java.util.Scanner;

public class Symmetric

{

    public static void main(String[] args)

    {

        Scanner sc = new Scanner(System.in);

        System.out.println("Enter the no. of rows : ");

        int rows = sc.nextInt();

        System.out.println("Enter the no. of columns : ");
```

```
int cols = sc.nextInt();

int matrix[][] = new int[rows][cols];

System.out.println("Enter the elements :");

for (int i = 0; i < rows; i++)

{

for (int j = 0; j < cols; j++)

{

matrix[i][j] = sc.nextInt();

}

}

System.out.println("Printing the input matrix :");

for (int i = 0; i < rows; i++)

{

for (int j = 0; j < cols; j++)

{

System.out.print(matrix[i][j]+"\\t");

}

System.out.println();

}

if(rows != cols)

{

System.out.println("The given matrix is not a

square matrix, so it can't be symmetric.");

}
```

```
}

boolean symmetric = true;

for (int i = 0; i < rows; i++)

{

for (int j = 0; j < cols; j++)

{

if(matrix[i][j] != matrix[j][i])

{

symmetric = false;

break;

}

}

}

if(symmetric)

{

System.out.println("The given matrix is

symmetric...");

}

else

{

System.out.println("The given matrix is not

symmetric...");

}
```



```
}
```

```
sc.close();
```

```
}
```

```
}
```

OUTPUT

```
E:\javalab>javac Symmetric.java

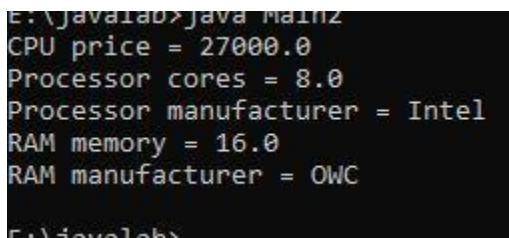
E:\javalab>java Symmetric
Enter the no. of rows :
3
Enter the no. of columns :
3
Enter the elements :
1
2
3
2
4
5
3
5
8
Printing the input matrix :
1      2      3
2      4      5
3      5      8
The given matrix is symmetric...
```

PROGRAM NO:5

A/M: 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.

```
class CPU {  
    double price=27000; class Processor{  
        double cores=8;  
        String manufacturer="Intel";  
    }  
    protected class RAM{ double memory=16;  
        String manufacturer="OWC";  
    }  
}  
  
public class Main2 {  
    public static void main(String[] args) { CPU cpu = new CPU();  
        CPU.Processor processor = cpu.new Processor();  
        CPU.RAM ram = cpu.new RAM();  
        System.out.println("CPU price = " + cpu.price);  
        System.out.println("Processor cores = " + processor.cores);  
        System.out.println("Processor manufacturer = " + processor.manufacturer);  
        System.out.println("RAM memory = " + ram.memory); System.out.println("RAM  
manufacturer = " + ram.manufacturer);  
    }  
}
```

OUTPUT

A screenshot of a terminal window showing the output of a Java program. The text is as follows:
E:\java1a0>java Main2
CPU price = 27000.0
Processor cores = 8.0
Processor manufacturer = Intel
RAM memory = 16.0
RAM manufacturer = OWC
E:\java1a0>

PROGRAM NO: 6

AIM: Program to Sort strings

```
public class sortstring1 {  
    public static void main(String[] args)  
    {  
        String  
        names[]={ "amal", "jyothi", "college", "of", "engineering"};  
        String temp;  
        Int n= names.length;  
        int i;  
        int j;  
        for(i=0;i<n;i++)  
        {  
            for(j=i+1;j<n;j++)  
            {  
                if(names[i].compareTo(names[j])>0)  
                {  
                    temp=names[i];  
                    names[i]=names[j];  
                    names[j]=temp;  
                }  
            }  
        }  
        System.out.println("the    sorted    array    of    stringis:");  
        for(i=0;i<n;i++)  
        {  
            System.out.println(names[i]);  
        }  
    }  
}
```

OUTPUT

```
E:\javalab>java sortstring1
the sorted array of string is :
amal
college
engineering
jyothi
of
```

PROGRAM NO: 7

AIM: Search an element in an array.

```
import java.util.*;

public class searchelee{

public static void main(String[] args)

{

int n,i,b,flag=0;

Scanner s=new Scanner(System.in);

System.out.println("enter the number of elements for the array:");

n=s.nextInt();

int a[]=new int[n];

System.out.println("enter the elements of the array:");

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

{

a[i]=s.nextInt();

}

System.out.println("enter the element u want to search :");

b=s.nextInt();

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

{

if(a[i]==b)

{

flag=1;

break;

}

else

{

flag=0;

}

}

}
```

```
if(flag==1)
{
    System.out.println("element found at position :"+(i+1));
}
else
{
    System.out.println("element not found");
}
}
}
```

OUTPUT

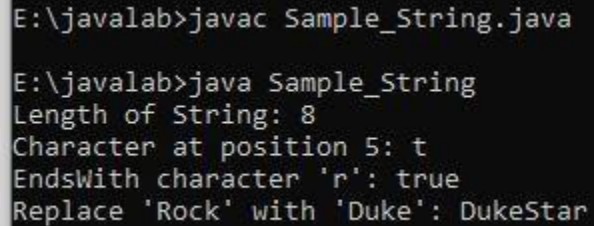
```
E:\javalab>java searchelee
enter the number of elements for the array :
2
enter the elements of the array :
12 13
enter the element u want to search :
13
element found at position :2
```

PROGRAM NO: 8

AIM: Perform string manipulations

```
public class Sample_String{  
    public static void main(String[] args){  
        String str_Sample = "RockStar";  
  
        System.out.println("Length of String: " + str_Sample.length());  
        System.out.println("Character at position 5: " + str_Sample.charAt(5));  
        System.out.println("EndsWith character 'r': " + str_Sample.endsWith("r"));  
        System.out.println("Replace 'Rock' with 'Duke': " + str_Sample.replace("Rock",  
            "Duke"));  
    }  
}
```

OUTPUT



```
E:\javalab>javac Sample_String.java  
  
E:\javalab>java Sample_String  
Length of String: 8  
Character at position 5: t  
EndsWith character 'r': true  
Replace 'Rock' with 'Duke': DukeStar
```

PROGRAM NO:9

A/M: Area of different shapes using overloaded functions

```
public class shape
{
    int s,as,ar;

    public void area(int a)//area of square
    {
        s=a; as=a*a;

        System.out.println("area of square is"+as);
    }

    public void area(double r)//area of circle
    {

        double radi=r;
        double ac=(22/7)*radi*radi;

        System.out.println("area circle is"+ac);
    }

    public void area(int l,int w)//area of rectangle
    {
        int len=l; int
        wid=w;
        ar=len*wid;

        System.out.println("area of rectangle"+ar);
    }

    public void area(int h,double r)//area of cylinder
    {
        int he=h;

        double rad=r;
        double acy=(2*(22/7)*rad*he)+((22/7)*rad*rad);
```



```
System.out.println("area of cylinder"+acy);
```

```
}
```

```
public static void main(String[] args)
```

```
{
```

```
    shape o=new shape();
```

```
    o.area(8);//area of square
```

```
    o.area(3.42);//area of circle
```

```
    o.area(6,2);//area of rectangle
```

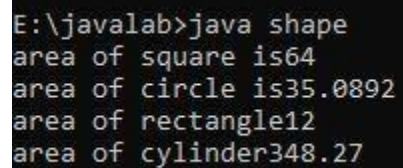
```
    o.area(10,4.7); //area of
```

```
    cylinder
```

```
}
```

```
}
```

OUTPUT

A screenshot of a Java command prompt window showing the execution of a program. The prompt is 'E:\javalab>java shape'. The output consists of four lines: 'area of square is64', 'area of circle is35.0892', 'area of rectangle12', and 'area of cylinder348.27'.

```
E:\javalab>java shape
area of square is64
area of circle is35.0892
area of rectangle12
area of cylinder348.27
```

PROGRAM NO:10

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.

```
import java.util.*;

class Employee
{
    int empid;
    String
    name,address;
    double salary;
    public Employee(int empid, String name, String address, double salary) {

        this.empid = empid;
        this.name = name;
        this.address =address;
        this.salary = salary;
    }
}

public class Teacher extends Employee
{
    String subject,department;
    public Teacher(int empid, String name, String address, double salary,String
    department,Stringsubject ) {
        super(empid, name, address, salary);

        this.subject=subject;
        this.department = department;
    }
    void display()
```

```
{
System.out.println("Empid : "+this.empid+" Name : "+this.name+" Salary : "+this.salary+"
Address : "+this.address+" department : "+this.department+" Subjects : "+this.subject);
}
public static void main(String[] args) {
    stub Scanner sc=new
Scanner(System.in);

    int n;
System.out.println("Enter number of Teachers");
n=sc.nextInt();
Teacher obj[]=new Teacher[n];
for(int i=0;i<n;i++) {
int j = i+1;
System.out.print("Enter Empid of teacher"+j+":");

int Empid = sc.nextInt();
System.out.print("Enter Name of teacher "+j+" : ");
```

```

String Name = sc.next();

System.out.print("Enter Salary of teacher "+j+" :");

double Salary=sc.nextDouble(); System.out.print("Enter
Address of teacher"+j+":");String Address = sc.next();
System.out.print("Enter department of teacher "+j+" ");
String department =sc.next();
System.out.print("Enter Subjects of teacher "+j+":");
String Subjects =sc.next();
obj[i] = new Teacher(Empid, Name, Address, Salary, department, Subjects);
}
System.out.println("\n.....\n");

System.out.println("Teacher's List \n");
for(int i=0;i<n;i++)
{ obj[i].display();
}
}
}
}

```

Output:

```

E:\javalab>java Teacher
Enter number of Teachers :
2
Enter Empid of teacher 1 : 12
Enter Name of teacher 1 : anu
Enter Salary of teacher 1 : 10000
Enter Address of teacher 1 : anuvilla kochi
Enter department of teacher 1 : Enter Subjects of teacher 1 : maths
Enter Empid of teacher 2 : 34
Enter Name of teacher 2 : sonu
Enter Salary of teacher 2 : 70000
Enter Address of teacher 2 : sonuvilla
Enter department of teacher 2 : cs
Enter Subjects of teacher 2 : english

-----
Teacher's List

Empid : 12 Name : anu Salary : 10000.0 Address : anuvilla department : kochi Subjects : maths
Empid : 34 Name : sonu Salary : 70000.0 Address : sonuvilla department : cs Subjects : english

```

PROGRAM NO:11

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.

```
import java.util.Scanner;

class Person
{
    String name,gender,address;

    int age;

    public Person(String name, String gender, String address, int
age) { super();
    this.name  =  name;
    this.gender =  gender;
    this.address = address;
    this.age = age;
    }
}

class Employee extends Person
{ int empid;
    String
    company_name,qualification;
    double salary;

    public Employee(String name, String gender, String address, int age, int empid,
Stringcompany_name,
    String qualification, double 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 subject,department;
int teacherid;

public Teacher(String name, String gender, String address, int age, int empid, String
company_name, String qualification, double salary, String subject, String department, int
teacherid) {
super(name, gender, address, age, empid, company_name, qualification,
salary);this.subject = subject;
this.department = department;
this.teacherid = teacherid;
}
void display()
{
System.out.println("....Personal details...");
System.out.println(" Name : "+this.name+" Gender : "+this.gender+" Age
:"+this.age);System.out.println("...Employee details.          ");
System.out.println("Empid : "+this.empid + " company_name : "+this.company_name+"
Salary :"+this.salary+" Address : "+this.address+" qualification : "+this.qualification);
System.out.println("...Teacher's details. ");
System.out.println(" teacherid : "+this.teacherid+ " department : "+this.department+"
Subjects :"+this.subject);
}
}

public class Main3 {
public static void main(String[]
args)    {    Scanner    s=new
Scanner(System.in); int n;

System.out.println("Enter number of Teachers :");

```

```

n=s.nextInt();
Teacher  obj[]=new  Teacher[n];
for(int i=0;i<n;i++) {
System.out.println("Enter the person name:");
System.out.println("Enter the Gender: ");
String gen1=s.next();
System.out.println("Enter the Address: ");
String adr1=s.next();
System.out.println("Enter the Age:");
int age1=s.nextInt();
System.out.println("Enter the Employee id: ");
int id1=s.nextInt();
System.out.println("Enter the Company name: ");
String cname1=s.next();
System.out.println("Enter the Salary:");
double sal1=s.nextDouble();
System.out.println("Enter the Qualification:");
String qu1=s.next();
System.out.println("Enter the Teacher id: ");
int tid1=s.nextInt();
System.out.println("Enter the Department:");
String dept1=s.next();
System.out.println("Enter the Subject:");
String sub1=s.next();
obj[i]=new Teacher(nam1,gen1,adr1,age1,id1,cname1,qu1,sal1,sub1,dept1,tid1);
}
System.out.println("\n-----\n");
for(int i=0;i<n;i++) {
obj[i].display();
}
}
}

```

OUTPUT:

```

-----
....Personal details...
Name : rani Gender : female Age :23
...Employee details....
Empid : 12 company_name : wipro Salary : 10000.0 Address : ranivila qualification : cs
...Teacher's details...
teacherid : 34 department : cs Subjects : maths
....Personal details...
Name : raju Gender : male Age :44
...Employee details....
Empid : 10 company_name : tcs Salary : 2.0 Address : rajubhavan qualification : bcom
...Teacher's details...
teacherid : 24 department : chemistry Subjects : chemistry
E:\java\lab\

```

PROGRAM NO:12

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.

```

import
java.util.Scanner;

class Publisher {
String Pubname;
Publisher()
{
Scanner s=new Scanner(System.in);
System.out.println("Enter publisher name");

Pubname=s.next();
}
}

class Book extends Publisher
{
String title, author;
int price;

```



```

Book()
{
Scanner      s=new      Scanner(System.in);
System.out.println("Enter Title of the book");
title=s.next();
System.out.println("Enter Author's name");
author=s.next();
System.out.println("Enter price");
price=s.nextInt();
}
}

class Literature extends Book
{
Literature()
{
System.out.println("Literature Books");
}
void display()
{
System.out.println("Publisher name: "+Pubname);System.out.println("Title of the
book:      "+title);      System.out.println("Author's      name:      "+author);
System.out.println("Price: "+price);
}
}

class Fiction extends Literature
{
Fiction()
{
System.out.println("Friction Books");
}
}

```

```
void display()
{
super.display();
}

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

System.out.println ("Enter the No of literature book: ");

int a=s.nextInt();
Literature L[]=new
Literature[a];
for(int i=0;i<a;i++)
{
L[i]=new Literature();
}
System.out.println("Enter the No of Fiction book:");

int b=s.nextInt();
Fiction F[]=new
Fiction[b];
for(int i=0;i<b;i++)
{
F[i]=new Fiction();
}
int no;

System.out.println("Enter your choice of book");
no=s.nextInt();
int type =no;
switch (no)

{ case 1:

System.out.println("... Details of literature books");
```

```
For(int i=0;i<a;i++)
L[i].display(
);
break;
case 2:
System.out.println("... Details of fiction books");
for(int i=0;i<b;i++)
F[i].display();
break;
default:
System.out.println("Wrong
input");
}
}
}
```

OUTPUT:

```
E:\javalab>java Fiction
Enter the No of literature book:
2
Enter publisher name
sreeravi
Enter Title of the book
mazha
Enter Author's name
renjith
Enter price
200
Literature Books
Enter publisher name
gopi
Enter Title of the book
yatra
Enter Author's name
viswam
Enter price
300
Literature Books
Enter the No of Fiction book:
1
Enter publisher name
sancharam
Enter Title of the book
newone
Enter Author's name
steeven
Enter price
150
Literature Books
Friction Books
Enter your choice of book
2
.....Details of fiction books
Publisher name: sancharam
Title of the book: newone
Author's name: steeven
Price: 150
```

PROGRAM NO:13

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

```
interface student{
void acad();
}
interface sports{
void score();
}
Class result implements
student,sports{

public void acad()
{
System.out.println("Student details");
int physics=30;
int
chemistry=40; int
maths=45;
int english=50;
int computer=50;
System.out.println("Marks");
System.out.println("Physics:"+physics);
System.out.println("Chemistry:"+chemistry);
System.out.println("Mathematics:"+maths);
System.out.println("English:"+english);
System.out.println("Computer:"+computer);
}
public void score()
{
System.out.println("Sports
details");
```

```

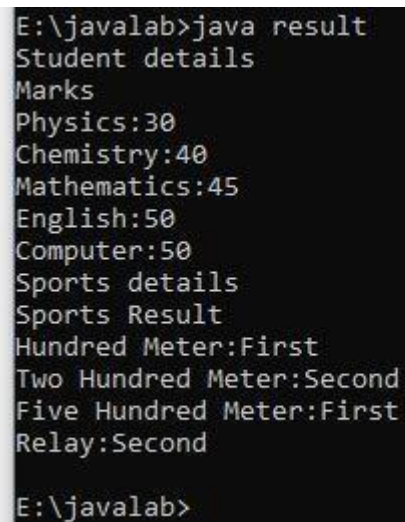
String hundred="First";
String twohundred="Second";
String fivehundred="First";

String    relay="Second";
System.out.println("Sports
Result");
System.out.println("Hundre
d Meter:"+hundred);
System.out.println("Two Hundred Meter:"+twohundred);
System.out.println("Five Hundred Meter:"+fivehundred);System.out.println("Relay:"+relay);
}
public static void main(String[] args)
{
result o= new result();
o.acad();
o.score();
}

}
}

```

OUTPUT:



```

E:\javalab>java result
Student details
Marks
Physics:30
Chemistry:40
Mathematics:45
English:50
Computer:50
Sports details
Sports Result
Hundred Meter:First
Two Hundred Meter:Second
Five Hundred Meter:First
Relay:Second
E:\javalab>

```

PROGRAM NO:14

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

```
import    java.util.Scanner;

interface Shape

{

Shape

{

int l = 0, b = 0; double ar,per;

public void input()

{

    Scanners=new Scanner(System.in); System.out.print("Enter length  of rectangle:");

    l = s.nextInt();

    System.out.print("Enter breadth of rectangle:");

    b = s.nextInt();

}

public void area()

{

    ar = l * b;

    System.out.println("Area of rectangle:"+ar);

}

public void perimeter()

{

    per = 2 * (l + b);

    System.out.println("Perimeter of rectangle:"+per);

}

}

public class shape1

{
```

```
public static void main(String[] args)
{
    int n;
    Scanner s = new Scanner(System.in);
    Rectangle obj1 = new Rectangle();
    Circle obj2 = new Circle();
    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 your option:");
    n= s.nextInt();
    switch(n) {
        case 1:
            obj2.input();
            obj2.area();
            break;
        case 2:
            obj2.input();
            obj2.perimeter();
            break;
        case 3:
            obj1.input();
            obj1.area();
            break;
        case 4:
            obj1.input();
            obj1.perimeter();
            break;
        default:
            System.out.println("Invalid option");
    }
```



```
}  
  
}
```

OUTPT:

```
E:\javalab>java shape1  
1.Area of circle  
2.Perimeter of circle  
3.Area of rectangle  
4.Perimeter of rectangle  
Enter your option:  
2  
Enter radius of circle:4  
Perimeter of circle:25.12
```

PROGRAM NO:15

A/M: Prepare bill with the given format using calculate method from interface.Order No.

Date :

Product Id Name Quantity unit price
Total101 A 2 25 50
102 B 1 100 100
Net. Amount 150

```
interface bill
```

```
{
```

```
int productdetails();
```

```
}
```

```
class product1 implements
```

```
bill{
```

```
int id = 101,quantity=2,unit=25,total=0;
```

```
String name="A";
```

```
public int productdetails()
```

```
{
```

```
total = quantity * unit;
```

```
System.out.println("Product Id :"+id);
```

```

System.out.println("Name :"+name);
System.out.println("Quantity
:"+quantity);
System.out.println("Unit price
:"+unit);
System.out.println("Total
:"+total); return(total);

}
}
class product2 implements
bill{
int id = 102,quantity=
1,unit=100,total=0; String
name="B";

public int productdetails()

{

total = quantity * unit;
System.out.println("Product Id :"+id);
System.out.println("Name :"+name);
System.out.println("Quantity :"+quantity);
System.out.println("Unit price :"+unit);
System.out.println("Total :"+total); return(total);
}
}
public class productbill
{

public static void main(String[] args)
{

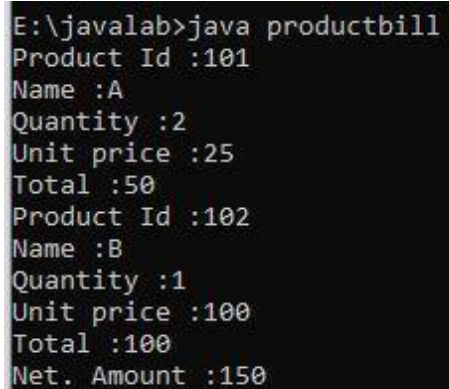
```

```
product1 p1 = new product1();  
product2 p2 = new product2();  
int t1= p1.productdetails();  
int t2= p2.productdetails();  
int t3=t1+t2;
```

```
System.out.println("Net.  
Amount :"+t3);
```

```
}  
}
```

OUTPUT:



```
E:\javalab>java productbill  
Product Id :101  
Name :A  
Quantity :2  
Unit price :25  
Total :50  
Product Id :102  
Name :B  
Quantity :1  
Unit price :100  
Total :100  
Net. Amount :150
```

PROGRAM NO:16

AIM: Java program to create generic stack and do the push and pop operation.

```
import java.io.*;
import java.util.*;
class stack<T> {
    ArrayList<T> A;
    int top = -1;
    int size;
    {
        this.size = size;
        this.A = new
ArrayList<T>(size);
    }
    void push(T X)
    {
        if (top + 1 == size)
        {
            System.out.println("Stack Overflow");
        }
        else
        {
            top = top + 1;
            if (A.size() > top)
            A.set(top, X);
            else
            A.add(X);
        }
    }
}
```

```
T top()
{
if (top == -1)
{
System.out.println("Stack
Underflow");
return null;
}
else
return A.get(top);
}

void pop()
{
if (top == -1)
{
System.out.println("Stack
Underflow");
}
else
top--;
}

boolean empty() {
return top == -1;
}

public String toString()
{
String Ans = "";
for (int i = 0; i < top; i++)
{
```

```
Ans +=
```

```
String.val
```

```
ueOf(A.g
```

```
et(i)) + "-
```

```
>";
```

```
}
```

```
Ans+=
```

```
String.valueOf(A.get(top));
```

```
return
```

```
Ans;
```

```
}
```

```
}
```

```
public class GFG {
```

```
    public static void
```

```
main(String[] args)
```

```
{
```

```
    stack<Integer> s1 = new
```

```
stack<>(3);
```

```
    s1.push(10);
```

```
    s1.push(20);
```

```
    s1.push(30);
```

```
    System.out.println("s1 after
```

```
pushing 10, 20 and 30 :\n" + s1);
```

```
    s1.pop();
```

```
    System.out.println("s1 after pop
```

```
:\n" + s1);
```

```
    stack<String> s2 = new
```

```
stack<>(3);
```

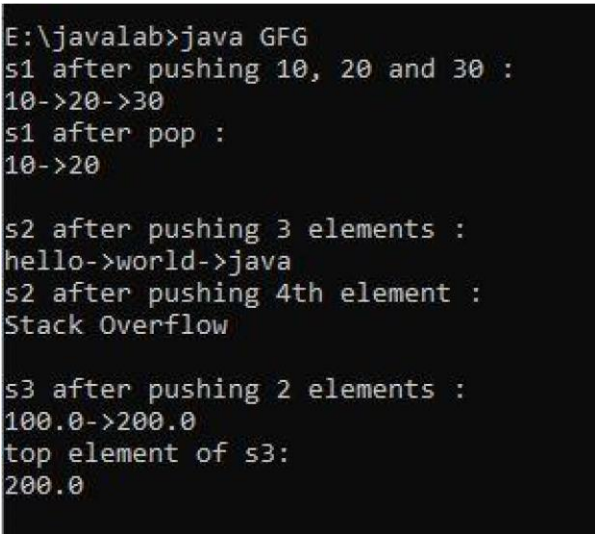
```
    s2.push("hello");
```

```

        s2.push("world");
        s2.push("java");
System.out.println("\ns2  after
pushing 3 elements :\n" + s2);
System.out.println("s2      after
pushing 4th element :");
s2.push("GFG");
stack<Float>  s3  =  new
stack<>(2);
s3.push(100.0f);
s3.push(200.0f);
System.out.println("\ns3  after
pushing 2 elements :\n" + s3);
System.out.println("top element
of s3:\n" + s3.top());
}
}

```

OUTPUT:



```

E:\javalab>java GFG
s1 after pushing 10, 20 and 30 :
10->20->30
s1 after pop :
10->20

s2 after pushing 3 elements :
hello->world->java
s2 after pushing 4th element :
Stack Overflow

s3 after pushing 2 elements :
100.0->200.0
top element of s3:
200.0

```

PROGRAM NO:17

AIM: Generic method implement bubble sort

```
public class BubbleSort
{
    static void bubbleSort(int[] arr)
    {
        int n =
        arr.length; int
        temp = 0;

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

        {

            if(arr[j-1] > arr[j])

            {

                temp = arr[j-
                1]; arr[j-1] =
                arr[j]; arr[j] =
                temp;

                }

            }

        }

        public static void main(String[] args)
        {

            int arr[] = { 2, 5, -2, 6, -3, 8, 0, -7, -9, 4 };

            System.out.println("Array Before Bubble Sort");
```



```

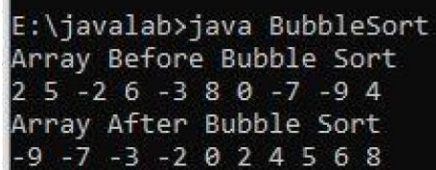
        for(int i = 0; i < arr.length; i++)
        {
            System.out.print(arr[i] + " ");
        }
        System.out.println(
);bubbleSort(arr);

        System.out.println("Array After Bubble Sort");

        for(int i = 0; i < arr.length; i++)
        {
            System.out.print(arr[i] + " ");
        }
    }
}

```

OUTPUT:



```

E:\javalab>java BubbleSort
Array Before Bubble Sort
2 5 -2 6 -3 8 0 -7 -9 4
Array After Bubble Sort
-9 -7 -3 -2 0 2 4 5 6 8

```

PROGRAM NO:18

AIM: Maintain a list of string using arraylist from a collection of framework, perform built-in operation

```

import java.util.*;
import java.util.Collections;

```

```

class JavaExample{

```

```
public static void main(String args[]){

    ArrayList<String>      alist=new
    ArrayList<String>();alist.add("Steve");

    alist.add("Tim");

    alist.add("Lucy");

    alist.add("Pat");

    alist.add("Angela)

    alist.add("Tom");

    System.out.print("original list >");

    System.out.println(alist);

    System.out.println()

    alist.add(3, "Steve");

    System.out.print("after adding element--->");

    System.out.println(alist);

    System.out.println();

    alist.set(0, "Lucy");

    System.out.print("after updating element--->");

    System.out.println(alist);

    System.out.println();

    alist.remove("tom");

    alist.remove("Angela");

    System.out.print("after  removing  element
    >");System.out.println(alist);

    System.out.println();

    System.out.print("after sorting elements---
    >");Collections.sort(alist);
```

```
for (String str : alist) {  
  
    System.out.println(  
  
    );  
  
    System.out.println(str  
  
    );  
  
}  
  
}  
  
}
```

OUTPUT:

```
E:\javalab>java JavaExample  
original list-->[Steve, Tim, Lucy, Pat, Angela, Tom]  
  
after adding element--->[Steve, Tim, Lucy, Steve, Pat, Angela, Tom]  
  
after updating element--->[Lucy, Tim, Lucy, Steve, Pat, Angela, Tom]  
  
after removing element--->[Lucy, Tim, Lucy, Steve, Pat, Tom]  
  
after sorting elements--->  
Lucy  
  
Lucy  
  
Pat  
  
Steve  
  
Tim  
  
Tom
```

PROGRAM NO:19

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

```
import java.util.Scanner;

import
java.util.InputMismatchException;

public class TestDemo
{
    public static void main(String args[])
    {
        double total = 0, N, userInput;
        Scanner    input    =    new
        Scanner(System.in); while (true)
        {
            System.out.print("Enter  how  many  numbers(N)  to  calculate
            average:"); userInput = input.nextDouble();
            if (userInput > 0)
            {
                N=userInput;
                break;
            }
            else
                System.out.println("N must be positive.");
        }
        for (int i = 0; i < N; i++)
        {
            while (true)
            {
                System.out.print("Enter number:"); try
                {
                    userInput = input.nextDouble(); total += userInput;
                    break;
                }
            }
        }
    }
}
```

```
        }
        catch (InputMismatchException e)
        {
            input.nextLine();
            System.out.println("Input must be a number. Try again");
        }
    }
}

System.out.println("Average: " + total / N);
}
}
```

OUTPUT:

```
E:\javalab>java TestDemo
Enter how many numbers(N) to calculate average:4
Enter number:2
Enter number:2
Enter number:3
Enter number:5
Average: 3.0
```

PROGRAM NO:20

AIM: Write a user defined exception class to authentication the user name and password.

```
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 CheckLoginCredential {

    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

```

E:\javalab>java CheckLoginCredential
Enter username :: hari
Enter password :: qwer
UsernameException: Username must be greater than 6 characters ???
    at CheckLoginCredential.main(CheckLoginCredential.java:33)
The finally statement is executed

```

PROGRAM NO:21

AIM: Define 2 classes one for generating fibanocci numbers and other for displaying even numbers in agiven range.implement using threads(runnable interface)

```
public class Mythread {  
    public static void main(String[]  
        args) { Runnable r = new  
        Runnable1(); Thread t = new  
        Thread(r); t.start();  
        Runnable r2 = new  
        Runnable2(); Thread t2 =  
        new Thread(r2);  
  
        t2.start();  
    }  
}
```

```
class Runnable2 implements  
    Runnable{ public void run(){  
        for(int  
            i=0;i<11;i++)  
            { if(i%2 == 1)  
                System.out.println(i);  
            }  
        }  
}
```

```
class Runnable1 implements  
    Runnable{ public void run(){  
        int n1=0,n2=1,n3,i,count=10;  
        System.out.print(n1+" "+n2);//printing 0 and 1
```

```
        for(i=2;i<count;++i)//loop starts from 2 because 0 and 1 are already printed
```



```

{
    n3=n1+n2;
    System.out.print("
    "+n3);n1=n2;
    n2=n3;
}
}
}

```

OUTPUT

```
}
```

```
}
```

```
class PrimeNo extends
```

```
Thread{ public void run()
```

```
{
```

```
int i, j, flag;
```

```
Scanner s = new Scanner(System.in);
```

```
System.out.println("\n___To generate first N prime  
numbers_____");System.out.println("Enter the limit
```

```
(N):");
```

```
int N = s.nextInt();
```

```
System.out.println("Prime numbers between 1 and " + N + "
```

```
are:");for (i = 1; i <= N; i++)
```

```
{
```

```
if (i == 1 || i == 0)
```

```
contin
```

```
ue;
```

```
flag =
```

```
1;
```

```
for (j = 2; j <= i / 2; ++j)
```

```
{
```

```
if (i % j == 0)
```

```
{
```

```
flag
```

```
= 0;
```

```
break
```

```
};
```

```
}
```

```
}
```

```
if (flag == 1)
```

```
System.out.print(i +
```

```

" ");
}
}
}

```

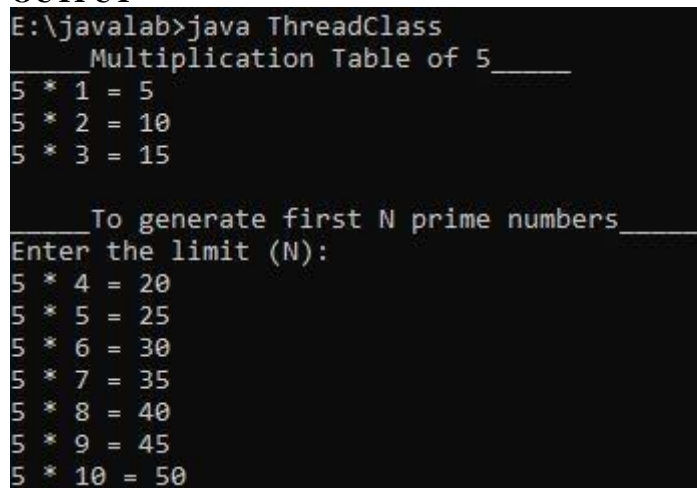
```

public class ThreadClass {
    public static void main(String[] args)

    throws InterruptedException { MulTable m = new
    MulTable();
    m.start();
    m.sleep(200);
    PrimeNo p = new
    PrimeNo();p.start();
    p.sleep(200);
    // TODO Auto-generated method stub
    }
}

```

OUTPUT



```

E:\javab>java ThreadClass
      Multiplication Table of 5_____
5 * 1 = 5
5 * 2 = 10
5 * 3 = 15

      To generate first N prime numbers_____
Enter the limit (N):
5 * 4 = 20
5 * 5 = 25
5 * 6 = 30
5 * 7 = 35
5 * 8 = 40
5 * 9 = 45
5 * 10 = 50

```

PROGRAM NO: 22

AIM: Maintain a list of string using arraylist from a collection of framework, perform built- in operation

```

import java.util.*;
import java.util.Collections;

```

```
class JavaExample{  
    public static void main(String args[]){  
        ArrayList<String> alist=new ArrayList<String>();  
        alist.add("Steve");  
        alist.add("Tim");  
        alist.add("Lucy");  
        alist.add("Pat");  
        alist.add("Angela");  
        alist.add("Tom");  
  
        //displaying elements  
        System.out.print("original list-->");  
        System.out.println(alist);  
        System.out.println();  
  
        //Adding "Steve" at the fourth position  
        alist.add(3, "Steve");  
  
        //displaying elements  
        System.out.print("after adding element--->");  
        System.out.println(alist);  
        System.out.println();  
  
        //update element  
        alist.set(0, "Lucy");  
  
        //displaying elements  
        System.out.print("after updating element--->");  
        System.out.println(alist);  
        System.out.println();  
  
        //remove elements  
        alist.remove("tom");
```

```

alist.remove("Angela");

//displaying elements

    System.out.print("after removing element--->");
System.out.println(alist);

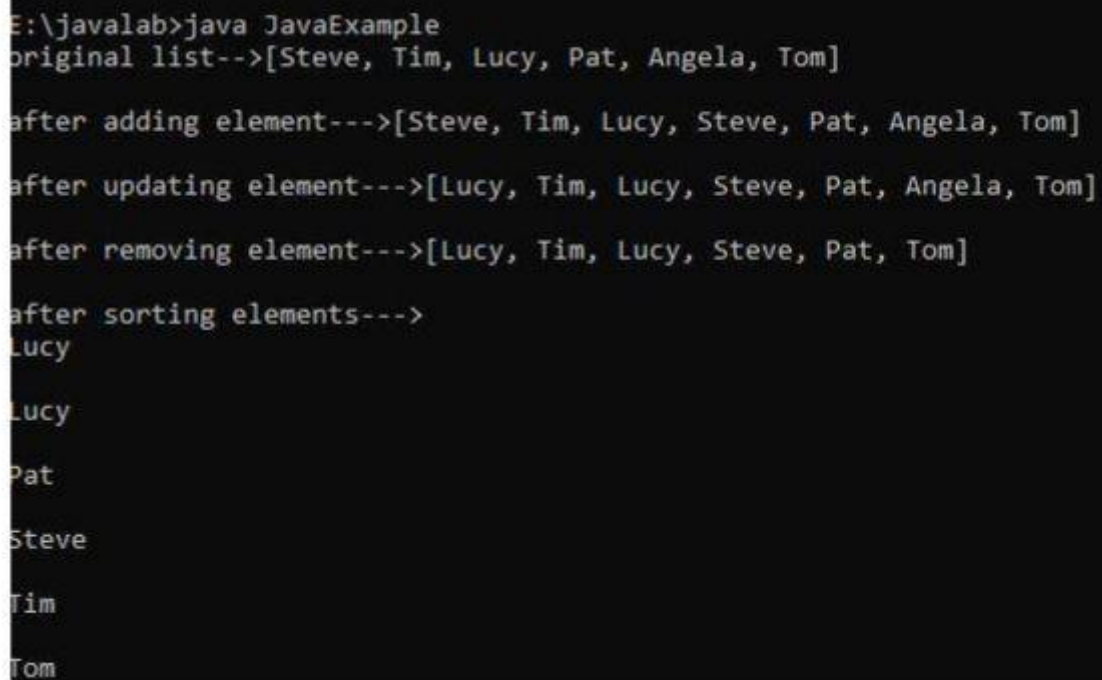
    System.out.println();

//sorting arraylist

    System.out.print("after sorting elements--->");
Collections.sort(alist);
    for (String str : alist) {
        System.out.println();
        System.out.println(str);
    }
}
}
}

```

OUTPUT



```

E:\javalab>java JavaExample
original list-->[Steve, Tim, Lucy, Pat, Angela, Tom]

after adding element--->[Steve, Tim, Lucy, Steve, Pat, Angela, Tom]

after updating element--->[Lucy, Tim, Lucy, Steve, Pat, Angela, Tom]

after removing element--->[Lucy, Tim, Lucy, Steve, Pat, Tom]

after sorting elements--->
Lucy
Lucy
Pat
Steve
Tim
Tom

```

PROGRAM NO: 23

AIM : Program to remove all the elements from a linked list

```
import java.io.*;
import java.util.LinkedList;

public class elementDelete {
    public static void main(String args[])
    {

        // Creating an empty LinkedList
        LinkedList<String> list = new LinkedList<String>();

        // Using add() method to add elements in the list
        list.add("Hanna");
        list.add("Tony");
        list.add("Ali");
        list.add("30");
        list.add("2100");

        // Output the list
        System.out.println("LinkedList:" + list);

        // Remove the head using remove()
        list.remove();

        // Print the final list
        System.out.println("Final LinkedList:" + list);
    }
}
```

OUTPUT

```
E:\javalab>
E:\javalab>java elementDelete
LinkedList:[Hanna, Tony, Ali, 30, 2100]
Final LinkedList:[Tony, Ali, 30, 2100]
```

PROGRAM NO: 24

AIM : program to demonstrate the addition and deletion of elements in dequeue

```
import java.util.*;

public class DequeueExample {

    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 last element.

        deque.removeFirst();

        deque.removeLast();

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

    }

}

```

OUTPUT:

```

E:\javab>java DequeExample
[Element 6 (Head), Element 4 (Head), Element 2 (Head), Element 1 (Tail), Element 3 (Tail), Element 5 (Tail)]
Deque after removinfirst and last: [Element 4 (Head), Element 2 (Head), Element 1 (Tail), Element 3 (Tail)]

```

PROGRAM NO: 25

AIM : program to demonstrate the working of map interface by adding ,removing,changing

```

import java.util.*;

class hashmap {

    public static void main(String args[])

    {

        Map<String, Integer> hm

            = new HashMap<String, Integer>();

        hm.put("a", new Integer(200));

        hm.put("b", new Integer(400));

        hm.put("c", new Integer(600));

        hm.put("d", new Integer(800));

        // Traversing through the map

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

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

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

        }

    }

}

```



```
}
```

OUTPUT:

```
E:\javalab>java hashmap  
a:200  
b:400  
c:600  
d:800
```

PROGRAM NO: 26

AIM : program to convert hash map to tree map

```
import java.util.*;  
import java.util.stream.*;  
public class HT {  
    public static void main(String args[]) {  
        Map<String, String> map = new HashMap<>();  
        map.put("1", "One");  
        map.put("2", "Two");  
        map.put("3", "Three");  
        map.put("4", "Four");  
        map.put("5", "Five");  
        map.put("6", "Six");  
        map.put("7", "Seven");  
        map.put("8", "Eight");  
        map.put("9", "Nine");  
        System.out.println("HashMap = " + map);  
        Map<String, String> treeMap = new TreeMap<>();  
        treeMap.putAll(map);  
        System.out.println("TreeMap (HashMap to TreeMap) " + treeMap);  
    }  
}
```

```
E:\javalab>java HT  
HashMap = {1=One, 2=Two, 3=Three, 4=Four, 5=Five, 6=Six, 7=Seven, 8=Eight, 9=Nine}  
TreeMap (HashMap to TreeMap) {1=One, 2=Two, 3=Three, 4=Four, 5=Five, 6=Six, 7=Seven, 8=Eight, 9=Nine}
```

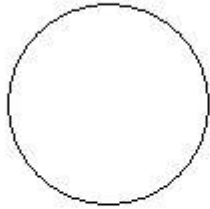
PROGRAM NO:27

AIM: Program to draw Circle, Rectangle, Line in Applet.

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

public class figures extends Applet
{
    public void paint(Graphics g)
    {
        g.drawLine(30,30,300,30);
        g.drawOval(100,100,100,100);
        g.drawRect(250, 250, 200, 100);
    }
}
```

OUTPUT:



PROGRAM NO:28

AIM: Program to find maximum of three numbers using AWT.

```
import java.awt.*;
import java.awt.event.*;
import java.applet.*;
public class largest extends Applet implements ActionListener
{
    int a, b, c, result;

    String str;
    TextField Txt1 = new TextField(10);
    TextField Txt2 = new TextField(10);
    TextField Txt3 = new TextField(10);
    TextField t4 = new TextField(10);
    Label l2 = new Label("enter number 1: ");
```

```
Label l3 = new Label("enter number 2: ");
Label l5 = new Label("enter number 3: ");
Label l4 = new Label("largest : ");
Button b1 = new Button("click");
public void init()
{
    add(l2);
    add(txt1);
    add(l3);
    add(txt2);
    add(l5);
    add(txt3);

    add(b1);
    add(l4);
    add(t4);
    b1.addActionListener(this);
}
public void actionPerformed(ActionEvent e)
{
    if (e.getSource() == b1)
    {
        str = txt1.getText();
        a = Integer.parseInt(str);
        str = txt2.getText();
        b = Integer.parseInt(str);
        str = txt3.getText();
        c = Integer.parseInt(str);
        if (a >= b && a >= c)
        {
            result = a;
            t4.setText(String.valueOf(a));
            repaint();
        }
    }
}
```

```

    }
    else if (b >= a && b >= c)

{
    result = b;
    t4.setText(String.valueOf(b));
    repaint();
}
else
{
    result = c;
    t4.setText(String.valueOf(c));
    repaint();
}
}
}
}
}

```

OUTPUT

Applet

enter number 1:	<input type="text" value="78"/>	enter number 2:	<input type="text" value="45"/>	enter number 3:	<input type="text" value="99"/>	<input type="button" value="click"/>	largest :
			<input type="text" value="99"/>				

PROGRAM NO:29

AIM:. Find the percentage of marks obtained by a student in 5 subjects. Display a happy face if he secures above 50% or a sad face if otherwise.

```

import java.awt.*;
import java.awt.event.*;
import java.applet.*;
public class myline extends Applet implements ActionListener {
    private int SMILE = 0;
    private float k;
    int i;

```

```
float j;
TextField T1 = new TextField(10);
TextField T2 = new TextField(10);
TextField t3 = new TextField(10);

Label l2 = new Label("enter total marks obtained : ");
Label l3 = new Label("enter total Marks : ");
Label l4 = new Label("percentage : ");
Button b = new Button("percentage");
public void init() {
    add(l2);
    add(T1);
    add(l3);
    add(T2);
    add(l4);
    add(t3);
    add(b);
    b.addActionListener(this);
}
public void actionPerformed(ActionEvent e) {
    if (e.getSource() == b)
        i = Integer.parseInt(T1.getText());
        j = Integer.parseInt(T2.getText());
        k = i / j;
        k = k * 100;
        if (k >= 50) {
            SMILE = 1;
        } else {
            SMILE = 0;
        }
        t3.setText(String.valueOf(k) + " %");
        repaint();
    }
}
```

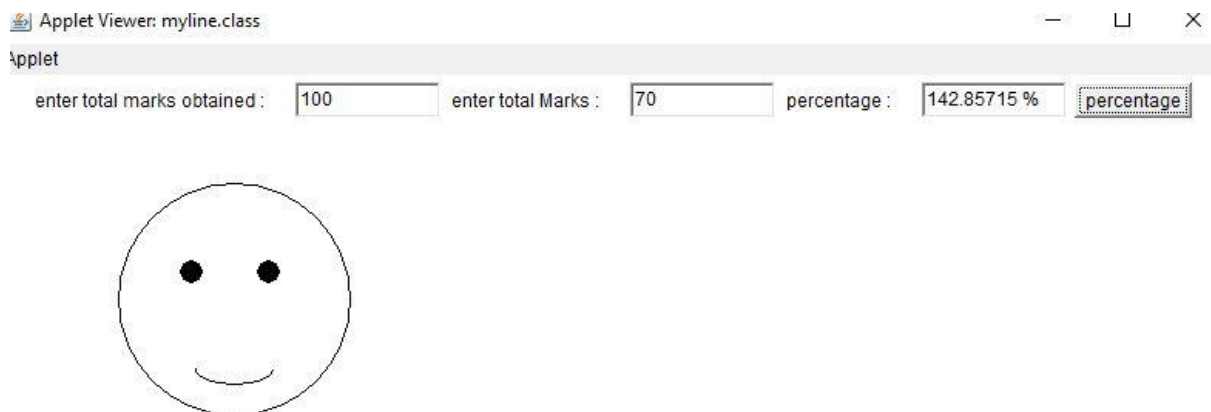
```

public void paint(Graphics g) {
    g.drawOval(80, 70, 150, 150);
    g.setColor(Color.black);
    g.fillOval(120, 120, 15, 15);
    g.fillOval(170, 120, 15, 15);

    if (SMILE == 1) {
        g.drawArc(130, 180, 50, 20, 180, 180);
        SMILE = 0;
    } else {
        g.drawArc(130, 180, 50, 20, 180, -180);
    }
}
}
}

```

OUTPUT



PROGRAM NO:30

AIM: .Using 2D graphics commands in an Applet, construct a house. On mouse click event, change the color of the door from blue to red.

```

import java.applet.*;
import java.awt.*;
import java.awt.event.MouseEvent;
import java.awt.event.MouseListener;

public class house extends Applet implements MouseListener, Runnable {
    private Color door = Color.blue;

```

```
public void paint(Graphics g) {
int x[] = { 150, 300, 225 };
int y[] = { 150, 150, 25 };
g.setColor(Color.orange);
g.fillRect(150, 150, 150, 200);
g.drawRect(150, 150, 150, 200);
g.setColor(door);

g.fillRect(200, 200, 50, 150);
g.drawRect(200, 200, 50, 150);
g.setColor(Color.red);
g.fillPolygon(x, y, 3);
g.drawPolygon(x, y, 3);
}

public void init() {
this.setSize(200, 200);
addMouseListener(this);
}

public void run() {
while (true) {
repaint();
try {
Thread.sleep(5);
} catch (InterruptedException e) {
e.printStackTrace();
}
}
}

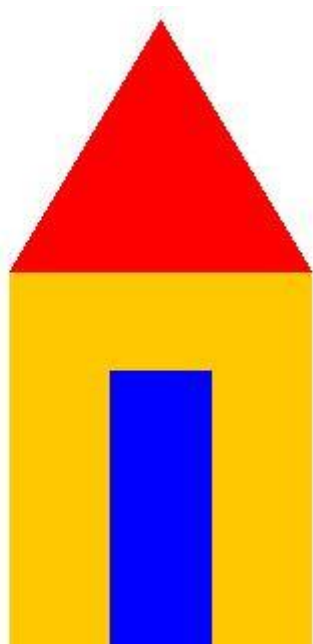
public void mouseClicked(MouseEvent e) {
int x = e.getX(), y = e.getY();
if (x <= 300)
door = Color.red;
else
door = Color.blue;
```



```
repaint();
}
public void mousePressed(MouseEvent e) {
}
public void mouseReleased(MouseEvent e) {
}
public void mouseEntered(MouseEvent e) {

}
public void mouseExited(MouseEvent e) {
}
}
```

OUTPUT



PROGRAM NO:31

AIM: Program to list the sub directories and files in a given directory and also search for a file Name

```
import java.io.File;
import java.util.*;
```

```

import java.io.*;

public class p1 {

    public static final String RED="\033[0;31m";

    public static final String RESET="\033[0m";

    static void RecursivePrint(File[] arr, int index, int level, String searchfor) {

        // exit condition

        if (index == arr.length)

            return;

        // space for internbal level

        for (int i = 0; i < level; i++)

            System.out.print("\t");

        if(arr[index].getName().toLowerCase().contains(searchfor))
            System.out.print(RED);
        else

            System.out.print(RESET);

        // for files

        if (arr[index].isFile())

            System.out.println(arr[index].getName());

        else if (arr[index].isDirectory()) {

            System.out.println "[" + arr[index].getName() + "]";

            RecursivePrint(arr[index].listFiles(), 0, level + 1, searchfor);

        }

        RecursivePrint(arr, ++index, level, searchfor);

    }

    public static void main(String[] args) {

        Scanner scan = new Scanner(System.in);

        System.out.println("Enter the directory path");

        String maindirpath = scan.nextLine();

        System.out.println("Enter the file/directory name to search");

        String searchfor = scan.nextLine();

        File maindir = new File(maindirpath);

        if (maindir.exists() && maindir.isDirectory()) {

            File arr[] = maindir.listFiles();

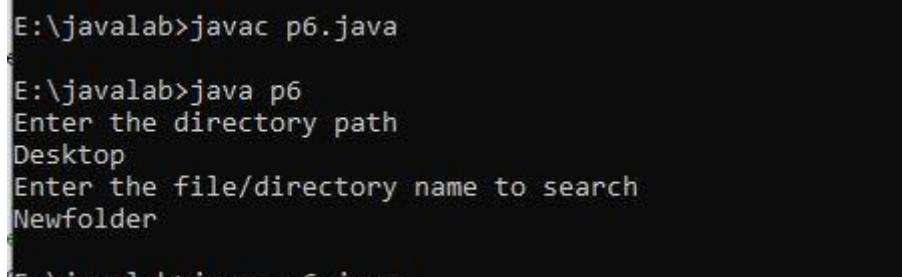
```

```

System.out.println("#####");
System.out.println("Files from main directory" + maindir);
System.out.println("#####");
RecursivePrint(arr, 0, 0, searchfor.toLowerCase()); // array,index
}
}
}

```

OUTPUT:



```

E:\javalab>javac p6.java
E:\javalab>java p6
Enter the directory path
Desktop
Enter the file/directory name to search
Newfolder

```

PROGRAM NO: 32

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

```

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 exit to stop");

while (!var.endsWith("exit\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) {
// 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:
```

```
Enter the text to create file : type exit to stop
hw
program
desktop
backspace
enter
name
exit
Reading File content
hw
program
desktop
backspace
enter
```

PROGRAM NO: 33

AIM: Write a program to copy one file to another Pre-requisite Create a text file with content where the java program is running for reading

```
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();

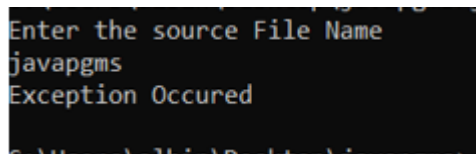
        try {
            FileReader fr=new FileReader(source);
            String str = "";
            int i;
            System.out.println("Reading from file "+source);
            while ((i = fr.read()) != -1) {
                // Storing every character in the string
```

```

str += (char) i;
}
System.out.println(str);
System.out.println("\nEnter 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) {
System.out.println("Exception Occured");
}
}
}

```

OUTPUT:



The screenshot shows a command prompt window with the following text:

Enter the source File Name

javaprgms

Exception Occured

C:\Users\j\bin\Debug\javaprgms

PROGRAM NO: 34

AIM: Write a program that reads from a file having integers. Copy even numbers and odd numbers to separate files Pre-requisite Create a text file with content of numbers where the java program is running for reading numbers

```

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");
}

}

}

```

.Client server communication using Socket – TCP/IP

Program

server

```

import java.io.*;
import java.net.*;

```

```

public class Myserver {
public static void main(String[] args) {
try{
ServerSocket ss=new ServerSocket(6666);
Socket s=ss.accept(); //establishes connection
DataInputStream dis=new DataInputStream(s.getInputStream());
String str=(String)dis.readUTF();
System.out.println("message= "+str);
ss.close();
}catch(Exception e) { System.out.println(e);}
}
}

Client
import java.io.*;
import java.net.*;

public class Myclient {
public static void main(String[] args) {

try{
Socket s=new Socket("localhost",6666);

DataOutputStream dout=new DataOutputStream(s.getOutputStream());

dout.writeUTF("Hello Server"); // Writes a string to the underlying output stream using
modified
UTF-8 encoding
dout.flush();
dout.close();
s.close();
}catch(Exception e){System.out.println(e);}
}
}

```

Output


```
1 To copy a file
```

```
1
```

PROGRAM NO: 35

AIM: Client server communication using Socket – TCP/IP Program server

```
import java.io.*;
import java.net.*;
public class Myserver {
public static void main(String[] args) {
try{
ServerSocket ss=new ServerSocket(6666);
Socket s=ss.accept(); //establishes connection
DataInputStream dis=new DataInputStream(s.getInputStream());
String str=(String)dis.readUTF();
System.out.println("message= "+str);
ss.close();
```

```
}catch(Exception e) { System.out.println(e);}
}
}
```

Client

```
import java.io.*;
import java.net.*;

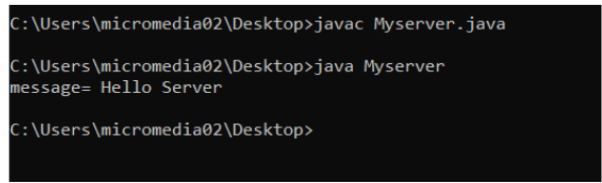
public class Myclient {
    public static void main(String[] args) {
        try{
            Socket s=new Socket("localhost",6666);

            DataOutputStream dout=new DataOutputStream(s.getOutputStream());

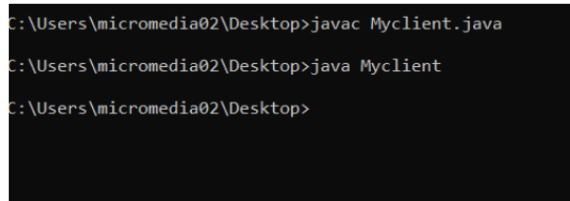
            dout.writeUTF("Hello Server"); // Writes a string to the underlying output stream using
            modified
            UTF-8 encoding

            dout.flush();
            dout.close();
            s.close();
        }catch(Exception e){System.out.println(e);}
    }
}
```

OUTPUT



```
C:\Users\micromedia02\Desktop>javac Myserver.java
C:\Users\micromedia02\Desktop>java Myserver
message= Hello Server
C:\Users\micromedia02\Desktop>
```



```
C:\Users\micromedia02\Desktop>javac Myclient.java
C:\Users\micromedia02\Desktop>java Myclient
C:\Users\micromedia02\Desktop>
```

