20MCA132 - OBJECT ORIENTED PROGRAMMING LAB

RECORD

TOM JOSEPH
ROLL NO : 36
S2-REG-MCA-B

TABLE OF CONTENTS

EXP NO.	EXPERIMENT	PAGE NO.
1	Create 3 objects of the class	5
2	matrix addition	7
3	Add complex numbers	9
4	Read a matrix and check whether it is symmetric or not	10
5	Program to Sort strings	12
6	Search an element in an array	13
7	Perform string manipulations	15
8	Program using Array of Objects	16
9	Area of different shapes using overloaded functions	18
10	Use array of objects to display details of N teachers	20
11	Use array of objects to display details of N teachers	23
12	Program to read and print book information using inheritance	26
13	Multiple inheritance using interface	28
14	Create a menu driven program to find area and perimeter of objects using interface	30
15	Program to calculate method from interface	33
16	Create a graphic package and test it	35
17	Create an arithmetic package and test it	39
18	Write a user defined exception class to authenticate the user name and password	42

19	**Find average of n positive integer and raise exception for each negative integer**	**44**
20	**Arithmetic operation using thread**	**46**
21	**Fibonacci series and even numbers using thread**	**48**
22	**Program to draw Circle, Rectangle, Line in Applet**	**50**
23	**Program to find maximum of three numbers using AWT.**	**51**
24	**Display happy face and sad face using applet based on marks secured**	**53**
25	**Program using Mouse event**	**57**
26	**Implement a simple calculator using AWT components.**	**59**
27	**Draw shapes for given parameters as per user's choice**	**62**
28	**Develop a program to handle all window events**	**65**
29	**Develop a program to handle all mouse events**	**67**
30	**Develop a program to handle Key events**	**69**
31	**Producer/Consumer using ITC**	**71**
32	**Program to create a generic stack and do the Push and Pop operations.**	**74**
33	**Using generic method perform Bubble sort**	**77**
34	**Program to demonstrate the creation of queue object using the PriorityQueue class**	**79**
35	**Program to remove all the elements from a linked list**	**81**
36	**program to demonstrate the addition and deletion of elements in dequeue**	**82**
37	**list of Strings using ArrayList from collection framework, perform built-in operations.**	**83**
38	**Program to demonstrate the working of map interface by adding ,removing,changing.**	**84**
39	**program to convert hash map to tree map.**	**85**

40	**Program list subdirectory and files , perform search operation**	**86**
41	**Program to write content to a file and display it on the console**	**88**

1 .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.

```
class productpro
{
 int pcode;
 float price;
 String pname;
 void getdata(int x,float y,String z)
 {
 pcode=x;
 price=y;
 pname=z;
 }
 void show()
 {
 System.out.println("Product Code : "+pcode);
 System.out.println("Product price : "+price);
 System.out.println("Product name : "+pname);
 }
}
public class Product
{
 public static void main(String[] args)
 {
 productpro p1=new productpro();
 productpro p2=new productpro();  productpro p3=new productpro();

 p1.getdata(11,55000,"dell insperion");
 p2.getdata(12,80000,"Asus Rog");
 p3.getdata(13,75000,"Acer Nitro 7");
 System.out.println("The Product has lowest Price ");
 System.out.println("");
 if((p3.price<p2.price) && (p3.price<p1.price))
 {
 p3.show();
 }
 else
 {
 if(p1.price<p2.price)
 p1.show();
 else
```

```
 p2.show();
 }
 }
}
```

OUTPUT

```
The Product has lowest Price

Product Code : 11
Product price : 55000.0
Product name : dell insperion
PS C:\Users\tomma>
```

RESULT : The program has been executed and the output was verified.

2. Read 2 matrices from the console and perform matrix addition.

```
import java.util.Scanner;

public class AddMatrix {

    public static void main(String args[])
    {
        int row, col,i,j;
        Scanner in = new Scanner(System.in);

        System.out.println("Enter the number of rows : ");
        row = in.nextInt();

        System.out.println("Enter the number columns : ");
        col = in.nextInt();

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

        System.out.println("Enter the elements of matrix 1  : ");

        for ( i= 0 ; i < row ; i++ )
        {

            for ( j= 0 ; j < col ;j++ )
                mat1[i][j] = in.nextInt();

            System.out.println();
        }
        System.out.println("Enter the elements of matrix 2  : ");

        for ( i= 0 ; i < row ; i++ )
        {

            for ( j= 0 ; j < col ;j++ )
                mat2[i][j] = in.nextInt();

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

```
        for ( i= 0 ; i < row ; i++ )
            for ( j= 0 ; j < col ;j++ )
                res[i][j] = mat1[i][j] + mat2[i][j] ;

        System.out.println("Sum of matrices : ");

        for ( i= 0 ; i < row ; i++ )
        {
            for ( j= 0 ; j < col ;j++ )
                System.out.print(res[i][j]+"\t");

            System.out.println();
        }

    }

}
```

OUTPUT

```
C:\java programs>javac AddMatrix.java

C:\java programs>java AddMatrix
Enter the number of rows :
2
Enter the number columns :
2
Enter the elements of matrix 1  :
1
2

3
4

Enter the elements of matrix 2  :
4
3

2
1

Sum of matrices :
5       5
5       5
```

RESULT : The program has been executed and the output was verified.

3. Add complex numbers .

```
public class Demo{
  double my_real;
  double my_imag;
  public Demo(double my_real, double my_imag){
    this.my_real = my_real;
    this.my_imag = my_imag;
  }
  public static void main(String[] args){
    Demo n1 = new Demo(76.8, 24.0),
    n2 = new Demo(65.9, 11.23),
    temp;
    temp = add(n1, n2);
    System.out.printf("The sum of two complex numbers is %.1f + %.1fi",
temp.my_real,
    temp.my_imag);
  }
  public static Demo add(Demo n1, Demo n2){
    Demo temp = new Demo(0.0, 0.0);
    temp.my_real = n1.my_real + n2.my_real;
    temp.my_imag = n1.my_imag + n2.my_imag;
    return(temp);
  }
}
```

OUTPUT

```
The sum of two complex numbers is 142.7 + 35.2i
PS C:\Users\tomma>
```

RESULT : The program has been executed and the output was verified.

4. 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.");
      }
      else
      {
         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

```
D:\MCA NOTES\Sem 2\Object Oriented Programming Lab\24-05-2021>javac Symmetric.java

D:\MCA NOTES\Sem 2\Object Oriented Programming Lab\24-05-2021>java Symmetric
Enter the no. of rows :
3
Enter the no. of columns :
3
Enter the elements :
2
3
5
6
7
8
3
6
8
Printing the input matrix :
2       3       5
6       7       8
3       6       8
The given matrix is not symmetric...
```

RESULT : The program has been executed and the output was verified.

5.Program to Sort strings

```
public class sortstring {
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 string is :");
for(i=0;i<n;i++)
{
System.out.println(names[i]);
}
}
}
```

OUTPUT

```
Problems  @ Javadoc  Declaration  Console
<terminated> sortstring [Java Application] C:\Users\HOMEPC\.p2\po
the sorted array of string is :
amal
college
engineering
jyothi
of
```

RESULT : The program has been executed and the output was verified.

6. Search an element in an array.

```
import java.util.*;

public class searcharray {
   public static void main(String[] args) {
      int n,i,b,flag=0;
     Scanner s=new Scanner(System.in);
     System.out.println("enter the size of the array");
     n=s.nextInt();
     int a[]=new int[n];
     System.out.println("enter the elements");
     for(i=0;i<n;i++){
        a[i]=s.nextInt();

     }
     System.out.println("enter the element 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 postion :"+(i+1));
     }
     else{
        System.out.println("not found");
     }
   }

}
```

OUTPUT

```
C:\java programs>java searcharray
enter the size of the array
3
enter the elements
4
5
2
enter the element to search
5
element  found at postion :2
```

RESULT : The program has been executed and the output was verified.

7. Perform string manipulations

```
public class string {
   public static void main(String[] args) {
      String x="hello! how are you";
      int a=x.length();
      System.out.println("lenght of string is :" +a);
      System.out.println(x.toUpperCase());
      System.out.println(x.toLowerCase());
      System.out.println(x.indexOf("are"));
      String y="amal";
      String z="Jyothi";
      System.out.println(y.concat(z));
      System.out.println(y+" "+z);


   }

}
```

OUTPUT

```
lenght of string is :18
HELLO! HOW ARE YOU
hello! how are you
11
amalJyothi
amal Jyothi
PS C:\Users\tomma>
```

RESULT : The program has been executed and the output was verified.

8. 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.

```
import java.util.*;
public class employee {
int eNo;
String eName;
int eSalary;

public void read(){
 Scanner sc= new Scanner(System.in);
 System.out.print("Enter ID : ");
 eNo = Integer.parseInt(sc.nextLine());
 System.out.print("Enter Name : ");
 eName = sc.nextLine();
 System.out.print("Enter monthly salary : ");
 eSalary = Integer.parseInt(sc.nextLine());
}
public void display(){
 System.out.println("Emp Name : "+ eName );
}

public static void main(String[] args) {
int i,n=3;
int No;
employee emp[] = new employee[n];

for(i=0;i<n;i++){
 emp[i] = new employee();
 emp[i].read();
}
System.out.println("Search using emp id");

while(true){
 Scanner sc= new Scanner(System.in);
 System.out.print("Enter ID : ");
 No = Integer.parseInt(sc.nextLine());
 for(i=0;i<n;i++){
  if(emp[i].eNo == No){
   emp[i].display();
```

```
  break;
 }
 }
}
}
}
```

OUTPUT

```
Problems  @ Javadoc  Declaration  Search  Console  Debu
employee [Java Application] C:\Program Files\Java\jdk1.7.0_80\bin\javaw.exe
Enter ID : 100
Enter Name : Kiran
Enter monthly salary : 100
Enter ID : 105
Enter Name : sabu
Enter monthly salary : 1200
Enter ID : 106
Enter Name : Silju
Enter monthly salary : 250
Search using emp id
Enter ID : 106
Emp Name : Silju
```

RESULT : The program has been executed and the output was verified.

9.Area of different shapes using overloaded functions

```
class areofshaping
{
   void area(float x)
   {
      System.out.println("Area of the square: "+x*x+" sq units");
   }
   void area(float x, float y)
   {
      System.out.println("Area of the rectangle: "+x*y+" sq units");
   }
   void area(double r)
   {
      double area = 3.14*r*r;
      System.out.println("Area of the circle: "+area+" sq units");
   }
   void area(int x)
   {
      //float area=6*z*z;
      System.out.println("area of cube:"+6*x*x);
   }
   void area(int x,int y)
   {
      System.out.println("the surface area of cone : "+3.14*x*2 + 3.14*x*y);
   }
   void area(double x,double y)
   {
      System.out.println("the volume of cylinder : "+ 3.14*x*x*y);
   }
   public static void main(String args[]){

      areofshaping ob = new areofshaping();


     ob.area(6.1);

     ob.area(10,22);
     ob.area(6.1);
    ob.area(4);
    ob.area(2,8);
    ob.area(4.5,7.8);
```

```
    }
}
```

OUTPUT

```
C:\java programs>java areofshaping
Area of the circle: 116.8394 sq units
the surface area of cone : 62.800000000000004690.8000000000001
Area of the circle: 116.8394 sq units
area of cube:96
the surface area of cone : 12.5650.24
the volume of cylinder : 495.963

C:\java programs>
```

RESULT : The program has been executed and the output was verified.

10 . 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;
float salary;
String address;
	public employee(int empid,String name,float salary,String address)
	{
		this.empid=empid;
		this.name=name;
		this.salary=salary;
		this.address=address;
	}

}


public class Teacher extends employee
{
	String department;
	String subject;
	public Teacher(int empid,String name,float salary,String address,String
department,String subject)
	{
		super(empid,name,salary,address);
		this.department=department;
		this.subject=subject;
	}
	void show()
	{
		System.out.println("the id of the employee"+this.empid+"name :
"+this.name+"salary :"+this.salary+"address :"+this.address+"department
:"+this.department+"subject :"+this.subject);
```

```
        }

        public static void main(String[] args)
        {

        Scanner sc=new Scanner(System.in);

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

                System.out.println("enter id :"+j+" : ");
                int empid=sc.nextInt();

                System.out.println("enter name :"+j+" : ");
                String name=sc.next ();
                System.out.println("enter salary :"+j+" : ");
                float salary=sc.nextInt();
                System.out.println("enter address :"+j+" : ");
                String address=sc.next ();
                System.out.println("enter department :"+j+" : ");
                String department=sc.next ();
                System.out.println("enter subject :"+j+" : ");
                String subject=sc.next ();
                obj[i]=new Teacher(empid,name,salary,address,department,subject);
                }

        System.out.println("list");
                {
                for(int i=0;i<n;i++)
                obj[i].show();
                }
        }
}
```

OUTPUT

```
C:\Users\tomma\OneDrive\Desktop\javaworker>java Teacher
the number of teacher
2
enter id :1 :
01
enter name :1 :
ajo
enter salary :1 :
10000
enter address :1 :
kottayam
enter department :1 :
cs
enter subject :1 :
maths
enter id :2 :
02
enter name :2 :
bjo
enter salary :2 :
20000
enter address :2 :
idukki
enter department :2 :
maths
enter subject :2 :
cs
list
the id of the employee1name : ajosalary :10000.0address :kottayamdepartment :cssubject :maths
the id of the employee2name : bjosalary :20000.0address :idukkidepartment :mathssubject :cs

C:\Users\tomma\OneDrive\Desktop\javaworker>
```

RESULT : The program has been executed and the output was verified.

11. 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,
String company_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 Main {
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:"); String nam1=s.next();
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

```
D:\MCA NOTES\Sem 2\Object Oriented Programming Lab\08-06-2021>java Main
Enter number of Teachers :
1
Enter the person name:
Rini
Enter the Gender:
Female
Enter the Address:
Alappuzha
Enter the Age:
26
Enter the Employee id:
5
Enter the Company name:
AJCE
Enter the Salary:
40000
Enter the Qualification:
MCA
Enter the Teacher id:
2
Enter the Department:
MCA
Enter the Subject:
Network

---------------------------------------------------------------------------

....Personal details...
 Name : Rini Gender : Female Age :26
...Employee details....
Empid : 5 company_name : AJCE Salary : 40000.0 Address : Alappuzha qualification : MCA
...Teacher's details...
 teacherid : 2 department : MCA Subjects : Network
```

RESULT : The program has been executed and the output was verified.

12. 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.

PROGRAM

```
import java.util.*;
class publisher{
   public void display() {
      Scanner sc= new Scanner(System.in);

      System.out.print("Enter name of the book  ");
      String str= sc.nextLine();
      System.out.println(" The publisher of the book :" + str);
   }
}
class book extends publisher{

   public void show() {

   Scanner sc= new Scanner(System.in);

   System.out.print("Enter title of the book ");
   String as= sc.nextLine();
   System.out.println(" The title of the book :" + as);



   }
}
class litertater extends book{
   public void  find(){

   Scanner sc= new Scanner(System.in);

   System.out.print("Enter price of the book ");
   int a= sc.nextInt();
   System.out.println(" The price of the book :" + a);
   }
}
class fiction extends litertater{
   public void visible() {
```

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

    System.out.print("Enter the type of  fiction in the book ");
    String bc= sc.nextLine();
    System.out.println(" the type of fiction in the book :" + bc);



    }

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

        fiction f=new fiction();

        f.display();
        f.show();
        f.visible();
        f.find();

    }
}
```

OUTPUT

```
nExceptionMessages' '-Dfile.encoding=UTF-8' '-cp' 'C:\User
Enter name of the book  alchemist
 The publisher of the book :alchemist
Enter title of the book dreamer
 The title of the book :dreamer
Enter the type of  fiction in the book science
 the type of fiction in the book :science
Enter price of the book 20000
 The price of the book :20000
PS C:\Users\tomma>
```

RESULT : The program has been executed and the output was verified.

13.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 stresullt(); }
interface sports
{       void spresult();   }
class result implements student,sports{
          public void spresult()  {
          String hundred="First";
          String twohundred="Second";
          String fivehundred="First";
          String relay="Second";
          System.out.println("Sports Result");
        System.out.println("Hundred Meter:"+hundred);
          System.out.println("Two Hundred Meter:"+twohundred);
        System.out.println("Five Hundred Meter:"+fivehundred);
        System.out.println("Relay:"+relay);   }
          public void stresullt()   {
          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 static void main(String[] args)
          {    result r = new result();
                   r.stresullt();
                   r.spresult();   }  }
```

OUTPUT

```
D:\MCA NOTES\Sem 2\Object Oriented Programming Lab\15-6-2021>java result
Marks
Physics:30
Chemistry:40
Mathematics:45
English:50
Computer:50
Sports Result
Hundred Meter:First
Two Hundred Meter:Second
Five Hundred Meter:First
Relay:Second
```

RESULT : The program has been executed and the output was verified.

14.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
{
   void input();
   void area();
   void perimeter();
}
class Circle implements Shape
{

   int r = 0;
   double pi = 3.14, ar = 0,per=0;
   public void input()
   {       Scanner s = new Scanner(System.in);
            System.out.print("Enter radius of circle:");
          r= s.nextInt();
   }
   public void area()
   {
      ar = pi * r * r;
      System.out.println("Area of circle:"+ar);
   }
   public void  perimeter()
   {
           per = 2 * pi * r;
           System.out.println("Perimeter of circle:"+per);
   }

}
class Rectangle implements Shape
{
   int l = 0, b = 0;
   double ar,per;
   public void input()
   {    Scanner s = 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 shapes
{
    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:
           obj2.input();
             obj2.area();
        break;
        case 4:
           obj2.input();
             obj2.perimeter();
        break;
        default:
         System.out.println("Invalid option");
}
```

```
 }
  }
```

OUTPUT

```
D:\MCA NOTES\Sem 2\Object Oriented Programming Lab\Java>javac shapes.java

D:\MCA NOTES\Sem 2\Object Oriented Programming Lab\Java>java shapes
1.Area of circle
2.Perimeter of circle
3.Area of rectangle
4.Perimeter of rectangle
Enter your option:
1
Enter radius of circle:3
Area of circle:28.259999999999998
```

RESULT : The program has been executed and the output was verified.

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

```
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

```
D:\MCA NOTES\Sem 2\Object Oriented Programming Lab\15-6-2021>javac productbill.java

D:\MCA NOTES\Sem 2\Object Oriented Programming Lab\15-6-2021>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
```

RESULT : The program has been executed and the output was verified.

16.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.

```
package
Graphics;
interface Area1
{
        public void
        Rectangle();public
        void Triangle();
        public void Square();
        public void Circle();
        public void getRect();
        public void getTri();
        public void getSqr();
        public void getCrl();
}

//shapes.java

package
Graphiccs;
import
java.util.*;


public class shapess implements Area1

{

        double
        lr,lb,ra,th,tb,ta,saa,sa,cr,cc;
        public void getrect()
        {

                Scanner ab= new Scanner(System.in);
                System.out.println("Enter the length of the
                rectangle");lr=ab.nextInt();
```

```
        System.out.println("Enter the breadth of the
        rectangle");lb=ab.nextInt();


    }
    public void rectangle()
    {
        ra=lr*lb;
        System.out.println("Area of Rectangle is "+ra);
    }
    public void getTri()
    {
        Scanner cb= new Scanner(System.in);
        System.out.println("Enter the height of the
        Triangle");th=cb.nextInt();
        System.out.println("Enter the base of the
        Triangle");tb=cb.nextInt();
    }
    public void Triangle()
    {
        ta=0.5*th*tb;
        System.out.println("Area of Triangle angle is "+ta);
    }
    public void getSqr()
    {
        Scanner sq= new Scanner(System.in);
        System.out.println("Enter the Side of the
        Square");sa=sq.nextInt();
    }
```

```
public void Square()
{
        saa=sa*sa;
        System.out.println("Area of Square is "+saa);
}

public void getCrl()
{
        Scanner sc= new Scanner(System.in);
        System.out.println("Enter the radius of the
        Circle");cc=sc.nextInt();
}
public void Circle()
{
        cr=3.14*cc*cc;
        System.out.println("Area of Square is "+cr);


}
public static void main(String[] args)
{
        shapess o= new
        shapess();o.getrect();
        o.rectangle();
        o.getTri();
        o.Triangle();
        o.getSqr();
        o.Square();
        o.getCrl();
```

```
            o.Circle();
        }
}
```

OUTPUT

```
D:\java_lab>javac -d . Area1.java

D:\java_lab>javac -d . shapess.java

D:\java_lab>java Graphiccs.shapess
Enter the length of the rectangle
5
Enter the breadth of the rectangle
2
Area of Rectangle is 10.0
Enter the height of the Triangle
9
Enter the base of the Triangle
2
Area of Triangle angle is 9.0
Enter the Side of the Square
4
Area of Square is 16.0
Enter the radius of the Circle
6
Area of Square is 113.03999999999999
```

RESULT : The program has been executed and the output was verified.

17. Create an Arithmetic package that has classes and interfaces for the 4 basic arithmetic operations. Test the package by implementing all operations on two given numbers.

```
package Arithmetic;

interface operations

{

        public void input();
        public void add();
        public void
        substract();public
        void multiply();
        public void
        division();


}

package
Aarithmetic;import
java.util.*;



public class basic implements operations

{

        double a,b,ad,dif,mult,div;
        public void input()
        {

                Scanner ab=new
                Scanner(System.in);
                System.out.println("Enter two
                numbers");a=ab.nextInt();
                b=ab.nextInt();

        }
```

```
public void add()
{
        ad=a+b;
        System.out.println("Sum is "+ad);
}
public void substract()
{

        dif=a-b;
        System.out.println("Difference is "+dif);
}
public void multiply()
{
        mult=a*b;
        System.out.println("Product is "+mult);
        }
public void division()
{
        div=a/b;
        System.out.println("Quotient is "+div);
}
public static void main(String[] args)
{
        basic o=new basic();
        o.input();
        o.add();
```

```
            o.substract();
            o.multiply();
            o.division();

        }

    }
```

OUTPUT

RESULT : The program has been executed and the output was verified.

18. Write a user defined exception class to authenticate 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

```
D:\MCA NOTES\Sem 2\Object Oriented Programming Lab\14-08-2021(Bijimol)>java CheckLoginCredential
Enter username :: Antony
Enter password :: 12345
PasswordException: Incorrect password
Type correct password ???
        at CheckLoginCredential.main(CheckLoginCredential.java:35)
The finally statement is executed

D:\MCA NOTES\Sem 2\Object Oriented Programming Lab\14-08-2021(Bijimol)>
```

RESULT : The program has been executed and the output was verified.

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

```
import java.util.Scanner;
class NegInputException extends Exception {
private static final long serialVersionUID = 1L;
NegInputException(String s){
super(s);
}

}
public class avgException {

public static void main(String[] args) {
int N;

float Sum = 0,avg;
Scanner sc=new Scanner(System.in);
System.out.println("Enter the number of inputs required:");
N = sc.nextInt();
float[] numbers = new float[N];
System.out.println("Start inputting the data:");

for( int i=0; i < N ; i++)
{
numbers[i]=sc.nextInt();
try{ if(numbers[i]<0)
{
throw new NegInputException("Negative inputs not allowed!");
}
else
{
Sum += numbers[i];
}
}catch(NegInputException e)
{
System.out.println("Exception Occurred. . "+e); System.exit(0);
}
}
sc.close();

avg = Sum / N;
```

```
System.out.println("Average = "+ avg);
}
}
```

OUTPUT

```
Problems  @ Javadoc  Declaration  Console
<terminated> avgException [Java Application] C:\Program Files\Jav
Enter the number of inputs required:
4
Start inputting the data:
2
6
4
8
Average = 5.0
```

RESULT : The program has been executed and the output was verified.

20.Define 2 classes; one for generating multiplication table of 5 and other for displaying first N prime numbers. Implement using threads. (Thread class) .

```
import java.util.Scanner;

class MulTable extends Thread{
public void run() {
int num = 5;
System.out.printf(" _Multiplication Table of 5 \n");
for(int i = 1; i <= 10; ++i)
{
System.out.printf("%d * %d = %d \n", num, i, num * i);
}
}

}

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)
continue;
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);
}

}
```

OUTPUT

```
Problems  @ Javadoc  Declaration  Console
<terminated> ThreadClass [Java Application] C:\Program Files\Java\jdk
____Multiplication Table of 5_____
5 * 1 = 5
5 * 2 = 10
5 * 3 = 15
5 * 4 = 20
5 * 5 = 25
5 * 6 = 30
5 * 7 = 35
5 * 8 = 40
5 * 9 = 45
5 * 10 = 50

_____To generate first N prime numbers_____
Enter the limit (N):
20
Prime numbers between 1 and 20 are:
2 3 5 7 11 13 17 19
```

RESULT : The program has been executed and the output was verified.

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

```
import java.util.Scanner;
class Fibonacci implements Runnable{
public void run(){
int first = 0, second = 1, next;
Scanner sc= new Scanner(System.in);
System.out.println("___TO GENERATE FIBONACCI SERIES____");
System.out.println("Enter the no.of terms required:");
int n=sc.nextInt();
System.out.println("Series Generated!!!");
for (int i = 1; i <= n; ++i){
System.out.print(first + " ");
 next = first + second;
first = second;
second = next;
  }
 }
}
class EvenNo implements Runnable{
public void run(){
Scanner sc= new Scanner(System.in);
int lower, upper;
System.out.println("\n\n_____TO GENERATE EVEN NUMBERS OF GIVEN
RANGE_____");
System.out.println("Enter the lower limit:");
lower=sc.nextInt();
System.out.println("Enter the upper limit:");
upper=sc.nextInt();
System.out.println("Even numbers from " + lower + " and " + upper + " are:");
for (int i = lower; i <= upper; i++){
 if (i%2!=0)
 continue;
 else
 {
 System.out.print(i+" ") ;
 }
 }
 }
```

```
 }
public class ThreadRunner {
public static void main(String arg[]) throws InterruptedException
{
Fibonacci obj1 = new Fibonacci();
Thread a=new Thread(obj1);
a.start();
a.sleep(2000);
EvenNo obj2 = new EvenNo();
Thread b= new Thread(obj2);
b.start();
b.sleep(1000);
}
}
```

OUTPUT

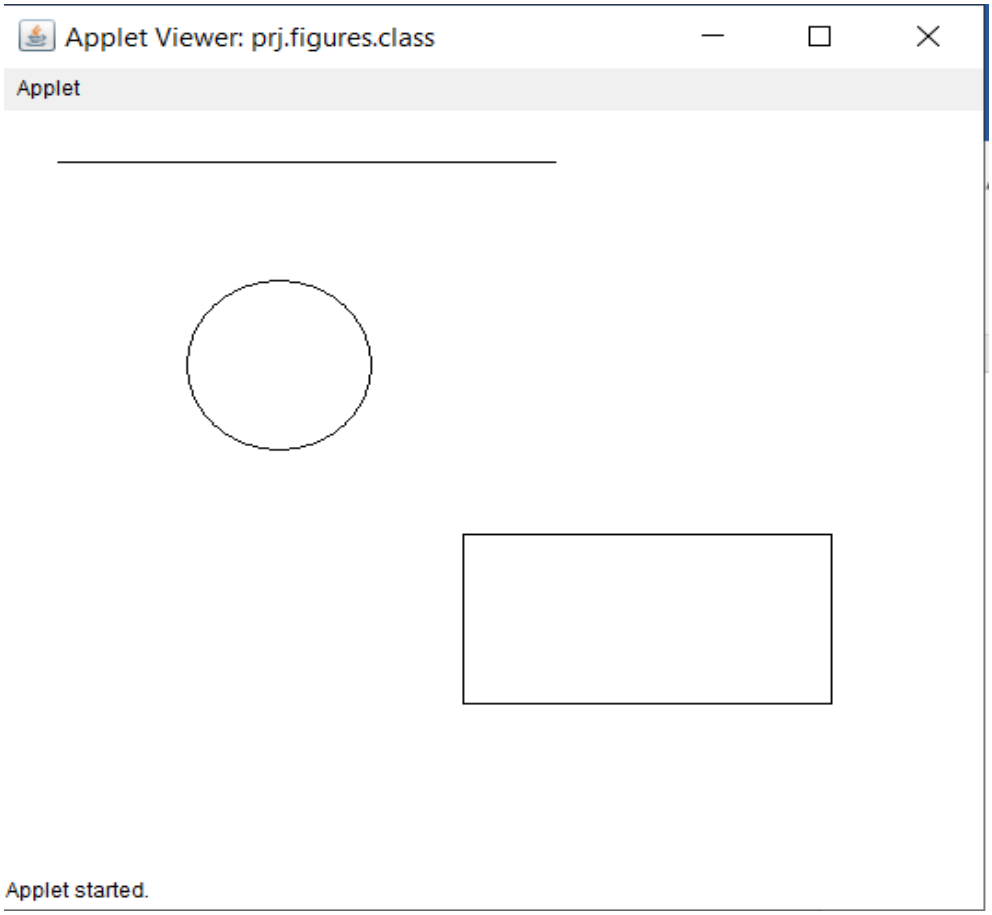
```
Problems  @ Javadoc  Declaration  Console
<terminated> ThreadRunner [Java Application] C:\Program Files\Java\jdk-16.0.
___TO GENERATE FIBONACCI SERIES____
Enter the no.of terms required:
4
Series Generated!!!
0  1  1  2

_____TO GENERATE EVEN NUMBERS OF GIVEN RANGE_____
Enter the lower limit:
2
Enter the upper limit:
9
Even numbers from 2 and 9 are:
2 4 6 8
```

RESULT : The program has been executed and the output was verified.

22. Program to draw Circle, Rectangle, Line in Applet.

```
package prj;
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



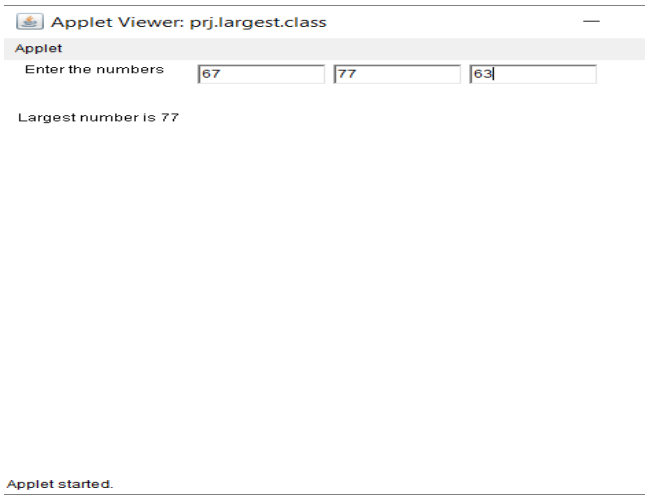
RESULT : The program has been executed and the output was verified.

23. Program to find maximum of three numbers using AWT.

```
package prj;
import java.awt.*;
import java.awt.Event;
import java.applet.*;
public class largest extends Applet
{
 TextField Txt1,Txt2,Txt3;
 public void init(){
 Txt1 = new TextField(10);
 Txt2 = new TextField(10);

 Txt3 = new TextField(10);
 add(Txt1);
 add(Txt2);
 add(Txt3);
 }
 public void paint(Graphics g){
 int a, b, c,result;
 String str;
 g.drawString("Enter the numbers ",15,15);
 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;
 }
 else if(b>=a && b>=c)
 {
 result=b;
 }
 else
 {
 result=c;
 }
 g.drawString("Largest number is "+result,10,70);
 }
```

```
 public boolean action(Event e, Object o){
 repaint();
 return true;
 }

}
```

OUTPUT



RESULT : The program has been executed and the output was verified.

24. 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.

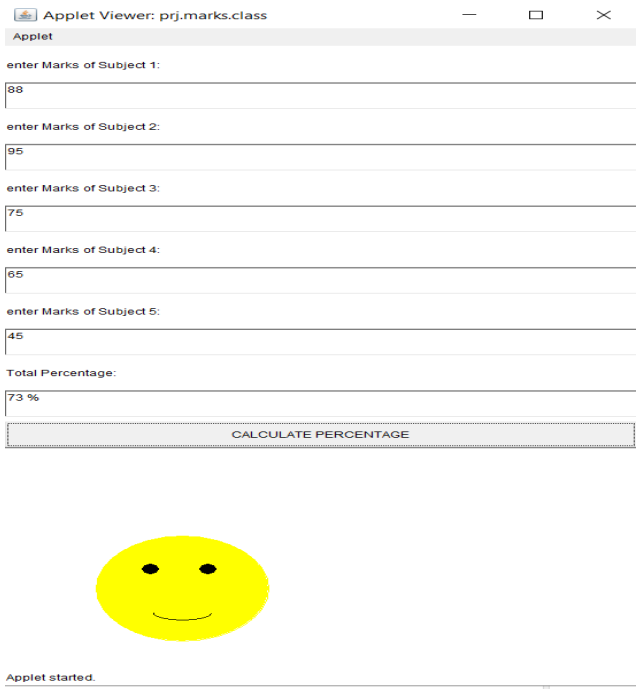
```
import java.awt.*;
import java.awt.event.*;
import java.applet.*;
public class marks extends Applet implements ActionListener {
public int per =0;
Label l1 = new Label("enter Marks of Subject 1: ");
 Label l2 = new Label("enter Marks of Subject 2: ");
 Label l3 = new Label("enter Marks of Subject 3: ");
 Label l4 = new Label("enter Marks of Subject 4: ");
 Label l5 = new Label("enter Marks of Subject 5: ");
 Label l6 = new Label("Total Percentage: ");

 TextField t1 = new TextField(10);
 TextField t2 = new TextField(10);
 TextField t3 = new TextField(10);
 TextField t4 = new TextField(10);
 TextField t5 = new TextField(10);
 TextField t6 = new TextField(10);

 Button b1 = new Button("CALCULATE PERCENTAGE");

 public marks()
 {
 l1.setBounds(50, 100, 280, 20);

 l2.setBounds(50, 150, 280, 20);
 l3.setBounds(50, 200, 280, 20);
 l4.setBounds(50, 250, 280, 20);
 l5.setBounds(50, 300, 280, 20);
 l6.setBounds(50, 350, 280, 20);
 t1.setBounds(200, 100, 300, 20);
 t2.setBounds(200, 150, 300, 20);
 t3.setBounds(200, 200, 300, 20);
 t4.setBounds(200, 250, 300, 20);
 t5.setBounds(200, 300, 300, 20);
 t6.setBounds(200, 350, 300, 20);

 b1.setBounds(200,400, 200, 20);
```

```
 GridLayout g1 = new GridLayout(20, 2, 5, 5);
 setLayout(g1);
 add(l1);
 add(t1);
 add(l2);
 add(t2);
 add(l3);
 add(t3);
 add(l4);
 add(t4);
 add(l5);
 add(t5);
 add(l6);
 add(t6);
 add(b1);
 b1.addActionListener(this);
 }
@Override
public void actionPerformed(ActionEvent e) {
 int m1 = Integer.parseInt(t1.getText());
 int m2= Integer.parseInt(t2.getText());
 int m3= Integer.parseInt(t3.getText());
 int m4= Integer.parseInt(t4.getText());
 int m5= Integer.parseInt(t5.getText());

 if(e.getSource()==b1)
 {
 int add=m1+m2+m3+m4+m5;
 per=add/5;
 t6.setText(String.valueOf(per)+" %");

 repaint();
 }
}
public void paint(Graphics g)
 {

if(per>=50)
{
g.setColor(Color.yellow);
g.drawOval(80, 700, 150, 150);
g.fillOval(80, 700, 150, 150);
```

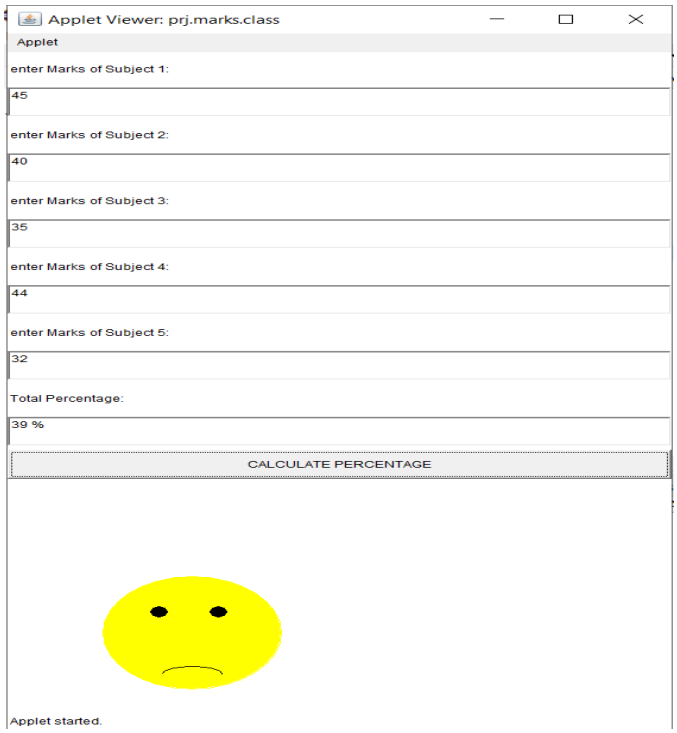
```
g.setColor(Color.BLACK);
g.fillOval(120, 740, 15, 15);
g.fillOval(170, 740, 15, 15);
g.drawArc(130, 800, 50, 20, 180, 180);
}
else if(per>0 && per<50)
{
g.setColor(Color.yellow);
g.drawOval(80, 700, 150, 150);
g.fillOval(80, 700, 150, 150);
 g.setColor(Color.BLACK);
 g.fillOval(120, 740, 15, 15);
 g.fillOval(170, 740, 15, 15);
g.drawArc(130,820,50,20,0,180);
 }
 }
public static void main(String args[]) {
 new marks();
 }
}
```

OUTPUT

Case 1



Case 2



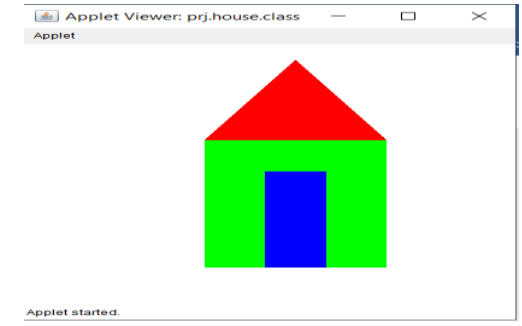
RESULT : The program has been executed and the output was verified.

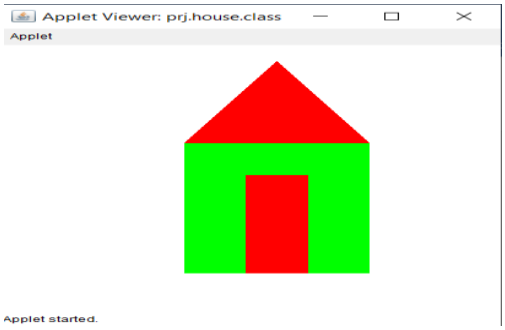
25. Using 2D graphics commands in an Applet, construct a house. On mouse click event, change the color of the door from blue to red.

```
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.green);
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





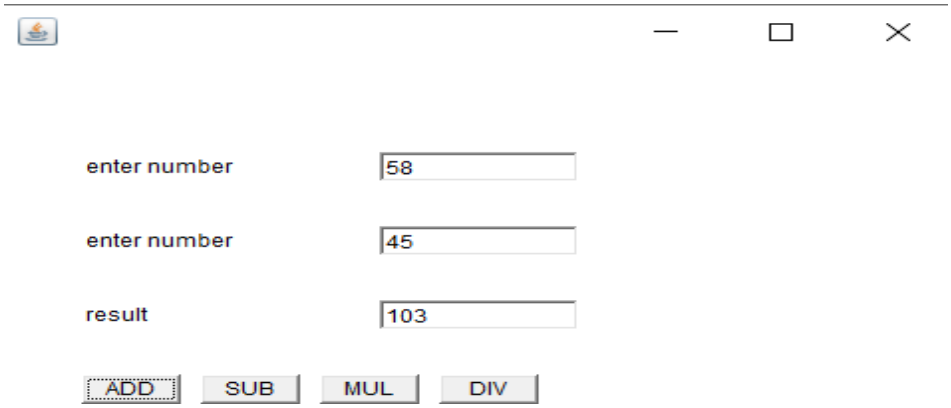
RESULT : The program has been executed and the output was verified.

26. Implement a simple calculator using AWT components.

```
import java.awt.*;
import java.awt.event.*;
import java.applet.*;
public class calc extends Applet implements ActionListener {
 Frame f = new Frame();
 Label l1 = new Label("enter number");
 Label l2 = new Label("enter number");
 Label l3 = new Label("result");

 TextField t1 = new TextField(10);
 TextField t2 = new TextField(10);
 TextField t3 = new TextField(10);

 Button b1 = new Button("ADD");
 Button b2 = new Button("SUB");
 Button b3 = new Button("MUL");
 Button b4 = new Button("DIV");
 calc()
 {
 l1.setBounds(50, 100, 100, 20);
 l2.setBounds(50, 150, 100, 20);
 l3.setBounds(50, 200, 100, 20);
 t1.setBounds(200, 100, 100, 20);
 t2.setBounds(200, 150, 100, 20);
 t3.setBounds(200, 200, 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);
 f.add(l1);
 f.add(t1);
 f.add(l2);
 f.add(t2);
 f.add(l3);
 f.add(t3);
 f.add(b1);
 f.add(b2);
 f.add(b3);
 f.add(b4);
```

```
b1.addActionListener(this);
b2.addActionListener(this);
b3.addActionListener(this);
b4.addActionListener(this);
f.setLayout(null);
f.setVisible(true);
f.setSize(500, 500);
}
public void actionPerformed(ActionEvent e) {
int i = Integer.parseInt(t1.getText());
int j = Integer.parseInt(t2.getText());
if (e.getSource() == b1) {
t3.setText(String.valueOf(i + j));
}
if (e.getSource() == b2) {
t3.setText(String.valueOf(i - j));
}
if (e.getSource() == b3) {


t3.setText(String.valueOf(i * j));
}
if (e.getSource() == b4) {
t3.setText(String.valueOf(i / j));
}
}

public static void main(String args[]) {
new calc();
}
}
```

OUTPUT



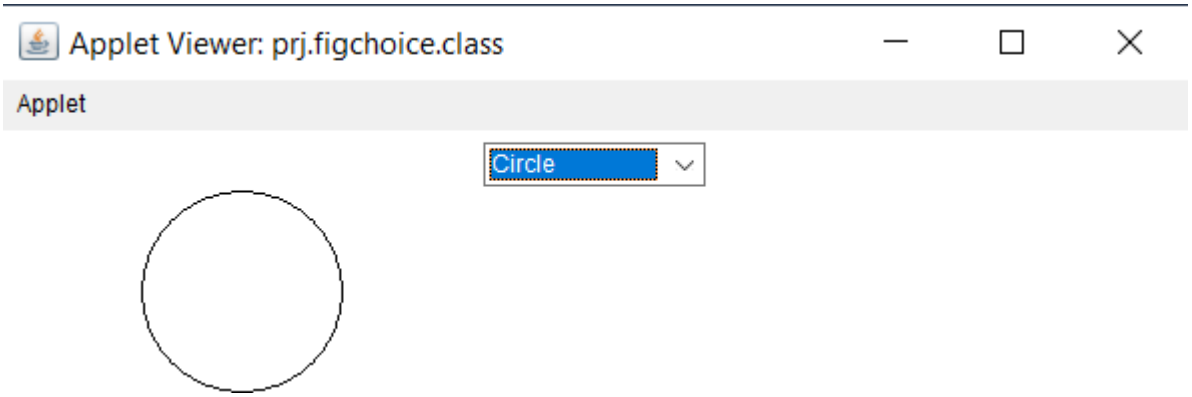
RESULT : The program has been executed and the output was verified.

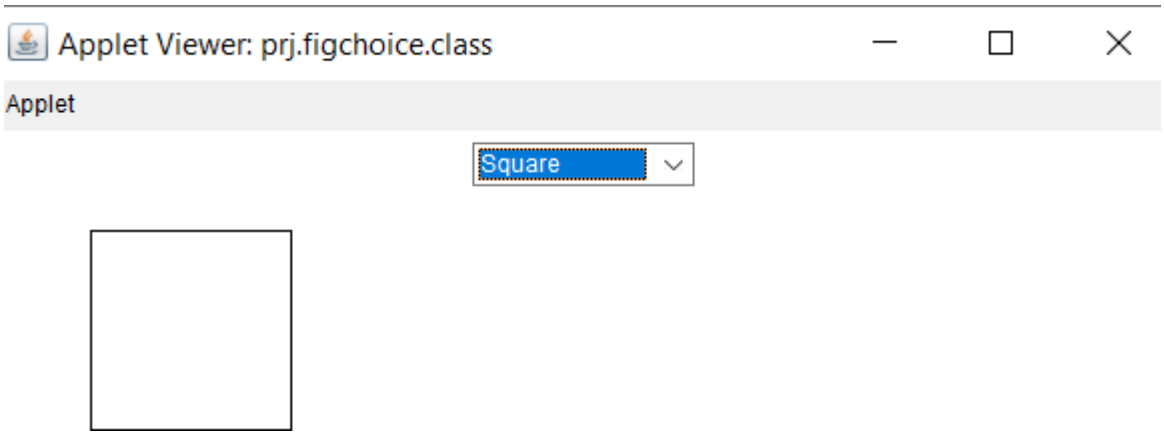
27.Develop a program that has a Choice component which contains the names of shapes such as rectangle, triangle, square and circle. Draw the corresponding shapes for given parameters as per user's choice.

```
import java.applet.Applet;
import java.awt.*;
import java.awt.Graphics;
import java.awt.event.*;
public class figchoice extends Applet implements ItemListener {
 Choice ch;
 int x1[]= {50,120,220,20};
int y1[]= {50,120,20,20};
int n=4;
 int Selection;
 public void init()
 {
 ch = new Choice();
 ch.addItem("Select a Shape");

 ch.addItem("Rectangle");
 ch.addItem("Triangle");
 ch.addItem("Square");
 ch.addItem("Circle");
 add(ch);
 ch.addItemListener(this);
 }
public void itemStateChanged (ItemEvent e)
 {
 Selection = ch.getSelectedIndex();
 repaint();
 }
public void paint(Graphics g)
{
 super.paint(g);
 if (Selection == 1)
 {
 g.drawRect(50,50,100,150);
 }
 if (Selection == 2)
 {
 g.drawPolygon(x1,y1,n);
 }
```

```
if (Selection == 3)
{
g.drawRect(50,50,100,100);
}
if (Selection == 4)
{
g.drawOval(70,30,100,100);
}
}
}
```

OUTPUT





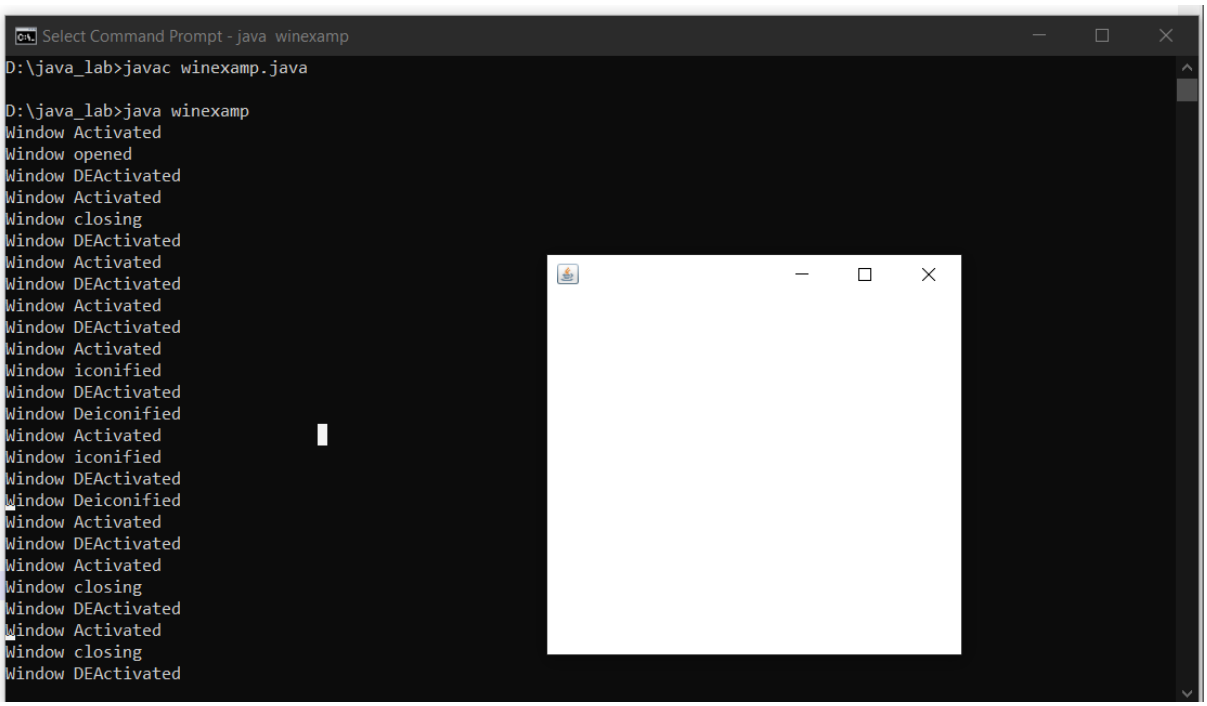
RESULT : The program has been executed and the output was verified.

28. Develop a program to handle all window events

```
import java.awt.*;
import java.awt.event.WindowEvent;
import java.awt.event.WindowListener;
public class winexamp extends Frame implements WindowListener
{
winexamp()
{
addWindowListener(this);
setSize(400,400);
setLayout(null);
setVisible(true);
}
public static void main(String args[])
{
new winexamp();
}
public void windowActivated(WindowEvent arg0)
{
System.out.println("Window Activated");
}
public void windowClosed(WindowEvent args0)
{
System.out.println("Window closed");
}
public void windowClosing(WindowEvent arg0)
{
System.out.println("Window closing");
}
public void windowDeactivated(WindowEvent arg0)
{
System.out.println("Window DEActivated");
}
public void windowDeiconified(WindowEvent arg0)
{
System.out.println("Window Deiconified");
```

```
}
public void windowIconified(WindowEvent arg0)
{
System.out.println("Window iconified");
}
public void windowOpened(WindowEvent arg0)
{
System.out.println("Window opened");
}
}
```

OUTPUT



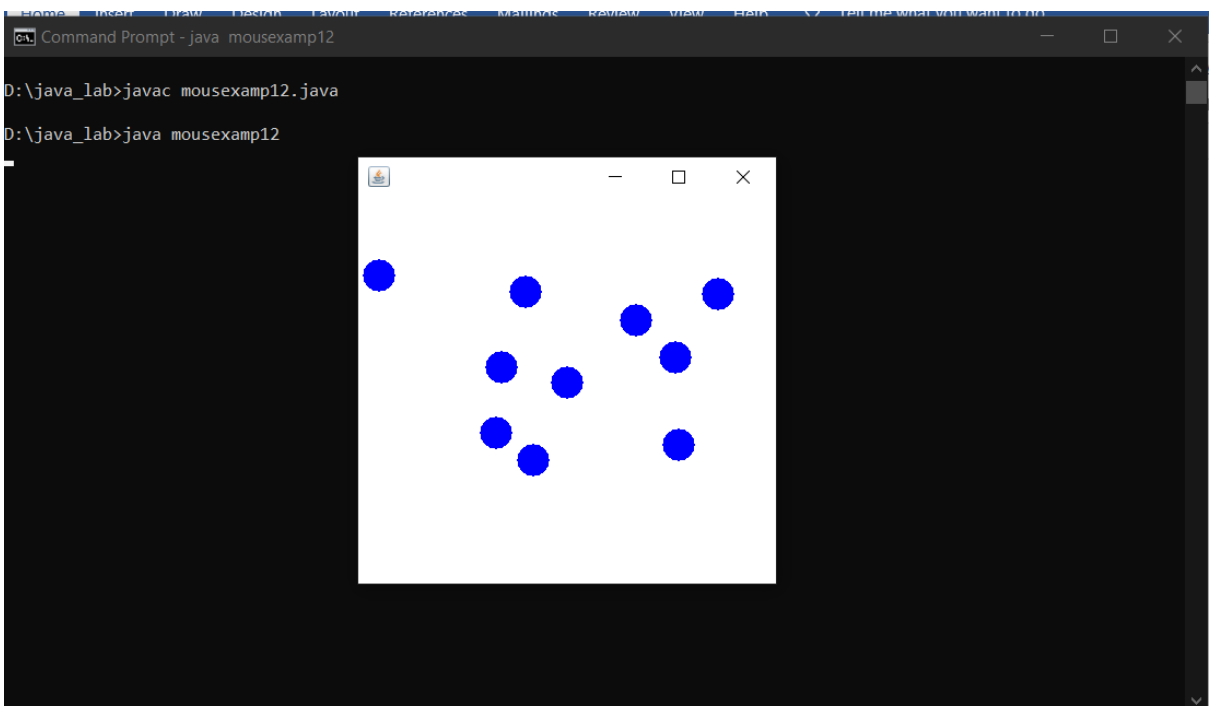
RESULT : The program has been executed and the output was verified.

29. Develop a program to handle all mouse events

```
import java.awt.*;
import java.awt.event.*;
public class mousexamp12 extends Frame implements MouseListener
{
        mousexamp12()
        {
        addMouseListener(this);
        setSize(400,400);
        setLayout(null);
        setVisible(true);
        }
        public void mouseClicked(MouseEvent e)
        {
        Graphics g=getGraphics();
        g.setColor(Color.blue);
        g.fillOval(e.getX(),e.getY(),30,30);
        }
public void mouseEntered(MouseEvent e)
{
}
public void mouseExited(MouseEvent e)
{
}
public void mousePressed(MouseEvent e)
{
```

```
}
public void mouseReleased(MouseEvent e){
}
public static void main(String args[])
{
new mousexamp12();
}
}
```

OUTPUT



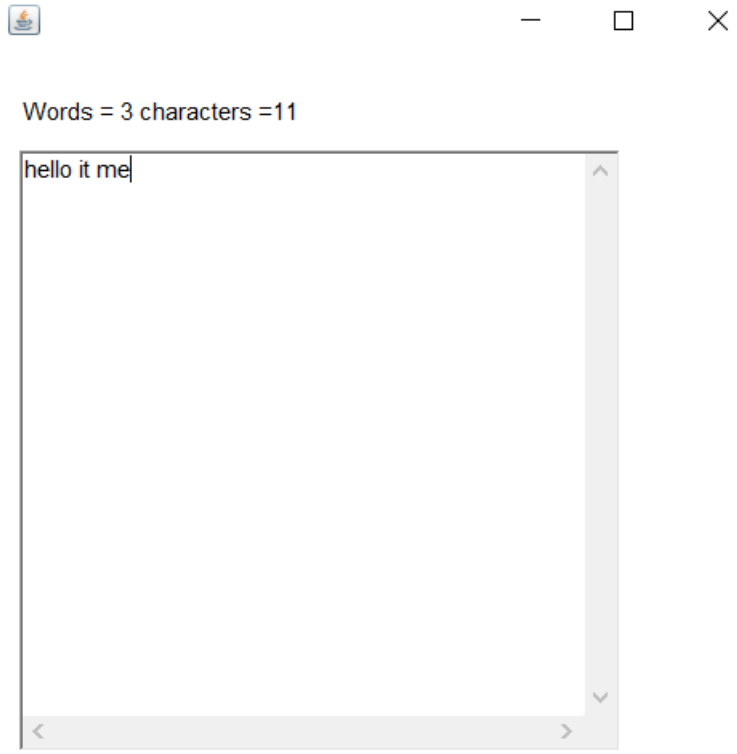
RESULT : The program has been executed and the output was verified.

31. Develop a program to handle Key events.

PROGRAM

```
import  java.awt.*;
import java.awt.event.*;
public class keyex extends Frame implements KeyListener{
   Label l;
   TextArea a;
   keyex()
   {
      l=new Label();
      l.setBounds(20,50,200,20);
      a=new TextArea();
      a.setBounds(20,80,300,300);
      a.addKeyListener(this);
      add(l);
      add(a);
      setSize(400,400);
      setLayout(null);
      setVisible(true);
   }
   public void keyPressed(KeyEvent e)
   {}
   public void keyReleased(KeyEvent e){
      String t=a.getText();
      String w[]=t.split("\\s");//split words
      l.setText("Words = "+w.length+" characters ="+t.length());

   }
   public void keyTyped(KeyEvent e)
   {}

   public static void main(String[] args) {
      new keyex();
   }
}
```

OUTPUT



RESULT : The program has been executed and the output was verified.

31. Producer/Consumer using ITC

```
import java.util.*;
class Q
{
int n;
boolean statusFlag=false;
synchronized void put(int n)
{
try
{
while(statusFlag)
{
wait();
}
}
catch(InterruptedException e){}
this.n=n;
System.out.println("Put :"+n);
statusFlag=true;
notify();
}
synchronized int get()
{
try{
while(!statusFlag)
{
wait();
}
}
catch(InterruptedException e){}
statusFlag=false;
System.out.println("Got :"+n);
notify();
return n;
}
}
class Producer implements Runnable
{
Q q;
Producer(Q q)
{
this.q=q;
```

```
new Thread(this, "Producer").start();
}
public void run()
{
int i=0;
while(true)
{
q.put(i++);
}
}
}
class Consumer implements Runnable
{
Q q;
Consumer(Q q)
{
this.q=q;
new Thread(this,"Consumer").start();
}
public void run()
{
while(true)
{
q.get();
}
}
}
public class D
{
public static void main(String[] args)
{
Q q=new Q();
Producer p=new Producer(q);
Consumer c=new Consumer(q);
}
}
```

OUTPUT

```
Got: 699
Put: 700
Got: 700
Put: 701
Got: 701
Put: 702
Got: 702
Put: 703
Got: 703
Put: 704
Got: 704
Put: 705
Got: 705
Put: 706
Got: 706
Put: 707
Got: 707
Put: 708
Got: 708
Put: 709
Got: 709
Put: 710
Got: 710
Put: 711
Got: 711
Put: 712
Got: 712
Put: 713
Got: 713
Put: 714
Got: 714
Put: 715
Got: 715
Put: 716
Got: 716
Put: 717
Got: 717
Put: 718
Got: 718
Put: 719
```

RESULT : The program has been executed and the output was verified.

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

```
import java.util.Scanner;
public class GenericStack {
private int[] arr;
private int top;
private int capacity;
GenericStack(int size) {
arr = new int[size];
capacity = size;
top = -1;

}
public void push(int x) {
if (isFull()) {
System.out.println("OverFlow");
}
else{
System.out.println("Inserting " + x);
arr[++top] = x;
}
}
public int pop() {
if (isEmpty()) {
System.out.println("STACK EMPTY");
return -1;
}
else {
return arr[top--];
}
}
public int size() {
return top + 1;
}
public Boolean isEmpty() {
return top == -1;
}
public Boolean isFull() {
return top == capacity - 1;
}
public void printStack() {
```

```
for (int i = 0; i <= top; i++) {
System.out.println(arr[i]);
}
}
public static void main(String[] args) {
Scanner scanner = new Scanner(System.in);
System.out.println("Enter size of stack : ");
int count = scanner.nextInt();
GenericStack stack = new GenericStack(count);
while (true) {
System.out.println("Enter operation : 1)Push 2)Pop 3)Display : ");
int choice = scanner.nextInt();
switch (choice) {
case 1:
System.out.println("item to insert :");
int item = scanner.nextInt();


stack.push(item);
break;
case 2:
stack.pop();
break;
case 3:
stack.printStack();
}
}
}
}
```

OUTPUT

```
GenericStack [Java Application] C:\Program Files\Java\jdk-10.0.1\bin\jav
Enter size of stack :
4
Enter operation : 1)Push 2)Pop 3)Display :
1
item to insert :
20
Inserting 20
Enter operation : 1)Push 2)Pop 3)Display :
1
item to insert :
40
Inserting 40
Enter operation : 1)Push 2)Pop 3)Display :
3
20
40
Enter operation : 1)Push 2)Pop 3)Display :
2
Enter operation : 1)Push 2)Pop 3)Display :
3
20
Enter operation : 1)Push 2)Pop 3)Display :
```

RESULT : The program has been executed and the output was verified.

33 . Using generic method perform 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[] = { 1, 6, 4, -1, -3, 9, 0, 7, -9, 5 };
    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

```
C:\java programs>java BubbleSort
Array Before Bubble Sort
1 6 4 -1 -3 9 0 7 -9 5
Array After Bubble Sort
-9 -3 -1 0 1 4 5 6 7 9
C:\java programs>
```

RESULT : The program has been executed and the output was verified.

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

```
import java.util.PriorityQueue;
import java.util.Scanner;

public class Main {

   public static void main(String[] args)
   {
      PriorityQueue<String> queue=new PriorityQueue<String>();

Scanner scanner=new Scanner(System.in);
      System.out.println("Enter the size of the queue : ");
      int n=scanner.nextInt();
      System.out.println("Enter the elements : ");
      for(int i =0;i<n;i++)
      {
         String st=scanner.next();
         queue.add(st);

      }
      System.out.println("head:"+queue.element());
      System.out.println("head:"+queue.peek());
      System.out.println("Iterating the queue elements\n : ");
      for (String value : queue) {
         System.out.println(value);
      }
      queue.remove();
      queue.poll();
      System.out.println("After removing two elements \n");
      for (String s : queue) {
         System.out.println(s);
      }
   }

}
```

OUTPUT

```
Problems  @ Javadoc  Declaration  Console
<terminated> Client [Java Application] C:\Program Files (x86)\Java\jre1.8.0_301\bin\javaw.exe
Enter the size of the queue :
3
Enter the elements :
blue
green
yellow
head:blue
head:blue
Iterating the queue elements
 :
blue
green
yellow
After removing two elements

yellow
```

RESULT : The program has been executed and the output was verified.

35. Program to remove all the elements from a linked list

```
import java.util.*;
public class Main {
public static void main(String[] args) {
int n;
String data;
LinkedList<String> linkedList = new LinkedList<String>();
System.out.println("Enter number of strings to be inserted : ");
Scanner scanner = new Scanner(System.in);
n = scanner.nextInt();
System.out.println("Enter the string(s) : ");
scanner.nextLine();

for (int i = 0; i < n; i++) {
data = scanner.nextLine();
linkedList.add(data);
}
System.out.println("LinkedList: " + linkedList);
System.out.println("Removing all the elements....");
linkedList.clear();
System.out.println(linkedList);
}
}
```

OUTPUT

```
Enter number of strings to be inserted :
2
Enter the string(s) :
tom
joseph
LinkedList: [tom, joseph]
Removing all the elements....
[]
PS C:\Users\tomma>
```

RESULT : The program has been executed and the output was verified.

36. Program to demonstrate the addition and deletion of elements in dequeue

```
import java.util.ArrayDeque;
import java.util.Deque;

public class JavaDequeExample1 {
  public static void main(String[] args) {
    Deque<Integer> deque = new ArrayDeque<Integer>();

    deque.add(5);
    deque.add(6);
    deque.add(7);
    System.out.println("Inserting three elements : ");
    for (Integer integer : deque) {
    System.out.println(integer);
    }

    deque.pop();
    System.out.println("After popping : ");
    for (Integer integer : deque) {
        System.out.println(integer);
    }
    deque.remove(7);
    System.out.println("Removing the element 7 :"+deque);
  }
}
```

OUTPUT

```
Inserting three elements :
5
6
7
After popping :
6
7
Removing the element 7 :[6]
PS C:\Users\tomma>
```

RESULT : The program has been executed and the output was verified.

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

```
import java.util.*;
class arrayjava{
public static void main(String args[]){
 ArrayList<String> alist=new ArrayList<String>();
alist.add("appu");
alist.add("ammu");
alist.add("minnu");      alist.add("thomu");
alist.add("pinky");    alist.add("Tom");
//displaying elements
 System.out.println(alist);
//Adding "appu" at the fourth position alist.add(3, "appu");
 //displaying elements
 System.out.println(alist);
}    }
```

OUTPUT

```
D:\MCA NOTES\Sem 2\Object Oriented Programming Lab\20-09-2021(Sr. Elsin)>javac arrayjava.java

D:\MCA NOTES\Sem 2\Object Oriented Programming Lab\20-09-2021(Sr. Elsin)>java arrayjava
[appu, ammu, minnu, thomu, pinky, Tom]
[appu, ammu, minnu, thomu, pinky, Tom]
```

RESULT : The program has been executed and the output was verified.

38 .Program to demonstrate the working of map interface by adding ,removing,changing.

```
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

```
D:\java_lab>javac hashmap.java

D:\java_lab>java hashmap
Jinu : 3
Anu  : 1
sinu : 2
```

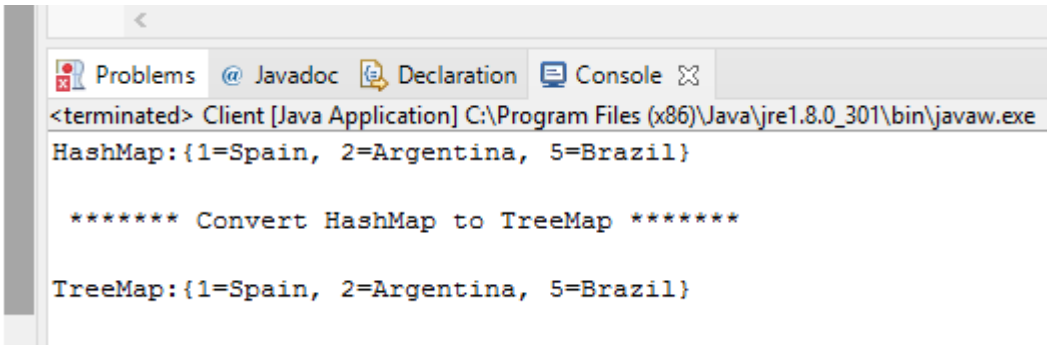
RESULT : The program has been executed and the output was verified.

39. Program to Convert HashMap to TreeMap

PROGRAM

```
import java.util.*;

public class Main {
   public static void main(String[] args) {
      Map<Integer,String> hm=new LinkedHashMap<>();
      hm.put(1,"England");
      hm.put(1,"Spain");
      hm.put(2,"France");
      hm.put(5,"Brazil");
      hm.put(2,"Argentina");
      System.out.println("HashMap:"+hm);
      Map<Integer,String> tm=new TreeMap<>(hm);
      System.out.println("\n ******* Convert HashMap to TreeMap
*******\n");
      System.out.println("TreeMap:"+tm);
   }
}
```

OUTPUT



RESULT : The program has been executed and the output was verified.

40. 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 internal level
for(int i=0;i<level;i++)
System.out.print("\t");
      if(arr[index].getName().toLowerCase().contains(searchfor))
      System.out.println(RED);
      else
      System.out.println(RESET);
      //for file
      if(arr[index].isFile())
      System.out.println(arr[index].getName());
         //check for sub folder and pint files in subfolder
      else if(arr[index].isDirectory())
      {
         System.out.println("["+arr[index].getName()+"]");

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

```
    public static void main(String[] args) {
        Scanner scan=new Scanner(System.in);
        System.out.println("Enter the directory path");
        String path=scan.nextLine();
        System.out.println("Enter the file name or directory name for search ");
        String searchfor=scan.nextLine();
        File maindir=new File(path);
        if(maindir.exists()&&maindir.isDirectory())//check it exist and is a folder
        {
            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,level-
folder(0) inside another folder(1),search ,,,call funtio


        }
    }
}
```

OUTPUT

```
C:\java programs>java p1
Enter the directory path
D:\Ubantu
Enter the file name or directory name for search
ubantu.txt
#####################################
files from main directoryD:\Ubantu
#####################################
←[0;31m
ubantu.txt
        ←[0m
ubuntu-21.04-desktop-amd64.iso
                ←[0m
VirtualBox-6.1.22-144080-Win.exe

C:\java programs>_
```

RESULT : The program has been executed and the output was verified.

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

```
import java.io.BufferedReader;
      import java.io.FileReader;
      import java.io.FileWriter;
      import java.io.IOException;

public class Main {

   public static void main(String[] args) {

      try {
         FileWriter writer = new FileWriter("java_write.txt",true);
         writer.write("new file is created");
         writer.close();
         FileReader reader = new FileReader("java_write.txt");
         BufferedReader br= new BufferedReader(reader);
         String line;
         System.out.println("Data read from the file");
         while ((line = br.readLine()) != null) {
            System.out.println(line);
         }
         reader.close();

      } catch (IOException e) {
         System.out.println("-----Error-----");
      }
   }
}
```

OUTPUT

```
Problems  @ Javadoc  Declaration  Console
<terminated> Main (6) [Java Application] C:\Program Files (x86)\Java\jre1.8.0_301\bin\javaw.exe
Data read from the file
new file is created
```

RESULT : The program has been executed and the output was verified.