OBJECT ORIENTED PROGRAMMING LAB

Name: SHAMJAD MAZOOD NAZER
Roll No: 36
Batch: B
Date: 29/03/2022

Experiment No.: 1

Aim

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

Procedure

```
class Product{
    String pcode, pname;
    double price;
    void details(){
            System.out.println("PRODUCT DETAILS");
            System.out.println("PCode : "+pcode);
            System.out.println("PName : "+pname);
            System.out.println("Price : "+price);
    }
}
public class ProductDetails{
    public static void main(String args[]){
            Product p1 = new Product();
            p1.pcode = "M200J9PI";
            p1.pname = "POCO M2";
            p1.price = 10999;
            System.out.println("\nProduct 1:-");
            p1.details();

            Product p2 = new Product();
            p2.pcode = "XMSH05HM";
            p2.pname = "Mi Band 3";
```

```
        p2.price = 1799;
        System.out.println("\nProduct 2:-");
        p2.details();

        Product p3 = new Product();
        p3.pcode = "EPSP5248";
        p3.pname = "Camlin Scale";
        p3.price = 5;
        System.out.println("\nProduct 3:-");
        p3.details();

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

Output

```
ajc@ubuntu20-04:~/Desktop/S2/Java/29-03-2022$ javac ProductDetails.java
ajc@ubuntu20-04:~/Desktop/S2/Java/29-03-2022$ java ProductDetails

Product 1:-
PRODUCT DETAILS
PCode : M200J9PI
PName : POCO M2
Price : 10999.0

Product 2:-
PRODUCT DETAILS
PCode : XMSH05HM
PName : Mi Band 3
Price : 1799.0

Product 3:-
PRODUCT DETAILS
PCode : EPSP5248
PName : Camlin Scale
Price : 5.0


Product with lowest price is :
PRODUCT DETAILS
PCode : EPSP5248
PName : Camlin Scale
Price : 5.0
ajc@ubuntu20-04:~/Desktop/S2/Java/29-03-2022$
```

OBJECT ORIENTED PROGRAMMING LAB

Name: SHAMJAD MAZOOD NAZER

Roll No: 36

Batch: B

Date: 05/04/2022

Experiment No.: 2

Aim

Read 2 matrices from the console and perform matrix addition.

Procedure

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

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

Output

```
ajc@ubuntu20-04:~/Desktop/S2/Java/LAB CYCLE 1$ javac MatrixAddition.java
ajc@ubuntu20-04:~/Desktop/S2/Java/LAB CYCLE 1$ java MatrixAddition
Enter the number of rows for the Matrices : 2
Enter the number of columns for the Matrices : 2
Enter the 2 elements for the Matrix A :
10 20
20 10


Enter the 2 elements for the Matrix B :
90 80
80 90


Matrix A is :
10 20

20 10

Matrix B is :
90 80

80 90

MatrixSum is :
100 100

100 100

ajc@ubuntu20-04:~/Desktop/S2/Java/LAB CYCLE 1$
```

OBJECT ORIENTED PROGRAMMING LAB

Name: SHAMJAD MAZOOD NAZER

Roll No: 36

Batch: B

Date: 05/04/2022

Experiment No.: 3

Aim

Java program to add complex numbers.

Procedure

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

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

Output

```
ajc@ubuntu20-04:~/Desktop/S2/Java/LAB CYCLE 1$ javac ComplexNumbers.java
ajc@ubuntu20-04:~/Desktop/S2/Java/LAB CYCLE 1$ java ComplexNumbers
Enter the real value of the 1st complex number : 3
Enter the imaginary value of the 1st complex number : 2
The entered complex number is : 3 + 2i


Enter the real value of the 2nd complex number : 2
Enter the imaginary value of the 2nd complex number : 3
The entered complex number is : 2 + 3i


The sum of the complex numbers are : 5 + 5i
ajc@ubuntu20-04:~/Desktop/S2/Java/LAB CYCLE 1$
```

OBJECT ORIENTED PROGRAMMING LAB

Name: SHAMJAD MAZOOD NAZER
Roll No: 36
Batch: B
Date: 06/04/2022

Experiment No.: 4

Aim

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

Procedure

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

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

Output

```
ajc@ubuntu20-04:~/Desktop/S2/Java/LAB CYCLE 1$ javac SymmetricMatrix.java
ajc@ubuntu20-04:~/Desktop/S2/Java/LAB CYCLE 1$ java SymmetricMatrix
Enter the number of rows for the Matrices : 3
Enter the number of columns for the Matrices : 3
Enter the elements for the Matrix :
1 0 0
0 1 0
0 0 1


The entered matrix is :
1   0   0

0   1   0

0   0   1

The entered matrix is Symmetric Matrix
ajc@ubuntu20-04:~/Desktop/S2/Java/LAB CYCLE 1$
```

OBJECT ORIENTED PROGRAMMING LAB

Name: SHAMJAD MAZOOD NAZER
Roll No: 36
Batch: B
Date: 17/05/2022

Experiment No.: 5

Aim

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

Procedure

```
class CPU{
    int price = 29999;
    class processor{
            int cores = 8;
            String mgf = "AMD RAYZEN 5";
    }
    static class RAM{
            int memory = 16;
            String mgf = "SAMSUNG";
    }
}
public class cpuInfo{
    public static void main(String args[]){
            CPU CPU_Obj = new CPU();
            CPU.processor processor_Obj = CPU_Obj.new processor();
            CPU.RAM RAM_Obj = new CPU.RAM();
            System.out.println("Number of CPU Cores          : "+processor_Obj.cores);
            System.out.println("Number of CPU Manufacturer  : "+processor_Obj.mgf);
            System.out.println("Volume of Memory            : "+RAM_Obj.memory);
            System.out.println("Number of Memory Manufacturer      : "+RAM_Obj.mgf);
            System.out.println("Price                       : "+CPU_Obj.price);
    }
}
```

Output

```
student@U36:~/Desktop/RMCA-B/OOP/17-05-2022$ java cpuInfo
Number of CPU Cores               : 8
Number of CPU Manufacturer        : AMD RAYZEN 5
Volume of Memory                  : 16
Number of Memory Manufacturer     : SAMSUNG
Price                             : 29999
student@U36:~/Desktop/RMCA-B/OOP/17-05-2022$
```

OBJECT ORIENTED PROGRAMMING LAB

Name: SHAMJAD MAZOOD NAZER

Roll No: 36

Batch: B

Date: 23/04/2022

Experiment No.: 6

Aim

Write a java program to Sort strings.

Procedure

```
import java.util.*;
public class ArrSort{
    public static void main(String args[]){
            int size;
            Scanner sc=new Scanner(System.in);
            System.out.print("Enter the size of the Array : ");
            size = sc.nextInt();
            String[] Names = new String[size];
            System.out.println("Enter the Elements of the Array : ");
            for(int i=0; i<size; i++){
                    Names[i] = sc.nextLine();
            }
            Arrays.sort(Names);
            for(int i=0; i<size; i++){
                    System.out.print(Names[i]);
            }
    }
}
```

Output

```
ajc@ubuntu20-04:~/Desktop/S2/Java/CO2$ javac ArrSort.java
ajc@ubuntu20-04:~/Desktop/S2/Java/CO2$ java ArrSort
Enter the size of the Array : 5
Enter the Elements of the Array :
SHAMJAD
JADSPRIT
HARISH
ANWAR

ANWAR
HARISH
JADSPRIT
SHAMJAD
ajc@ubuntu20-04:~/Desktop/S2/Java/CO2$
```

OBJECT ORIENTED PROGRAMMING LAB

Name: SHAMJAD MAZOOD NAZER

Roll No: 36

Batch: B

Date: 21/04/2022

Experiment No.: 7

Aim

Write a java program to search for an element in the array.

Procedure

```
import java.util.*;
class ArrSearch{
    public static void main(String[] args){
            Scanner sc= new Scanner(System.in);
            int size, item, flag = 0;
            System.out.print("Enter the size for the array : ");
            size = sc.nextInt();
            int[] arr = new int[size];
            System.out.println("\nEnter the elements for the array : ");
            for(int i=0; i<size; i++)
                    arr[i]= sc.nextInt();
            System.out.print("Enter the item to search : ");
            item = sc.nextInt();
            for(int i=0; i<size; i++){
                    if(arr[i] == item)
                            flag = 1;
            }
            if(flag == 1)
                    System.out.println(item+" found in the Array!");
            else
                    System.out.println(item+" not found in the Array!");
    }
}
```

Output

```
ajc@ubuntu20-04:~/Desktop/S2/Java/CO2$ javac ArrSearch.java
ajc@ubuntu20-04:~/Desktop/S2/Java/CO2$ java ArrSearch
Enter the size for the array : 5

Enter the elements for the array :
2
3
5
4
1
Enter the item to search : 5
5 found in the Array!
ajc@ubuntu20-04:~/Desktop/S2/Java/CO2$
```

OBJECT ORIENTED PROGRAMMING LAB

Name: SHAMJAD MAZOOD NAZER

Roll No: 36

Batch: B

Date: 23/04/2022

Experiment No.: 8

Aim

Write a java program to perform string manipulations.

Procedure

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

        String str1= "This is String1 ", str2="This is String2.";
        System.out.println("The string 01 is : "+str1+"\nString 02 is : "+str2);

        String strconcat= str1+str2;
        System.out.println("\nConcatenation of two Strings is : "+strconcat);

        String strUppercase= str1.toUpperCase();
        System.out.println("\nNormal String1 to uppercase string is : "+strUppercase);

        String strLowercase= str2.toLowerCase();
        System.out.println("\nNormal String2 to lowercase string is : "+strLowercase);

        String strsubString= str1.substring(5);
        System.out.println("\nSubstring of the string1 is : "+strsubString);

        String strtrim= str1.trim();
        System.out.println("\nString1 trim is given by : "+strtrim);

        boolean strcontains= str1.contains("String1");
        System.out.println("\nCheck if the String1 contains -'String1' : "+strcontains);
```

```
        int strlength= str2.length();
        System.out.println("\nThe length of the String2 is : "+strlength);
    }
}
```

Output

```
student@U36:~/Desktop/RMCA-B/OOP/23-04-2022$ java StringManipulation
The string 01 is : This is String1
String 02 is : This is String2.

Concatenation of two Strings is : This is String1 This is String2.

Normal String1 to uppercase string is : THIS IS STRING1

Normal String2 to lowercase string is : this is string2.

Substring of the string1 is : is String1

String1 trim is given by : This is String1

Check if the String1 contains -'String1' : true

The length of the String2 is : 16
student@U36:~/Desktop/RMCA-B/OOP/23-04-2022$
```

OBJECT ORIENTED PROGRAMMING LAB

Name: SHAMJAD MAZOOD NAZER

Roll No: 36

Batch: B

Date: 23/04/2022

Experiment No.: 9

Aim

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

Procedure

```
import java.util.*;
class Employee{
    int eno, esalary;
    String ename;
    Employee(){
    }

    Employee(int eno, String ename, int esalary){
            this.eno= eno;
            this.ename= ename;
            this.esalary= esalary;
    }
}

public class EmpArray {
    public static void main(String[] args){
            int size, search_emp, eno, esalary;
            String ename;
            Scanner sc = new Scanner(System.in);
            System.out.print("Enter the Limit : ");
            size = sc.nextInt();
            Employee[] emp_arr = new Employee[size];
```

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

                System.out.print("Enter the employee number of "+(i+1)+" : ");
                eno= sc.nextInt();

                System.out.print("Enter the name for the employee "+(i+1)+": ");
                ename= sc.next();

                System.out.print("Enter the salary for the employee "+(i+1)+": ");
                esalary= sc.nextInt();

                Employee emp= new Employee(eno,ename,esalary);
                emp_arr[i]= emp;

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

        System.out.print("Enter the emp no of the employee that you want to search : ");
        search_emp = sc.nextInt();
        for(int i=0; i<size; i++){
                if(emp_arr[i].eno == search_emp){
                        System.out.println("RESULT FOUND.\n");
                        System.out.println("Emp no : "+emp_arr[i].eno);
                        System.out.println("Emp name : "+emp_arr[i].ename);
                        System.out.println("Emp salary : "+emp_arr[i].esalary);
                }
                else{
                        System.out.println("RESULT NOT FOUND!");
                }
        }
        sc.close();
}
```

}

Output

```
student@U36:~/Desktop/RMCA-B/OOP/23-04-2022$ java EmpArray
Enter the Limit : 2
Enter the employee number of 1 : 36
Enter the name for the employee 1: SHAMJAD
Enter the salary for the employee 1: 25000


Enter the employee number of 2 : 93
Enter the name for the employee 2: JADSPRIT
Enter the salary for the employee 2: 20000


Enter the emp no of the employee that you want to search : 93
RESULT NOT FOUND!
RESULT FOUND.

Emp no : 93
Emp name : JADSPRIT
Emp salary : 20000
student@U36:~/Desktop/RMCA-B/OOP/23-04-2022$
```

OBJECT ORIENTED PROGRAMMING LAB

Name: SHAMJAD MAZOOD NAZER
Roll No: 36
Batch: B
Date: 17/05/2022

Experiment No.: 10

Aim

Write a java program to find the area of different shapes using overloaded functions.

Procedure

```
import java.util.*;
class Area{
    int l,b,h;
    void area(int l)
    {
            System.out.println("Area of Square is : "+(l*l));
    }
    void area(int l, float b)
    {
            System.out.println("Area of Rectangle is : "+(l*b));
    }
    void area(float b, float h)
    {
            System.out.println("Area of Triangle is : "+(b*h)/2);
    }
    void area(float r)
    {
            System.out.println("Area of Circle is : "+(3.14*r*r));
    }
}
public class functionOverloading{
    public static void main(String[] args){
            int l;
```

```
        float h, r, b;
        Area ar = new Area();
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter the length : ");
        l = sc.nextInt();
        System.out.println("Enter the breadth : ");
        b = sc.nextInt();
        System.out.println("Enter the height : ");
        h = sc.nextInt();
        System.out.println("Enter the radius : ");
        r = sc.nextInt();
        ar.area(l);
        ar.area(l, b);
        ar.area(b, h);
        ar.area(r);
    }
}
```

Output

```
student@U36:~/Desktop/RMCA-B/OOP/17-05-2022$ java functionOverloading
Enter the length :
2
Enter the breadth :
2
Enter the height :
5
Enter the radius :
6
Area of Square is : 4
Area of Rectangle is : 4.0
Area of Triangle is : 5.0
Area of Circle is : 113.03999999999999
student@U36:~/Desktop/RMCA-B/OOP/17-05-2022$
```

OBJECT ORIENTED PROGRAMMING LAB

Name: SHAMJAD MAZOOD NAZER

Roll No: 36

Batch: B

Date: 17/05/2022

Experiment No.: 11

Aim

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

Procedure

```
import java.util.*;
class Employee
{
   int empid;
   double salary;
   String name, address;
   Employee(int emp_id, String emp_name, String emp_address, double emp_salary){
            empid = emp_id;
            name = emp_name;
            address = emp_address;
            salary = emp_salary;
   }
}
class Teacher extends Employee
{
   String department, subject;
   Teacher(int empid, String name, String address,  String dept, String sub, double salary)
   {
            super(empid, name, address, salary);
            department = dept;
```

```
            subject = sub;
        }
        void display()
        {
            System.out.println("EMP ID : "+empid);
            System.out.println("EMP NAME : "+name);
            System.out.println("EMP ADDRESS : "+address);
            System.out.println("EMP DEPARTMENT : "+department);
            System.out.println("SUBJECT TAUGHT : "+subject);
            System.out.println("EMP SALARY : "+salary);
        }
    }

    public class InheritEmployee{
        public static void main(String args[]){
            int n, empid;
            String name, address, department, subject;
            double salary;
            Scanner sc = new Scanner(System.in);
            System.out.println("Enter the limit : ");
            n = sc.nextInt();
            Teacher[] arr = new Teacher[n];
            for(int i=0; i<n; i++){
                System.out.print("Enter the Employee ID : ");
                empid = sc.nextInt();
                System.out.print("Enter the Employee Name : ");
                name = sc.next();
                System.out.print("Enter the Employee Address : ");
                address = sc.next();
                System.out.print("Enter the Employee Department : ");
                department = sc.next();
                System.out.print("Enter the Employee Subject : ");
```

```
                subject = sc.next();
                System.out.print("Enter the Employee Salary : ");
                salary = sc.nextDouble();

                arr[i] = new Teacher(empid, name, address, department, subject, salary);
            }
            for(int i=0; i<n; i++){
                arr[i].display();
            }
    }
}
```

Output

```
student@U36:~/Desktop/RMCA-B/OOP/24-05-2022$ javac InheritEmployee.java
student@U36:~/Desktop/RMCA-B/OOP/24-05-2022$ java InheritEmployee
Enter the limit :
2
Enter the Employee ID : 36
Enter the Employee Name : SHAMAJD
Enter the Employee Address : MELEVEETTIL
Enter the Employee Department : MCA
Enter the Employee Subject : OOP
Enter the Employee Salary : 25000.00
Enter the Employee ID : 93
Enter the Employee Name : JADSPRIT
Enter the Employee Address : JAD
Enter the Employee Department : ME
Enter the Employee Subject : ADS
Enter the Employee Salary : 49999.99
EMP ID : 36
EMP NAME : SHAMAJD
EMP ADDRESS : MELEVEETTIL
EMP DEPARTMENT : MCA
SUBJECT TAUGHT : OOP
EMP SALARY : 25000.0
EMP ID : 93
EMP NAME : JADSPRIT
EMP ADDRESS : JAD
EMP DEPARTMENT : ME
SUBJECT TAUGHT : ADS
EMP SALARY : 49999.99
student@U36:~/Desktop/RMCA-B/OOP/24-05-2022$
```

OBJECT ORIENTED PROGRAMMING LAB

Name: SHAMJAD MAZOOD NAZER

Roll No: 36

Batch: B

Date: 24/05/2022

Experiment No.: 12

Aim

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

Procedure

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

class Employee extends Person{
    int empid, salary;
    String company_name, qualification;
    Employee(String Name, String Gender, String Address, int Age, int EmpId, String
CompanyName, String Qualify, int Salary){
            super(Name, Gender, Address, Age);
            empid = EmpId;
            company_name = CompanyName;
```

```
            qualification = Qualify;
            salary = Salary;
        }
    }

    class Teacher extends Employee{
        int tid;
        String dept, sub;
        Teacher(String Name, String Gender, String Address, int Age, int empid, String company_name,
    String qualification, int salary, int teacherid, String department, String subject){
            super(Name, Gender, Address, Age, empid, company_name, qualification, salary);
            tid = teacherid;
            dept = department;
            sub = subject;
        }

        void Display(int i){
            System.out.println("DETAILS ("+(i+1)+"):-\n----------\n");
            System.out.println("EMP-ID : "+empid);
            System.out.println("TEACHER-ID : "+tid);
            System.out.println("NAME : "+Name);
            System.out.println("ADDRESS : "+Address);
            System.out.println("GENDER : "+Gender);
            System.out.println("AGE : "+Age);
            System.out.println("COMANY NAME : "+company_name);
            System.out.println("QUALIFICATION : "+qualification);
            System.out.println("DEPARTMENT : "+dept);
            System.out.println("SUBJECT : "+sub);
            System.out.println("SALARY : "+salary);
        }
    }

    public class PersonEmployee{
```

```
    public static void main(String args[]){
        int n, empid, age, teacherid, salary;
        String name, address, gender, companyname, qualify, dept, sub;
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter the Limit : ");
        n = sc.nextInt();
        Teacher[] arr = new Teacher[n];
        for(int i=0; i<n; i++){
            System.out.println("Enter the ["+(i+1)+"] Employee ID : ");
            empid = sc.nextInt();

            System.out.println("Enter the Teacher ID : ");
            teacherid = sc.nextInt();

            System.out.println("Enter the Name : ");
            name = sc.next();

            System.out.println("Enter the Address : ");
            address = sc.next();

            System.out.println("Enter the Gender : ");
            gender = sc.next();

            System.out.println("Enter the Age : ");
            age = sc.nextInt();

            System.out.println("Enter the Company Name : ");
            companyname = sc.next();

            System.out.println("Enter the Qualification : ");
            qualify = sc.next();
```

```
                System.out.println("Enter the Deaprtment : ");
                dept = sc.next();

                System.out.println("Enter the Subject : ");
                sub = sc.next();

                System.out.println("Enter the Salary : ");
                salary = sc.nextInt();

                arr[i] = new Teacher(name, gender, address, age, empid, companyname, qualify,
salary, teacherid, dept, sub);
            }
            for(int i=0; i<n; i++){
                arr[i].Display(i);
            }
    }
}
```

Output

```
student@U36:~/Desktop/RMCA-B/OOP/24-05-2022$ javac PersonEmployee.java
student@U36:~/Desktop/RMCA-B/OOP/24-05-2022$ java PersonEmployee
Enter the Limit :
2
Enter the [1] Employee ID :
36
Enter the Teacher ID :
101
Enter the Name :
SHAMJAD
Enter the Address :
MELEVEETTIL
Enter the Gender :
M
Enter the Age :
22
Enter the Company Name :
AJC
Enter the Qualification :
BCA
Enter the Deaprtment :
MCA
Enter the Subject :
JAVA
Enter the Salary :
490000
Enter the [2] Employee ID :
93
Enter the Teacher ID :
102
Enter the Name :
JADSPRIT
Enter the Address :
JADALAYAM
```

```
Enter the Gender :
M
Enter the Age :
22
Enter the Company Name :
IND
Enter the Qualification :
BE
Enter the Deaprtment :
BOWLER
Enter the Subject :
FAST
Enter the Salary :
100000
```

```
DETAILS (1):-
--------------

EMP-ID : 36
TEACHER-ID : 101
NAME : SHAMJAD
ADDRESS : MELEVEETTIL
GENDER : M
AGE : 22
COMANY NAME : AJC
QUALIFICATION : BCA
DEPARTMENT : MCA
SUBJECT : JAVA
SALARY : 490000
DETAILS (2):-
--------------

EMP-ID : 93
TEACHER-ID : 102
NAME : JADSPRIT
ADDRESS : JADALAYAM
GENDER : M
AGE : 22
COMANY NAME : IND
QUALIFICATION : BE
DEPARTMENT : BOWLER
SUBJECT : FAST
SALARY : 100000
```

OBJECT ORIENTED PROGRAMMING LAB

Name: SHAMJAD MAZOOD NAZER

Roll No: 36

Batch: B

Date: 24/05/2022

Experiment No.: 13

Aim

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

Procedure

```
import java.util.*;
class publisher{
    String pubname;
    publisher(String pub){
            this.pubname=pub;
    }
}
class Books extends publisher{
    String Bookname;
    Books(String pub,String B){
            super(pub);
            Bookname=B;
    }
}
class Literature extends Books{
    String Litname;
    Literature(String pub,String B,String L){
            super(pub, B);
            Litname=L;
    }
    void display(){
            System.out.println(".....");
```

```
            System.out.println("Publisher name :+super.pubname");
            System.out.println("Book Name :+super.Bookname");
            System.out.println("Literature Name :+Litname");
            System.out.println(".....");
    }
}
class Fiction extends Books{
    String ficname;
    Fiction(String pub,String B,String F){
            super(pub, B);
            ficname=F;
    }
    void display(){
    System.out.println(".....");
    System.out.println("Publisher name : "+super.pubname);
    System.out.println("Book Name : "+super.Bookname);
    System.out.println("Fiction Name : "+ficname);
    System.out.println(".....");
    }
}
class BookInherit{
    public static void main(String[] args){
            String pub_name, book_name,lit_name,fic_name,p_name, b_name;
            Scanner s=new Scanner(System.in);
            System.out.println("Enter the Publication Name : ");
            pub_name=s.next();
            System.out.println("Enter the Book Name : ");
            book_name=s.next();
            System.out.println("Enter the Literature Name : ");
            lit_name=s.next();
            System.out.println("Enter the Publication Name : ");
            p_name=s.next();
```

```
            System.out.println("Enter the Book Name : ");
            b_name=s.next();
            System.out.println("Enter the Fiction Name : ");
            fic_name=s.next();
            Literature obj1=new Literature(pub_name,book_name,lit_name);
            Fiction obj2=new Fiction(p_name,b_name,fic_name);
            obj1.display();
            obj2.display();
    }
}
```

Output

```
ajc@36:~/Desktop/S2/Java/22-04-2022$ javac BookInherit.java
ajc@36:~/Desktop/S2/Java/22-04-2022$ java BookInherit
Enter the Publication Name :
MCU
Enter the Book Name :
EndGame
Enter the Literature Name :
Comics
Enter the Publication Name :
DC
Enter the Book Name :
JusticeLeague
Enter the Fiction Name :
Sci
.........................................
Publisher name :+super.pubname
Book Name :+super.Bookname
Literature Name :+Litname
.........................................
.........................................
Publisher name : DC
Book Name : JusticeLeague
Fiction Name : Sci
.........................................
ajc@36:~/Desktop/S2/Java/22-04-2022$
```

OBJECT ORIENTED PROGRAMMING LAB

Name: SHAMJAD MAZOOD NAZER

Roll No: 36

Batch: B

Date: 24/05/2022

Experiment No.: 14

Aim

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

Procedure

```
import java.util.*;
class Student{
    int sid, mark1, mark2;
    String name, dept;
    Student(int sid, String name, String dept, int mark1, int mark2){
            this.sid = sid;
            this.name = name;
            this.dept = dept;
            this.mark1 = mark1;
            this.mark2 = mark2;
    }
}
class Sports extends Student{
    int rank;
    String item;
    Sports(int sid, String name, String dept, int mark1, int mark2, int rank, String item){
            super(sid, name, dept, mark1, mark2);
            this.rank = rank;
            this.item = item;
    }
}
```

```
class Result extends Sports{
    int total;
    Result(int sid, String name, String dept, int mark1, int mark2, int rank, String item){
            super(sid, name, dept, mark1, mark2, rank, item);
            total = mark1+mark2;
    }
    void display(){
            System.out.println("Student ID \t\t: "+sid);
            System.out.println("Student Name \t\t: "+name);
            System.out.println("Student Department \t: "+dept);
            System.out.println("Student Mark1 \t\t: "+mark1);
            System.out.println("Student Mark2 \t\t: "+mark2);
            System.out.println("Sports rank \t\t: "+rank);
            System.out.println("Sport Item \t\t: "+item);
            System.out.println("Student Total Mark \t: "+total);
    }
}

public class Academic{
    public static void main(String args[]){
            int n, i, sid, mark1, mark2, rank;
            String name, dept, item;
            Scanner sc = new Scanner(System.in);
            System.out.print("Enter the Limit : ");
            n = sc.nextInt();
            Result[] arr = new Result[n];
            for(i=0;i<n;i++){
                    System.out.println("Enter the Details of
Student["+(i+1)+"]\n----------------------------------------\n");
                    System.out.print("Enter Student ID \t\t: ");
                    sid = sc.nextInt();
                    System.out.print("Enter Student Name \t\t: ");
                    name = sc.next();
```

```
                System.out.print("Enter Student Department \t: ");
                dept = sc.next();
                System.out.print("Enter Student Mark1 \t\t: ");
                mark1 = sc.nextInt();
                System.out.print("Enter Student Mark2 \t\t: ");
                mark2 = sc.nextInt();
                System.out.print("Enter Sport rank \t\t: ");
                rank = sc.nextInt();
                System.out.print("Enter Sport item \t\t: ");
                item = sc.next();
                arr[i] = new Result(sid, name, dept, mark1, mark2, rank, item);
        }
        for(i=0; i<n; i++){
                System.out.println("\nDetails of the Student["+(i+1)+"] is : \n");
                arr[i].display();
        }
        sc.close();
    }
}
```

Output

```
student@U36:~/Desktop/RMCA-B/OOP/25-05-2022$ javac Academic.java
student@U36:~/Desktop/RMCA-B/OOP/25-05-2022$ java Academic
Enter the Limit : 2
Enter the Details of Student[1]
----------------------------------------

Enter Student ID                : 36
Enter Student Name              : SHAMJAD
Enter Student Department        : MCA
Enter Student Mark1             : 28
Enter Student Mark2             : 40
Enter Sport rank                : 1
Enter Sport item                : VOLEYBALL
Enter the Details of Student[2]
----------------------------------------

Enter Student ID                : 93
Enter Student Name              : JADSPRIT
Enter Student Department        : ME
Enter Student Mark1             : 40
Enter Student Mark2             : 39
Enter Sport rank                : 2
Enter Sport item                : CRICKET
```

```
Details of the Student[1] is :

Student ID              : 36
Student Name            : SHAMJAD
Student Department      : MCA
Student Mark1           : 28
Student Mark2           : 40
Sports rank             : 1
Sport Item              : VOLEYBALL
Student Total Mark      : 68

Details of the Student[2] is :

Student ID              : 93
Student Name            : JADSPRIT
Student Department      : ME
Student Mark1           : 40
Student Mark2           : 39
Sports rank             : 2
Sport Item              : CRICKET
Student Total Mark      : 79
student@U36:~/Desktop/RMCA-B/OOP/25-05-2022$
```

OBJECT ORIENTED PROGRAMMING LAB

Name: SHAMJAD MAZOOD NAZER

Roll No: 36

Batch: B

Date: 24/05/2022

Experiment No.: 15

Aim

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

Procedure

```
import java.util.Scanner;
interface calc{
    final double pi = 3.14;
    void area();
    void perimeter();
}

class Circle implements calc{
    int r;
    Circle(int r){
            this.r = r;
    }

    public void area(){
            System.out.println("Area of Circle is : "+(pi*this.r*this.r));
    }
    public void perimeter(){
            System.out.println("Perimeter of Circle is : "+(2*pi*this.r));
    }
}
class Rectangle implements calc{
    int l, b;
```

```
    Rectangle(int l, int b){
        this.l = l;
        this.b = b;
    }
    public void area(){
        System.out.println("Area of Circle is : "+(this.l*this.b));
    }
    public void perimeter(){
        System.out.println("Perimeter of Circle is : "+(2*(this.l+this.b)));
    }
}

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

        Scanner sc = new Scanner(System.in);
        int rad, l, b, i, ch;
        do{
            System.out.print("1) Circle\n2) Rectangle\n3) Exit\nEnter Your Choice : ");
            ch = sc.nextInt();
            switch(ch){
                case 1:
                    System.out.print("Enter the radius : ");
                    rad = sc.nextInt();
                    calc c = new Circle(rad);
                    c.area();
                    c.perimeter();
                break;

                case 2:
                    System.out.print("Enter the Length : ");
                    l = sc.nextInt();
```

```
                    System.out.print("Enter the Breadth : ");
                    b = sc.nextInt();
                    calc r = new Rectangle(l, b);
                    r.area();
                    r.perimeter();
                break;
                case 3:
                System.exit(0);
                default:
                    System.out.print("Invalid Choice!");
            }
        }while(ch!=3);
    }
}
```

Output

```
ajc@36:~/Desktop/S2/Java/interface$ javac InterfaceAreaPerimeter.java
ajc@36:~/Desktop/S2/Java/interface$ java InterfaceAreaPerimeter
1) Circle
2) Rectangle
3) Exit
Enter Your Choice : 1
Enter the radius : 2
Area of Circle is : 12.56
Perimeter of Circle is : 12.56
1) Circle
2) Rectangle
3) Exit
Enter Your Choice : 2
Enter the Length : 3
Enter the Breadth : 6
Area of Circle is : 18
Perimeter of Circle is : 18
1) Circle
2) Rectangle
3) Exit
Enter Your Choice : 3
ajc@36:~/Desktop/S2/Java/interface$
```

OBJECT ORIENTED PROGRAMMING LAB

Name: SHAMJAD MAZOOD NAZER

Roll No: 36

Batch: B

Date: 17/05/2022

Experiment No.: 16

Aim

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

Procedure

```
import java.util.Scanner;
interface calc{
   void calculate();
}
class bill implements calc{
   String date,name,p_id;
   int quantity;
   double unit_price,total,namount=0;
   Scanner sc = new Scanner(System.in);
   public void getdata(){
      System.out.println("\nEnter product id:");
      p_id = sc.nextLine();
      System.out.println("Enter product name:");
      name = sc.nextLine();
      System.out.println("Enter the Quantity:");
      quantity = sc.nextInt();
      System.out.println("Enter the unit price:");
      unit_price = sc.nextDouble();
   }
   public void calculate(){
      total = quantity * unit_price;
   }
   public void display(){
```

```
        System.out.println(p_id+"\t\t"+name+"\t\t"+quantity+"\t\t"+unit_price+"\t"+total);
    }
}
public class Bill {
    public static void main(String[] args) {
        int n,i;
        double namount=0,t;
        int ran;
        String date;
        t = Math.random() *1000000;
        ran = (int) t;
        Scanner sc = new Scanner(System.in);
        System.out.println("Order no. #"+ran);
        System.out.print("Enter the date:");
        date = sc.nextLine();
        System.out.println("Enter how many products are there:");
        n = sc.nextInt();
        bill ob[] = new bill[n];
        for(i=0;i<n;i++)
            ob[i] = new bill();
        for(i=0;i<n;i++){
            ob[i].getdata();
            ob[i].calculate();
        }
        System.out.println("\nDate:"+date);
        System.out.println("Product Id  Name   Quantity  unit price  Total ");
        System.out.println("---------------------------------------------------------------");
        for(i=0;i<n;i++){
            ob[i].display();
            namount += ob[i].total;
        }
        System.out.println("---------------------------------------------------------------");
```

```
        System.out.println("\t\t\t\tNet.Amount\t\t"+ namount);


    }
}
```

Output

```
ajc@36:~/Desktop/S2/Java$ java Bill
Order no. #192214
Enter the date:20-04-2022
Enter how many products are there:
2

Enter product id:
101
Enter product name:
POCO M2(6/128)
Enter the Quantity:
2
Enter the unit price:
11499

Enter product id:
201
Enter product name:
Mi Band3
Enter the Quantity:
5
Enter the unit price:
1799

Date:20-04-2022
Product Id  Name   Quantity  unit price  Total
-------------------------------------------------------------
101             POCO M2(6/128)          2               11499.0 22998.0
201             Mi Band3                5               1799.0  8995.0
-------------------------------------------------------------
                                Net.Amount              31993.0
ajc@36:~/Desktop/S2/Java$
```

OBJECT ORIENTED PROGRAMMING LAB

Name: SHAMJAD MAZOOD NAZER

Roll No: 36

Batch: B

Date: 31/05/2022

Experiment No.: 17

Aim

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

Procedure

```
import java.util.*;
import GraphicPackage.*;
public class PackageProgram{
    public static void main(String[] args){
            int ch;
            double l, b, a, r, h;
            Scanner sc = new Scanner(System.in);

            do{
                    System.out.print("1. Rectangle\n2. Circle\n3. Triangle\n4. Square\n5. Exit\nEnter Your Choice : ");
                    ch = sc.nextInt();
                    switch(ch){

                            case 1:
                                    System.out.print("Enter the Length : ");
                                    l = sc.nextDouble();
                                    System.out.print("Enter the Breadth : ");
                                    b = sc.nextDouble();
                                    Rectangle rect = new Rectangle(l, b);
                                    rect.Area();
                            break;
```

```
                    case 2:
                        System.out.print("Enter the Radius : ");
                        r = sc.nextDouble();
                        Circle cir = new Circle(r);
                        cir.Area();
                    break;

                    case 3:
                        System.out.print("Enter the Base : ");
                        b = sc.nextDouble();
                        System.out.print("Enter the Height : ");
                        h = sc.nextDouble();
                        Triangle tri = new Triangle(b,h);
                        tri.Area();
                    break;

                    case 4:
                        System.out.print("Enter the Side : ");
                        a = sc.nextDouble();
                        Square sq = new Square(a);
                        sq.Area();
                    break;

                    case 5:
                        System.exit(0);

                    default:
                        System.out.println("Invalid Choice!");
                }
            }while(ch != 5);
        }
    }
```

Output

```
ajc@36:~/Desktop/S2/Java/PKG$ javac PackageProgram.java
ajc@36:~/Desktop/S2/Java/PKG$ java PackageProgram
1. Rectangle
2. Circle
3. Triangle
4. Square
5. Exit
Enter Your Choice : 1
Enter the Length : 10
Enter the Breadth : 20
Area of the Rectangle is : 200.0
1. Rectangle
2. Circle
3. Triangle
4. Square
5. Exit
Enter Your Choice : 2
Enter the Radius : 5
Area of the Circle is : 78.5
1. Rectangle
2. Circle
3. Triangle
4. Square
5. Exit
Enter Your Choice : 3
Enter the Base : 5
Enter the Height : 8
Area of the Triangle is : 20.0
1. Rectangle
2. Circle
3. Triangle
4. Square
5. Exit
Enter Your Choice : 4
Enter the Side : 3
```

```
Enter Your Choice : 4
Enter the Side : 3
Area of the Square is : 9.0
1. Rectangle
2. Circle
3. Triangle
4. Square
5. Exit
Enter Your Choice : 5
ajc@36:~/Desktop/S2/Java/PKG$
```

OBJECT ORIENTED PROGRAMMING LAB

Name: SHAMJAD MAZOOD NAZER
Roll No: 36
Batch: B
Date: 31/05/2022

Experiment No.: 18

Aim

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

Procedure

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

class IncompleteException extends Exception{
    public IncompleteException(String error){
            super(error);
    }
}

public class UnamePwdException{
    public static void main(String args[]){
            Scanner sc = new Scanner(System.in);
            String uname, pwd;
            int i;
            System.out.print("Enter the user name : ");
            uname = sc.next();
            System.out.print("Enter the password : ");
            pwd = sc.next();
            System.out.println("Your user name is : "+uname);
            System.out.print("Your password is : ");
            for(i=0; i<pwd.length(); i++){
                    System.out.print("*");
            }
```

```
            System.out.println();
            try{
                    if(uname == null && uname == " "){
                            throw new IncompleteException("USERNAME is Incomplete or
NULL!");
                    }
                    else if(pwd == null && pwd == " "){
                            throw new IncompleteException("PASSWORD is missing or NULL!");
                    }
                    else if(pwd.length() <= 6){
                            throw new IncompleteException("PASSWORD character must be greater
than 6 characters!");
                    }
            }
            catch(Exception e){
                    System.out.println(e);
            }
    }
}
```

Output

```
ajc@36:~/Desktop/S2/Java/Exception$ javac UnamePwdException.java
ajc@36:~/Desktop/S2/Java/Exception$ java UnamePwdException
Enter the user name : JADSPRIT
Enter the password : sorry
Your user name is : JADSPRIT
Your password is : *****
IncompleteException: PASSWORD character must be greater than 6 characters!
ajc@36:~/Desktop/S2/Java/Exception$
```

OBJECT ORIENTED PROGRAMMING LAB

Name: SHAMJAD MAZOOD NAZER

Roll No: 36

Batch: B

Date: 01/06/2022

Experiment No.: 19

Aim

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

Procedure

```
import java.io.*;
import java.util.Scanner;
class negativeInput extends Exception{
    public negativeInput(String msg){
            super(msg);
    }
}

public class avgNExc{
    public static void main(String args[]){
            Scanner sc = new Scanner(System.in);
            int n, i, temp, sum=0;
            double avg=0;
            System.out.print("Enter the limit : ");
            n = sc.nextInt();
            int[] a = new int[n];
            System.out.println("Enter the elements : ");
            for(i=0; i<n; i++){
                    temp = sc.nextInt();
                     try{
                            if(temp < 0){
                                    i--;
```

```
                                throw new negativeInput("Negative number cannot be
calculated!");
                        }
                        else{
                                a[i] = temp;
                        }
                }
                catch(Exception e){
                        System.out.println(e);
                }
        }
        System.out.print("Array is : ");
        for(i=0; i<n; i++){
                System.out.print(a[i]+" ");
                sum += a[i];
        }
        avg = sum/n;
        System.out.println();
        System.out.println("Average of positive numbers is : "+avg);
    }
}
```

Output

```
ajc@36:~/Desktop/S2/Java/Exception$ javac avgNExc.java
ajc@36:~/Desktop/S2/Java/Exception$ java avgNExc
Enter the limit : 4
Enter the elements :
10
-8
negativeInput: Negative number cannot be calculated!
80
-1
negativeInput: Negative number cannot be calculated!
5
3
Array is : 10 80 5 3
Average of positive numbers is : 24.0
```

OBJECT ORIENTED PROGRAMMING LAB

Name: SHAMJAD MAZOOD NAZER

Roll No: 36

Batch: B

Date: 01/06/2022

Experiment No.: 20

Aim

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

Procedure

```
import java.util.Scanner;
class Fibonacci extends Thread{
    int size;
    Fibonacci(int size){
            this.size=size;
    }
    public void run(){
            int num1=0, num2=1;
            System.out.println("Fibonacci series : 0, 1, ");
            for(int i=2;i<size;i++){
                    int temp=num1;
                    num1= num2;
                    num2= temp+num1;
                    System.out.println("Fibonacci series : "+num2+", ");
            }
    }
}
class EvenNumber extends Thread{
    int range;
    EvenNumber(int range){
    this.range= range;
    }
```

```
    public void run(){
            for(int i=0;i<range;i++){
                    if(i%2==0){
                            System.out.println("Even number are  : "+i);
                    }
            }
    }
}
public class EvenFiboThread{
    public static void main(String[] args) {
            int size, range;
            Scanner sc= new Scanner(System.in);
            System.out.print("Enter the limit of fibonacci series : ");
            size= sc.nextInt();
            System.out.println("Enter the range of even numbers : ");
            range= sc.nextInt();
            Fibonacci fib= new Fibonacci(size);
            EvenNumber even= new EvenNumber(range);
            fib.start();
            even.start();
            sc.close();
    }
}
```

Output

```
ajc@36:~/Desktop/S2/Java/Threads$ javac EvenFiboThread.java
ajc@36:~/Desktop/S2/Java/Threads$ java EvenFiboThread
Enter the limit of fibonacci series : 5
Enter the range of even numbers :
5
Fibonacci series : 0, 1,
Even number are  : 0
Even number are  : 2
Even number are  : 4
Fibonacci series : 1,
Fibonacci series : 2,
Fibonacci series : 3,
ajc@36:~/Desktop/S2/Java/Threads$
```

OBJECT ORIENTED PROGRAMMING LAB

Name: SHAMJAD MAZOOD NAZER

Roll No: 36

Batch: B

Date: 07/06/2022

Experiment No.: 21

Aim

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

Procedure

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

public class StackOpr
{
   public static void main(String args[])
   {
            int i, n, e;
            Scanner sc = new Scanner(System.in);
            System.out.print("Enter the Limit : ");
            n = sc.nextInt();
            Stack<Integer> s = new Stack<>();
            boolean result = s.empty();
            System.out.println("Is the Stack Empty? : "+result);
            System.out.print("Enter "+n+" elements : ");
            for(i=0; i<n; i++)
            {
                     e = sc.nextInt();
                     s.push(e);
            }
            System.out.print("Elements in the Stack : ");
            System.out.print(s+" ");
            System.out.println("\nPOP last element : "+s.pop());
```

```
            result = s.empty();
            System.out.print("Now the Elements in the Stack : ");
            System.out.print(s+" ");
            System.out.println("\nIs the Stack Empty? : "+result);
    }
}
```

Output

```
ajc@36:~/Desktop/S2/Java/07-06-2022$ javac StackOpr.java
ajc@36:~/Desktop/S2/Java/07-06-2022$ java StackOpr
Enter the Limit : 5
Is the Stack Empty? : true
Enter 5 elements : 1 2 3 4 5
Elements in the Stack : [1, 2, 3, 4, 5]
POP last element : 5
Now the Elements in the Stack : [1, 2, 3, 4]
Is the Stack Empty? : false
ajc@36:~/Desktop/S2/Java/07-06-2022$
```

OBJECT ORIENTED PROGRAMMING LAB

Name: SHAMJAD MAZOOD NAZER

Roll No: 36

Batch: B

Date: 07/06/2022

Experiment No.: 22

Aim

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

Procedure

```
import java.util.*;
class ArrayListFramework
{
   public static void main(String args[])
   {
            String name, del, check;
            char clean;
            int i, n;
            Scanner sc = new Scanner(System.in);
            ArrayList<String> arrList = new ArrayList<String>();
            System.out.print("Enter the limit : ");
            n = sc.nextInt();
            System.out.println("Enter the names :");
            for(i=0; i<n; i++)
            {
                  name = sc.next();
                  arrList.add(name);
            }
            System.out.println("\nElements on the Array List are :");
                  System.out.println(arrList);

            Collections.sort(arrList);
            System.out.println("\nSort using Collection Package is :");
```

```
                System.out.println(arrList);
            System.out.print("\nEnter the name you wish to remove : ");
            del = sc.next();
                arrList.remove(del);
            System.out.println("Updated Array List by removing of "+del+" is : "+arrList);
            System.out.print("\nEnter a name you wish to search : ");
                check = sc.next();

            if(arrList.contains(check))
            {
                System.out.println(check+" is present in the Array List.");
            }
            else
            {
                System.out.println(check+" is not present in the Array List.");
            }

            System.out.print("\nClear the arrayList ? [y/N] : ");
            clean = sc.next().charAt(0);
            if(clean == 'y')
            {
                System.out.println("Clearing the Array List!...");
                        arrList.clear();
                System.out.print("Now Array List is : "+arrList);
                System.out.println();
            }
    }
}
```

Output

```
ajc@36:~/Desktop/S2/Java/07-06-2022$ javac ArrayListFramework.java
ajc@36:~/Desktop/S2/Java/07-06-2022$ java ArrayListFramework
Enter the limit : 4
Enter the names :
SHAMJAD
JADSPRIT
DQ
JAD

Elements on the Array List are :
[SHAMJAD, JADSPRIT, DQ, JAD]

Sort using Collection Package is :
[DQ, JAD, JADSPRIT, SHAMJAD]

Enter the name you wish to remove : DQ
Updated Array List by removing of DQ is : [JAD, JADSPRIT, SHAMJAD]

Enter a name you wish to search : JADSPRIT
JADSPRIT is present in the Array List.

Clear the arrayList ? [y/N] : y
Clearing the Array List!...
Now Array List is : []
ajc@36:~/Desktop/S2/Java/07-06-2022$
```

OBJECT ORIENTED PROGRAMMING LAB

Name: SHAMJAD MAZOOD NAZER
Roll No: 36
Batch: B
Date: 07/06/2022

Experiment No.: 23

Aim

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

Procedure

```
import java.util.*;
class PriorityQueue1{
    public static void main(String args[]){
            PriorityQueue<String> queue = new PriorityQueue<String>();
            queue.add("Shamjad");
            queue.add("Jadsprit");
            queue.add("Jad");
            System.out.println("Head : "+queue.element());
            System.out.println("Peek : "+queue.peek());
            System.out.print("Iterating through Queue Elements : ");
            Iterator itr = queue.iterator();
            while(itr.hasNext()){
                    System.out.print(itr.next()+" ");
            }

            queue.remove();
            queue.poll();
            System.out.println("\nAfter removing 2 elements : ");
            Iterator<String> itr2 = queue.iterator();
            while(itr2.hasNext()){
                    System.out.println(itr2.next()+" ");
            }
```

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

Output

```
ajc@36:~/Desktop/S2/Java/07-06-2022$ javac PriorityQueue1.java
ajc@36:~/Desktop/S2/Java/07-06-2022$ java PriorityQueue1
Head : Jad
Peek : Jad
Iterating through Queue Elements : Jad Shamjad Jadsprit
After removing 2 elements :
Shamjad

ajc@36:~/Desktop/S2/Java/07-06-2022$
```

OBJECT ORIENTED PROGRAMMING LAB

Name: SHAMJAD MAZOOD NAZER

Roll No: 36

Batch: B

Date: 07/06/2022

Experiment No.: 24

Aim

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

Procedure

```
import java.util.*;
class DeQueue{
    public static void main(String[] args)
    {
            Deque<String> deque = new LinkedList<String>();
            deque.add("Element 1(Tail)");
            deque.addFirst("Element 2(Head)");
            deque.addLast("Element 3(Tail)");
            deque.push("Element 4(Head)");
            deque.offer("Element 5(Tail)");
            deque.offerFirst("Element 6(Head)");
            System.out.println(deque+"\n");
            deque.removeFirst();
            deque.removeLast();
            System.out.println("Dequeue after removing " + " first and last : " +deque);
    }
}
```

Output

```
ajc@36:~/Desktop/S2/Java/07-06-2022$ javac DeQueue.java
ajc@36:~/Desktop/S2/Java/07-06-2022$ java DeQueue
[Element 6(Head), Element 4(Head), Element 2(Head), Element 1(Tail), Element 3(Tail), Element 5(Tail)]

Dequeue after removing  first and last : [Element 4(Head), Element 2(Head), Element 1(Tail), Element 3(Tail)]
ajc@36:~/Desktop/S2/Java/07-06-2022$
```

OBJECT ORIENTED PROGRAMMING LAB

Name: SHAMJAD MAZOOD NAZER

Roll No: 36

Batch: B

Date: 07/06/2022

Experiment No.: 25

Aim

Write a Java program to compare two hash sets.

Procedure

```
import java.util.*;
class Compare2HashSet{
   public static void main(String[] args)
   {
            HashSet<String> hset1 = new HashSet<>();
            hset1.add("RED");
            hset1.add("GREY");
            hset1.add("BLACK");
            hset1.add("ORANGE");
            System.out.print("HASHSET 1 : "+hset1);

            HashSet<String> hset2 = new HashSet<String>();
            hset2.add("RED");
            hset2.add("BLUE");
            hset2.add("GREEN");
            hset2.add("ORANGE");
            System.out.println("\nHASHSET 2 : "+hset2);

            HashSet<String> result = new HashSet<String>();
            System.out.println("HASHSET COMAPRISON\n----------");
            for(String elements : hset1){
                  System.out.print(hset2.contains(elements) ? "[YES]\n" : "[NO]\n");
            }
```

```
    }
}
```

Output

```
ajc@36:~/Desktop/S2/Java/07-06-2022$ javac Compare2HashSet.java
ajc@36:~/Desktop/S2/Java/07-06-2022$ java Compare2HashSet
HASHSET 1 : [RED, BLACK, GREY, ORANGE]
HASHSET 2 : [RED, BLUE, GREEN, ORANGE]
HASHSET COMAPRISON
---------------
[YES]
[NO]
[NO]
[YES]
ajc@36:~/Desktop/S2/Java/07-06-2022$
```

OBJECT ORIENTED PROGRAMMING LAB

Name: SHAMJAD MAZOOD NAZER

Roll No: 36

Batch: B

Date: 07/06/2022

Experiment No.: 26

Aim

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

Procedure

```
import java.util.*;
class HashMapping
{
   public static void main(String args[]){
            Map<String, Integer> HM = new HashMap<String, Integer>();
            HM.put("SHAMJAD", new Integer(21));
            HM.put("JADSPRIT", new Integer(93));
            HM.put("JAD", new Integer(9));
            for(Map.Entry<String, Integer> ME : HM.entrySet()){
                     System.out.print(ME.getKey()+" : ");
                     System.out.println(ME.getValue());
            }
   }
}
```

Output

```
student@U36:~/Desktop/RMCA-B/OOP/07-06-2022$ java HashMapping
SHAMJAD : 21
JAD : 9
JADSPRIT : 93
student@U36:~/Desktop/RMCA-B/OOP/07-06-2022$
```

OBJECT ORIENTED PROGRAMMING LAB

Name: SHAMJAD MAZOOD NAZER

Roll No: 36

Batch: B

Date: 05/07/2022

Experiment No.: 27

Aim

Program to find the maximum of three numbers using AWT.

Procedure

```
import java.awt.*;
import java.awt.event.*;
public class AwtLarge3 implements ActionListener{
    Frame f=new Frame();
    Label l1=new Label("First Number : ");
    Label l2=new Label("Second Number : ");
    Label l3=new Label("Third Number : ");
    Label res=new Label("Which one will? : ");
    TextField t1=new TextField();
    TextField t2=new TextField();
    TextField t3=new TextField();
    Button b1=new Button("Compare!");
    AwtLarge3(){
            l1.setBounds(50,100,100,20);
            l2.setBounds(50,140,100,20);
            l3.setBounds(50,180,100,20);
            t1.setBounds(150,100,100,20);
            t2.setBounds(150,140,100,20);
            t3.setBounds(150,180,100,20);
            b1.setBounds(50,220,100,20);
            res.setBounds(50,260,100,20);
            f.add(l1);
            f.add(l2);
```

```
            f.add(l3);
            f.add(t1);
            f.add(t2);
            f.add(t3);
            f.add(res);
            f.add(b1);
            b1.addActionListener(this);
            f.setLayout(null);
            f.setVisible(true);
            f.setSize(400,400);
    }
    public static void main(String[] args){
            new AwtLarge3();
    }
    public void actionPerformed(ActionEvent e){
            if(e.getSource()==b1){
                    int n1=Integer.parseInt(t1.getText());
                    int n2=Integer.parseInt(t2.getText());
                    int n3=Integer.parseInt(t3.getText());
                    if(n1 > n2 && n1 > n3)
                            res.setText(String.valueOf("largest is : "+n1));
                    else if(n2 > n3)
                            res.setText(String.valueOf("largest is : "+n2));
                    else
                            res.setText(String.valueOf("largest is : "+n3));
            }
    }
}
```

Output

```
ajc@36:~/Desktop/S2/Java/AWT$ javac AwtLarge3.java
ajc@36:~/Desktop/S2/Java/AWT$ java AwtLarge3
```

OBJECT ORIENTED PROGRAMMING LAB

Name: SHAMJAD MAZOOD NAZER

Roll No: 36

Batch: B

Date: 05/07/2022

Experiment No.: 28

Aim

Implement a simple calculator using AWT components.

Procedure

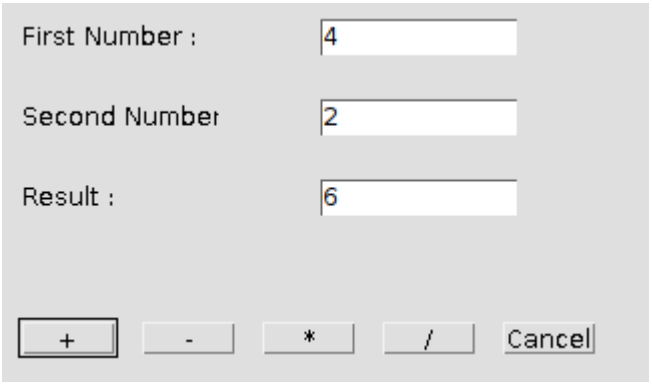
```
import java.awt.*;
import java.awt.event.*;
public class AwtCalc implements ActionListener{
    Frame f=new Frame();
    Label l1=new Label("First Number : ");
    Label l2=new Label("Second Number : ");
    Label l3=new Label("Result : ");
    TextField t1=new TextField();
    TextField t2=new TextField();
    TextField t3=new TextField();
    Button b1=new Button("+");
    Button b2=new Button("-");
    Button b3=new Button("*");
    Button b4=new Button("/");
    Button b5=new Button("Cancel");
    AwtCalc(){
            l1.setBounds(50,100,100,20);
            l2.setBounds(50,140,100,20);
            l3.setBounds(50,180,100,20);
            t1.setBounds(200,100,100,20);
            t2.setBounds(200,140,100,20);
            t3.setBounds(200,180,100,20);
            b1.setBounds(50,250,50,20);
```

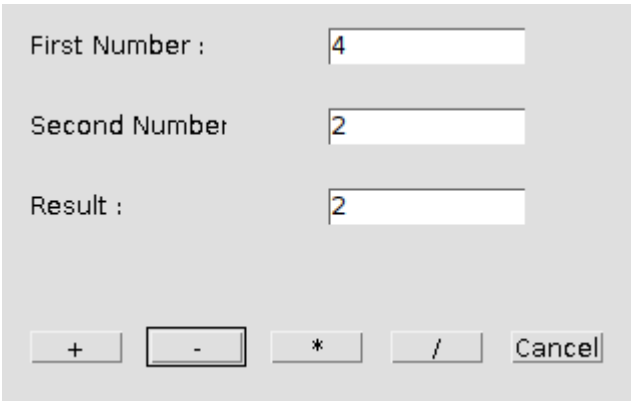
```
        b2.setBounds(110,250,50,20);
        b3.setBounds(170,250,50,20);
        b4.setBounds(230,250,50,20);
        b5.setBounds(290,250,50,20);
        f.add(l1);
        f.add(l2);
        f.add(l3);
        f.add(t1);
        f.add(t2);
        f.add(t3);
        f.add(b1);
        f.add(b2);
        f.add(b3);
        f.add(b4);
        f.add(b5);
        b1.addActionListener(this);
        b2.addActionListener(this);
        b3.addActionListener(this);
        b4.addActionListener(this);
        b5.addActionListener(this);
        f.setLayout(null);
        f.setVisible(true);
        f.setSize(400,350);
    }
    public void actionPerformed(ActionEvent e){
        int n1=Integer.parseInt(t1.getText());
        int n2=Integer.parseInt(t2.getText());
        if(e.getSource()==b1){
            t3.setText(String.valueOf(n1+n2));
        }
        if(e.getSource()==b2){
            t3.setText(String.valueOf(n1-n2));
```

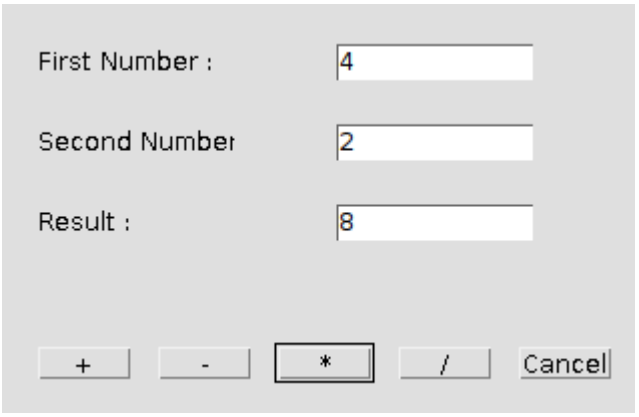
```
            }
            if(e.getSource()==b3){
                t3.setText(String.valueOf(n1*n2));
            }
            if(e.getSource()==b4){
                t3.setText(String.valueOf(n1/n2));
            }
            if(e.getSource()==b5){
                System.exit(0);
            }
    }
    public static void main(String[] args){
            new AwtCalc();
    }
}
```

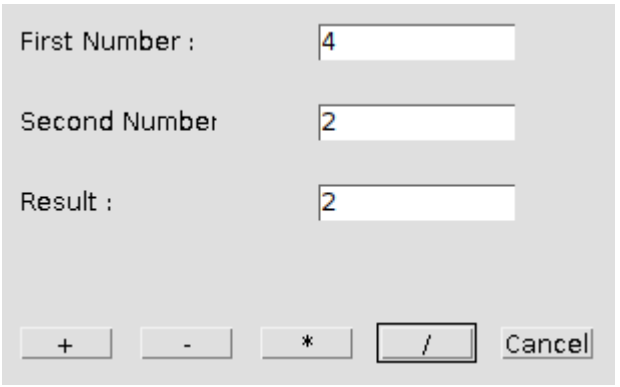
Output

```
ajc@36:~/Desktop/S2/Java/AWT$ javac AwtCalc.java
ajc@36:~/Desktop/S2/Java/AWT$ java AwtCalc
ajc@36:~/Desktop/S2/Java/AWT$
```




First Number :
4
Second Number
2
Result :
2
+
-
*
/
Cancel


First Number :
4
Second Number
2
Result :
8
+
-
*
/
Cancel


First Number :
4
Second Number
2
Result :
2
+
-
*
/
Cancel

OBJECT ORIENTED PROGRAMMING LAB

Name: SHAMJAD MAZOOD NAZER

Roll No: 36

Batch: B

Date: 05/07/2022

Experiment No.: 29

Aim

Develop a program to handle all mouse events and window events

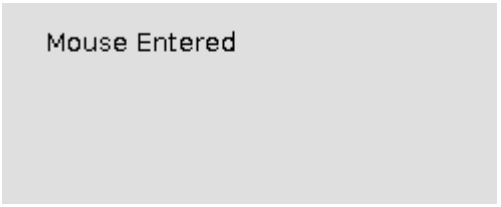
Procedure

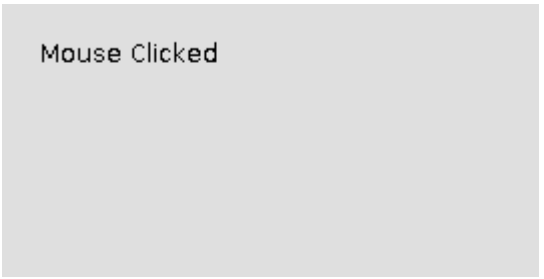
```
import java.awt.*;
import java.awt.event.*;
public class AwtMouseEvent extends Frame implements MouseListener{
    Label l;
    AwtMouseEvent(){
            addMouseListener(this);
            l=new Label();
            l.setBounds(20,50,100,20);
            add(l);
            setSize(300,300);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

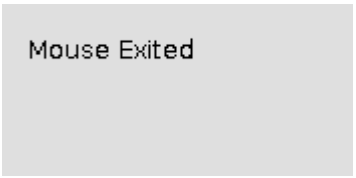
```
            l.setText("Mouse Pressed");
    }
    public void mouseReleased(MouseEvent e){
            l.setText("Mouse Released");
    }
    public static void main(String[] args){
            new AwtMouseEvent();
    }
}
```

Output

```
ajc@36:~/Desktop/S2/Java/AWT$ javac AwtMouseEvent.java
ajc@36:~/Desktop/S2/Java/AWT$ java AwtMouseEvent
```







OBJECT ORIENTED PROGRAMMING LAB

Name: SHAMJAD MAZOOD NAZER

Roll No: 36

Batch: B

Date: 05/07/2022

Experiment No.: 30

Aim

Develop a program to handle Key events.

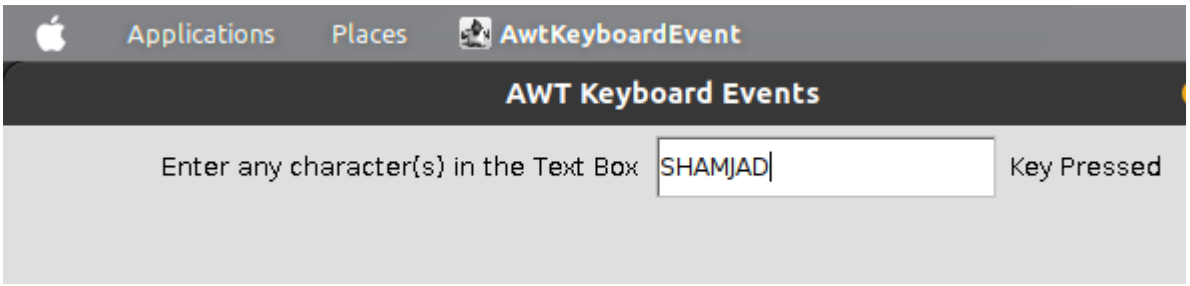
Procedure

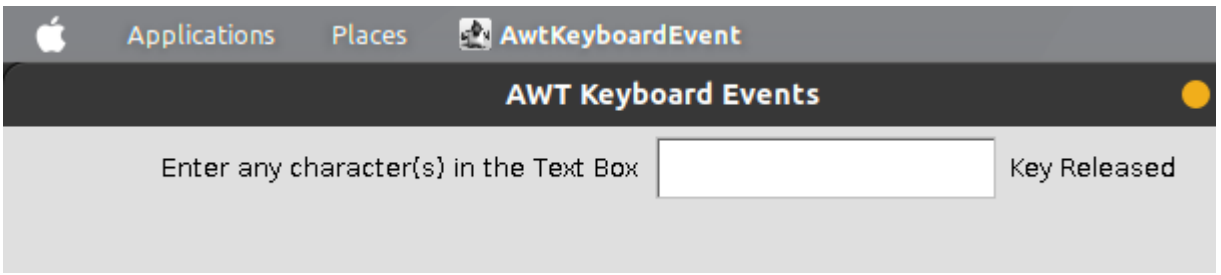
```
import java.awt.FlowLayout;
import java.awt.Frame;
import java.awt.Label;
import java.awt.TextField;
import java.awt.event.KeyEvent;
import java.awt.event.KeyListener;
public class AwtKeyboardEvent implements KeyListener{
    Label lb1, lbl2, lb;
    TextField tf1;
    Frame fr;
    String s;
    AwtKeyboardEvent(){
            fr = new Frame("AWT Keyboard Events");
            lb1= new Label("Enter any character(s) in the Text Box", Label.CENTER);
            lbl2= new Label();
            lb= new Label();
            tf1 = new TextField(20);
            fr.setLayout(new FlowLayout());
            fr.add(lb1);
            fr.add(tf1);
            fr.add(lbl2);
            tf1.addKeyListener(this);
            fr.setSize(660,450);
```

```
            fr.setVisible(true);
    }
    public void keyPressed(KeyEvent ev){
            lbl2.setText("Key Pressed");
    }
    public void keyReleased(KeyEvent ev){
            lbl2.setText("Key Released");
    }
    public void keyTyped(KeyEvent ev){
            lbl2.setText("Key is Typed");
            fr.setVisible(true);
    }
    public static void main(String[] args){
            new AwtKeyboardEvent();
    }
}
```

Output

```
ajc@36:~/Desktop/S2/Java/AWT$ javac AwtKeyboardEvent.java
ajc@36:~/Desktop/S2/Java/AWT$ java AwtKeyboardEvent
```





OBJECT ORIENTED PROGRAMMING LAB

Name: SHAMJAD MAZOOD NAZER

Roll No: 36

Batch: B

Date: 31/05/2022

Experiment No.: 31

Aim

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

Procedure

```
import java.io.FileReader;
import java.io.FileWriter;
import java.io.IOException;
import java.io.*;
import java.util.*;
import java.io.File;
class FileRead{
    public static void main(String[] args) {
            // initialize String
            String var = "";
            Scanner scan = new Scanner(System.in);
            System.out.println("Enter the text to create file : type ENTER key 3 times to stop");
            while (!var.endsWith("\n\n\n"))
            var = var + scan.nextLine() + "\n";
            try{
                    // create file object
                    File file = new File("output.txt");
                    // create filewriter object
                    FileWriter fw = new FileWriter(file);
                    fw.write(var);
                    fw.close();
                    System.out.println("Reading File content");
                    FileReader fr = new FileReader("output.txt");
```

```
                String str = "";
                int i;
                while ((i = fr.read()) != -1) {
                        // Storing every character in the string
                        str += (char) i;
                }
                System.out.println(str);
                fr.close();
            }catch (IOException e){
                System.out.println("There are some exception");
            }
    }
}
```

Output

```
ajc@36:~/Desktop/S2/Java/FILE$ javac FileRead.java
ajc@36:~/Desktop/S2/Java/FILE$ java FileRead
Enter the text to create file : type ENTER key 3 times to stop
Shamjad Mazood Nazer
OOP using JAVA
AJC21MCA-2095


Reading File content
Shamjad Mazood Nazer
OOP using JAVA
AJC21MCA-2095



ajc@36:~/Desktop/S2/Java/FILE$
```

OBJECT ORIENTED PROGRAMMING LAB

Name: SHAMJAD MAZOOD NAZER

Roll No: 36

Batch: B

Date: 31/05/2022

Experiment No.: 32

Aim

Write a Java program to write a program to copy one file to another.

Procedure

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

```
                File file=new File(destination);
                FileWriter fw = new FileWriter(file);
                fw.write(str);
                fr.close();
                fw.close();
                System.out.println("Copied from "+source+" to "+destination+ "
Successfully..!");
            }catch (Exception e){
                //TODO: handle exception
                System.out.println("Exception Occured");
            }
    }
}
```

Output

```
ajc@36:~/Desktop/S2/Java/FILE$ javac FileCopy.java
ajc@36:~/Desktop/S2/Java/FILE$ java FileCopy
Enter the source File Name
output.txt
Reading from file output.txt
Shamjad Mazood Nazer
OOP using JAVA
AJC21MCA-2095



 Enter the filename to copy
FileCopied.txt
Copied from output.txt to FileCopied.txt Successfully..!
ajc@36:~/Desktop/S2/Java/FILE$
```

OBJECT ORIENTED PROGRAMMING LAB

Name: SHAMJAD MAZOOD NAZER

Roll No: 36

Batch: B

Date: 31/05/2022

Experiment No.: 33

Aim

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

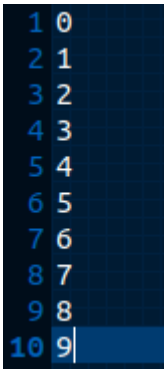
Procedure

```
import java.io.FileReader;
import java.io.FileWriter;
import java.io.IOException;
import java.io.*;
import java.util.*;
import java.io.File;
public class FileOddEven{
    public static void main(String[] args){
            try {
                    FileReader fr = new FileReader("FileNumbers.txt");
                    BufferedReader br = new BufferedReader(fr);
                    File file1 = new File("FileOdd.txt");
                    FileWriter fw1 = new FileWriter(file1);
                    File file2 = new File("FileEven.txt");
                    FileWriter fw2 = new FileWriter(file2);
                    String num;
                    while ((num = br.readLine()) != null){
                            if (Integer.parseInt(num) % 2 == 0){
                                    fw2.write(num + "\n");
                            }
                            else{
                                    fw1.write(num + "\n");
```

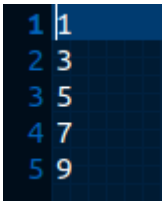
```
                    }
                }
                fw1.close();
                fw2.close();
            } catch (Exception e){
                // TODO: handle exception
                System.out.println("Error");
            }
    }
}
```

Output

FileNumbers.txt



FileOdd.txt



FileEven.txt