CHRIST
(DEEMED TO BE UNIVERSITY)
PUNE LAVASA CAMPUS
The Hub of Analytics

CHRIST (Deemed to be University) Pune Lavasa Campus

School of Sciences

Department of Data Science

MDS273 Java Programming

Lab Record

Karan Punjabi

22122140

May 2023

CONTENTS

Sl. No.	Title	Page No.
1.	Question 1	3
2.	Question 2	7
3.	Question 3	16
4.	Question 4	26
5.	Question 5	37
6.	Question 6	47
7.	Question 7	59

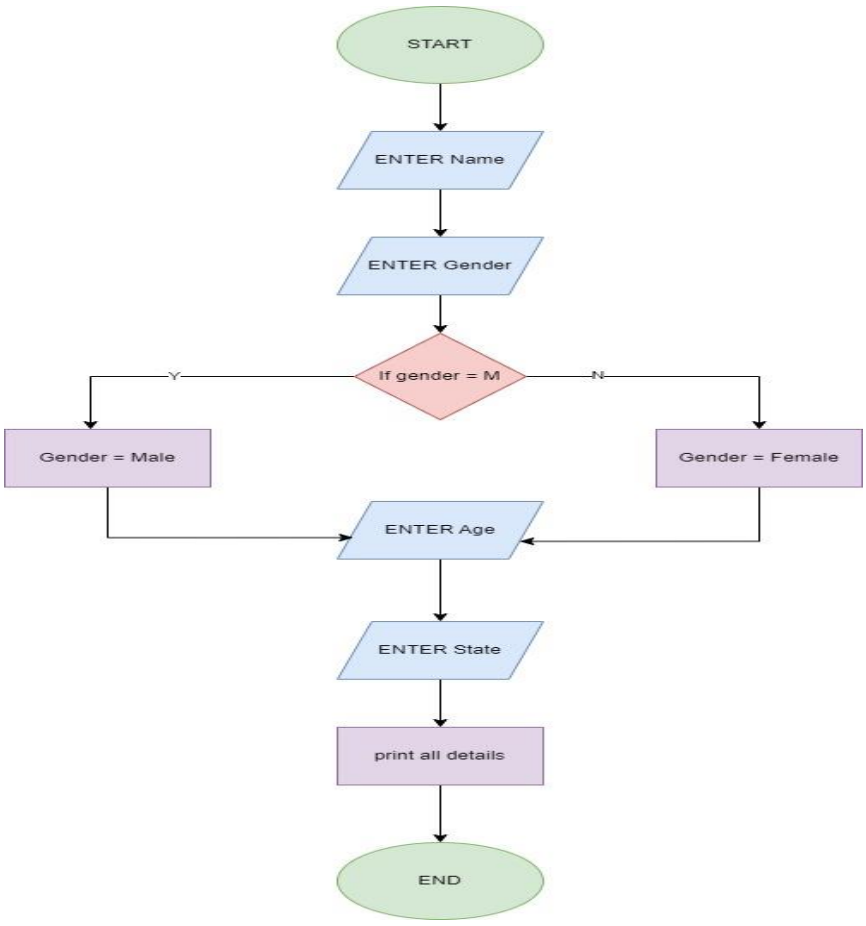
Question 1.

Write a Java Program that will collect your basic details that fall into different data types and displays them.

After the details have been displayed, with the help of conditional statements, check if the gender of the user is 'm' or 'f'. It should print "MALE" for 'm' and "FEMALE" for 'f'.

Assume that you can divide the states among India into four different regions (North, South, East, and West). If you are from the southern part of India; with the help of a switch statement, it should display The Student is from the southern states of India", along with the basic details.

Flow Chart



Solution

```
// package 22122140-MDS273L-JAVA.Lab1;
import java.util.*;

public class Lab1 {
    public static void main(String[] args) {
        Scanner scan = new Scanner(System.in);
        System.out.println("=========================================");
        System.out.println("Enter your details:");
        System.out.println("=========================================");
        System.out.print("Enter your First Name: ");
        String fname = scan.nextLine();
        System.out.print("Enter your Last Name: ");
        String lname = scan.nextLine();
        System.out.print("Enter your Gender (M|F): ");
        char gender = Character.toUpperCase(scan.nextLine().charAt(0));
        System.out.print("Enter your Age: ");
        int age = Integer.parseInt(scan.nextLine());
        System.out.print("Enter your State: ");
        String state = scan.nextLine().toLowerCase();
        System.out.println("=========================================");
        System.out.println("=========================================");
        System.out.println("Your Details are listed below:");
        System.out.println("=========================================");
        System.out.println("Full Name: " + fname + " " + lname);
        if (gender == 'M') {
            System.out.println("Gender: Male");
        } else if (gender == 'F') {
            System.out.println("Gender: Female");
        }
        System.out.println("Age: " + age);
        switch (state) {
            case "andhra pradesh":
            case "karnataka":
            case "kerala":
            case "tamil nadu":
            case "telangana":
                System.out.println("State name: " + state + "\nState Region:
South");
                break;
            case "jammu & kashmir":
            case "himachal pradesh":
            case "punjab":
```

```
            case "uttarakhand":
            case "haryana":
            case "delhi":
            case "rajasthan":
            case "uttar pradesh":
            case "chandigarh":
                System.out.println("State name: " + state + "\nState Region:
North");
                break;
            case "arunachal pradesh":
            case "assam":
            case "manipur":
            case "meghalaya":
            case "mizoram":
            case "nagaland":
            case "sikkim":
            case "tripura":
                System.out.println("State name: " + state + "\nState Region:
East");
                break;
            case "goa":
            case "gujarat":
            case "maharashtra":
                System.out.println("State name: " + state + "\nState Region:
West");
                break;
            case "madhya pradesh":
                System.out.println("State name: " + state + "\nState Region:
Central");
                break;
            default:
                System.out.println("State name: " + state + "\nState Region:
Cannot Recognize Region");
        }
        System.out.println("========================================");
        scan.close();
    }
}
```

Output

```
PS D:\Karan\CHRIST UNIVERSITY\SEM2\JAVA\22122140-MDS273L-JAVA\lab1> java Lab1
========================================
Enter your details:
========================================
Enter your First Name: Karan
Enter your Last Name: Punjabi
Enter your Gender (M|F): M
Enter your Age: 23
Enter your State: Gujarat
========================================
========================================
Your Details are listed below:
========================================
Full Name: Karan Punjabi
Gender: Male
Age: 23
State name: gujarat
State Region: West
========================================
PS D:\Karan\CHRIST UNIVERSITY\SEM2\JAVA\22122140-MDS273L-JAVA\lab1>
```

Question 2.

You are supposed to create a menu-driven program that has the following menu options:

Enter a name

Search for a name

Remove a name

Show all names

Note:

The menu-driven program has to be made with the help of a do-while loop and switch-case statements.

Constraints:

The names collected must be stored in an array with a max length of 1024.

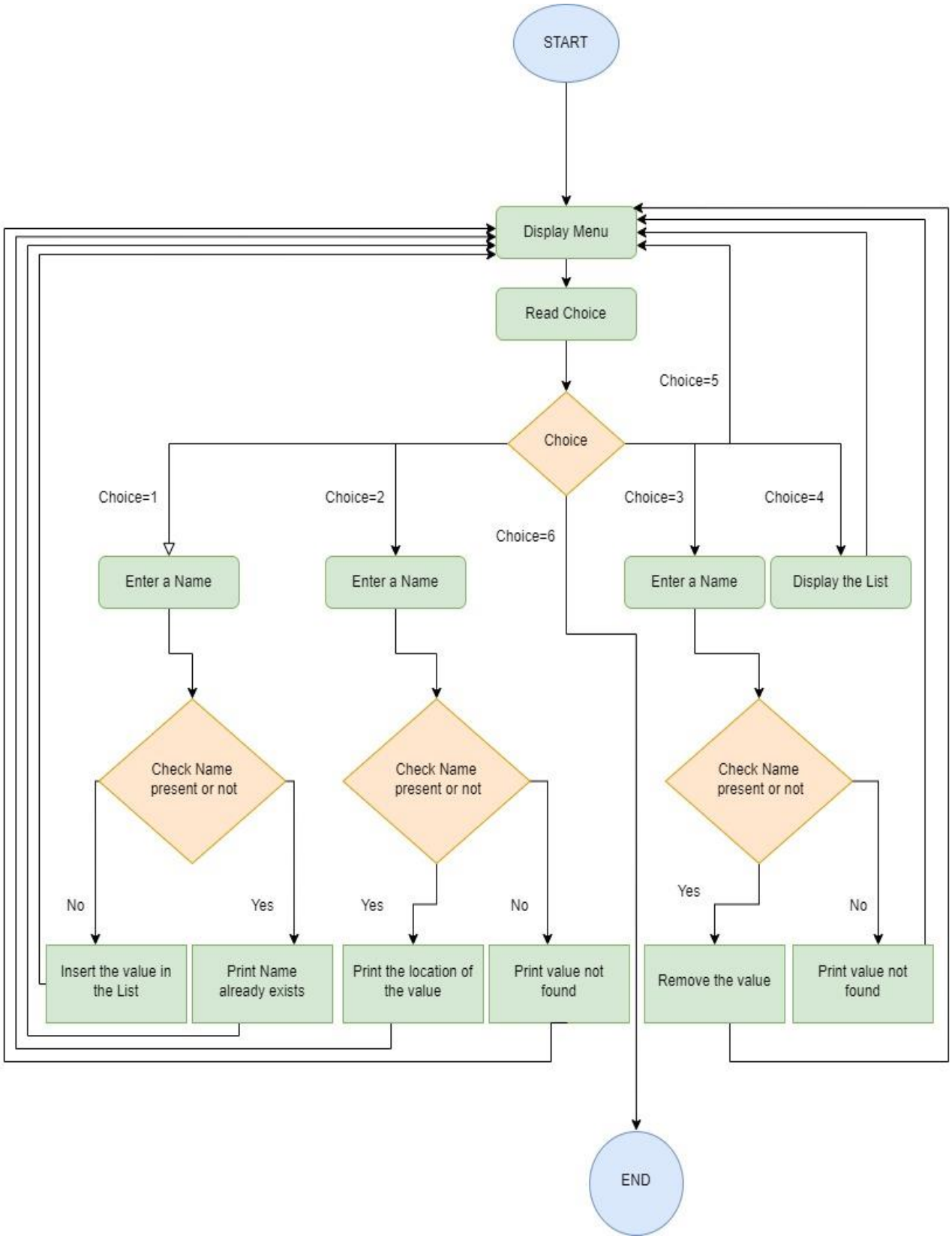
The names in the array should be UNIQUE; no duplicate entries are expected!

Provide necessary validations that the user enters only valid names that are not going to be repeated.

Removing a name should not create empty space inside the array!

Format your results properly!!

Flow Chart


START
Display Menu
Read Choice
Choice
Choice=1
Choice=2
Choice=3
Choice=4
Choice=5
Choice=6
Enter a Name
Enter a Name
Enter a Name
Display the List
Check Name present or not
Check Name present or not
Check Name present or not
No
Yes
Yes
No
Yes
No
Insert the value in the List
Print Name already exists
Print the location of the value
Print value not found
Remove the value
Print value not found
END

Solution

```
import java.util.List;
import java.util.Scanner;

public class Lab2 {
    public static void main(String[] args) {
        Scanner scan=new Scanner(System.in);
        String[] nameList=new String[1024];
        int choice;
        int index=0;
        do {
            System.out.println("============================================
==");
            System.out.println("1. Enter a name:");
            System.out.println("2. Search for a name:");
            System.out.println("3. Remove a name:");
            System.out.println("4. Show all names:");
            System.out.println("5. Continue:");
            System.out.println("6. Exit:");
            System.out.println("============================================
==");
            System.out.print("Please Enter your choice:");
            choice=Integer.parseInt(scan.nextLine());
            System.out.println("============================================
==");
            switch (choice) {
                case 1:
                    String name;
                    System.out.print("Enter a Name: ");
                    name=scan.nextLine();
                    name=name.toLowerCase();
                    int flag2=0;
                    for(int i=0;i<nameList.length;i++){
                        if(name.equals(nameList[i])){
                            System.out.println("============================
================");
                            System.out.println("Name Already Exists.");
                            System.out.println("============================
================");
                            flag2=1;
                            break;
                        }
                    }
                    if(flag2==0){
```

```
                        nameList[index]=name;
                        index++;
                    }
                    break;
                case 2:
                    String search;
                    System.out.println("==================================
==========");
                    System.out.print("Enter a Name you want to search: ");
                    search=scan.nextLine();
                    search=search.toLowerCase();
                    System.out.println("==================================
==========");
                    int flag1=0;
                    for(int i=0;i<nameList.length;i++){
                        if(search.equals(nameList[i])){
                            System.out.println("============================
==================");
                            System.out.println(search+" is present in
"+(i+1)+" place.");
                            System.out.println("============================
==================");
                            flag1=1;
                            break;
                        }
                    }
                    if(flag1==0){
                        System.out.println("================================
==============");
                        System.out.println("Sorry!, name "+search+" is not
present.");
                        System.out.println("================================
==============");
                    }
                    break;
                case 3:
                    String remove;
                    System.out.println("==================================
==========");
                    System.out.print("Enter a Name you want to Remove: ");
                    remove=scan.nextLine();
                    remove=remove.toLowerCase();
                    System.out.println("==================================
==========");
                    String[] newNameList=new String[nameList.length];
                    int flag=0;
```

```
                    for(int i=0,k=0;i<nameList.length;i++){
                        if(remove.equals(nameList[i])){
                            flag=1;
                            index--;
                        }
                        else{
                            newNameList[k]=nameList[i];
                            k++;
                        }
                    }
                    if(flag==0){
                        System.out.println("==================================
==============");
                        System.out.println("Sorry!, name "+remove+" is not
present.");
                        System.out.println("==================================
==============");
                    }
                    else{
                        nameList=newNameList;
                    }
                    break;
                case 4:
                    System.out.println("======================================
==========");
                    System.out.println("Values are as follows:");
                    System.out.println("======================================
==========");
                    for (String string : nameList) {
                        if(string==null){
                            break;
                        }
                        System.out.println(string);
                    }
                    System.out.println("======================================
==========");
                    break;
                case 5:
                    break;
                case 6:
                    break;
                default:
                    System.out.println("======================================
==========");
                    System.out.println("Please provide a valid choice!");
```

```
                    System.out.println("==================================
==========");
                    break;
                }
            } while (choice!=6);
        }
}
```

Output

```
================================================
1. Enter a name:
2. Search for a name:
3. Remove a name:
4. Show all names:
5. Continue:
6. Exit:
================================================
Please Enter your choice:1
================================================
Enter a Name: Karan
================================================
1. Enter a name:
2. Search for a name:
3. Remove a name:
4. Show all names:
5. Continue:
6. Exit:
================================================
Please Enter your choice:1
```

```
================================================
Enter a Name: Akshay
================================================
1. Enter a name:
2. Search for a name:
3. Remove a name:
4. Show all names:
5. Continue:
6. Exit:
================================================
Please Enter your choice:1
================================================
Enter a Name: Reena
================================================
1. Enter a name:
2. Search for a name:
3. Remove a name:
4. Show all names:
5. Continue:
6. Exit:
================================================
Please Enter your choice:1
================================================
Enter a Name: kaRAn
================================================
Name Already Exists.
================================================
```

```
================================================
1. Enter a name:
2. Search for a name:
3. Remove a name:
4. Show all names:
5. Continue:
6. Exit:
================================================
Please Enter your choice:4
================================================
================================================
Values are as follows:
================================================
karan
akshay
reena
================================================
```

```
================================================
1. Enter a name:
2. Search for a name:
3. Remove a name:
4. Show all names:
5. Continue:
6. Exit:
================================================
Please Enter your choice:2
================================================
================================================
Enter a Name you want to search: Reena
================================================
================================================
reena is present in 3 place.
================================================
```

```
================================================
1. Enter a name:
2. Search for a name:
3. Remove a name:
4. Show all names:
5. Continue:
6. Exit:
================================================
Please Enter your choice:3
================================================
================================================
Enter a Name you want to Remove: Akshay
================================================
```

```
================================================
1. Enter a name:
2. Search for a name:
3. Remove a name:
4. Show all names:
5. Continue:
6. Exit:
================================================
Please Enter your choice:4
================================================
================================================
Values are as follows:
================================================
karan
reena
================================================
```

```
================================================
1. Enter a name:
2. Search for a name:
3. Remove a name:
4. Show all names:
5. Continue:
6. Exit:
================================================
Please Enter your choice:1
================================================
Enter a Name: Anujit
================================================
1. Enter a name:
2. Search for a name:
3. Remove a name:
4. Show all names:
5. Continue:
6. Exit:
================================================
Please Enter your choice:4
================================================
================================================
Values are as follows:
================================================
karan
reena
anujit
================================================
```

```
================================================
1. Enter a name:
2. Search for a name:
3. Remove a name:
4. Show all names:
5. Continue:
6. Exit:
================================================
Please Enter your choice:2
================================================
================================================
Enter a Name you want to search: Akshay
================================================
================================================
Sorry!, name akshay is not present.
================================================
```

Question 3.

Create a java program that performs the following operation:

The program should collect an integer array from the user

After the array is entered, you need to create a menu of items

Find the Mean Value

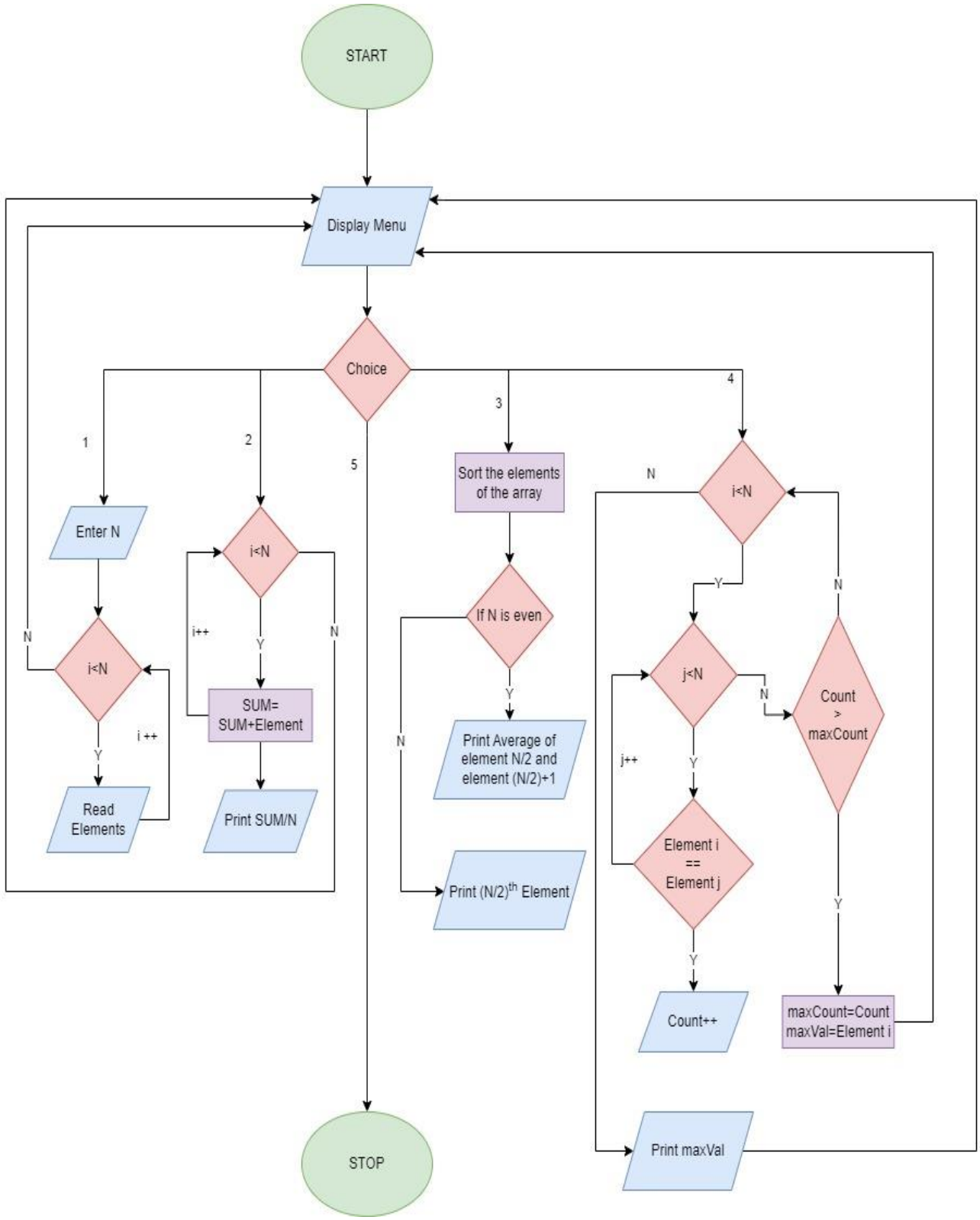
Find the Median Value

Find the Mode Value

From the user-given array.

The program should have a minimum of 3 functions apart from the main function; each of the functions implements the Mean, Median, and Mode computation by accepting the array user has entered as a parameter and returning the value as a result. From the main method, you will display the result.

Flow Chart


START
Display Menu
Choice
1
2
3
4
5
Enter N
i<N
Y
N
i ++
Read Elements
i<N
i++
Y
N
SUM= SUM+Element
Print SUM/N
Sort the elements of the array
If N is even
Y
N
Print Average of element N/2 and element (N/2)+1
Print (N/2)th Element
N
i<N
Y
N
j<N
j++
Y
N
Element i == Element j
Y
Count++
Count > maxCount
Y
maxCount=Count
maxVal=Element i
Print maxVal
STOP

Solution

```java
// importing the java.util.Scanner package to use Scanner class
import java.util.Scanner;

public class Lab3 {
    static int n = 0;

    // Creating a static function to find mean of the array.
    static float findMean(float[] arr) {
        float sum = 0;
        for (int i = 0; i < n; i++) {
            sum += arr[i];
        }
        return sum / n;
    }

    // Creating a static function to find median of the array.
    static float findMedian(float[] arr) {
        // Sorting the array in ascending order.
        for (int i = 0; i < n; i++) {
            float temp = 0;
            for (int j = 0; j < n; j++) {
                if (arr[i] < arr[j]) {
                    temp = arr[i];
                    arr[i] = arr[j];
                    arr[j] = temp;
                }
            }
        }
        // location variable
        int loc = 0;
        // if there are even number of observations
        if (n % 2 == 0) {
            loc = n / 2;
            return (arr[loc - 1] + arr[loc]) / 2;
        }
        // if there are odd number of observations
        else {
            loc = n / 2;
            return arr[loc];
        }
    }
```

```
    // Creating a static function to find mode of the array.
    static float findMode(float[] arr) {
        // maxVal will store our mode.
        float maxVal = 0;
        // maxCount will store the maximum count of each element.
        int maxCount = 0;
        for (int i = 0; i < n; i++) {
            int count = 0;
            for (int j = 0; j < n; j++) {
                if (arr[i] == arr[j]) {
                    count++;
                }
            }
            if (count > maxCount) {
                maxCount = count;
                maxVal = arr[i];
            }
        }
        return maxVal;
    }

    public static void main(String[] args) {
        Scanner scan = new Scanner(System.in);
        int choice = 0;
        System.out.println("================================================");
        System.out.println("-----------------WELCOME!--------------------");
        System.out.println("================================================");
        float[] arr = new float[1024];
        // Displaying Menu.
        do {
            System.out.println("================================================"
);
            System.out.println("Please choose from the following:");
            System.out.println("1) Enter the values:");
            System.out.println("2) Find Mean:");
            System.out.println("3) Find Median");
            System.out.println("4) Find Mode");
            System.out.println("5) Exit:");
            System.out.println("================================================"
);
            System.out.print("Please enter your choice:");
            // Reading the choice of the user.
            choice = Integer.parseInt(scan.nextLine());
            System.out.println("================================================"
);
            switch (choice) {
```

```
                case 1:
                    arr = new float[1024];
                    System.out.print("How many values you want to enter?:");
                    // Reading the value of n.
                    n = Integer.parseInt(scan.nextLine());
                    // Reading and Storing the values of array.
                    for (int i = 0; i < n; i++) {
                        System.out.print("Enter value " + (i + 1) + ":");
                        arr[i] = Float.parseFloat(scan.nextLine());
                    }
                    break;
                case 2:
                    // Calling findMean function to calculate the mean of the
values.
                    float mean = findMean(arr);
                    System.out.println("==================================
=======");
                    System.out.println("Mean of the values is " + mean);
                    System.out.println("==================================
=======");
                    break;
                case 3:
                    // Calling findMedian function to calculate the mean of
the values.
                    float median = findMedian(arr);
                    System.out.println("==================================
=======");
                    System.out.println("Median of the values is " + median);
                    System.out.println("==================================
=======");
                    break;
                case 4:
                    // Calling findMode function to calculate the mean of the
values.
                    float mode = findMode(arr);
                    System.out.println("==================================
=======");
                    System.out.println("Mode of the values is " + mode);
                    System.out.println("==================================
=======");
                    break;
                case 5:
                    System.out.println("==================================
=======");
                    System.out.println("----------------Thank You!----------
-------");
```

```
                    System.out.println("=================================
=======");
                    break;
                default:
                    // For any other choices other than 1,2,3,4,5
                    System.out.println("=================================
=======");
                    System.out.println("Please enter a valid choice!!:");
                    System.out.println("=================================
=======");
                    break;
            }
        } while (choice != 5);
        // Closing the scanner object
        scan.close();
    }
}
```

Output

```
================================================
----------------WELCOME!--------------------
================================================
================================================
Please choose from the following:
1) Enter the values:
2) Find Mean:
3) Find Median
4) Find Mode
5) Exit:
================================================
Please enter your choice:1
================================================
How many values you want to enter?:5
Enter value 1:140
Enter value 2:26.01
Enter value 3:1
Enter value 4:44
Enter value 5:1
================================================
Please choose from the following:
1) Enter the values:
2) Find Mean:
3) Find Median
4) Find Mode
5) Exit:
================================================
```

```
================================================
Please enter your choice:2
================================================
================================================
Mean of the values is 42.402
================================================
================================================
Please choose from the following:
1) Enter the values:
2) Find Mean:
3) Find Median
4) Find Mode
5) Exit:
================================================
```

```
==========================================
Please enter your choice:3
==========================================
==========================================
Median of the values is 26.01
==========================================
==========================================
Please choose from the following:
1) Enter the values:
2) Find Mean:
3) Find Median
4) Find Mode
5) Exit:
==========================================
Please enter your choice:4
==========================================
==========================================
Mode of the values is 1.0
==========================================
```

```
==========================================
Please choose from the following:
1) Enter the values:
2) Find Mean:
3) Find Median
4) Find Mode
5) Exit:
==========================================
Please enter your choice:1
==========================================
How many values you want to enter?:6
Enter value 1:1
Enter value 2:2
Enter value 3:3
Enter value 4:11
Enter value 5:11
Enter value 6:1
==========================================
```

```
==============================================
Please choose from the following:
1) Enter the values:
2) Find Mean:
3) Find Median
4) Find Mode
5) Exit:
==============================================
Please enter your choice:3
==============================================
==============================================
Median of the values is 2.5
==============================================
```

```
==============================================
Please choose from the following:
1) Enter the values:
2) Find Mean:
3) Find Median
4) Find Mode
5) Exit:
==============================================
Please enter your choice:4
==============================================
==============================================
Mode of the values is 1.0
==============================================
```

```
==============================================
Please choose from the following:
1) Enter the values:
2) Find Mean:
3) Find Median
4) Find Mode
5) Exit:
==============================================
Please enter your choice:2
==============================================
==============================================
Mean of the values is 4.8333335
==============================================
```

```
==============================================
Please choose from the following:
1) Enter the values:
2) Find Mean:
3) Find Median
4) Find Mode
5) Exit:
==============================================
Please enter your choice:5
==============================================
==============================================
----------------Thank You!--------------------
==============================================
```

Question 4.

Write a JAVA Menu driven program that does the following:

You can ONLY have the below variables as global variables

Account Number

Account Holder Name

Account Balance

You MUST have the below as functions

To initialize the customer

To deposit money

To withdraw money

To print the transactions

To print account summary

Your menu will have the following operations once the customer is created

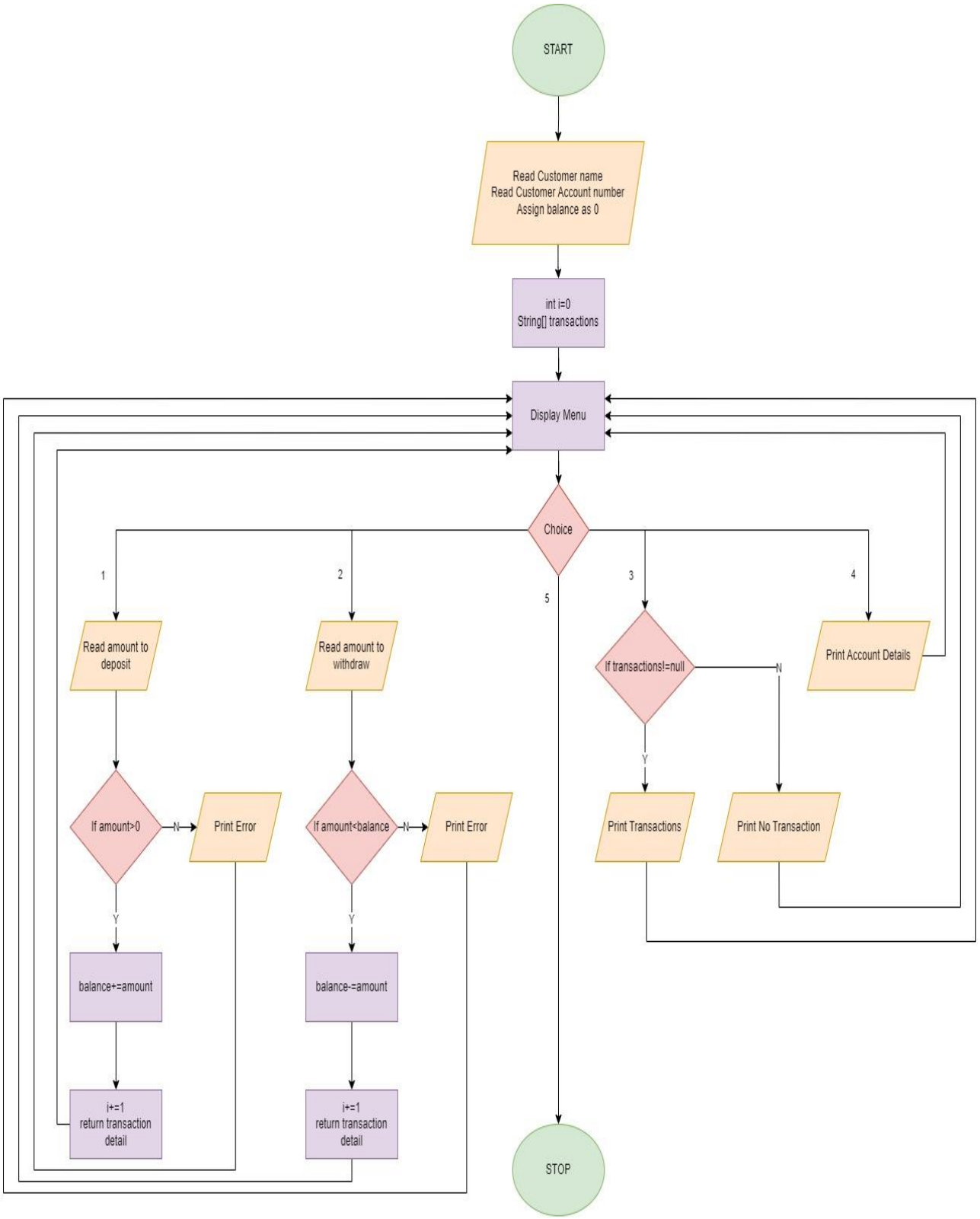
To deposit money

To withdraw money

To print the transactions

To print account summary

Flow Chart


START
Read Customer name
Read Customer Account number
Assign balance as 0
int i=0
String[] transactions
Display Menu
Choice
1
2
3
4
5
Read amount to deposit
Read amount to withdraw
If transactions!=null
Print Account Details
If amount>0
Print Error
If amount<balance
Print Error
Print Transactions
Print No Transaction
N
Y
balance+=amount
balance-=amount
i+=1
return transaction detail
i+=1
return transaction detail
STOP

Solution

```
// Importing the necessary classes. Scanner class for reading the inputs, Date
class to create an date object, Timestamp class to get the timestamp of the
transactions.
import java.util.Scanner;
import java.util.Date;
import java.sql.Timestamp;

public class Lab4 {
    // Declaring all the necessary global variables.
    static Scanner scan = new Scanner(System.in);
    static String accNo;
    static String holderName;
    static double balance = 0.0;

    // This function will be called only once in the programme at the
beginning.
    static void initCustomer() {
        System.out.println("==============================================
================================");
        System.out.print("Enter the Name of the Customer:");
        holderName = scan.nextLine();
        System.out.print("Enter the Account Number of the Customer:");
        accNo = scan.nextLine();
        balance = 0.0;
        System.out.println("==============================================
================================");
    }

    // This function will add the deposited amount to the balance and it will
return
    // a String with all the transaction details.
    static String deposit() {
        System.out.println("==============================================
================================");
        System.out.print("Enter the amount you want to deposit:");
        // Type casting the input string to double value.
        double deposit = Double.parseDouble(scan.nextLine());
        System.out.println("==============================================
================================");
        Date date = new Date();
        // date.getTime() will give you the current timestamp.
        Timestamp ts = new Timestamp(date.getTime());
```

```
        String str = "";
        if (deposit > 0.0) {
            // Updating our string variable so that we can return the
transaction details.
            str += "DateTime:" + ts.toString() + " ";
            str += "Previous Balance:" + balance + " ";
            balance += deposit;
            str += "Updated Balance: " + balance + " ";
            str += "Type: Deposit ";
            str += "Amount: " + deposit + " ";
            str += "Status: Success";
            System.out.println("==========================================
==================================");
            System.out.println("Deposit Successfull");
            System.out.println("Updated Balance: " + balance);
            System.out.println("==========================================
==================================");
            return str;
        } else {
            System.out.println("==========================================
==================================");
            System.out.println("Error! Please enter a valid amount.");
            System.out.println("==========================================
==================================");
            return "";
        }
    }

    // This function will deduct the withdrawn amount from the balance and it
will
    // return a String with all the transaction details.
    static String withdraw() {
        System.out.println("==========================================
==================================");
        System.out.print("Enter the amount you want to withdraw:");
        // Type casting the input string to double value.
        double withdraw = Double.parseDouble(scan.nextLine());
        System.out.println("==========================================
==================================");
        Date date = new Date();
        Timestamp ts = new Timestamp(date.getTime());
        String str = "";
        if (withdraw <= balance) {
            // Updating our string variable so that we can return the
transaction details.
            str += "DateTime:" + ts.toString() + " ";
```

```
            str += "Previous Balance:" + balance + " ";
            balance -= withdraw;
            str += "Updated Balance:" + balance + " ";
            str += "Type: Withdraw ";
            str += "Amount: " + withdraw + " ";
            str += "Status: Success";
            System.out.println("==========================================
==================================");
            System.out.println("Withdraw Successfull");
            System.out.println("Updated Balance: " + balance);
            System.out.println("==========================================
==================================");
            return str;
        } else {
            System.out.println("==========================================
==================================");
            System.out.println("Error! Insufficient Balance.");
            System.out.println("==========================================
==================================");
            return "";
        }
    }
    // This function will only print all the valid transaction details of the
account and it won't return anything
    static void printTransactions(String[] transactions) {
        System.out.println("==========================================
==============================");
        for (int i = 0; i < transactions.length; i++) {
            if (transactions[0] == null) {
                System.out.println("==========================================
======================================");
                System.out.println("No Transactions to print.");
                System.out.println("==========================================
======================================");
                break;
            } else if (transactions[i] == null) {
                break;
            } else if (transactions[i].equalsIgnoreCase("")) {
                continue;
            } else {
                System.out.println(transactions[i]);
            }
        }
        System.out.println("==========================================
==============================");
    }
```

```
    static void printSummary() {
        System.out.println("================================================
==============================");
        System.out.println("Account Holder Name: " + holderName);
        System.out.println("Account Number: " + accNo);
        System.out.println("Total Balance: " + balance);
        System.out.println("================================================
==============================");
    }

    public static void main(String[] args) {
        String[] transactions = new String[1024];
        initCustomer();
        int choice = 0;
        int i = 0;
        do {
            System.out.println("============================================
==================================");
            System.out.println("1) Deposit Money.");
            System.out.println("2) Withdraw Money.");
            System.out.println("3) Print Transactions.");
            System.out.println("4) Print Account Summary.");
            System.out.println("5) Exit.");
            System.out.println("============================================
==================================");
            System.out.print("Please Enter your choice:");
            choice = Integer.parseInt(scan.nextLine());
            System.out.println("============================================
==================================");
            switch (choice) {
                case 1:
                    transactions[i] = deposit();
                    i += 1;
                    break;
                case 2:
                    transactions[i] = withdraw();
                    i += 1;
                    break;
                case 3:
                    printTransactions(transactions);
                    break;
                case 4:
                    printSummary();
                    break;
                case 5:
```

```
                    break;
                default:
                    System.out.println(
                            "================================================
================================");
                    System.out.println("Please enter a valid choice.");
                    System.out.println(
                            "================================================
================================");
                    break;
            }
        } while (choice != 5);
    }
}
```

Output

```
================================================================================
Enter the Name of the Customer:Karan Punjabi
Enter the Account Number of the Customer:22122140
================================================================================
================================================================================
1) Deposit Money.
2) Withdraw Money.
3) Print Transactions.
4) Print Account Summary.
5) Exit.
================================================================================
Please Enter your choice:3
================================================================================
================================================================================
================================================================================
No Transactions to print.
================================================================================
================================================================================
================================================================================
1) Deposit Money.
2) Withdraw Money.
3) Print Transactions.
4) Print Account Summary.
5) Exit.
================================================================================
```

```
================================================================================
Please Enter your choice:3
================================================================================
================================================================================
================================================================================
No Transactions to print.
================================================================================
================================================================================
```

```
================================================================================
1) Deposit Money.
2) Withdraw Money.
3) Print Transactions.
4) Print Account Summary.
5) Exit.
================================================================================
Please Enter your choice:1
================================================================================
================================================================================
Enter the amount you want to deposit:1000
================================================================================
================================================================================
Deposit Successfull
Updated Balance: 1000.0
================================================================================
================================================================================
1) Deposit Money.
2) Withdraw Money.
3) Print Transactions.
4) Print Account Summary.
5) Exit.
================================================================================
```

```
================================================================================
Please Enter your choice:3
================================================================================
================================================================================
DateTime:2023-02-22 12:25:39.596 Previous Balance:0.0 Updated Balance: 1000.0 Type: Deposit Amount: 1000.0 Status: Success
================================================================================
================================================================================
1) Deposit Money.
2) Withdraw Money.
3) Print Transactions.
4) Print Account Summary.
5) Exit.
================================================================================
Please Enter your choice:2
================================================================================
================================================================================
Enter the amount you want to withdraw:2000
================================================================================
================================================================================
Error! Insufficient Balance.
================================================================================
```

```
================================================================================
1) Deposit Money.
2) Withdraw Money.
3) Print Transactions.
4) Print Account Summary.
5) Exit.
================================================================================
Please Enter your choice:2
================================================================================
================================================================================
Enter the amount you want to withdraw:699
================================================================================
================================================================================
Withdraw Successfull
Updated Balance: 301.0
================================================================================
```

```
================================================================================
1) Deposit Money.
2) Withdraw Money.
3) Print Transactions.
4) Print Account Summary.
5) Exit.
================================================================================
Please Enter your choice:3
================================================================================
================================================================================
DateTime:2023-02-22 12:25:39.596 Previous Balance:0.0 Updated Balance: 1000.0 Type: Deposit Amount: 1000.0 Status: Success
DateTime:2023-02-22 12:26:15.119 Previous Balance:1000.0 Updated Balance:301.0 Type: Withdraw Amount: 699.0 Status: Success====
================================================================================
```

```
================================================================================
1) Deposit Money.
2) Withdraw Money.
3) Print Transactions.
4) Print Account Summary.
5) Exit.
================================================================================
Please Enter your choice:1
================================================================================
================================================================================
Enter the amount you want to deposit:140
================================================================================
================================================================================
Deposit Successfull
Updated Balance: 441.0
================================================================================
================================================================================
1) Deposit Money.
2) Withdraw Money.
3) Print Transactions.
4) Print Account Summary.
5) Exit.
================================================================================
```

```
================================================================================
Please Enter your choice:3
================================================================================
================================================================================
DateTime:2023-02-22 12:25:39.596 Previous Balance:0.0 Updated Balance: 1000.0 Type: Deposit Amount: 1000.0 Status: Success
DateTime:2023-02-22 12:26:15.119 Previous Balance:1000.0 Updated Balance:301.0 Type: Withdraw Amount: 699.0 Status: SuccessDate
================================================================================
================================================================================
1) Deposit Money.
2) Withdraw Money.
3) Print Transactions.
4) Print Account Summary.
5) Exit.
================================================================================
Please Enter your choice:4
================================================================================
================================================================================
Account Holder Name: Karan Punjabi
Account Number: 22122140
Total Balance: 441.0
================================================================================
```

Question 5.

Create a student class, that will store the details of the Student

Reg. No.

Name

Email

Phone

Class

Department

The class will have a constructor to initialize the values of the Student and a method to print the Details of the Student.

In the main-method class, create an array of Student Class to hold maximum details of 100 Students.

In the menu-driven program, the menu options will have

Add a student

Adds the details of 1 student to the array of Student

Search for a student

Search for the details of a student from the array of Student

(Optional) Searching can be done with Name & Register Number

Display all students

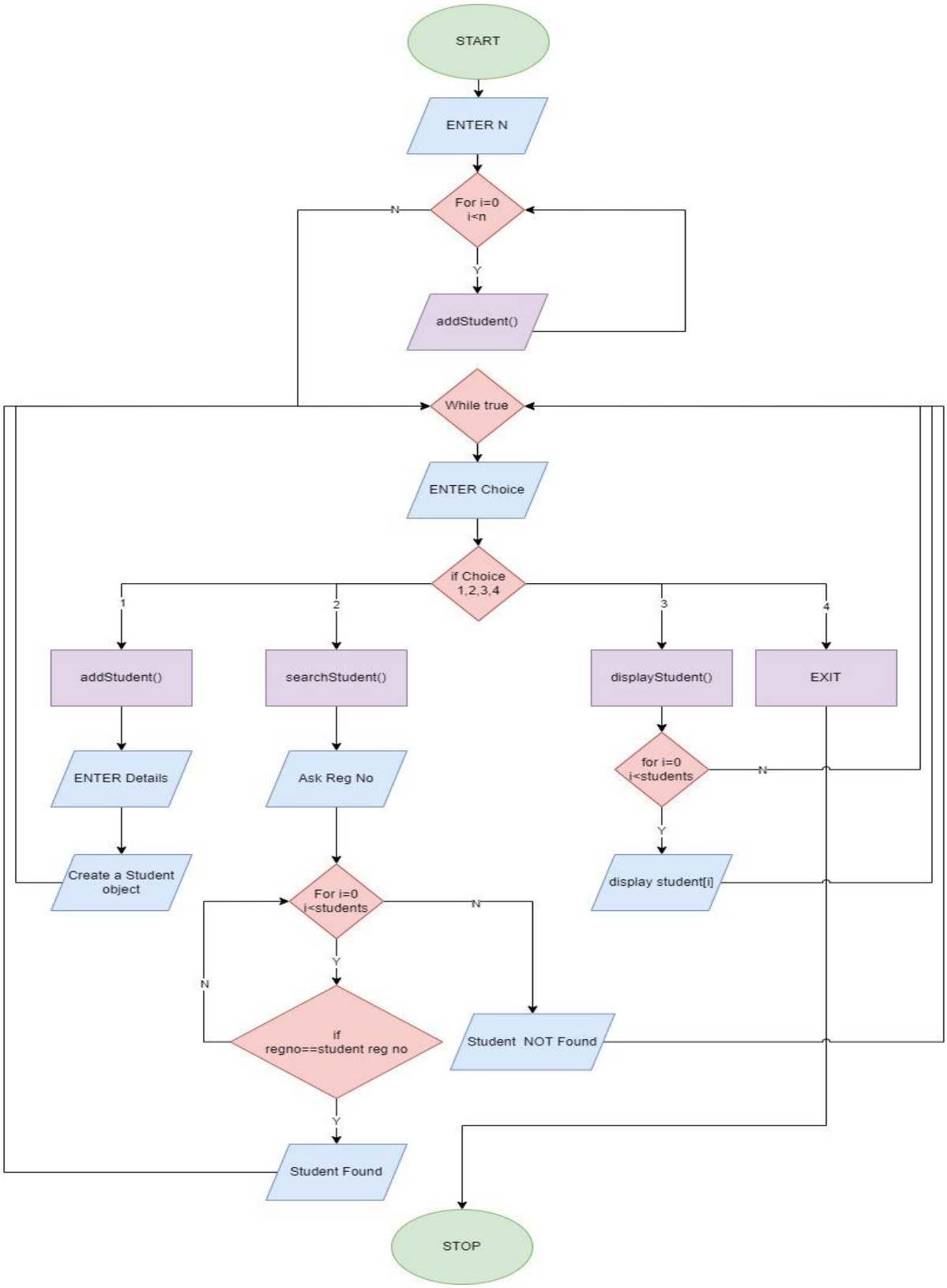
Displays the details of all students

Note: Make use of functions to implement the menu options

Extension (Optional):

Save the details of each student in a file, with his name with the help of a method in student class, when the student is getting created.

Flow Chart


START
ENTER N
For i=0
i<n
N
Y
addStudent()
While true
ENTER Choice
if Choice
1,2,3,4
1
2
3
4
addStudent()
searchStudent()
displayStudent()
EXIT
ENTER Details
Ask Reg No
for i=0
i<students
N
Y
Create a Student
object
display student[i]
For i=0
i<students
N
Y
N
if
regno==student reg no
Student NOT Found
Y
Student Found
STOP

Solution

```
import java.util.Scanner;
class Student {
    String regNo;
    String name;
    String email;
    String phone;
    String classs;
    String department;
    Student(String regNo, String name, String email, String phone, String
classs, String department){
        this.regNo=regNo;
        this.name=name;
        this.email=email;
        this.phone=phone;
        this.classs=classs;
        this.department=department;
    }
    void printDetails(){
        System.out.printf("%-25s %-15s %-50s %-12s %-10s %-
15s",this.regNo,this.name,this.email,this.phone,this.classs,this.department);
    }
    String getRegNo(){
        return this.regNo;
    }
    String getName(){
        return this.name;
    }
}
public class Lab5 {
    static int count=0;
    static Scanner scan = new Scanner(System.in);
    static Student[] student = new Student[100];
    static Student addStudent(){
        System.out.println("===============Enter details for student
"+(count+1)+":===============");
        System.out.println("==================================================
========");
        System.out.print("Registration Number: ");
        String regNo=scan.nextLine();
        System.out.print("Name: ");
        String name=scan.nextLine();
        System.out.print("Email: ");
        String email=scan.nextLine();
```

```
        System.out.print("Phone Number: ");
        String phone=scan.nextLine();
        System.out.print("Class: ");
        String classs=scan.nextLine();
        System.out.print("Department: ");
        String department=scan.nextLine();
        System.out.println("==================================================
========");
        Student student = new
Student(regNo,name,email,phone,classs,department);
        return student;
    }
    static void searchStudent(){
        System.out.println("====================Search a
student=======================");
        int flag=0;
        System.out.print("Registration Number: ");
        String regno=scan.nextLine();
        for(int i=0;i<count;i++){
            if(regno.equalsIgnoreCase(student[i].getRegNo())){
                flag=1;
                System.out.print("Name: ");
                String name=scan.nextLine();
                System.out.println("==========================================
================");
                if(name.equalsIgnoreCase(student[i].getName())){
                    System.out.println("The student with registration number
"+regno+ " and name "+name+" is present.");
                }
                else{
                    System.out.println("Name does not match!");
                }
                System.out.println("==========================================
================");
            }
        }
        if(flag==0){
            System.out.println("==========================================
============");
            System.out.println("The student with registration number "+regno+
" is not present!");
        }
    }
    public static void main(String[] args) {
        System.out.println("==================================================
========");
```

```
        System.out.print("How many students you want to add? : ");
        int n=Integer.parseInt(scan.nextLine());
        System.out.println("================================================
========");
        for(int i=0;i<n;i++){
            student[i]=addStudent();
            count++;
        }
        while(true){
            System.out.println("============================MENU==========
============");
            System.out.println("1. Add a new student.");
            System.out.println("2. Search a student.");
            System.out.println("3. Display all students.");
            System.out.println("4. Exit.");
            System.out.println("============================================
============");
            System.out.print("Enter your choice: ");
            int choice=Integer.parseInt(scan.nextLine());
            System.out.println("============================================
============");
            switch(choice){
                case 1:{
                    student[count]=addStudent();
                    count++;
                    break;
                }
                case 2:{
                    searchStudent();
                    break;
                }
                case 3:{
                    if(count==0){
                        System.out.println("====================No students
found!====================");
                        System.out.println("================================
========================");
                    }
                    else{
                        System.out.println("----------------------Student
Details----------------------");
                        System.out.println("--------------------------------
------------------------------------------------------------------------
-------------------------");
```

```
                        System.out.printf("%-25s %-15s %-50s %-12s %-10s %-
15s\n", "Registration Number", "Name", "Email", "Phone Number", "Class",
"Department");
                        System.out.println("---------------------------------
-------------------------------------------------------------------------
--------------------------");
                        for(int i=0;i<count;i++){
                            student[i].printDetails();
                            System.out.println();
                            System.out.println("-----------------------------
-------------------------------------------------------------------------
------------------------------");
                        }
                    }
                    break;
                }
                case 4:{
                    System.exit(0);
                    break;
                }
                default:{
                    System.out.println("Enter valid choice!");
                    break;
                }
            }
        }
    }
}
```

Output

```
============================================================
How many students you want to add? : 2
============================================================
===============Enter details for student 1:===============
============================================================
Registration Number: 22122140
Name: Karan
Email: punjabi.karan@msds.christuniversity.in
Phone Number: 7405387209
Class: 2MScDS-B
Department: Data Science
============================================================
===============Enter details for student 2:===============
============================================================
Registration Number: 22122103
Name: Akshay
Email: akshay.gangadhar@msds.christuniversity.in
Phone Number: 9976587678
Class: 2MScDS-B
Department: Data Science
============================================================
```

```
=========================MENU=========================
1. Add a new student.
2. Search a student.
3. Display all students.
4. Exit.
======================================================
Enter your choice: 3
======================================================
--------------------Student Details--------------------
-----------------------------------------------------------------------------------------------------------------
Registration Number      Name          Email                                    Phone Number Class      Department
-----------------------------------------------------------------------------------------------------------------
22122140                 Karan         punjabi.karan@msds.christuniversity.in   7405387209   2MScDS-B   Data Scien
-----------------------------------------------------------------------------------------------------------------
22122103                 Akshay        akshay.gangadhar@msds.christuniversity.in 9976587678  2MScDS-B   Data Scien
-----------------------------------------------------------------------------------------------------------------
```

```
============================MENU============================
1. Add a new student.
2. Search a student.
3. Display all students.
4. Exit.
============================================================
Enter your choice: 2
============================================================
====================Search a student=======================
Registration Number: 22122103
Name: AkshAY
============================================================
The student with registration number 22122103 and name AkshAY is present.
============================================================
```

```
============================MENU============================
1. Add a new student.
2. Search a student.
3. Display all students.
4. Exit.
============================================================
Enter your choice: 2
============================================================
====================Search a student=======================
Registration Number: 22122153
============================================================
The student with registration number 22122153 is not present!
============================MENU============================
1. Add a new student.
2. Search a student.
3. Display all students.
4. Exit.
============================================================
```

```
============================MENU============================
1. Add a new student.
2. Search a student.
3. Display all students.
4. Exit.
============================================================
Enter your choice: 2
============================================================
====================Search a student=======================
Registration Number: 22122140
Name: Punjabi
============================================================
Name does not match!
============================================================
```

```
============================MENU============================
1. Add a new student.
2. Search a student.
3. Display all students.
4. Exit.
============================================================
Enter your choice: 1
============================================================
===============Enter details for student 3:================
============================================================
Registration Number: 22122153
Name: Subhramanya
Email: subhramanya.patil@llb.christuniversity.in
Phone Number: 8789654376
Class: 2LLB-A
Department: LAW
============================================================
```

```
=========================MENU=========================
1. Add a new student.
2. Search a student.
3. Display all students.
4. Exit.
======================================================
Enter your choice: 3
======================================================
--------------------Student Details---------------------
-------------------------------------------------------------------------------------------------------------------
Registration Number     Name          Email                                    Phone Number Class     Department
-------------------------------------------------------------------------------------------------------------------
22122140                Karan         punjabi.karan@msds.christuniversity.in   7405387209   2MScDS-B  Data Scien
-------------------------------------------------------------------------------------------------------------------
22122103                Akshay        akshay.gangadhar@msds.christuniversity.in 9976587678  2MScDS-B  Data Scien
-------------------------------------------------------------------------------------------------------------------
22122153                Subhramanya   subhramanya.patil@llb.christuniversity.in 8789654376  2LLB-A    LAW
-------------------------------------------------------------------------------------------------------------------
```

```
============================MENU============================
1. Add a new student.
2. Search a student.
3. Display all students.
4. Exit.
============================================================
Enter your choice: 2
============================================================
====================Search a student========================
Registration Number: 22122153
Name: subhramanya
============================================================
The student with registration number 22122153 and name subhramanya is present.
============================================================
```

Question 6.

Write a JAVA program that reads a file (text file) and finds the following details.

Vowels and their count. Eg: [a:100, e:45, I:55, ..]

Digits and their count. Eg: [0:1000, 1:10, 2:40, ...]

Top Five repeated words and their count.

Least Five repeated words and their count.

Top Five repeated characters and their count.

Least FIve repeated characters and their count.

Note:

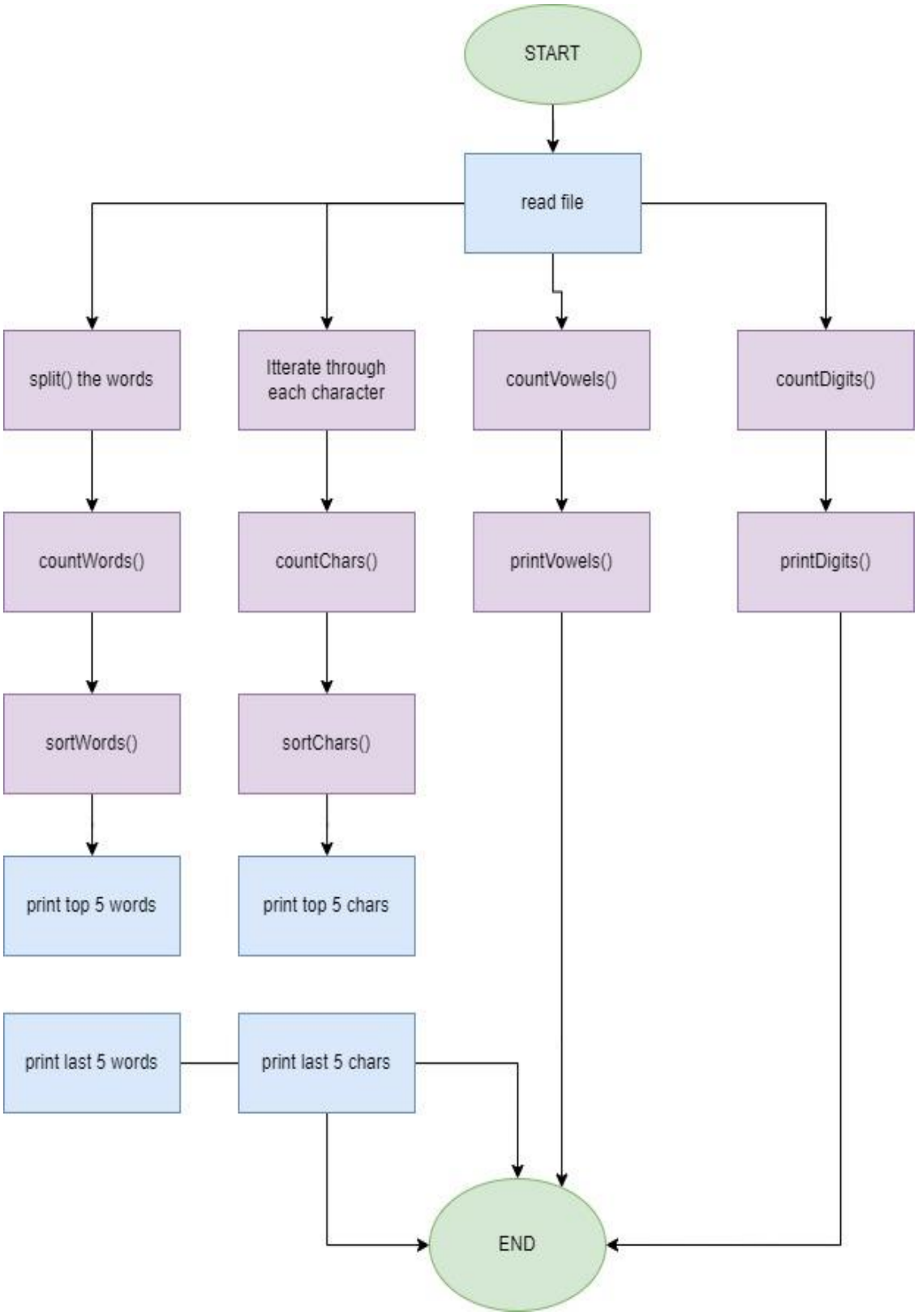
Use the file attached.

Use functions to implement the six functions asked.

Pass the file content to the functions and return the values accordingly.

No Global variables are allowed to be used.

Flow Chart


START
read file
split() the words
Itterate through each character
countVowels()
countDigits()
countWords()
countChars()
printVowels()
printDigits()
sortWords()
sortChars()
print top 5 words
print top 5 chars
print last 5 words
print last 5 chars
END

Solution

```
// Importing all the necessary packages
import java.io.BufferedReader;
import java.io.File;
import java.io.FileReader;
import java.io.FileWriter;
import java.io.IOException;
import java.util.Scanner;

public class Lab6 {
    // This function will count the vowels present in the entire string and
return an array like [2,3,1,6,2] representing values counts for each vowel
respectively.
    static int[] vowelCount(String s) {
        int[] vowelCount = { 0, 0, 0, 0, 0 };
        for (int i = 0; i < s.length(); i++) {
            switch (s.charAt(i)) {
                case 'A':
                case 'a':
                    vowelCount[0] += 1;
                    break;
                case 'E':
                case 'e':
                    vowelCount[1] += 1;
                    break;
                case 'I':
                case 'i':
                    vowelCount[2] += 1;
                    break;
                case 'O':
                case 'o':
                    vowelCount[3] += 1;
                    break;
                case 'U':
                case 'u':
                    vowelCount[4] += 1;
                    break;
            }
        }
        return vowelCount;
    }
    // This function will count the digits present in the entire string and
return an array like [23,3,11,6,2,6,7,3,2,6] representing values counts for
each digit respectively from [0:9].
```

```
    static int[] digitCount(String s) {
        int[] digitCount = { 0, 0, 0, 0, 0, 0, 0, 0, 0, 0 };
        for (int i = 0; i < s.length(); i++) {
            switch (s.charAt(i)) {
                case '0':
                    digitCount[0] += 1;
                    break;
                case '1':
                    digitCount[1] += 1;
                    break;
                case '2':
                    digitCount[2] += 1;
                    break;
                case '3':
                    digitCount[3] += 1;
                    break;
                case '4':
                    digitCount[4] += 1;
                    break;
                case '5':
                    digitCount[5] += 1;
                    break;
                case '6':
                    digitCount[6] += 1;
                    break;
                case '7':
                    digitCount[7] += 1;
                    break;
                case '8':
                    digitCount[8] += 1;
                    break;
                case '9':
                    digitCount[9] += 1;
                    break;
            }
        }
        return digitCount;
    }

    // This function will print all the vowels present in the string with
their respective counts and it will not return anything.
    static void printVowelCounts(String s) {
        String[] vowelsList = { "A", "E", "I", "O", "U" };
        int[] vowels = vowelCount(s);
        for (int i = 0; i < vowels.length; i++) {
            System.out.println(vowelsList[i] + " --> " + vowels[i]);
```

```
        }
    }

    // This function will print all the digits present in the string with
their respective counts and it will not return anything.
    static void printDigitCounts(String s) {
        String[] digitsList = { "0", "1", "2", "3", "4", "5", "6", "7", "8",
"9" };
        int[] digits = digitCount(s);
        for (int i = 0; i < digits.length; i++) {
            System.out.println(digitsList[i] + " --> " + digits[i]);
        }
    }

    // This function will take a word and array of words, if the word is
present in the array of words then it will return true. Else it will return
false.
    static boolean findWord(String word, String[] words) {
        for (String string : words) {
            if (word.equalsIgnoreCase(string)) {
                return true;
            }
        }
        return false;
    }

    // This function will return a array of words with their count in
decending order. Example: The function will return {"and 51","is 44","the
31"}, if the input is like {"is 44","and 51","the 31"}
    static String[] sortWordsInDescending(String[] words) {
        for (int i = 0; i < words.length; i++) {
            if (words[i] == null) {
                break;
            }
            String[] parts = words[i].split(" ");
            int num1 = Integer.parseInt(parts[1]);
            for (int j = 0; j < words.length; j++) {
                if (words[j] == null) {
                    break;
                }
                String[] parts1 = words[j].split(" ");
                int num2 = Integer.parseInt(parts1[1]);
                if (num1 <= num2) {
                    continue;
                } else {
                    String temp = words[j];
```

```
                    words[j] = words[i];
                    words[i] = temp;
                    num1 = num2;
                }
            }
        }
        return words;
    }

    // This function will take string as the input and pass the array of words
in sortWordsInDecending() so that it will return repeated words.
    static String[] wordCount(String s) {
        String[] words = s.split("[\\s,()]+");
        for (int i = 0; i < words.length; i++) {
            words[i] = words[i].replaceAll("\\.", "");
        }
        String[] uniqueWords = new String[words.length];
        String[] wordCount = new String[words.length];
        int[] count = new int[words.length];
        int k = 0;
        for (int i = 0; i < words.length; i++) {
            if (findWord(words[i], uniqueWords)) {
                continue;
            } else {
                uniqueWords[i] = words[i];
                count[i] = 1;
                for (int j = i + 1; j < words.length; j++) {
                    if (words[i].equalsIgnoreCase(words[j])) {
                        count[i] += 1;
                    }
                }
                wordCount[k] = words[i] + " " + count[i];
                k++;
            }

        }
        return sortWordsInDescending(wordCount);
    }

    // This function will take a character and array of character, if the
character is present in the array of characters then it will return true. Else
it will return false.
    static boolean findChar(char c, char[] s) {
        for (int i = 0; i < s.length; i++) {
            if (s[i] == c) {
                return true;
```

```
            }
        }
        return false;
    }

    // This function will take string as the input and pass the array of words
in sortWordsInDecending() so that it will return repeated characters.
    static String[] charCount(String s) {
        s = s.toLowerCase();
        char[] uniqueChars = new char[s.length()];
        String[] charCount = new String[s.length()];
        int[] count = new int[s.length()];
        int k = 0;
        for (int i = 0; i < s.length(); i++) {
            if (findChar(s.charAt(i), uniqueChars) || s.charAt(i) == ' ') {
                continue;
            } else {
                uniqueChars[i] = s.charAt(i);
                count[i] = 1;
                for (int j = i + 1; j < s.length(); j++) {
                    if (s.charAt(i) == s.charAt(j)) {
                        count[i] += 1;
                    }
                }
                charCount[k] = s.charAt(i) + " " + count[i];
                k++;
            }
        }
        return sortWordsInDescending(charCount);
    }

    // This function will print top 5 repeated words of your string.
    static void topFiveRepeatedWords(String[] s) {
        for (int i = 0; i < 5; i++) {
            System.out.println(s[i]);
        }
    }

    // This function will print last 5 repeated words of your string.
    static void lastFiveRepeatedWords(String[] s) {
        int length = 0;
        for (int i = 0; i < s.length; i++) {
            if (s[i] == null) {
                break;
            }
            length++;
```

```
        }
        for (int i = length - 5; i < length; i++) {
            System.out.println(s[i]);
        }
    }

    // This function will print top 5 repeated characters in your string.
    static void topFiveRepeatedChars(String[] s) {
        for (int i = 0; i < 5; i++) {
            System.out.println(s[i]);
        }
    }

    // This function will print last 5 repeated characters in your string.
    static void lastFiveRepeatedChars(String[] s) {
        int length = 0;
        for (int i = 0; i < s.length; i++) {
            if (s[i] == null) {
                break;
            }
            length++;
        }
        for (int i = length - 5; i < length; i++) {
            System.out.println(s[i]);
        }
    }

    public static void main(String[] args) throws IOException {
        System.out.println("==================================================
====================");
        String s="";
        try {
        File file=new File("text.txt");
        Scanner scan = new Scanner(file);
        while(scan.hasNextLine()){
        s+=scan.nextLine();
        }
        scan.close();
        } catch (Exception e) {
        System.out.println("An error occurred.");
        }
        String[] sortedWordsInDecending=wordCount(s);
        String[] sortedCharsInDecending=charCount(s);
        System.out.println("==================================================
====================");
        System.out.println("Vowels:");
```

```
        System.out.println("==================================================
====================");
        printVowelCounts(s);
        System.out.println("==================================================
====================");
        System.out.println("Digits:");
        System.out.println("==================================================
====================");
        printDigitCounts(s);
        System.out.println("==================================================
====================");
        System.out.println("Top 5 Repeated Words:");
        System.out.println("==================================================
====================");
        topFiveRepeatedWords(sortedWordsInDecending);
        System.out.println("==================================================
====================");
        System.out.println("Last 5 Repeated Words:");
        System.out.println("==================================================
====================");
        lastFiveRepeatedWords(sortedWordsInDecending);
        System.out.println("==================================================
====================");
        System.out.println("Top 5 Repeated Chars:");
        System.out.println("==================================================
====================");
        topFiveRepeatedChars(sortedCharsInDecending);
        System.out.println("==================================================
====================");
        System.out.println("Last 5 Repeated Chars:");
        System.out.println("==================================================
====================");
        lastFiveRepeatedChars(sortedCharsInDecending);
        System.out.println("==================================================
====================");



                                        // PART 2 (Additional Part)


        System.out.println("==================================================
====================");
        System.out.println("================================PART
2=================================");
```

```
        System.out.println("================================================
====================");
        String content="";
        String line="";
        try {
            File file = new File("htmlFile.txt");
            // As the file is a huge file, we use BufferedReader object
instead of Scanner object
            BufferedReader reader = new BufferedReader(new FileReader(file));
            while((line=reader.readLine())!=null){
                content+=line;
            }
            reader.close();
        } catch (Exception e) {
            System.out.println("An error occured!");
        }
        // Finding the starting index of Top Categories
        int start=content.indexOf("Top Categories");
        // Finding the starting index of the next section so that everything
between that is included.
        int end=content.indexOf("<div class=\"trending-head\">Trending
News</div>");
        // Extracting the substring from the full content.
        content=content.substring(start, end);
        // Again extracting the substrings based on our requirements.
        content=content.substring(content.indexOf("<ul"),content.indexOf("</ul
>"));
        String[] items = content.split("</a>");
        content="Top Categories\n\n";
        int count=0;
        for (String item : items) {
            // Spliting the content from <a href tag as our main categories
are present between the <a></a> tags.
            String[] itemSplitArray=item.split("<a href=\".*\">");
            if(itemSplitArray.length>1){
                count+=1;
                String itemName=item.split("<a href=\".*\">")[1];
                content+=count+ ") " +itemName+"\n";
            }
        }
        System.out.println(content);
        FileWriter writer=new FileWriter("Top Categories.txt");
        writer.write(content);
        writer.close();
        System.out.println("================================================
====================");
```

```
    }
}
```

Output

```
================================================================================
================================================================================
Vowels:
================================================================================
A --> 357
E --> 504
I --> 360
O --> 260
U --> 107
================================================================================
Digits:
================================================================================
0 --> 6
1 --> 6
2 --> 7
3 --> 4
4 --> 2
5 --> 4
6 --> 2
7 --> 4
8 --> 2
9 --> 0
================================================================================
```

```
================================================================================
Top 5 Repeated Chars:
================================================================================
e 504
i 360
a 357
t 337
n 334
================================================================================
Last 5 Repeated Chars:
================================================================================
6 2
z 2
4 2
8 2
j 2
================================================================================
```

```
================================================================================
================================PART 2==========================================
================================================================================
Top Categories

1) Explained News
2) Political Pulse
3) Latest Opinion
4) Mumbai News
5) Delhi News
6) Pune News
7) Bangalore News
8) Bollywood News
9) Health News
10) India News
11) Sports News
12) Lifestyle News
13) Jobs
14) Mobile &amp;amp; Tabs
15) Tech Reviews
16) Gadgets
17) Mobile &amp;amp; Tabs
18) Food &amp;amp; Wine
19) Elections 2023
20) Fitness

================================================================================
```

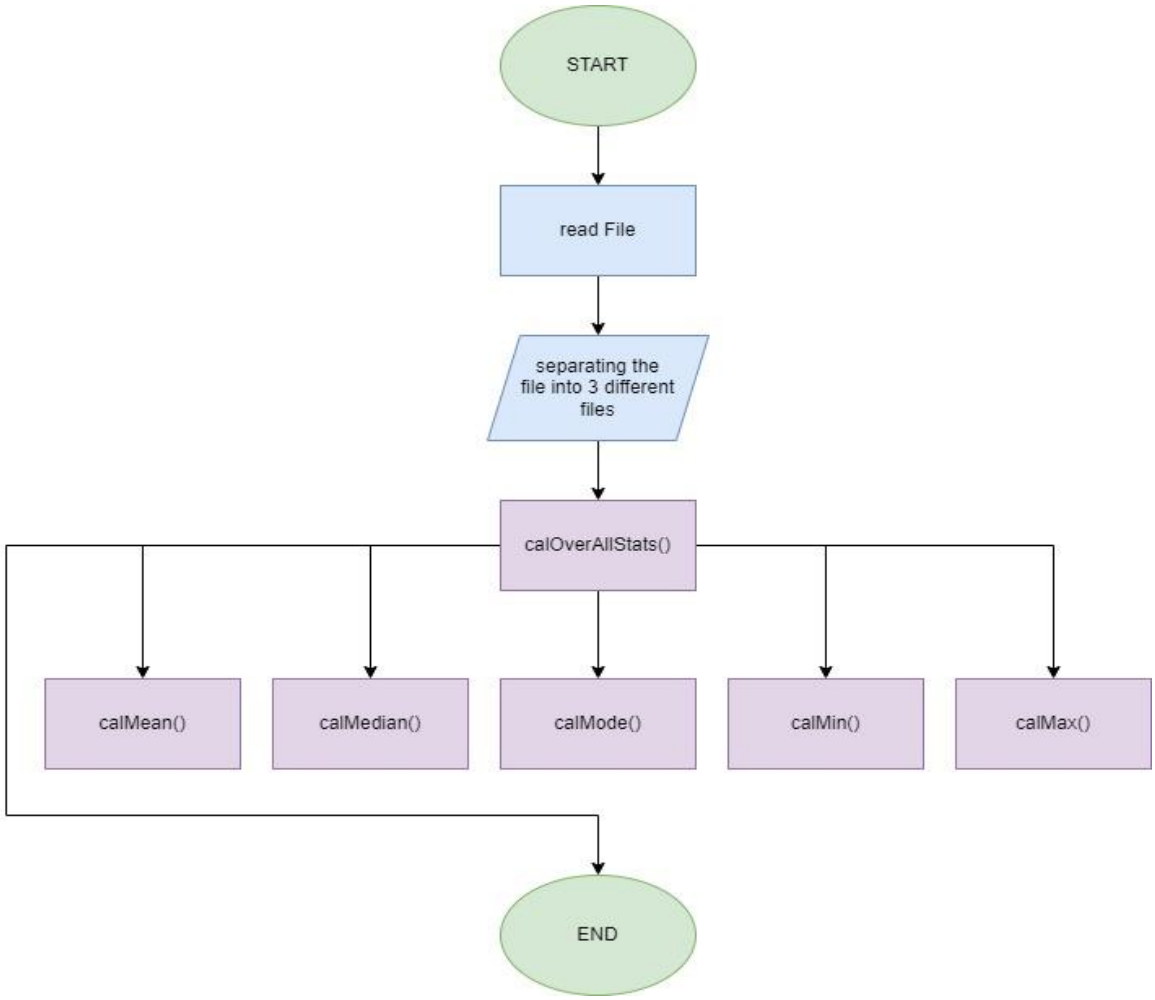
Question 7.

Given the famous iris dataset, find the 5-point summary [Mean, Median, Mode, Min, Max] for the attributes: SepalLengthCm, SepalWidthCm, PetalLengthCm, PetalWidthCm.

Once the overall summary statistics have been calculated, identify the summary statistics for each Species of iris flower [Iris-setosa, Iris-versicolor, Iris-virginica].

Present your results in the appropriate format and write the results in a file.

Flow Chart



Solution

```
import java.io.File;
import java.io.FileNotFoundException;
import java.io.FileWriter;
import java.util.Scanner;

public class Lab7 {
    // This function will take an array as an argument and it will return the
same
    // array after sorting in ascending order.
    static float[] sortArray(float[] arr) {
        float temp = 0;
        for (int i = 0; i < arr.length; i++) {
            for (int j = 0; j < arr.length; j++) {
                if (arr[j] > arr[i]) {
                    temp = arr[i];
                    arr[i] = arr[j];
                    arr[j] = temp;
                }
            }
        }
        return arr;
    }

    // This function will calculate and return the mean of the array which is
passed
    // in argument.
    static float calMean(float[] arr) {
        float sum = 0;
        for (int i = 0; i < arr.length; i++) {
            sum += arr[i];
        }
        return sum / arr.length;
    }

    // This function will calculate and return the median value of the array
which
    // is passed in argument.
    static float calMedian(float[] arr) {
        if (arr.length % 2 == 0) {
            return (arr[arr.length / 2] + arr[arr.length / 2 - 1]) / 2;
        } else {
            return arr[arr.length / 2];
        }
```

```
    }

    // This function will calculate and return the mode of the array which is
passed
    // in argument.
    static float calMode(float[] arr) {
        float mode = 0;
        int maxCount = 0;
        for (int i = 0; i < arr.length; i++) {
            int count = 0;
            for (int j = 0; j < arr.length; j++) {
                if (arr[i] == arr[j]) {
                    count++;
                }
            }
            if (count > maxCount) {
                maxCount = count;
                mode = arr[i];
            }
        }
        return mode;
    }

    // This function will calculate and return the minimum value of the array
which
    // is passed in argument.
    static float findMin(float[] arr) {
        return arr[0];
    }

    // This function will calculate and return the maximum value of the array
which
    // is passed in argument.
    static float findMax(float[] arr) {
        return arr[arr.length - 1];
    }

    // This function will take file as an argument and it will print all the
summary
    // statistics in proper format.
    static String calOverAllStats(File myFile) throws FileNotFoundException {
        String printContent = "";
        Scanner scan = new Scanner(myFile);
        // Skipping the first row of the file as it contains the header part.
        scan.nextLine();
        // Counting the number of lines the file contains.
```

```
        int countLines = 0;
        while (scan.hasNextLine()) {
            // Just reading the
            scan.nextLine();
            countLines += 1;
        }
        // Creating arrays of exact values.
        float[] sepalLength = new float[countLines];
        float[] sepalWidth = new float[countLines];
        float[] petalLength = new float[countLines];
        float[] petalWidth = new float[countLines];
        int i = 0;
        scan.close();
        // Again reading the file from starting.
        scan = new Scanner(myFile);
        // Skipping the first row of the file as it contains the header part.
        scan.nextLine();
        // Iterating through each row.
        while (scan.hasNextLine()) {
            String[] s;
            String line = scan.nextLine();
            // Splitting each row based on commas and assigning each value of
respected
            // column to its respected array.
            s = line.split(",");
            sepalLength[i] = Float.parseFloat(s[1]);
            sepalWidth[i] = Float.parseFloat(s[2]);
            petalLength[i] = Float.parseFloat(s[3]);
            petalWidth[i] = Float.parseFloat(s[4]);
            i++;
        }

        // Sorting all the values of the arrays in ascending order.
        sepalLength = sortArray(sepalLength);
        sepalWidth = sortArray(sepalWidth);
        petalLength = sortArray(petalLength);
        petalWidth = sortArray(petalWidth);

        // Storing all the details in good format and returning the content as
a string.

        printContent += String.format("%-15s %-15s %-15s %-15s %-15s %-15s\n",
" ", "Mean", "Median", "Mode", "Minimum",
                "Maximum");
        printContent += String.format("%-15s %-15s %-15s %-15s %-15s %-15s\n",
"Sepal Length", calMean(sepalLength),
```

```
                calMedian(sepalLength), calMode(sepalLength),
findMin(sepalLength), findMax(sepalLength));
        printContent += String.format("%-15s %-15s %-15s %-15s %-15s %-15s\n",
"Sepal Width", calMean(sepalWidth),
                calMedian(sepalWidth), calMode(sepalWidth),
findMin(sepalWidth), findMax(sepalWidth));
        printContent += String.format("%-15s %-15s %-15s %-15s %-15s %-15s\n",
"Petal Length", calMean(petalLength),
                calMedian(petalLength), calMode(petalLength),
findMin(petalLength), findMax(petalLength));
        printContent += String.format("%-15s %-15s %-15s %-15s %-15s %-15s\n",
"Petal Width", calMean(petalWidth),
                calMedian(petalWidth), calMode(petalWidth),
findMin(petalWidth), findMax(petalWidth));

        // Closing the scanner object.
        scan.close();
        return printContent;
    }

    public static void main(String[] args) {
        // Creating string objects to store the content of each file.
        String irisSetosa = "";
        String irisVersicolor = "";
        String irisVerginica = "";
        String fileContent = "";
        try {
            // Opening the file.
            File myFile = new File("Iris.csv");
            Scanner scan = new Scanner(myFile);
            // Readingband storing the first row of the file.
            String firstRow = scan.nextLine();
            irisSetosa = firstRow + "\n";
            irisVersicolor = firstRow + "\n";
            irisVerginica = firstRow + "\n";

            // Iterating through the file.
            while (scan.hasNextLine()) {
                String[] s;
                String line = scan.nextLine();
                s = line.split(",");
                // Storing the rows of differnt species in their respective
string objects.
                if (s[5].equalsIgnoreCase("Iris-setosa")) {
                    irisSetosa += line + "\n";
                } else if (s[5].equalsIgnoreCase("Iris-versicolor")) {
```

```
                    irisVersicolor += line + "\n";
                } else {
                    irisVerginica += line + "\n";
                }
            }

            // Creating separate files for each of the different species, so
that we can
            // pass these files directly to our calOverAllStats() function.
            FileWriter irisSetosaFile = new FileWriter("irisSetosa.csv");
            irisSetosaFile.write(irisSetosa);
            irisSetosaFile.close();

            FileWriter irisVersicolorFile = new
FileWriter("irisVersicolor.csv");
            irisVersicolorFile.write(irisVersicolor);
            irisVersicolorFile.close();

            FileWriter irisVerginicaFile = new
FileWriter("irisVerginica.csv");
            irisVerginicaFile.write(irisVerginica);
            irisVerginicaFile.close();

            // Storing the output in a decorated format.
            fileContent +=
"=============================================================================
==========\n====================================OVERALL
SUMMARY====================================\n==================================
=========================================================\n";
            fileContent += calOverAllStats(myFile);
            fileContent+="=============================================================
==================================\n=======================================IRI
S
SETOSA=====================================\n================================
=========================================================\n";
            File irisSetosaCsvFile = new File("irisSetosa.csv");
            fileContent += calOverAllStats(irisSetosaCsvFile);
            fileContent+="=============================================================
==================================\n=======================================IRIS
VERSICOLOR=================================\n================================
=========================================================\n";
            File irisVersicolorCsvFile = new File("irisVersicolor.csv");
            fileContent += calOverAllStats(irisVersicolorCsvFile);
            fileContent+="=============================================================
==================================\n=======================================IRIS
```

```
VERGINICA==================================\n==============================
============================================================\n";
            File irisVerginicaCsvFile = new File("irisVerginica.csv");
            fileContent += calOverAllStats(irisVerginicaCsvFile);
            fileContent+="==============================================================
================================";

            // Printing the final output.
            System.out.println(fileContent);

            // Creating and writing the final output in a txt file.
            FileWriter outputContent=new FileWriter("Output.txt");
            outputContent.write(fileContent);
            outputContent.close();

            // Closing the scanner object.
            scan.close();
        } catch (Exception e) {
            System.out.println("An Error Occured!" + e.getMessage());
        }
    }
}
```

Output

```
================================================================================
==================================OVERALL SUMMARY===============================
================================================================================
              Mean            Median          Mode            Minimum         Maximum
Sepal Length  5.8433356       5.8             5.0             4.3             7.9
Sepal Width   3.0540001       3.0             3.0             2.0             4.4
Petal Length  3.7586665       4.3500004       1.5             1.0             6.9
Petal Width   1.198667        1.3             0.2             0.1             2.5
================================================================================
```

```
=================================IRIS SETOSA====================================
================================================================================
              Mean            Median          Mode            Minimum         Maximum
Sepal Length  5.0060005       5.0             5.0             4.3             5.8
Sepal Width   3.418           3.4             3.4             2.3             4.4
Petal Length  1.4639996       1.5             1.5             1.0             1.9
Petal Width   0.24399994      0.2             0.2             0.1             0.6
================================================================================
```

```
================================IRIS VERSICOLOR=================================
================================================================================
              Mean            Median          Mode            Minimum         Maximum
Sepal Length  5.9360003       5.9             5.5             4.9             7.0
Sepal Width   2.77            2.8             3.0             2.0             3.4
Petal Length  4.26            4.3500004       4.5             3.0             5.1
Petal Width   1.3260001       1.3             1.3             1.0             1.8
================================================================================
```

```
================================IRIS VERGINICA==================================
================================================================================
              Mean            Median          Mode            Minimum         Maximum
Sepal Length  6.5880003       6.5             6.3             4.9             7.9
Sepal Width   2.9739997       3.0             3.0             2.2             3.8
Petal Length  5.5520005       5.55            5.1             4.5             6.9
Petal Width   2.0260003       2.0             1.8             1.4             2.5
================================================================================
```