**JAVA LAB CYCLE-2**

## Experiment 1: Use Array to Sort strings.

**Aim:** Program to Sort strings ,

**Algorithm**

1. Start
2. Declare an array String str[].
3. Declare the variable int count, String temp.
4. Ask the user to Enter number of strings that stored to –> count and Enter the Strings, initialize to the array str[i]
5. Perform sort operation: if (str[i].compareTo(str[j])>0) temp = str[i];str[i] = str[j];str[j] = temp;
6. Print the Strings in Sorted Order
7. stop

**Program:**

**Sort.java**

import java.util.Scanner;

public class Sort

{

public static void main(String[] args)

{

int count;

String temp;

Scanner scan = new Scanner(System.in);

System.out.print("Enter number of strings you would like to enter:");

count = scan.nextInt();

String str[] = new String[count];

Scanner scan2 = new Scanner(System.in);

System.out.println("Enter the Strings one by one:");

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

{

str[i] = scan2.nextLine();

}

scan.close();

scan2.close();

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

{

for (int j = i + 1; j < count; j++) {

if (str[i].compareTo(str[j])>0)

{

temp = str[i];

str[i] = str[j];

str[j] = temp;

}

}

}

System.out.print("Strings in Sorted Order:");

for (int i = 0; i <= count - 1; i++)

{

System.out.print(str[i] + ", ");

}

}

}

**Expected Output**



## Experiment 2: Search an element in an array.

**Aim:** Search an element in an array - write a menu driven program to perform both linear search and binary search of a string in a String array.

**Algorithm**

1. start.
2. Declare a class LinBin and initialize the variable int i,n,ch,flag=0; String search, str[],low=0; high=a.length-1, mid;
3. Ask the user to Enter no. of elements you want in array that stored to –> n and Enter all the elements , initialize to the array str[].
4. Ask the user to Enter the element you want to find that stored to –> str[i]=s.next();
5. Ask the user to select the choose that stored to –>ch and enter the element to be searched
6. Declare the switch case for linersearch

if(str[i].equals(search))flag=1;break; else flag=0;

if(flag==1) element +search+ found at position(i) else element not found

1. Declare the switch case for binarysearch

Arrays.sort(str);int searchIndex = binarySearch(str,search);

searchIndex != -1 ;str[searchIndex]+found at index +searchIndex : else element not found;while (low<=high);mid=(low + high)/2;

if (a[mid].compareTo(x)<0)

low = mid + 1;

else if (a[mid].compareTo(x)>0) high=mid-1;else return mid;return -1;

1. Print the output
2. stop

**Program:**

**LinBin.java**

import java.util.Arrays;

import java.util.Scanner;

import java.util.Arrays;

class LinBin

{

public static void main(String args[])

{

Scanner s=new Scanner(System.in);

int i,n,ch,flag=0;

String search;

System.out.print("enter the limit : ");

n=s.nextInt();

String str[]=new String[n];

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

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

{

str[i]=s.next();

}

do

{

System.out.println("\n\*\*\*ARRAY ELEMENT SEARCH\*\*\*");

System.out.println("\n 1.Linearsearch\n 2.Binarysearch\n 3.Exit");

System.out.println("\nenter your choice : " );

ch=s.nextInt();

switch(ch)

{

case 1:

System.out.print("enter the element to be searched : ");

search=s.next();

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

{

if(str[i].equals(search))

{

flag=1;

break;

}

else

{

flag=0;

}

}

if(flag==1)

{

System.out.println("element " +search+ " found at position "+(i)+" !!! ");

}

else

{

System.out.println("element not found!!!");

}

break;

case 2:

System.out.print("enter the element to be searched : ");

search=s.next();

Arrays.sort(str);

int searchIndex = binarySearch(str,search);

System.out.println(searchIndex != -1 ? str[searchIndex]+ " found at index "+searchIndex : "element not found !!!");

break;

case 3:

break;

default:

System.out.println("invalid option !!!");

break;

}

}

while(ch!=3);

}

public static int binarySearch(String a[], String x)

{

int low=0;

int high=a.length-1;

int mid;

while (low<=high)

{

mid=(low + high)/2;

if (a[mid].compareTo(x)<0)

{

low = mid + 1;

}

else if (a[mid].compareTo(x)>0)

{

high=mid-1;

}

else

{

return mid;

}

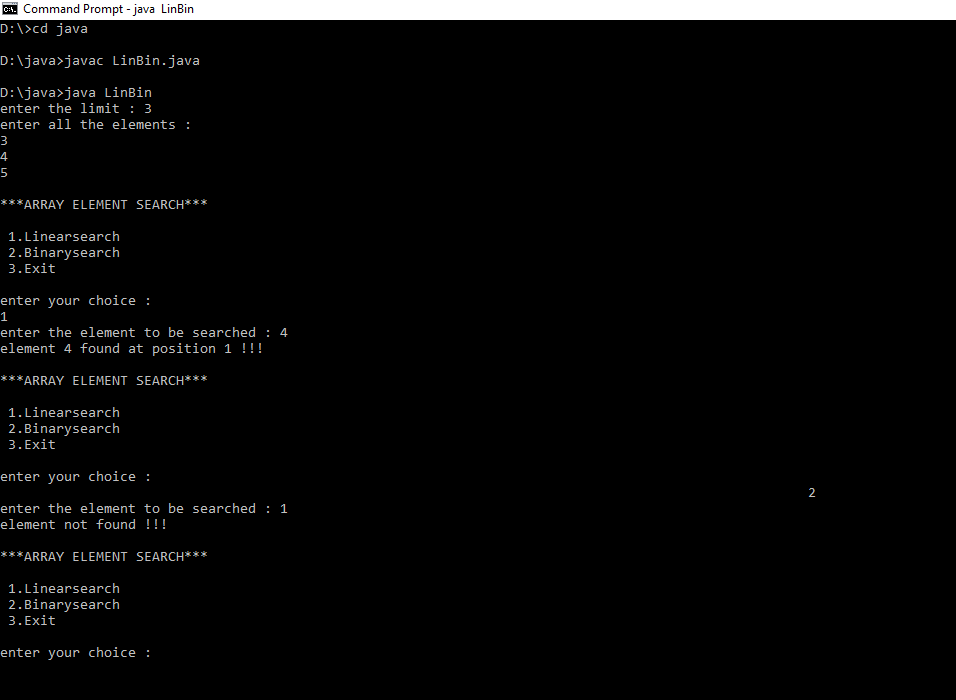
}

return -1;

}

}

**Expected Output**



**Experiment 3: Perform string manipulations**

**Aim:** Perform string manipulations.

* Create new strings using new.
* Getting String length
* String Concatenation
* Character extraction
* String Comparison
* Searching substrings
* Modifying a string
* Data conversion using value Of()

**Algorithm**

1. Start.
2. Declare an array char ch[].
3. Declare the String s1,s2,s3,s4.
4. Perform the string operation-> int length =s2.length(),

s2=s2.concat(" is immutable so assign it explicitly");

char s=s1.charAt(2); , (s1.compareTo(s2));

int firstIndex = s2.indexOf('s'); int lastIndex = s2.lastIndexOf('s');

String replaceString=s4.replace('a','e'); ,

String sample = String.valueOf(ch, 0, 3);

1. Print the output of string s1,s2,s3,s4,sample.
2. Stop

**Program:**

**StringExample.java**

public class StringExample{

public static void main(String args[]){

String s1="java";

char ch[]={'s','t','r','i','n','g','s'};

String s2=new String(ch);

String s3=new String("example");

System.out.println(s2);

System.out.println(s3);

int length =s2.length();

System.out.println("The length of the String \""+s2+"\" is: " +length);

s2=s2.concat(" is immutable so assign it explicitly");

System.out.println(s2);

char s=s1.charAt(2);

System.out.println(s);

System.out.println(s1.compareTo(s2));

int firstIndex = s2.indexOf('s');

System.out.println("First occurrence of char 's'" +

" is found at : " + firstIndex);

int lastIndex = s2.lastIndexOf('s');

System.out.println("Last occurrence of char 's' is" +

" found at : " + lastIndex);

String s4="javatpoint is a very good website";

String replaceString=s4.replace('a','e');//replaces all occurrences of 'a' to 'e'

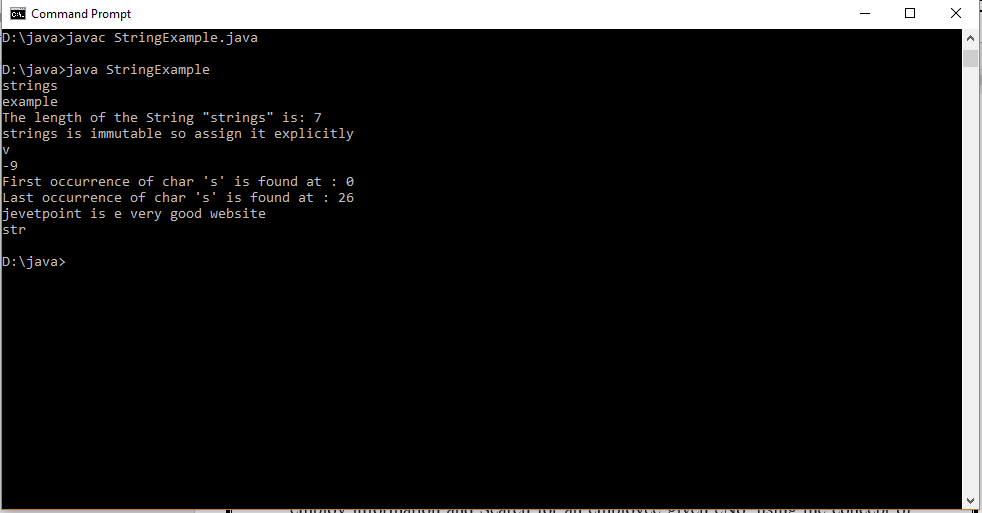
System.out.println(replaceString);

String sample = String.valueOf(ch, 0, 3);

System.out.println(sample);

}}

**Expected Output**



**Experiment 4: Using Array of Objects create a class for Employee and read n employ information and Search for an employee given eNo.**

**Aim:** 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.

**Algorithm**

1. Start.
2. Declare an array eNo[], eName[], eSalary [].
3. Declare and initialize the variable int count,flag = 0.
4. Ask the user to how many employees information you want enter that stored to –> count and Enter all the employees eno,ename,esalary and initialize to the array eNo[], eName[], eSalary [].
5. Ask the user to Enter the employee number you want to find that stored to –> x.
6. Check the condition for(i = 0; i < x; i++) if(no[i] == x) flag = 1;break;else flag = 0;
7. if(flag == 1) print Employee id "+x+" found otherwise Employee id not found
8. Stop

**Program:**

**employee.java**

import java.util.Scanner;

public class employee

{

public static void main(String[] args)

{

int count,esalarytemp,flag = 0;

Scanner scan = new Scanner(System.in);

System.out.print("how many employees information you want enter :");

count = scan.nextInt();

String eName[] = new String[count];

int salary[] = new int[count];

int no[] = new int[count];

Scanner scan2 = new Scanner(System.in);

Scanner scan3 = new Scanner(System.in);

Scanner scan4 = new Scanner(System.in);

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

{

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

no[i] = scan2.nextInt();

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

eName[i] = scan2.nextLine();

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

salary[i] = scan2.nextInt();

}

Scanner s = new Scanner(System.in);

System.out.print("Enter the employee number you want to find:");

int x = s.nextInt();

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

{

if(no[i] == x)

{

flag = 1;

break;

}

else

{

flag = 0;

}

}

if(flag == 1)

{

System.out.println("Employee id "+x+" found:");

}

else

{

System.out.println("Employee id not found");

}

scan.close();

scan2.close();

scan3.close();

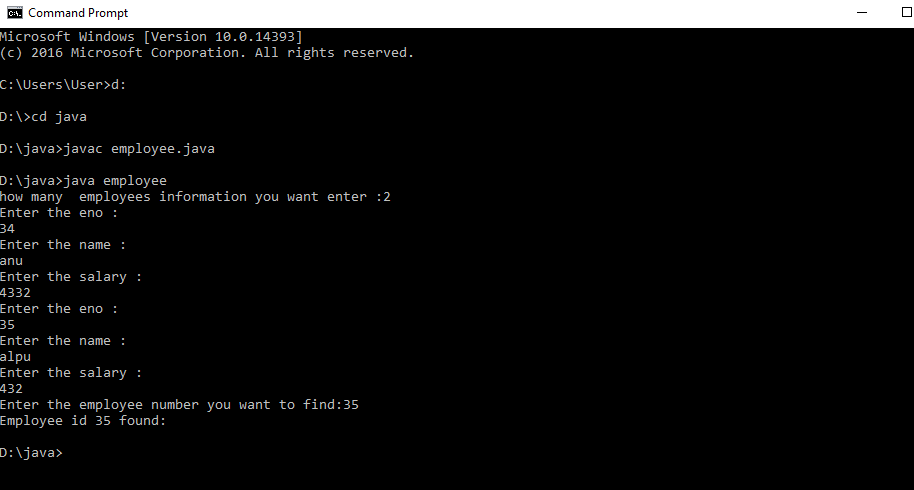
scan4.close();

s.close();

}

}

**Expected Output**



**Experiment 5: Design a class to represent a Student details.**

**Aim:** Design a class to represent a Student details. include the Student ID, name of the Student, branch, year and assign initial values, calculate average of marks of 6 subjects, and display grade. Also print the details of the students in the first and second position.

Program prints the grade based on this logic

1. If the average of marks is >= 80 then prints Grade ‘A’
2. If the average is <80 and >=60 then prints Grade ‘B’
3. If the average is <60 and >=40 then prints Grade ‘C’
4. else prints Grade ‘D’

**Algorithm**

1. Start.
2. Declare a class Student\_Details with data member int tot,i,avg,yr,sid , n=0,maxi1=0,maxi2=0,String sname,branch,int marks.Declare a member function void getdata()
3. Ask the user to enter information about id,name,branch,year and stored to –> yr,sname,branch,sid and Enter the marks of 6 subject and calculate the average
4. Declare a member function void display () to display all the details and compute average mark of each student

If the average of marks is >= 80 then prints Grade ‘A’

If the average is <80 and >=60 then prints Grade ‘B’

If the average is <60 and >=40 then prints Grade ‘C’

else prints Grade ‘D’

1. declare the main method and Ask the user to how many students information you want enter that stored to –> n, initialize a object and call the member function.

Student\_Details s[]=new Student\_Details[n];

1. Campare the average mark of each students: if(s[i].avg<s[i-1].avg)

max1=s[i].avg;maxi1=i;

else max1=s[i-1].avg;maxi1=i-1;

max2=max1;maxi2=maxi1;

1. Print first and second position of student
2. Stop

**Program:**

**Student\_Details.java**

import java.util.Scanner;

import java.util.Arrays;

public class Student\_Details

{

int tot,i,avg,yr,sid;

String sname,branch;

int marks[]= new int[6];

Scanner s = new Scanner(System.in);

public void getdata()

{

System.out.println("enter the student id : ");

sid=s.nextInt();

System.out.println("enter the student name : ");

sname=s.next();

System.out.println("enter the student branch : ");

branch=s.next();

System.out.println("enter the student year : ");

yr=s.nextInt();

for(i=0; i<6; i++)

{

System.out.print("enter marks of subject"+(i+1)+" : ");

marks[i]=s.nextInt();

tot=tot+marks[i];

}

avg=tot/6;

}

public void display()

{

System.out.println("\nstudent id : " +sid);

System.out.println("student name : " +sname);

System.out.println("student branch : " +branch);

System.out.println("student yr : " +yr);

if(avg>=80)

{

System.out.print("\nstudent grade is A.\n");

}

else if(avg>=60 && avg<80)

{

System.out.print("\nstudent grade is B.\n");

}

else if(avg>=40 && avg<60)

{

System.out.print("\nstudent grade is C.\n");

}

else

{

System.out.print("\nstudent grade is D.\n");

}

}

public static void main(String args[])

{

int n=0,maxi1=0,maxi2=0;

float max1=0,max2=0;

Scanner s1 = new Scanner(System.in);

System.out.print("enter the limit : ");

n=s1.nextInt();

Student\_Details s[]=new Student\_Details[n];

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

{

s[i]=new Student\_Details();

s[i].getdata();

}

System.out.println("\n\*\*\*STUDENT DETAILS\*\*\*");

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

{

s[i].display();

}

for(int i=1;i<n;i++)

{

if(s[i].avg<s[i-1].avg)

{

max1=s[i].avg;

maxi1=i;

}

else

{

max1=s[i-1].avg;

maxi1=i-1;

}

max2=max1;

maxi2=maxi1;

}

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

{

if(max1<=s[i].avg)

{

max1=s[i].avg;

maxi1=i;

}

}

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

{

if(max2<=s[i].avg && max1!=s[i].avg)

{

max2=s[i].avg;

maxi2=i;

}

}

System.out.println("\n\*\*\*FIRST POSITION\*\*\*");

s[maxi1].display();

System.out.println("average marks : "+max1);

System.out.println("\n\*\*\*SECOND POSITION\*\*\*");

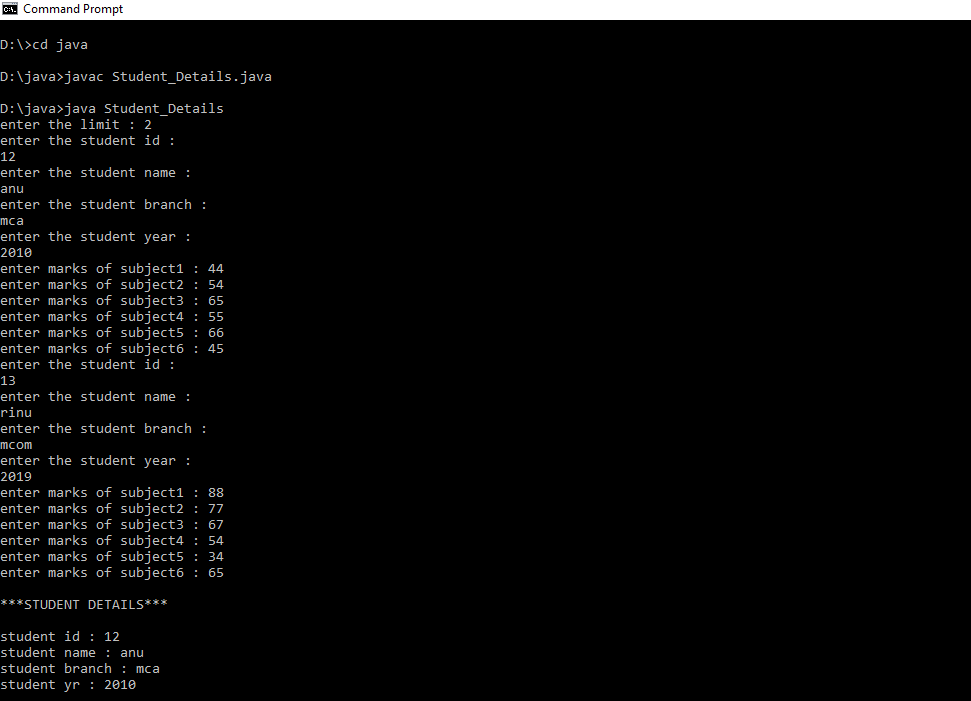
s[maxi2].display();

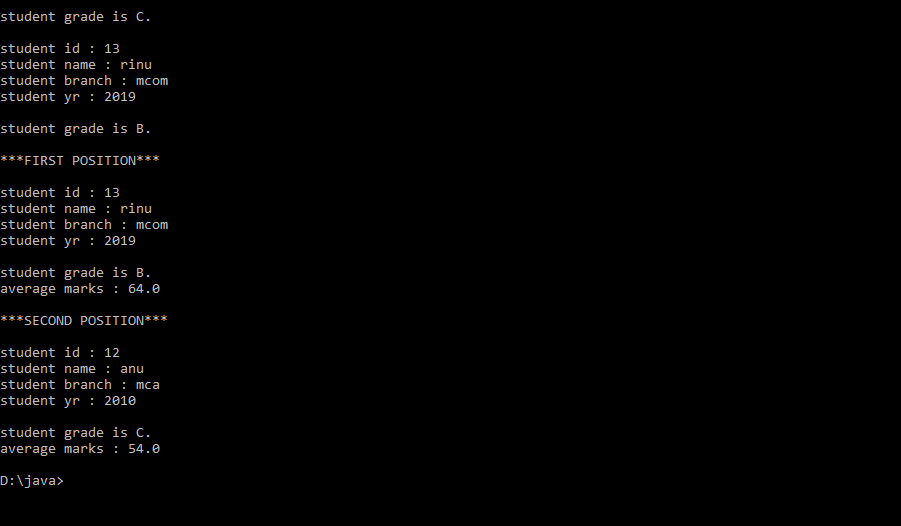
System.out.println("average marks : "+max2);

}

}

**Expected Output**





**Experiment 6: Design a class to represent a**  **bank account** **details.**

**Aim:** Design a class to represent a bank account which include account number, name of the depositor, type of the account and balance amount in the account. Define Methods, to assign initial values, to Deposit an amount, to Withdraw amount after checking balance, to display name and balance.

**Algorithm**

1. Start.
2. Declare a class Bank and initialize the variable int accno;String name,type;float amount;
3. Declare a member functions

void details(int n,String nm,String t,float a)

void show()

void deposit(float a)

void withdraw(float a)

void checkbalance()

1. Declare a class Customer\_Bank and initialize a object b
2. Calling the function by using the object.
3. Stop

**Program:**

**Customer\_Bank.java**

import java.util.\*;

class Bank{

int accno;

String name,type;

float amount;

void details(int n,String nm,String t,float a)

{

accno=n;

name=nm;

type=t;

amount=a;

}

void show()

{

System.out.println("Account number of the customer:"+accno);

System.out.println("Customer Name:"+name);

System.out.println("Account Type:"+type);

System.out.println("Balance Amount:" +amount);

}

void deposit(float a)

{

amount=amount+a;

System.out.println("deposit amount:"+amount);

}

void withdraw(float a)

{

if(amount<a)

System.out.println("insufficentt balance");

else

amount=amount-a;

System.out.println("withdraw amount is:"+a);

}

void checkbalance()

{

System.out.println("balance is:"+amount);

}

}

class Customer\_Bank{

public static void main(String args[])

{

Bank b =new Bank();

b.details(123456,"anu","savings",2000);

b.show();

b.deposit(40000);

b.checkbalance();

b.withdraw(10000);

b.checkbalance();

}

}

**Expected Output**

