**JAVA LABCYCLE-1**

**JAVA LABCYCLES-1**

## Experiment 1: Design and use ‘Array’

**AIM**

Java Program to Accept the Marks of a Student into a 1D Array and find total marks and percentage

**ALGORITHM**

1.start

2. Define class Studentstub with members sub1,sub2,sub3,sub4 and sub5(float),total(double) and percentage(float).

3. Get the data from the user marks secured in each out of 100 for 5 subject.

4. Store each marks in an 1D array .

5. Do the summation and return the value .

6.stop

**PROGRAM**

import java.util.Scanner;

class StudentSub

{

public static void main(String args[])

{

float sub1,sub2,sub3,sub4,sub5;

double total, percentage;

Scanner op=new Scanner(System.in);

System.out.println("Enter marks of five subjects");

System.out.print("Enter marks of subject1:");

sub1=op.nextFloat();

System.out.print("Enter marks of subject2:");

sub2=op.nextFloat();

System.out.print("Enter marks of subject3:");

sub3=op.nextFloat();

System.out.print("Enter marks of subject4:");

sub4=op.nextFloat();

System.out.print("Enter marks of subject5:");

sub5=op.nextFloat();

total = sub1+sub2+sub3+sub4+sub5;

percentage = (total / 500) \* 100;

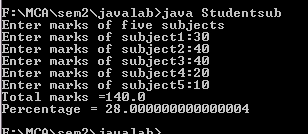
System.out.println("Total marks ="+total);

System.out.println("Percentage = "+percentage);

}

}

OUTPUT



## Experiment 2: Count the Number of Occurrencein an Array

**AIM**

Java Program to Count the Number of Occurrence of an Element in an Array

**ALGORITHM**

1. Start
2. Declare an array, initialize n, x, count = 0, i = 0
3. Ask the user to initialize the array
4. enter all elements in the array
5. for(i=0;i<n;i++) : a[i] store the elements In the array
6. Ask the user to initialize count number of occurrences
7. x = s.nextInt() store the count
8. for(i = 0; i< n; i++) if(a[i] == x)
9. count++;
10. print Number of Occurrence of the Element:"+count
11. Stop

**PROGRAM**

import java.util.Scanner;

public class Countocc

{

public static void main(String[] args)

{

int n, x, count = 0, i = 0;

Scanner s = new Scanner(System.in);

System.out.print("Enter no. of elements you want in array:");

n = s.nextInt();

int a[] = new int[n];

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

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

{

a[i] = s.nextInt();

}

System.out.print("Enter the element of which you want to count number of occurrences:");

x = s.nextInt();

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

{

if(a[i] == x)

{

count++;

}

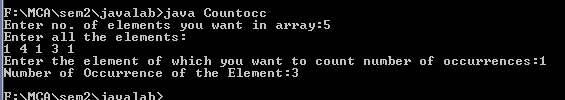
}

System.out.println("Number of Occurrence of the Element:"+count);

}

}

OUTPUT



## Experiment 3: Matrix Addition

**AIM**

Java Program to Add Two MXN Matrix from User Input

**ALGORITHM**

1. Start
2. declare two   matrices are of the same size.
3. Read row number,column number and initialize the  double dimensional arrays a[][],b[][],c[][] with same row number,column number.
4. Store the first matrix elements into the two-dimensional array a[][] using two for loops. i indicates row number, j indicates column index.Similarly matrix 2 elements in to b[][].
5. Add the two matrices using for loop

for i=0 to i<row

for j=0 to j<col

1. a[i][j] + b[i][j] and store it in to the matrix at c[i][j] .

c[i][j]=a[i][j]+b[i][j];

1. stop.

**PROGRAM**

import java.util.Scanner;

public class Addmatrix

{

public static void main(String[] args)

{

int p, q, m, n;

Scanner s = new Scanner(System.in);

System.out.print("Enter number of rows and columns in first matrix:");

p = s.nextInt();

q = s.nextInt();

System.out.print("Enter number of rows and columns in second matrix:");

m = s.nextInt();

n = s.nextInt();

if (p == m && q == n)

{

int a[][] = new int[p][q];

int b[][] = new int[m][n];

int c[][] = new int[m][n];

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

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

{

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

{

a[i][j] = s.nextInt();

}

}

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

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

{

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

{

b[i][j] = s.nextInt();

}

}

System.out.println("First Matrix:");

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

{

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

{

System.out.print(a[i][j]+" ");

}

System.out.println("");

} System.out.println("Second Matrix:");

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

{

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

{

System.out.print(b[i][j]+" ");

}

System.out.println("");

}

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

{

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

{

for (int k = 0; k < q; k++)

{

c[i][j] = a[i][j] + b[i][j];

}

}

}

System.out.println("Matrix after addition:");

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

{

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

{

System.out.print(c[i][j]+" ");

}

System.out.println("");

}

}

else

{

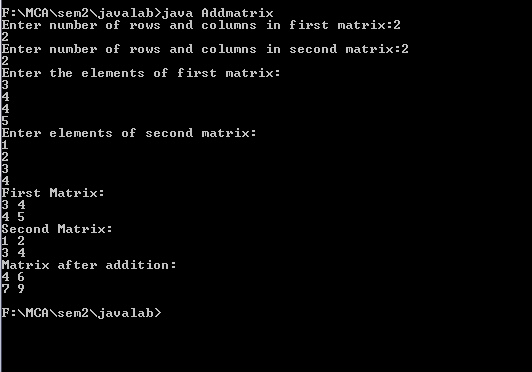
System.out.println("Addition would not be possible");

}

}

}

**OUTPUT**



## Experiment 4: Add Complex Numbers

**AIM**

Add complex numbers

**ALGORITHM**

1. Start

2. Define class AddComplex

3. Define the real and imaginary part of a complex number

4. Define a constructor to initialize real and img.

5. Create a temporary complex number to hold the sum of two numbers.

6. Do the Addition.

7. Return the result.

8. Stop

**PROGRAM**

import java.util.Scanner;

class Complex

{

float real1,imag1,real2,imag2;

void get()

{

Scanner op=new Scanner(System.in);

System.out.print("\n Enter the real part of first complex number: ");

real1=op.nextFloat();

System.out.print(" Enter the imaginary part of first complex number: ");

imag1=op.nextFloat();

System.out.print("\n Enter the real part of second complex number: ");

real2=op.nextFloat();

System.out.print(" Enter the imaginary part of second complex number : ");

imag2=op.nextFloat();

}

void display()

{

System.out.println(" \n Sum of complex numbers " + real1 + "+" + imag1 + "i and " + real2 + "+" + imag2 + "i is " + (real1+real2) + "+" + (imag1+imag2) + "i");

}

public static void main(String args[])

{

Complex cmp = new Complex();

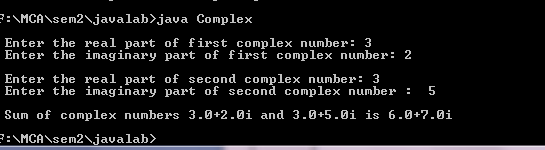
cmp.get();

cmp.display();

}

}

**OUTPUT**



**Experiment 5: Symmetric matrix**

**AIM**

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

**ALGORITHM**

1. Start
2. Declare array mat[][].
3. Read row number, column number and initialize row,col
4. Store the matrix elements into the two-dimensional array mat[i][j] using two for loops. i indicates row number, j indicates column index.
5. if(rows != cols) then the given matrix is not a square matrix, so it can't be symmetric.
6. Find transpose of the matrix
7. Compare two matrices.
8. If the two matrices is same then it is symmetric otherwise it's not.
9. Stop

**PROGRAM**

import java.util.Scanner;

class Symm{

public static void main(String args[]){

Scanner input = new Scanner(System.in);

System.out.println("Number of rows of Matrix:");

int m = input.nextInt();

System.out.println("Number of columns of Matrix:");

int n = input.nextInt();

int a[][] = new int[m][n];

System.out.println("Enter elements of The Matrix:");

for(int i =0;i < m;i++){

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

a[i][j] = input.nextInt();

}

}

if(m != n)

System.out.println("Matrix is not a Squre

matrix");

else{

int b[][] = new int[n][m];

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

for(int j = 0;j < m; j++){

b[i][j] = a[j][i];

}

}

int flag =0;

first:

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

for(int j = 0; j < m;j++){

if(a[i][j] != b[i][j]){

flag =1;

break first;

}

}

}

if(flag == 1)

System.out.println("Matrix is not

Symmetric");

else

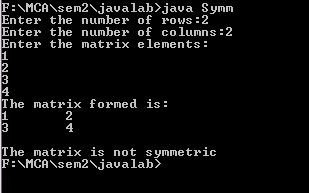
System.out.println("Matrix is Symmetric");

}

}

}

**OUTPUT**



## Experiment 6: Design and use ‘Product’ class

**AIM**

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

**ALGORITHM**

1. start
2. declare the class product
3. initialize data member pcode=0,pname=null,price=0 in default constructor product.
4. declare a member function void cal and compare the lowest price of product.
5. create 3 object in class product
6. Ask the user to enter 3 pcode and pname,price and stored using object of class product
7. performing comparison and print the output.
8. stop

**PROGRAM**

import java.util.\*;

public class Product

{

int pcode;

String pname;

int price;

Product()

{

pcode=0;

pname=null;

price=0;

}

public static void cal(int a,int b,int c)

{

int p1=a;

int p2=b;

int p3=c;

float lowest;

if(p1<p2)

{

if(p3<p1)

{

lowest = p3;

}

else

{

lowest = p1;

}

}

else

{

if(p2<p3)

{

lowest = p2;

}

else

{

lowest = p3;

}

}

System.out.println("The lowest price among the 3 Product is : "+lowest);

}

public static void main(String args[])

{

Scanner s = new Scanner(System.in);

Product p1= new Product();

Product p2= new Product();

Product p3= new Product();

System.out.print("Enter the Product 1 Code : ");

p1.pcode =s.nextInt();

System.out.print("Enter the Product 1 Name : ");

p1.pname = s.next();

System.out.print("Enter the Product 1 Price : ");

p1.price=s.nextInt();

System.out.print("Enter the product 2 Code : ");

p2.pcode =s.nextInt();

System.out.print("Enter the Product 2 Name : ");

p2.pname = s.next();

System.out.print("Enter the Product 2 Price : ");

p2.price = s.nextInt();

System.out.print("Enter the Product 3 Code : ");

p3.pcode =s.nextInt();

System.out.print("Enter the Product 3 Name : ");

p3.pname = s.next();

System.out.print("Enter the Product 3 Price : ");

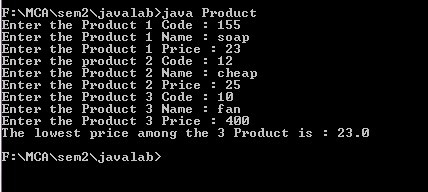
p3.price = s.nextInt();

Product.cal(p1.price,p2.price,p3.price);

}

}

**OUTPUT**



**Experiment 7: CPU and its details - Using Inner classes**

**AIM**

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

**ALGORITHM**

1. start
2. declare the class CPU
3. initialize data price in the class CPU.
4. declare the inner class Processor with data member no. of cores, manufacturer
5. declare a member function getCache in class Processor.
6. declare the static nested class RAM with data member memory, manufacturer
7. declare a member function getClockSpeed.
8. Create an object of CPU : CPU cpu = new CPU();
9. CPU.Processor processor = cpu.newProcessor();

CPU.RAM ram = cpu.newRAM();

1. print information of Processor and RAM.
2. stop

**PROGRAM**

class CPU

{

double price;

class Processor

{

double cores;

String manufacturer;

double getCache()

{

return 4.3;

}

}

protected class RAM

{

double memory;

String manufacturer;

double getClockSpeed()

{

return 5.5;

}

}

}

public class Main

{

public static void main(String[] args)

{

CPU cpu = new CPU();

CPU.Processor processor = cpu.new Processor();

CPU.RAM ram = cpu.new RAM();

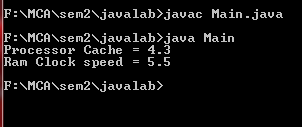
System.out.println("Processor Cache = " + processor.getCache());

System.out.println("Ram Clock speed = " + ram.getClockSpeed());

}

}

**OUTPUT**

****

**JAVA LABCYCLE-2**

**Experiment 1: Sort strings**

**AIM**

Program to Sort strings don’t use sort () method

**ALGORITHM**

**PROGRAM**

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++)

{

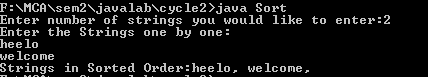
System.out.print(str[i] + ", ");

}

}

}

**OUTPUT**

****

**Experiment 2: Search element from an array**

**AIM**

Search an element in an 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

7) 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;

8) Print the output

9) stop

**PROGRAM**

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;

}

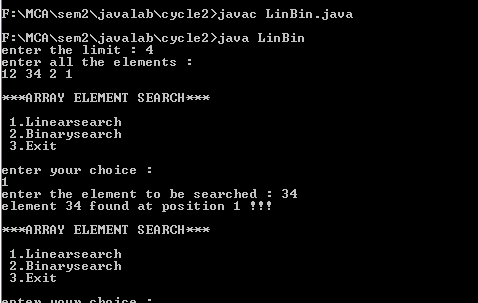
}

return -1;

}

}

**OUTPUT**

****

**Experiment 3: string manipulations**

**AIM**

. 3. 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 valueOf()

**ALGORITHM**

1. Start.
2. Declare an array char ch[].
3. Declare the String s1.
4. String s2=new String(ch); String s3=new String("example");
5. print string s1,s2,s3.
6. Stop

**PROGRAM**

public class Stringelement

{

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("objectoriented");

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');

System.out.println(replaceString);

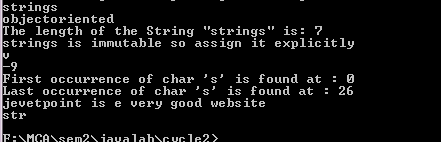
String sample = String.valueOf(ch, 0, 3);

System.out.println(sample);

}

}

**OUTPUT**

****

**Experiment 4: Array of objects**

**AIM**

Program to create a class for Employee having attributes eNo, eNameeSalary. 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**

import java.util.Scanner;

public class Employee

{

public static void main(String[] args)

{

int count,flag = 0, i;

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 eSalary[] = new int[count];

int eNo[] = new int[count];

Scanner scan2 = new Scanner(System.in);

Scanner scan3 = new Scanner(System.in);

Scanner scan4 = new Scanner(System.in);

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

{

System.out.println("Enter the eNo of Employee "+(i+1)+

":");

eNo[i] = scan2.nextInt();

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

eName[i] = scan3.nextLine();

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

eSalary[i] = scan4.nextInt();

}

Scanner s = new Scanner(System.in);

System.out.print("Enter the employee number you want to

find:");

int x = s.nextInt();

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

{

if(eNo[i] == x)

{

flag = 1;

break;

}

}

if(flag == 1)

{

int t = i;

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

System.out.println("Employee name : "+eName[t]);

System.out.println("Employee salary : "+eSalary[t]);

}

else

{

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

}

scan.close();

scan2.close();

scan3.close();

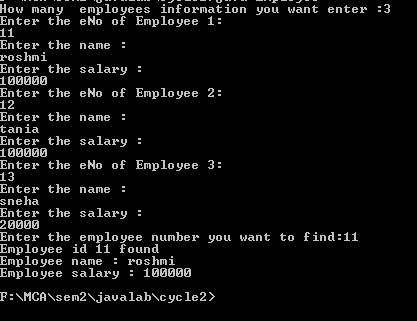
scan4.close();

s.close();

}

}

**OUTPUT**

****

**Experiment 5: Design a class represent 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.

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

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

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

d. else prints Grade ‘D’

**ALGORITHM**

1) Start.

2) Declare a class Student 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’

5) 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];

6) 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;

7) Print first and second position of student

8) Stop

**PROGRAM**

import java.util.\*;

class Student

{

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("\nenter 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 s[]=new Student[n];

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

{

s[i]=new Student();

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)

{

maxi2=i;

}

}

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

s[maxi1].display();

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

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

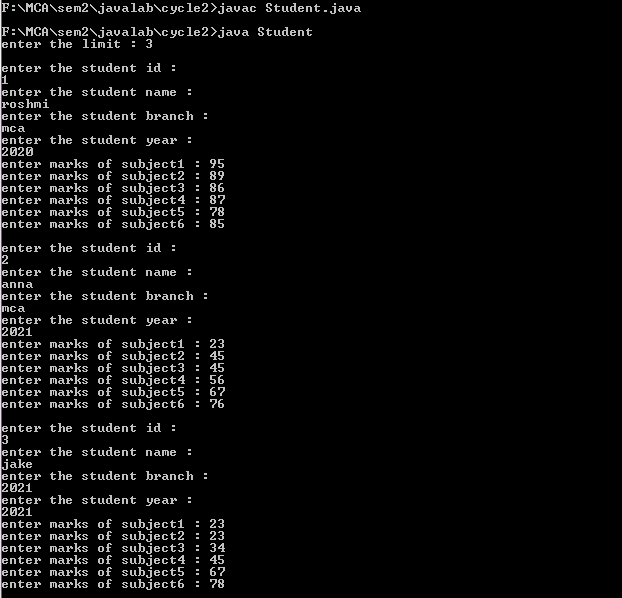
s[maxi2].display();

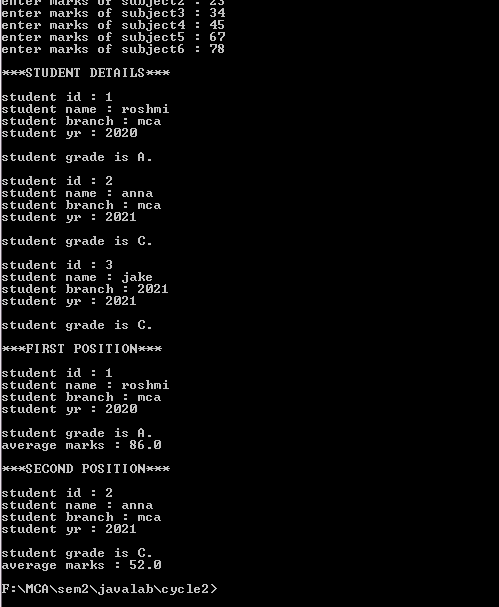
System.out.println("average marks : "+max2);

}

}

**OUTPUT**

****

****

**Experiment 6: Design a class to represent bank object**

**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. Create a class Bank with members accno, name, type and amount.

3. Define method details() to assign initial values.show() to display account details.deposit() to display deposit amount.withdraw() to display withdraw amount.checkbalance() to display balance amount.

4. Create class Customer\_Bank.

5. Define main(String String args[])

6. Create object of the class Bank

7. Call various methods to display account details.

8. Stop

**PROGRAM**

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 Customerbank{

public static void main(String args[])

{

Bank b =new Bank();

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

b.show();

b.deposit(40000);

b.checkbalance();

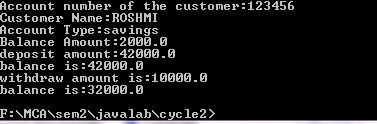
b.withdraw(10000);

b.checkbalance();

}

}

**OUTPUT**

****

**JAVA LABCYCLE-3**

2.Create a class ‘Employee’ with data members Empid, Name, Salary, Address andconstructors to initialize the data members. Create another class ‘Teacher’ that inherit theproperties of class employee and contain its own data members department, Subjects taughtand constructors to initialize these data members and also include display function todisplay all the data members. Use array of objects to display details of N teachers.

**PROGRAM**

import java.util.Scanner;

class EmployeeT {

int empid;

String name;

float salary;

String address;

EmployeeT() {

}

EmployeeT(int empid, String name, float salary, String address) {

this.empid = empid;

this.name = name;

this.salary = salary;

this.address = address;

}

}

class Teacher extends EmployeeT {

String department, subject;

Teacher(int empid, String name, float salary, String address,

String department, String subject) {

super(empid, name, salary, address);

this.department = department;

this.subject = subject;

}

public void display() {

System.out.println(&quot;Teacher\_id : &quot;+empid);

System.out.println(&quot;Teacher\_name : &quot;+name);

System.out.println(&quot;Teacher\_salary : &quot;+salary);

System.out.println(&quot;Teacher\_address : &quot;+address);

System.out.println(&quot;Teacher\_department : &quot;+department);

System.out.println(&quot;Teacher\_subject : &quot;+subject);

}

}

public class TeacherArrObjects {

public static void main(String[] args) {

System.out.println(&quot;Enter the number of Teachers:&quot;);

Scanner sc = new Scanner(System.in);

int n = sc.nextInt();

System.out.println(&quot;Enter Teacher Deatils one by one:&quot;);

Teacher teacher[] = new Teacher[n];

Scanner scT = new Scanner(System.in);

int tid;

String name;

float salary;

String address;

String department;

String subject;

for (int i = 0; i &lt; n; i++) {

System.out.println(&quot;Enter &quot; + i + &quot;teacher details :&quot;);

System.out.println(&quot;Enter teacher\_id(integer) :&quot;);

tid = scT.nextInt();

System.out.println(&quot;Enter teacher\_name(String) :&quot;);

name = scT.next();

System.out.println(&quot;Enter teacher\_salary(float) :&quot;);

salary = scT.nextFloat();

System.out.println(&quot;Enter teacher\_address(String) :&quot;);

address = scT.next();

System.out.println(&quot;Enter teacher\_department(String) :&quot;);

department = scT.next();

System.out.println(&quot;Enter teacher\_subject(String) :&quot;);

subject = scT.next();

Teacher t = new Teacher(tid, name, salary, address, department, subject);

teacher[i] = t;

}

System.out.println(&quot;Teachers are : \n&quot;);

for (Teacher x : teacher) {

x.display();

System.out.println(&quot;\n&quot;);

}

}

}

**Output**

Enter number o f Te ache r s :

3

Enter Teacher d e t a i l s one by one . .

Enter 0 t e a c h e r d e t a i l s . .

Enter t e a c h e r i d ( i n t e g e r ) :

1

Enter t e a c h e r name ( S t ri n g ) :

abc

Enter t e a c h e r s a l a r y ( f l o a t ) :

25000

Enter t e a c h e r a d d r e s s ( S t ri n g ) :

k e r al a

Enter t e a c h e r department ( S t ri n g ) :

computer

Enter t e a c h e r s u b j e c t ( S t ri n g ) :

j a v a

Enter 1 t e a c h e r d e t a i l s . .

Enter t e a c h e r i d ( i n t e g e r ) :

2

Enter t e a c h e r name ( S t ri n g ) :

d e f

Enter t e a c h e r s a l a r y ( f l o a t ) :

25000

Enter t e a c h e r a d d r e s s ( S t ri n g ) :

d e l h i

Enter t e a c h e r department ( S t ri n g ) :

maths

Enter t e a c h e r s u b j e c t ( S t ri n g ) :

c a l c u l u s

Enter 2 t e a c h e r d e t a i l s . .

Enter t e a c h e r i d ( i n t e g e r ) :

3

Enter t e a c h e r name ( S t ri n g ) :

g hi

Enter t e a c h e r s a l a r y ( f l o a t ) :

5.3. EXPERIMENT 12: MULTI-LEVEL INHERITANCE 49

25000

Enter t e a c h e r a d d r e s s ( S t ri n g ) :

chenn ai

Enter t e a c h e r department ( S t ri n g ) :

p h y si c s

Enter t e a c h e r s u b j e c t ( S t ri n g ) :

o p t i c s

Te ache r s a r e :

Teacher i d : 1

Teacher name : abc

Teacher s a l a r y : 2 5 0 0 0. 0

Teacher a d d r e s s : k e r al a

Teacher department : computer

Teacher s u b j e c t : j a v a

Teacher i d : 2

Teacher name : d e f

Teacher s a l a r y : 2 5 0 0 0. 0

Teacher a d d r e s s : d e l h i

Teacher department : maths

Teacher subject : calculus

Teacher id : 3

Teacher name : ghi

Teacher salary : 2 5 0 0 0. 0

Teacher address : chennai

Teacher department : physics

Teacher subject : optics

3. Create a class ‘Person’ with data members Name, Gender, Address, Age and a constructorto initialize the data members and another class ‘Employee’ that inherits the properties ofclass Person and also contains its own data members like Empid, Company\_name,Qualification, Salary and its own constructor. Create another class ‘Teacher’ that inherits

the properties of class Employee and contains its own data members like Subject,Department, Teacherid and also contain constructors and methods to display the datamembers. Use array of objects to display details of N teachers.

**Program**

import java . util . Scanner ;

class Person{

String Name,

Gender ,

Address ;

protected int Age; public Person (){} public Person ( String n, String g , String addr , int a){ this .Name = n;

this . Gender = g ; this . Address =addr ;

this .Age = a ;

}

public void displayPerson (){

System.out.println(&quot;Name :&quot; + Name);

System.out.println(&quot;Gender :&quot; + Gender);

System.out.println(&quot;Address :&quot; + Address);

System.out.println(&quot;Age :&quot; + Age);

}

}

class Employee extends Person{ int Empid, Salary ;

String Companyname ,

Qualification ;

public Employee(){

} public Employee( String n, String g , String addr , int a ,

int eid , String cname , String qual , int sal ){

super(n, g , addr , a );

Empid = eid ;

Companyname = cname ;

Qualification = qual ;

Salary = sal ;

}

public void displayEmployee (){ super . displayPerson ();

System.out.println(&quot;Empid :&quot; + Empid);

System.out.println(&quot;Companyname :&quot; + Companyname);

System.out.println(&quot;Qualification :&quot; + Qualification);

System.out.println(&quot;Salary :&quot; + Salary);

}

}

class Teacher1 extends Employee{

String Subject , Department ; int Teacherid ;

public Teacher1( String n, String g , String addr , int a , int eid , String cname , String qual , int sal ,

String sub , String dept , int tid ){ super(n, g , addr , a , eid , cname , qual , sal ); Subject = sub ;

Department = dept ;

Teacherid = tid ;

}

public void displayTeacher (){ super . displayEmployee ();

System.out.println(&quot;Teacherid :&quot; + Teacherid);

System.out.println(&quot;Subject :&quot; + Subject);

System.out.println(&quot;Department :&quot; + Department);

}

}

public class InheritancePersonExample {

public static void main( String args []){

System.out.println(&quot;Enter the number of Teachers&quot;);

Scanner sc = new Scanner (System . in );

int N = sc . nextInt ();

Teacher1 [ ] teacher1s = new Teacher1 [N] ; Scanner scs = new Scanner (System . in ); for ( int i = 0; i&lt;N;

i++){

System.out.println(&quot;Enter the name of Teachers&quot;);

String name = scs . nextLine ();

System.out.println(&quot;Enter the gender of the Teachers&quot;);

String gen = scs . nextLine ();

System.out.println(&quot;Enter the address of the Teachers&quot;);

String addr = scs . nextLine ();

System.out.println(&quot;Enter the age of the Teachers&quot;);

int ag = sc . nextInt ();

System.out.println(&quot;Enter the Empid of the Teachers&quot;);

int eid = sc . nextInt ();

System.out.println(&quot;Enter the Company name &quot;);

String cn = scs . nextLine ();

System.out.println(&quot;Enter the Qualification of the Teachers&quot;);

String quali = scs . nextLine ();

System.out.println(&quot;Enter the Salary of the Teachers&quot;);

int sal = sc . nextInt ();

System.out.println(&quot;Enter the Teacher id&quot;);

int tid = sc . nextInt ();

System.out.println(&quot;Enter the Subject of the Teachers&quot;);

String sub = scs . nextLine ();

System.out.println(&quot;Enter the Department of the Teachers&quot;);

String dept = scs . nextLine ();

Teacher1 t = new Teacher1(name,gen,addr,ag,eid,cn,quali,sal,sub,dept,tid );

teacher1s [ i ] = t ;

}

for (Teacher1 t :teacher1s ){

t.displayTeacher ();

}

}

}

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

**Program**

class Publisher{

String publisher;

Publisher(String publi){

this.publisher= publi;

}

}

class Book {

String name;

Publisher publisher;

Book(){}

public Book(String name,Publisher publisher){

this.name=name;

this.publisher=publisher;

}

}

class Literature extends Book

{

String Lit\_type =&quot;Literature&quot;;

Literature(String name,Publisher publisher){

super(name,publisher);

}

void display(){

System.out.println(&quot;Name: &quot;+ super.name);

System.out.println(&quot;Type: &quot;+ this.Lit\_type);

System.out.println(&quot;Publisher: &quot;+ this.publisher.publisher);

}

}

class Fiction extends Book {

String Lit\_type =&quot;Fiction&quot;;

Fiction( String name , Publisher publisher ) {

super ( name , publisher);

}

void display(){

System.out.println(&quot;Name: &quot;+ super.name);

System.out.println(&quot;Type: &quot;+ this.Lit\_type);

System.out.println(&quot;Publisher: &quot;+ this.publisher.publisher);

}

}

public class InheritanceBookExample{

public static void main(String [] args){

Publisher lp = new Publisher (&quot;S.chand&quot;) ;

Literature l = new Literature(&quot;As you like it&quot;,lp);

l.display();

Publisher fp = new Publisher (&quot;Tata MCGraw Hill&quot;);

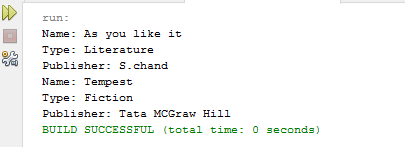
Fiction f = new Fiction(&quot;Tempest&quot;,fp);

f.display();

}

}

**Output**



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

**Program**

interface Student{

    int score = 10;

    void displayScore ();

}

interface Sports{

    int score =25;

    void displaySportScore ();

}

class Result implements Student , Sports{

    public void displayScore (){

    System.out.println("Academic score is :" + Student . score);

}

public void displaySportScore (){

    System.out.println("Sport score is :" + Sports . score);

}

}

public class SportStudentResult {

    public static void main( String [ ] args ){

        Result r = new Result ();

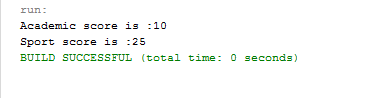
        r . displayScore ();

        r . displaySportScore ();

}

}

Output



6. Create an interface having prototypes of functions area() and perimeter(). Create twoclasses Circle and Rectangle which implements the above interface. Create a menu drivenprogram to find area and perimeter of objects.

**Program**

import java . util . Scanner ;

interface AP

{

void input ( ) ;

void area ( ) ;

void perimeter () ;

}

class Circle implements AP{

int r = 0 ;

double pi = 3.14 , area = 0 , perimeter = 0 ;

public void input ( ) {

Scanner c = new Scanner ( System . in ) ;

r = c.nextInt( );

}

public void area ( ) {

area = pi \* r \* r ;

System.out.println ( "Area of circle : " + area ) ; }

public void perimeter ( ) {

perimeter = 2 \* pi \* r ;

System.out.println ( "Perimeter of circle : " + perimeter ) ; }}

class Rectangle implements AP{

int l = 0 , b = 0 ;

double area , perimeter ;

public void input ( ) {

Scanner r = new Scanner ( System . in ) ;

l =r.nextInt ( ) ;

b =r.nextInt ( ) ;

}

public void area ( ) {

area = l \* b ;

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

}

public void perimeter ( ) {

perimeter = 2 \* ( l+b ) ;

System.out.println ( "Perimeter of Rectangle : " + perimeter ) ;

}

}

public class IFaceDemo{

public static void main ( String [ ] args ) {

int ch ;

Circle c = new Circle ( ) ;

Rectangle r = new Rectangle ( ) ;

Scanner sc = new Scanner ( System.in ) ;

System.out.println ( " 1 : Area of Circle " ) ;

System.out.println ( " 2 : Area o f Rectangle " ) ;

System.out.println ( " 3 : Perimeter o f Circle " ) ;

System.out.println ( " 4 : Perimeter o f Rectangle " ) ;

System.out.println ( " 5 : Program termination " ) ;

lp : while ( true ) {

System.out.println( "Make your choice : " ) ;

ch = sc.nextInt ( ) ;

switch ( ch ){

case 1 :

System.out.println ( "Enter the radius of the Circle \n" ) ;

c . input ( ) ;

c . area ( ) ;

break ;

case 2 :

System.out.println ( "Enter the length and breadth of the Rectangle\n") ;

 r . input ( ) ;

r.area ( ) ;

break ;

case 3 :

System.out.print ( "Enter the radius of the Circle \n" ) ;

c . input ( ) ;

c . perimeter ( ) ;

break ;

case 4 :

System.out.println ( "Enter the length and breadth of the Rectangle ");

 r . input ( );

r . perimeter ( ) ;

break ;

case 5 :

break lp ;

default :

System.out.println ( " Invalid choice ! Please make a valid choice") ;

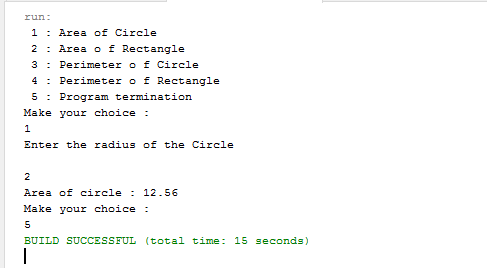
}

}

}

}

**Output**



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

Order No.

Date :

Product Id     Name    Quantity    unit price          Total

101                     A   2   25       50

102       B         1         100                 100

Net. Amount 150

**Program**

import java.util.\*;

interface BillGen{

    int calculate();

}

class ProductB implements BillGen {

String name ;

int prodid , quantity , unit\_price , total ;

ProductB( ) { }

ProductB(String n , int p , int q , int u ){

name = n ;

prodid = p ;

quantity = q ;

unit\_price = u ;

}

public int calculate( ) {

 total = quantity \* unit\_price;

 return total ;

}

}

public  class  Bill {

    public static void main ( String[ ] args ) {

      ProductB [ ][ ] order ;

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

      Scanner sc = new Scanner (System.in);

      int n = sc.nextInt ( ) ;

      order = new ProductB [n][ ] ;

      for ( int i = 0; i < n ; i ++)

{

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

int m = sc.nextInt( );

order[i] = new ProductB [m] ;

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

{

System.out.println("Enter Product" + j +"Name:");

String a = sc.next( );

System.out.println("Enter Product ID : ");

int b = sc.nextInt( );

System.out.println("Enter Product Quantity : ");

int c = sc.nextInt( );

System.out.println("Enter Product Unit Price : ");

int d = sc. nextInt( );

ProductB pb = new ProductB (a,b,c,d);

order[i][j] = pb ;

order[i][j].total = order[i][j].calculate( );

}

}

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

{

int sum = 0 ;

System.out.println("Order No :"+(i+1));

Date date=java.util.Calendar.getInstance( ).getTime( );

System.out.println(date);

System.out.println("-------------------------------------");

System.out.printf("%5s %20s %25s %10s %10s","Prod\_Id","Name","Quantity","Unit Price","Total");

System.out.println();

System.out.println("-------------------------------------");

for (int j =0; j<order[i].length; j++)

{

System.out.printf("%5s %20s %25s %10s %10s",

order[i][j].prodid,

order[i][j].name,

order[i][j].quantity,

order[i][j].unit\_price,

order[i][j].total);

System.out.println();

}

System.out.println("-------------------------------------");

for (int k=0; k<order[i].length; k++)

sum = sum + order[i][k].total;

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

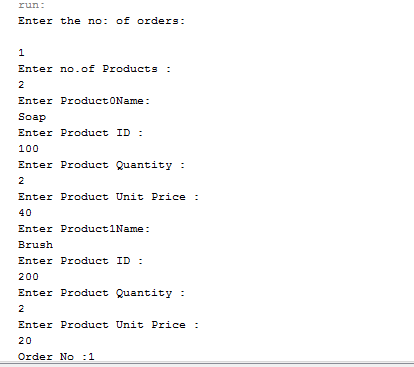
System.out.println("-------------------------------------");

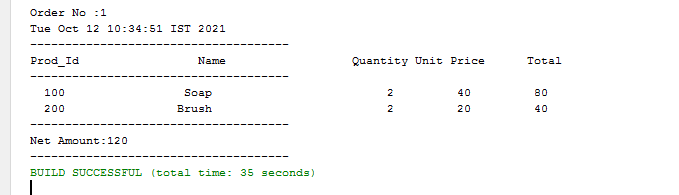
    }

}

}

**Output**





**JAVA LABCYCLE-5**

**Cycle5**

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

**Program**

import java.applet.Applet;

import java.awt.\*;

public class shape extends Applet{

public void paint(Graphics g){

g.setColor(Color.red);

g.drawLine(20,30,20,300);

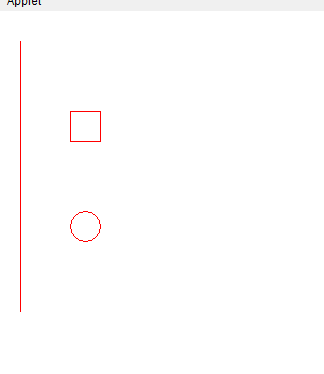
g.drawRect(70,100,30,30);

g.drawOval(70,200,30,30);

}

}

**Output**



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

**Program**

private void button1ActionPerformed(java.awt.event.ActionEvent evt) {

int a,b,c,result;

a=Integer.parseInt(textField1.getText());

b=Integer.parseInt(textField2.getText());

c=Integer.parseInt(textField3.getText());

if (a>b) {

if (a>c)

result=a;

else

result=c;

}

else{

if (b>c)

result=b;

else

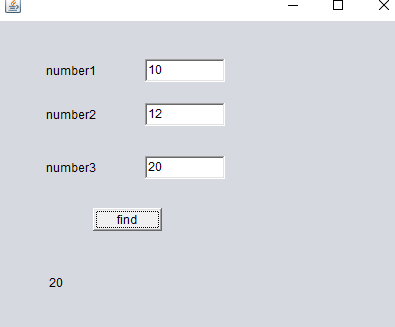
result=c;

}

label4.setText(" " +result);

}

**Output**

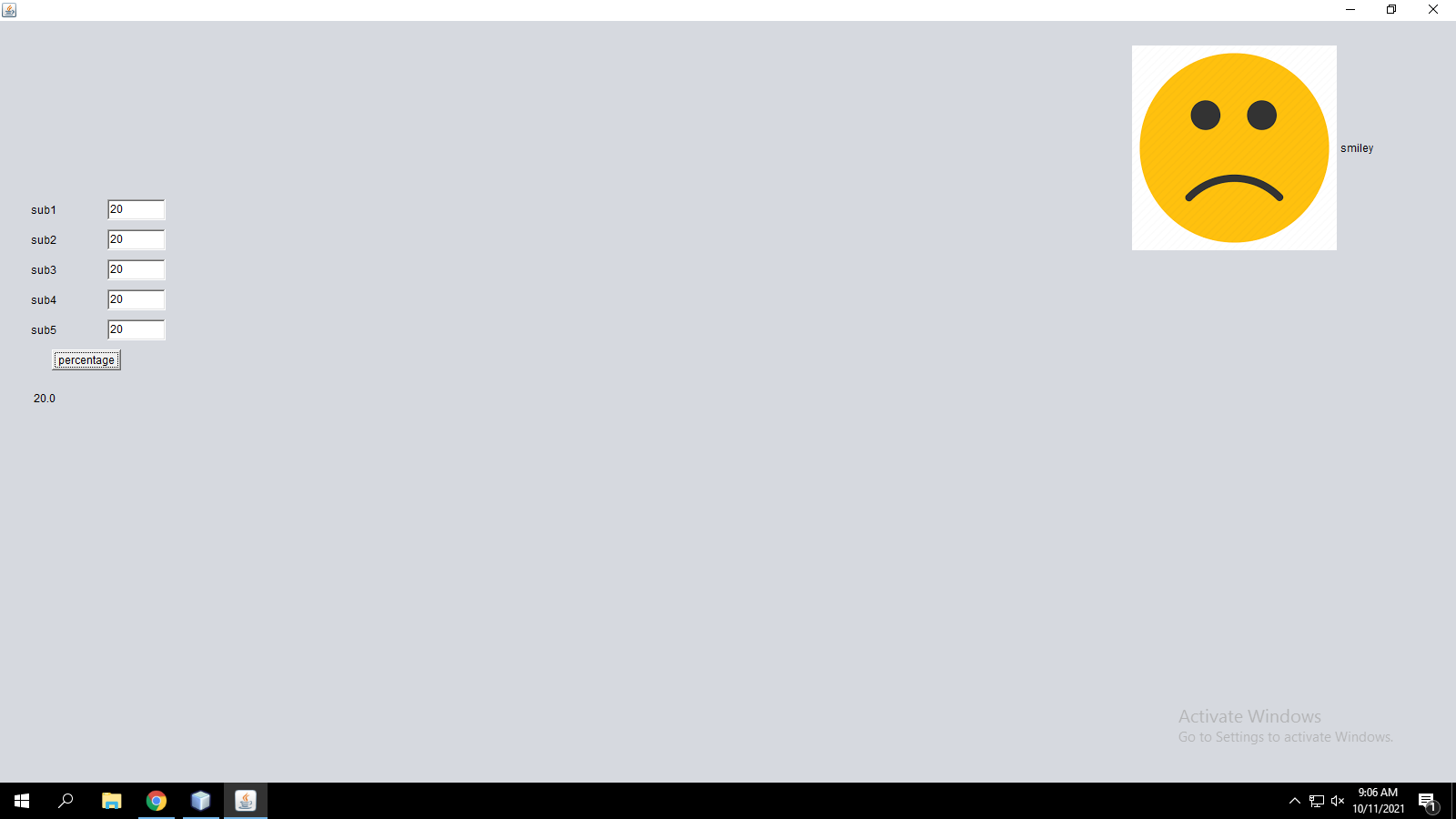
****

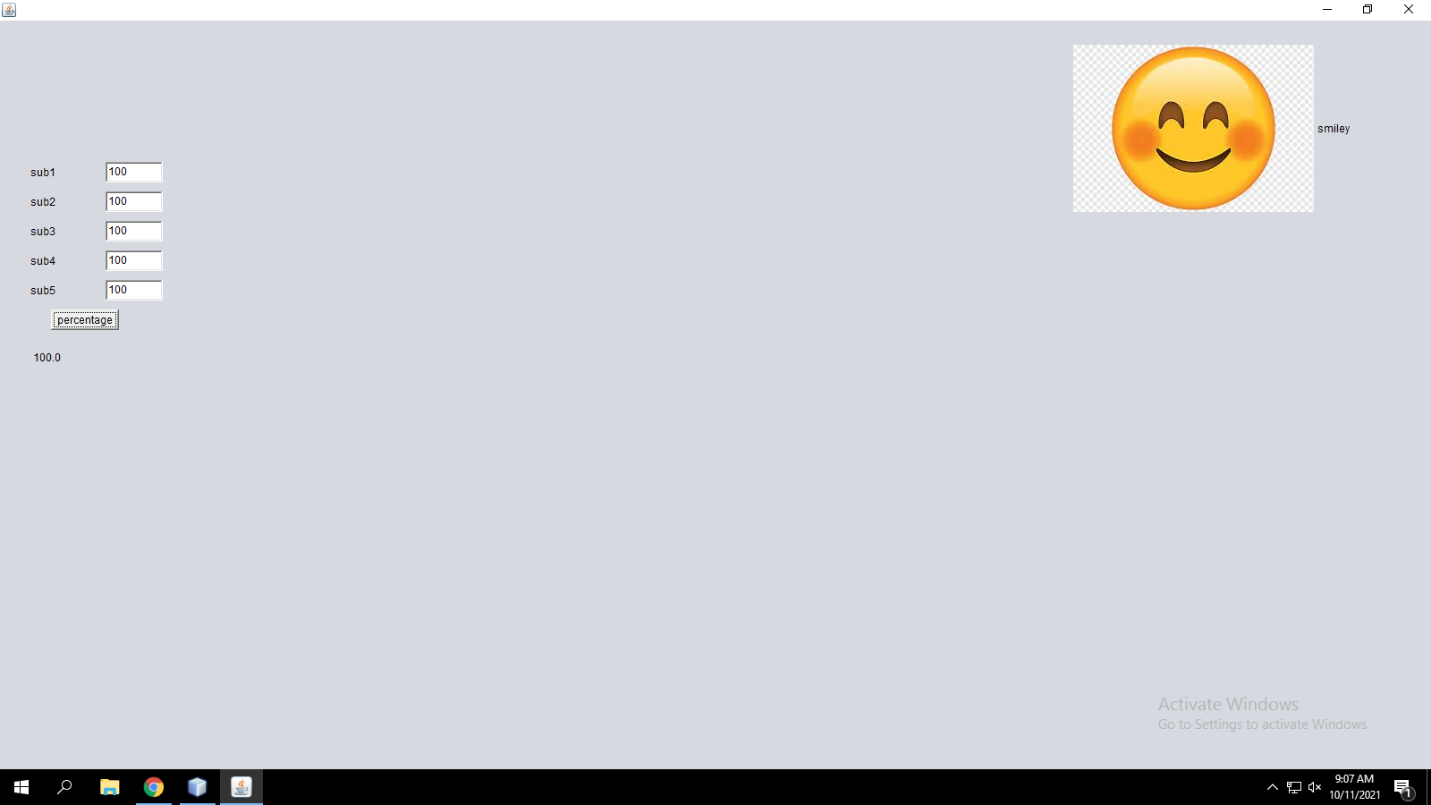
3.Find the percentage of marks obtained by a student in 5 subjects. Display a happy face if he secures above 50% or a sad face if otherwise.

**Program**

int a,b,c,d,e;  
       float p,r,k;  
       a= Integer.parseInt(textField1.getText());  
       b= Integer.parseInt(textField2.getText());  
       c= Integer.parseInt(textField3.getText());  
       d= Integer.parseInt(textField4.getText());  
       e= Integer.parseInt(textField5.getText());  
        p=a+b+c+d+e;  
        r =(p/500);    
        k = r\*100;  
       label6.setText(" " + k);  
ImageIcon image1;  
if(k >50)  
{  
    image1=new ImageIcon("C:\\Users\\student.MCALAB\\Documents\\NetBeansProjects\\cycle5\\src\\happy.jpg");  
}      
else  
{  
    image1=new ImageIcon("C:\\Users\\student.MCALAB\\Documents\\NetBeansProjects\\cycle5\\src\\sad.png");  
}  
smiley.setIcon(image1);

**Output**





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

**Program**

import java.applet.\*;

import java.awt.\*;

import java.awt.event.\*;

public class house extends Applet implements MouseListener {

Color color = Color.blue;

public void init()

{

addMouseListener(this);

}

public void paint(Graphics g)

{

int [] xCoords = { 40, 250, 460 };

int [] yCoords = { 200, 50, 200 };

super.paint(g);

g.drawRect(80, 200, 330, 260);

g.drawPolygon(xCoords , yCoords , 3);

g.setColor(this.color);

g.fillRect(190, 330, 100, 130);

}

public void mouseClicked(MouseEvent e)

{

this.color = color.red;

this.repaint();

}

public void mouseEntered(MouseEvent e) {}

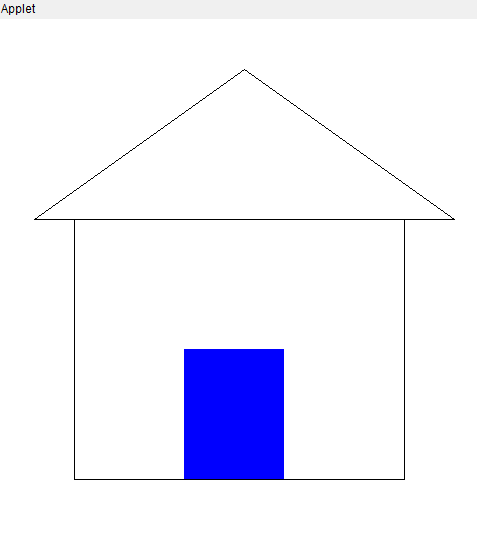
public void mouseExited(MouseEvent e) {}

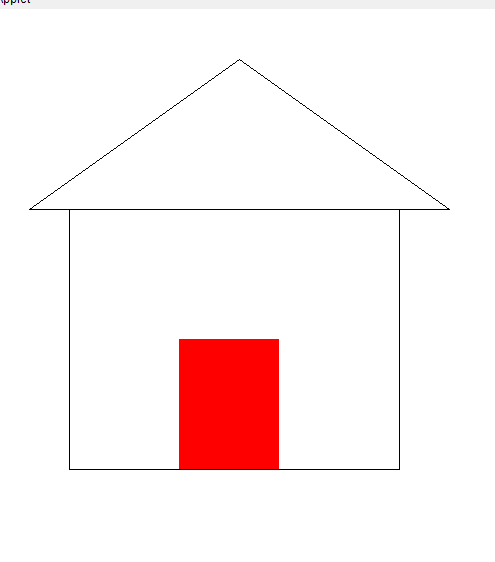
public void mousePressed(MouseEvent e) {}

public void mouseReleased(MouseEvent e) {}

}

**Output**





5. Implement a simple calculator using AWT components.

**Program**

private void button1ActionPerformed(java.awt.event.ActionEvent evt) {

int a,b,c;

a= Integer.parseInt(textField1.getText());

b= Integer.parseInt(textField2.getText());

c=a+b;

textField3 .setText(" " + c);

}

private void button2ActionPerformed(java.awt.event.ActionEvent evt) {

int a,b,c;

a= Integer.parseInt(textField1.getText());

b= Integer.parseInt(textField2.getText());

c=a-b;

textField3 .setText(" " + c);

}

private void button3ActionPerformed(java.awt.event.ActionEvent evt) {

int a,b,c;

a= Integer.parseInt(textField1.getText());

b= Integer.parseInt(textField2.getText());

c=a\*b;

textField3 .setText(" " + c);

}

private void button4ActionPerformed(java.awt.event.ActionEvent evt) {

int a,b,c;

a= Integer.parseInt(textField1.getText());

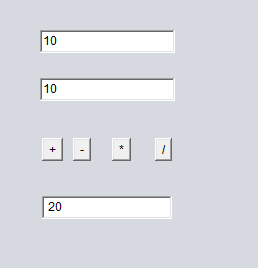
b= Integer.parseInt(textField2.getText());

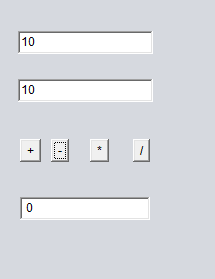
c=a/b;

textField3 .setText(" " + c);

}

**Output**





7.Develop a program to handle all mouse events and window events

**Program**

import java.awt.\*;

import java.applet.\*;

import java.awt.event.\*;

/\*<applet code="MouseDemo" width=300 height=300>

</applet>\*/

public class MouseDemo extends Applet implements MouseListener,MouseMotionListener

{

int mx=0;

int my=0;

String msg="";

public void init()

{

addMouseListener(this);

addMouseMotionListener(this);

}

public void mouseClicked(MouseEvent me)

{

mx=20;

my=40;

msg="Mouse Clicked";

repaint();

}

public void mousePressed(MouseEvent me)

{

mx=30;

my=60;

msg="Mouse Pressed";

repaint();

}

public void mouseReleased(MouseEvent me)

{

mx=30;

my=60;

msg="Mouse Released";

repaint();

}

public void mouseEntered(MouseEvent me)

{

mx=40;

my=80;

msg="Mouse Entered";

repaint();

}

public void mouseExited(MouseEvent me)

{

mx=40;

my=80;

msg="Mouse Exited";

repaint();

}

public void mouseDragged(MouseEvent me)

{

mx=me.getX();

my=me.getY();

showStatus("Currently mouse dragged"+mx+" "+my);

repaint(); }

public void mouseMoved(MouseEvent me)

{

mx=me.getX();

my=me.getY();

showStatus("Currently mouse is at"+mx+" "+my);

repaint();

}

public void paint(Graphics g)

{

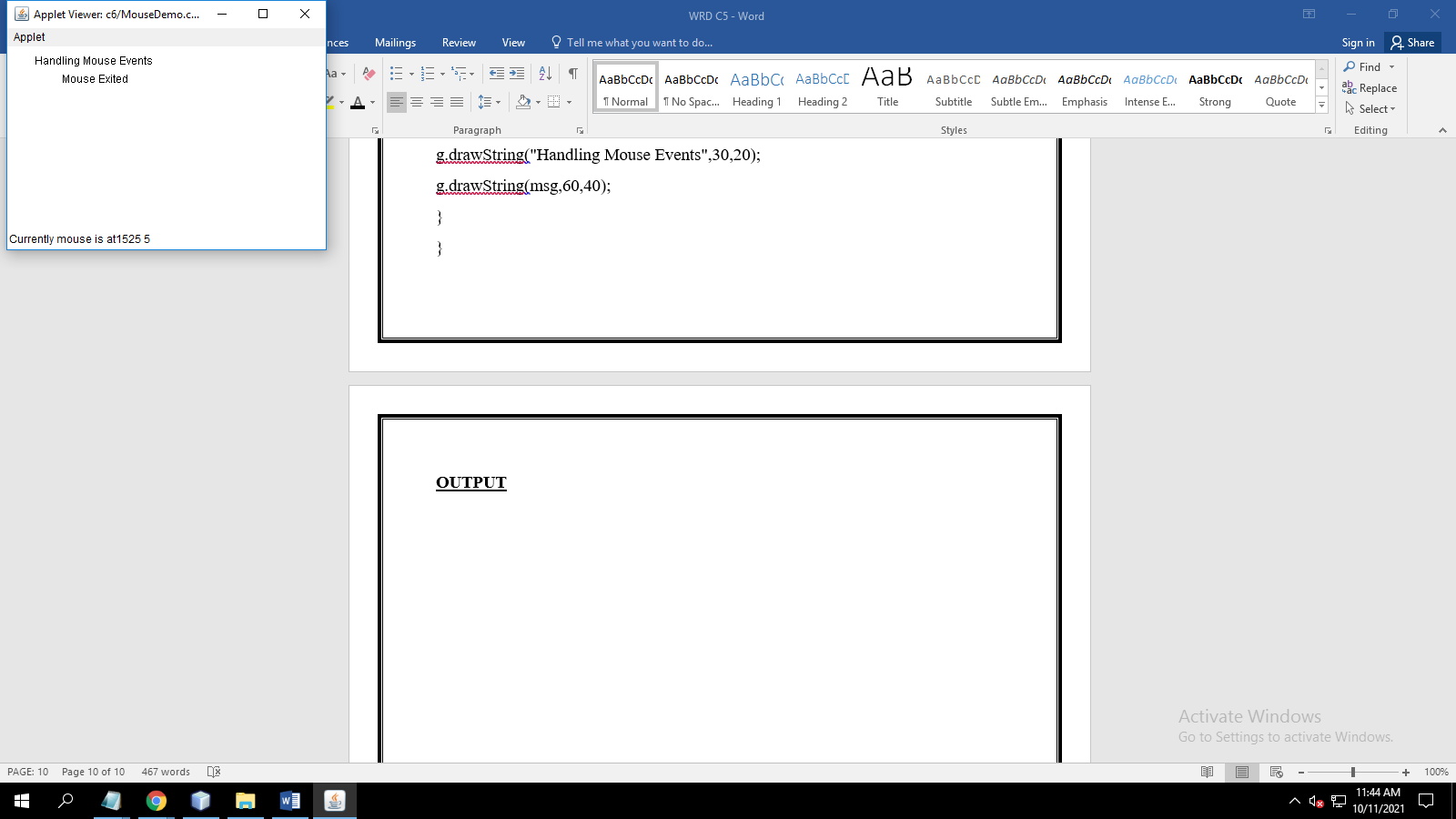
g.drawString("Handling Mouse Events",30,20);

g.drawString(msg,60,40);

}

}

**OUTPUT**



8. Develop a program to handle Key events.

**Program**

import java.awt.\*;

import java.applet.\*;

import java.awt.event.\*;

public class keyevent extends Applet implements KeyListener {

String msg="";

char ms;

public void init()

{

addKeyListener(this);

requestFocus();

}

public void paint(Graphics g)

{

g.drawString(msg,100,50);

}

public void keyTyped(KeyEvent ke)

{

msg="Key Typed";

repaint();

}

public void keyReleased(KeyEvent ke)

{

msg="Key Released";

repaint();

}

public void keyPressed(KeyEvent ke)

{

int k=ke.getKeyCode();

switch(k)

{

case KeyEvent.VK\_F1:msg="F1";

break;

case KeyEvent.VK\_F2:msg="F2";

break;

case KeyEvent.VK\_F3:msg="F3";

break;

case KeyEvent.VK\_PAGE\_UP:msg="Pageup";

break;

case KeyEvent.VK\_PAGE\_DOWN:msg="Pagedown";

break;

case KeyEvent.VK\_LEFT:msg="Left arrow";

break;

case KeyEvent.VK\_RIGHT:msg="Right arrow";

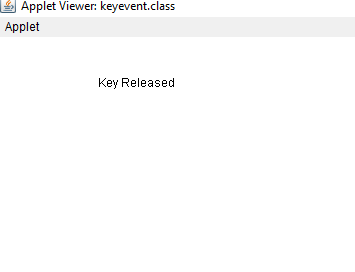
break;

}

repaint();

}

**OUTPUT**

****