**Experiment No.: 1**

**Aim**

Define a class product with data member pcode,pname and price .Create three

objects using the class and find the product having lowest price.

**Procedures**

**Source Code**

import java.util.Scanner;

public class product{

int pcode;

String pname;

int price;

public void get()

{

Scanner sc=new Scanner(System.in);

System.out.println("enter the pcode");

pcode =sc.nextInt();

System.out.println("enter the pname");

pname=sc.next();

System.out.println("enter the price");

price=sc.nextInt();

}

public void put(){

System.out.println("detail of product");

System.out.println("code of product"+pcode);

System.out.println("name of product"+pname);

System.out.println("price of product"+price);

}

public static void main(String args[]){

product p1=new product();

product p2=new product ();

product p3=new product();

p1.get();

p2.get();

p3.get();

p1.put();

p2.put();

p3.put();

if(p1.price<p2.price && p1.price<p3.price)

{

System.out.println("lowest price "+p1.price);

}

else if(p2.price<p1.price && p2.price<p3.price)

{

System.out.println("lowest price is"+p2.price);

}

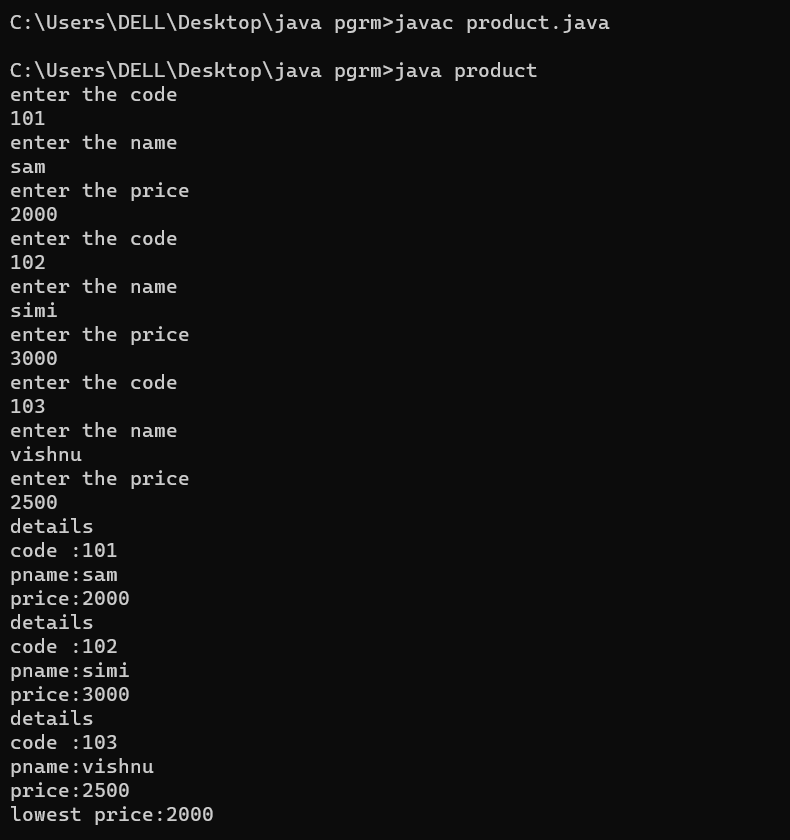
else

System.out.println("lowest price is"+p3.price);

}

}

**Output**

****

**Experiment No.: 2**

**Aim**

Read 2 matrices from the console and perform matrix addition.

**Procedure**

import java.util.Scanner;

public class MatrixAddition

{

public static void main(String[] args) {

Scanner s = new Scanner(System.in);

// Read the dimensions of the matrices

System.out.print("Enter the number of rows: ");

int rows = s.nextInt();

System.out.print("Enter the number of columns: ");

int columns = s.nextInt();

// Create the matrices

int[][] matrix1 = new int[rows][columns];

int[][] matrix2 = new int[rows][columns];

int[][] sumMatrix = new int[rows][columns];

// Read the elements of the first matrix

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

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

{

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

{

matrix1[i][j] = s.nextInt();

}

}

// Read the elements of the second matrix

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

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

{

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

{

matrix2[i][j] = s.nextInt();

}

}

// Perform matrix addition

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

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

sumMatrix[i][j] = matrix1[i][j] + matrix2[i][j];

}

}

// Display the sum matrix

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

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

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

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

}

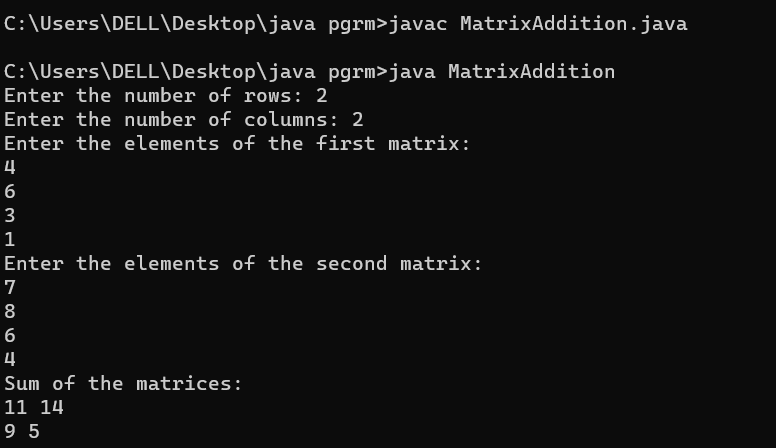
System.out.println();

}

}

}

**Output**

****

**Experiment No.: 3**

**Aim**

Add complex numbers.

**Procedure**

Source Code

import java.util.\*;

public class Complex{

public static void main(String[] args)

{

Scanner Snr = new Scanner(System.in);

System.out.println("Enter the real part of first imaginary number: ");

int r1 = Snr.nextInt();

System.out.println("Enter the coefficient of the first imaginary constant:");

int i1 = Snr.nextInt();

System.out.println("Enter the real part of second imaginary number: ");

int r2 = Snr.nextInt();

System.out.println("Enter the coefficient of the second imaginary constant: ");

int i2 = Snr.nextInt();

System.out.println("The first imaginary number is "+r1+" + "+i1+"i");

System.out.println("The second imaginary number is "+r2+" + "+i2+"i");

int r3=r1+r2;

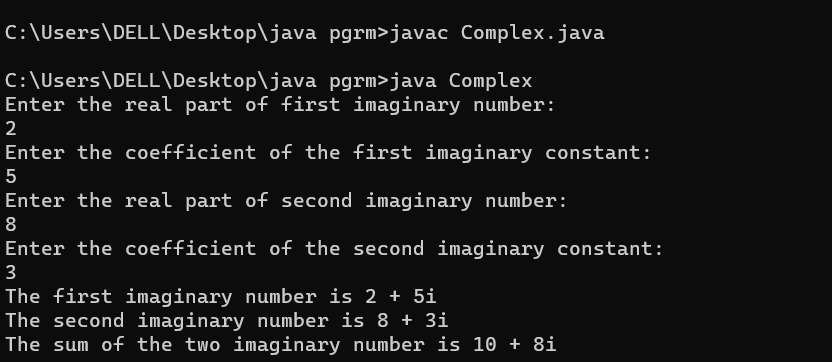
int i3=i1+i2;

System.out.println("The sum of the two imaginary number is "+ r3 +" + "+ i3 +"i");

}

}

**Output**

****

**Experiment No.: 4**

**Aim**

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

**Procedure**

**Source Code**

import java.util.\*;

public class SymmetricMatrix{

public static void main(String[] args){

Scanner Snr = new Scanner(System.in);

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

int sz = Snr.nextInt();

int Arr[][] = new int[sz][sz];

int Arr1[][] = new int[sz][sz];

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

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

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

Arr[i][j] = Snr.nextInt();

}

}

System.out.println("The matrix: ");

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

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

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

}

System.out.println(" ");

}

System.out.println("The Transpose of the matrix: ");

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

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

Arr1[i][j]=Arr[j][i];

}

}

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

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

System.out.print(Arr1[i][j] + " ");

}

System.out.println(" ");

}

int flag=0;

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

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

if(Arr[i][j] != Arr1[i][j]){

flag=1;

break;

}

}

}

if(flag==1){

System.out.println("The matrix is not symmetric");

}

else{

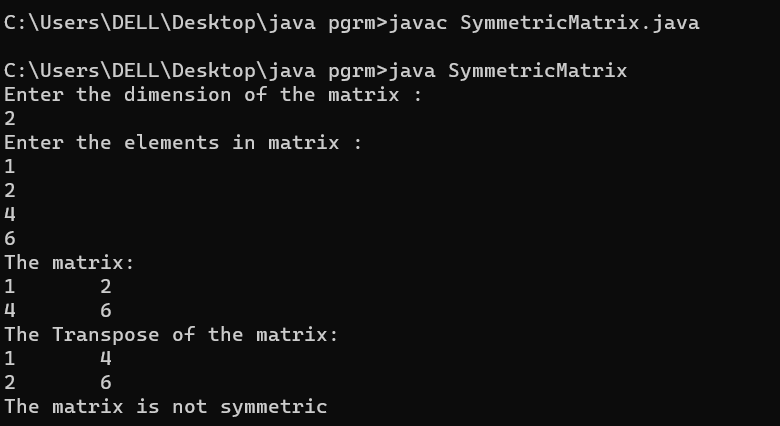
System.out.println("The matrix is symmetric");

}

}

}

**output**



**Experiment No.: 5**

**Aim**

Create CPU with attribute price create inner class processor(number of cores, manufacturer) and static nested class ram(memory,manufacturer). Create an object of CPU and print information of processor and ram.

**Procedure**

**Source Code**

import java.util.\*;

public class Cpu

{

int price;

public void price\_input()

{

Scanner in = new Scanner(System.in);

System.out.println("Enter the price of CPU : ");

price = in.nextInt();

}

public void print()

{

System.out.println("Price : "+price);

}

public class Processor

{

int no\_of\_cores;

String man;

Processor()

{

Scanner sc = new Scanner(System.in);

System.out.println("Enter the No:of cores : ");

no\_of\_cores = sc.nextInt();

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

man = sc.next();

}

void print()

{

System.out.println(" No:of cores : "+no\_of\_cores);

System.out.println(" Manufacturer : "+man);

}

}

static class RAM

{

static int mem\_size;

static String man;

RAM()

{

Scanner sc = new Scanner(System.in);

System.out.println("Enter the memory size ofRAM : ");

mem\_size = sc.nextInt();

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

man = sc.next();

}

void print()

{

System.out.println(" Memory Size : "+mem\_size);

System.out.println(" Manufacturer : "+man);

}

}

public static void main(String args[])

{

Cpu c = new Cpu();

Cpu.Processor p = c.new Processor();

Cpu.RAM r = new Cpu.RAM();

c.price\_input();

p.print();

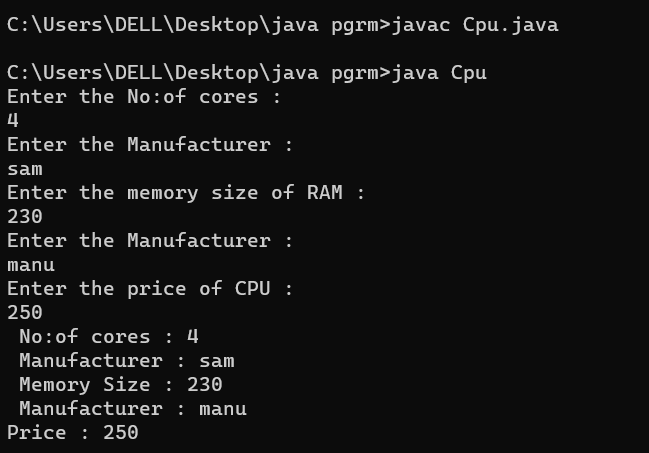
r.print();

c.print();

}

}

**output**

****

**Experiment No.: 6**

**Aim**

Program to sort strings.

**Procedure**

import java.util.\*;

public class sort {

public static void main(String[] args) {

Scanner value = new Scanner(System.in);

int i,j;

String temp;

System.out.println("Enter the size of Array: ");

int size = value.nextInt();

String array[] = new String[size];

System.out.println("Enter the elements of Array");

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

{

array[i] = value.nextLine();

}

System.out.println("The Array is: ");

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

{

System.out.println(array[i]);

}

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

{

for (j=i+1;j<size;j++)

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

{

temp = array[i];

array[i] = array[j];

array[j] = temp;

}

}

System.out.println("The Sorted Array is: ");

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

{

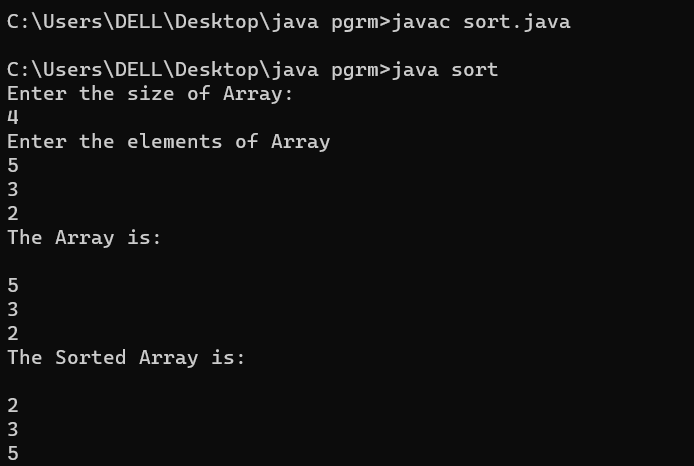
System.out.println(array[i]);

}

}

}

**output**

****

**Experiment No.: 8**

**Aim**

Perform string manipulations**.**

**Procedure**

import java.util.Scanner;

import java.lang.\*;

public class Manipulation{

public static void main(String [] args){

int a;

String b,c;

Scanner sc = new Scanner(System.in);

System.out.print(" Enter the string : ");

b = sc.nextLine();

while(true)

{

System.out.println("\n MENU:\n 1.String Length.\2.Uppercase.\n3.Lowercase.\n4.Concatenate.\n 5.Character index.\n6.Exit.");

System.out.print("\n Enter your option : ");

a = sc.nextInt();

switch(a)

{

case 1:

System.out.println(" String length = "+b.length());

break;

case 2:

System.out.println(" String in uppercase = "+b.toUpperCase());

break;

case 3:

System.out.println(" String in lowercase = "+b.toLowerCase());

break;

case 4:

{

System.out.print(" Enter the string to be concatenate = ");

c = sc.next();

System.out.println(" Concatenated string = "+b.concat(c));

break;

}

case 5:

{

System.out.print(" Enter the Character to be searched in the given string = ");

c = sc.next();

System.out.println(" The character is found at "+(b.indexOf(c)+1)+".");

break;

}

case 6: System.exit(0);

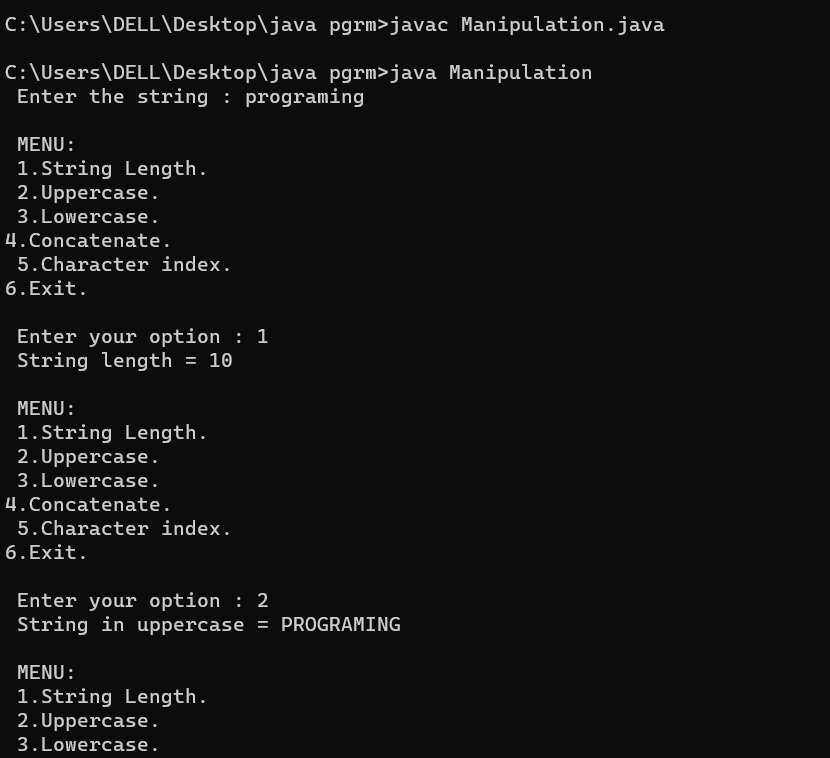
}

}

}

}

**Output**

****

**Experiment No.: 9**

**Aim**

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

**Procedure**

import java.util.\*;

public class employee\_search{

int eno;

String ename;

int salary;

public void get()

{

Scanner obj=new Scanner(System.in);

System.out.println("enter employee number :");

eno =obj.nextInt();

System.out.println("enter employee name :");

ename =obj.next();

System.out.println("enter employee salary :");

salary =obj.nextInt();

}

public void display()

{

System.out.println("employee number :"+eno);

System.out.println("employee name :"+ename);

System.out.println("employee salary :"+salary);

}

public static void main(String[] args)

{

int flag=0;

Scanner obj=new Scanner(System.in);

System.out.println("enter the size :");

int n=obj.nextInt();

employee\_search e1[]= new employee\_search[n];

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

{

e1[i]=new employee\_search();

e1[i].get();

}

for(int i=0;i<2;i++)

{

e1[i].display();

}

System.out.println("enter employee number to search for the employee details:");

int item=obj.nextInt();

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

{

if(e1[i].eno ==item)

{

e1[i].display();

flag++;

break;

}

}

if(flag==0)

{

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

}

}

}

Experiment : 10

Aim : Area of different shapes using overloaded functions

CO3: Implement object-oriented concepts like inheritance, overloading and interfaces

Procedure

import java.util.Scanner;

class Shape1{

public void area(double r){

System.out.println("Area of Circle "+ (3.14\*r\*r));

}

public void area(int b,int h){

System.out.println("Area of Triangle "+ (.5\*(b\*h)));

}

public void area(int a){

System.out.println("Area of Square "+ (a\*a));

}

public void area(int x,float y)

{

System.out.println("Area Of Reactangle "+(x\*y));

}

public static void main(String args []){

int b,c,x;

double a;

float y;

Scanner f=new Scanner(System.in);

Shape1 s =new Shape1();

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

a=f.nextDouble();

s.area(a);

System.out.println("Enter the Base of Triangle :");

c=f.nextInt();

System.out.println("Enter the Height of Triangle :");

b=f.nextInt();

s.area(c,b);

System.out.println("Enter the Length of Reactangle :");

x=f.nextInt();

System.out.println("Enter the width of Rectangle :");

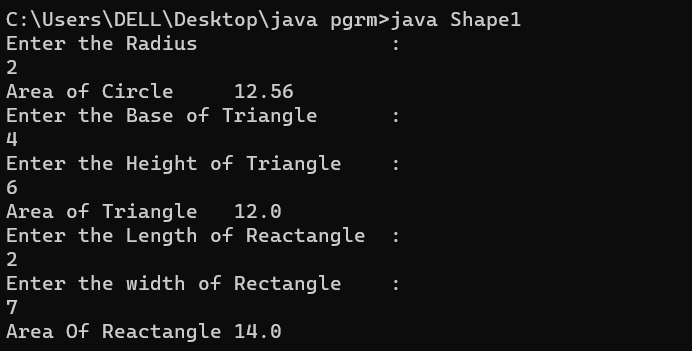
y=f.nextFloat();

s.area(x,y);

}

}

Output



Experiment : 11

Aim :

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

CO 3: Implement object-oriented concepts like inheritance, overloading and interfaces

Procedure

import java.util.Scanner;

class Employee {

int empid;

String name;

int salary;

String address;

Employee(int a, String b, int c, String d) {

empid = a;

name = b;

salary = c;

address = d;

}

}

class Teacher extends Employee {

String department;

String subject;

Teacher(int l, String m, int n, String o, String p, String q) {

super(l, m, n, o);

department = p;

subject = q;

}

public void display() {

System.out.println("\nEmployee Id :" + empid);

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

System.out.println("Employee Salary :" + salary);

System.out.println("Employee Address :" + address);

System.out.println("Teacher Department:" + department);

System.out.println("Teacher Subject :" + subject);

}

}

public class EmpCO3 {

public static void main(String[] args) {

int i;

Scanner cin = new Scanner(System.in);

System.out.print("Enter the limit of array :");

int n = cin.nextInt();

Teacher e[] = new Teacher[n];

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

System.out.print("Enter the Id of the Employee :");

int a = cin.nextInt();

System.out.print("Enter the Name of the Employee :");

String b = cin.next();

System.out.print("Enter the Salary of the Employee :");

int c = cin.nextInt();

System.out.print("Enter the Address of the Employee :");

String d = cin.next();

System.out.print("Enter the Department of the Teachers :");

String q = cin.next();

System.out.print("Enter the Subject of the Teachers :");

String f = cin.next();

e[i] = new Teacher(a, b, c, d, q, f);

}

System.out.println("\n\nPRINTING DETAILS");

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

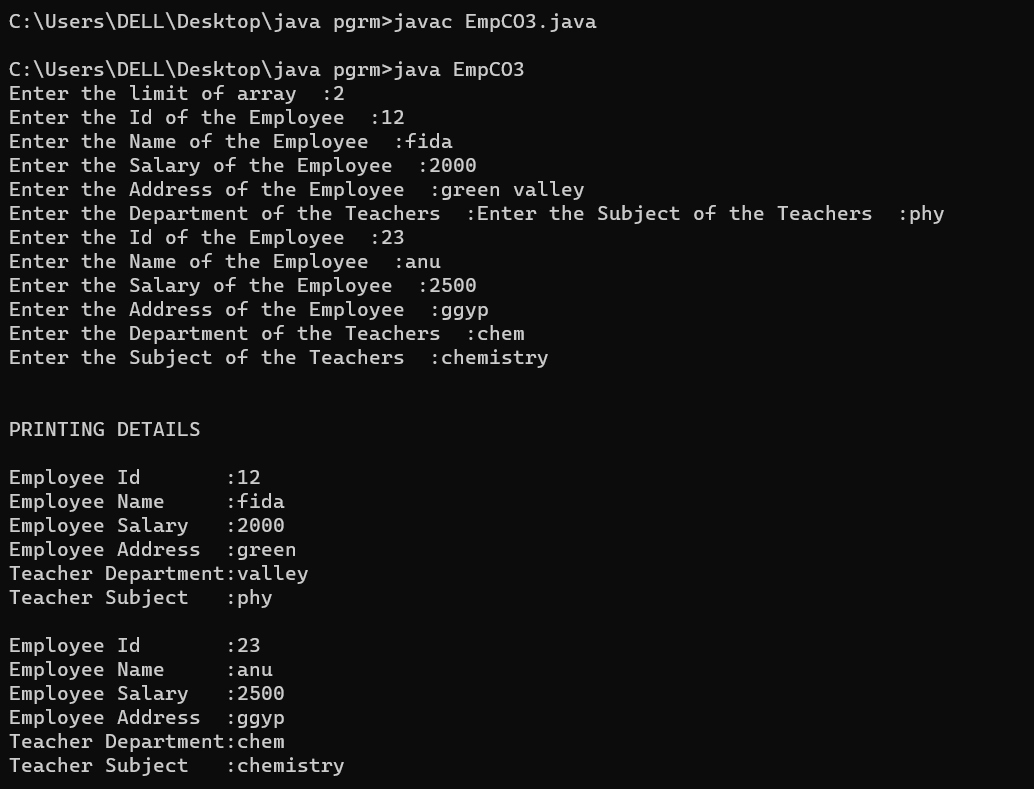
e[i].display();

}

}

}

Output



Experiment : 12

Aim :

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

CO 3: Implement object-oriented concepts like inheritance, overloading and interfaces

Procedure

import java.util.Scanner;

class person{

String pname;

String pgender;

String paddress;

int page;

person(String name,String gender,String address,int age){

pname=name;

pgender=gender;

paddress=address;

page=age;

}

}

class employee extends person{

int empid;

String cmpny\_name;

String qualificatiion;

int salary;

employee(String name,String gender,String address,int age,int eid,String cmpny,String qualif,int sal){

super(name,gender,address,age);

empid=eid;

cmpny\_name=cmpny;

qualificatiion=qualif;

salary=sal;

}

}

class teacher extends employee{

int teacherid;

String subject;

String department;

teacher(String name,String gender,String address,int age,int eid,String cmpny,String qualif,int sal,int tid,String sub,String dep){

super(name,gender,address,age,eid,cmpny,qualif,sal);

teacherid=tid;

subject=sub;

department=dep;

}

void display(){

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

System.out.println("Person name:"+pname);

System.out.println("Person gender:"+pgender);

System.out.println("Person address:\n"+paddress);

System.out.println("Person age:"+page);

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

System.out.println("Company name: "+cmpny\_name);

System.out.println("Employee qualification: "+qualificatiion);

System.out.println("Employee salary: "+salary);

System.out.println("Teacher id: "+teacherid);

System.out.println("Department: "+department);

System.out.println("Subject taught: "+subject);

}

}

public class SuperClass2{

public static void main(String[] args){

Scanner sc=new Scanner(System.in);

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

int limit=sc.nextInt();

teacher tcher[]=new teacher[limit];

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

{

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

String name=sc.next();

System.out.println("Enter the Person gender:");

String gender=sc.next();

System.out.println("Enter the Person address:");

String address=sc.next();

System.out.println("Enter the Person age:");

int age=sc.nextInt();

System.out.println("Enter the Employee id:");

int eid=sc.nextInt();

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

String cmpny=sc.next();

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

String qualif=sc.next();

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

int sal=sc.nextInt();

System.out.println("Enter the Teacher id:");

int tid=sc.nextInt();

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

String dep=sc.next();

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

String sub=sc.next();

tcher[i]=new teacher(name,gender,address,age,eid,cmpny,qualif,sal,tid,dep,sub);

}

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

{

tcher[i].display();

}

}

}

Output



Experiment : 04

Aim :

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

CO 3: Implement object-oriented concepts like inheritance, overloading and interfaces

Procedure

import java.util.\*;

class Publisher {

Scanner sc = new Scanner(System.in);

String publisher;

Publisher() {

System.out.print("\nEnter the Publisher name: ");

publisher = sc.next();

}

}

class Book extends Publisher {

String title;

Book() {

System.out.print("Enter the Title: ");

title = sc.next();

}

}

class Literature extends Book {

String category;

Literature() {

System.out.print("Enter the category: ");

category = sc.next();

}

}

class Fiction extends Book {

String category;

Fiction() {

System.out.print("Enter the category: ");

category = sc.next();

}

}

public class BookCO3 {

public static void main(String[] args) {

System.out.print("\nEnter the Details of Literature Book : ");

Literature l = new Literature();

System.out.print("\nEnter the Details of Fiction Book : ");

Fiction f = new Fiction();

System.out.println("\nPrinting Details of the book 1");

System.out.println("Publisher: " + l.publisher);

System.out.println("Title: " + l.title);

System.out.println("Category: " + l.category);

System.out.println("\nPrinting Details of the book 2");

System.out.println("Publisher: " + f.publisher);

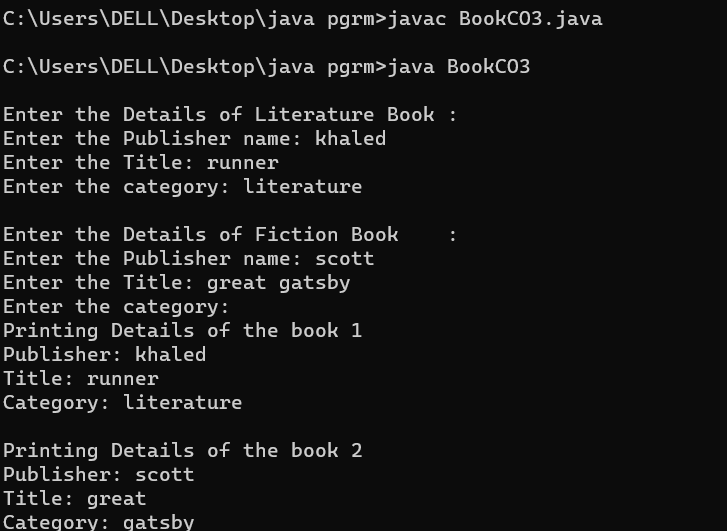
System.out.println("Title: " + f.title);

System.out.println("Category: " + f.category);

}

}

Output



Experiment : 14

Aim :

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

CO 3: Implement object-oriented concepts like inheritance, overloading and interfaces

Procedure

import java.util.Scanner;

interface Student {

void getAcademicMarks();

}

interface Sports {

void getSportsMarks();

}

class Result implements Student, Sports {

private int academicMarks;

private int sportsMarks;

public void getAcademicMarks() {

Scanner sc = new Scanner(System.in);

System.out.print("Enter the marks obtained for the examination: ");

academicMarks = sc.nextInt();

}

public void getSportsMarks() {

Scanner sc = new Scanner(System.in);

System.out.print("Enter the marks/position obtained for Sports: ");

sportsMarks = sc.nextInt();

}

public void displayScores() {

System.out.println("\nAcademic Score: " + academicMarks);

System.out.println("Sports Score: " + sportsMarks);

}

}

public class stuCO3 {

public static void main(String[] args) {

Result res = new Result();

res.getAcademicMarks();

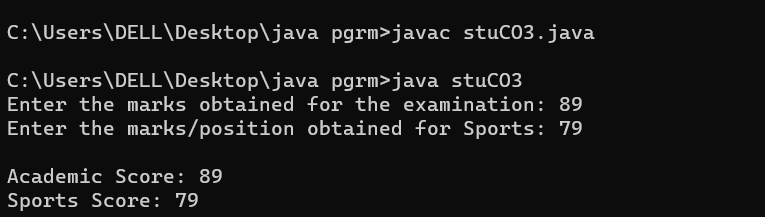
res.getSportsMarks();

res.displayScores();

}

}

Output



Experiment : 15

Aim :

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

CO 3:

Implement object-oriented concepts like inheritance, overloading and interfaces

Procedure

import java.util.\*;

interface prototype{

public void getdata();

public void area();

public void perimeter();

}

class circle implements prototype

{

Scanner obj=new Scanner(System.in);

int radius;

double z=3.14;

public void getdata()

{

System.out.println("enter the radius :");

radius=obj.nextInt();

}

public void area()

{

System.out.println("area of circle :"+z\*(radius\*radius));

}

public void perimeter()

{

System.out.println("perimeter of circle :"+(2\*z)\*radius);

}

}

class rectangle implements prototype

{

Scanner obj=new Scanner(System.in);

int l,b;

public void getdata()

{

System.out.println("enter the lenght :");

l=obj.nextInt();

System.out.println("enter the breadth :");

b=obj.nextInt();

}

public void area()

{

System.out.println("area of rectangle :"+l\*b);

}

public void perimeter()

{

System.out.println("perimeter of rectangle :"+l+b);

}

}

class shape{

public static void main(String[] args)

{

int ch,u=0;

circle cc=new circle();

rectangle jj=new rectangle();

while(u==0)

{

Scanner obj=new Scanner(System.in);

System.out.println("1.Circle \n2.Rectangle \n3.Exit");

System.out.println("choose one :");

ch=obj.nextInt();

switch(ch)

{

case 1:cc.getdata();

cc.area();

cc.perimeter();

break;

case 2:jj.getdata();

jj.area();

jj.perimeter();

break;

case 3:System.exit(0);

default:

System.out.println("choose valid one :");

break;

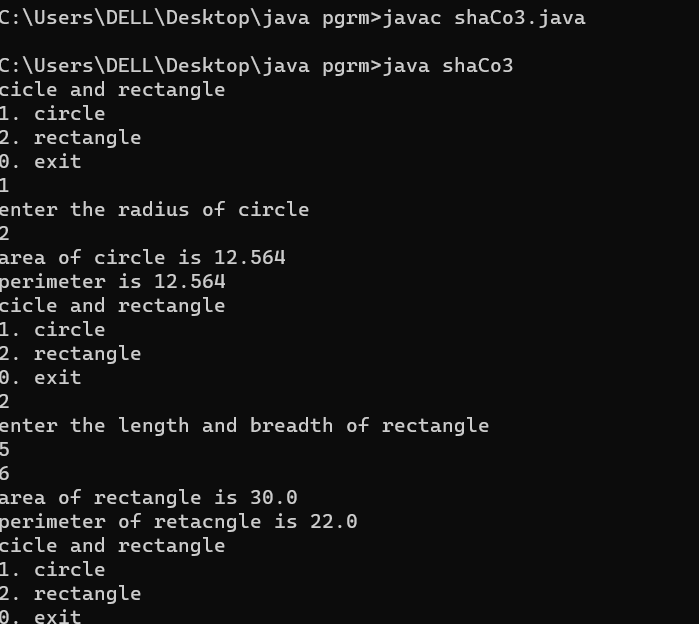
}

}

}

}

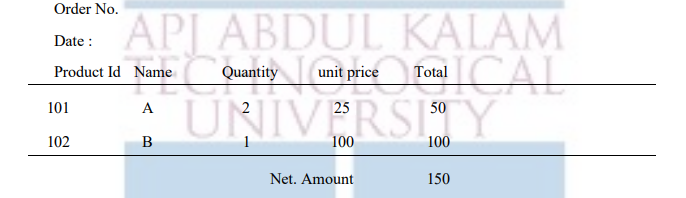
Output



Experiment : 15

Aim :

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



CO 3: Implement object-oriented concepts like inheritance, overloading and interfaces

Procedure

import java.util.\*;

interface product

{

public void getdata();

public void display();

public void calc();

}

class calculate implements product

{

int pid;

String name;

double quantity;

double unit\_price,total;

public void getdata()

{

Scanner obj=new Scanner(System.in);

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

pid=obj.nextInt();

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

name=obj.next();

System.out.println("enter the quantity :");

quantity=obj.nextDouble();

System.out.println("enter the unit\_price :");

unit\_price=obj.nextDouble();

}

public void calc()

{

total =quantity \* unit\_price;

}

public void display()

{ System.out.println(pid+"\t\t"+name+"\t\t"+quantity+"\t\t"+unit\_price+"\t\t"+total);

}

}

class bill{

public static void main(String[] args)

{

int i,odno,n;

String date;

double net\_amnt =0;

Scanner obj=new Scanner(System.in);

System.out.println("enter order number :");

odno=obj.nextInt();

System.out.println("enter the date :");

date=obj.next();

System.out.println("enter the total number products :");

n=obj.nextInt();

calculate arr[]= new calculate[n];

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

{

arr[i] = new calculate();

arr[i].getdata();

arr[i].calc();

}

System.out.println("Order no : #"+odno);

System.out.println("Date :"+date);

System.out.println("Product Id Name Quantity Unit Price Total"); System.out.println("=====================================");

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

{

arr[i].display();

net\_amnt +=arr[i].total;

}

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

System.out.println("\t\t\t\t\t\tNet Amount :"+net\_amnt);

}

}

Output

