Experiment : 01

Aim :

Area of different shapes using overloaded functions

CO 3:

Implement object-oriented concepts like inheritance, overloading and interfaces

Procedure

import java.util.\*;

class CO3\_Shape

{

int r,h,a,p1,p2,t2,t3;

double b,l,w,t1;

public void area(double r)

{

double c=3.14\*(r\*r);

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

}

public void area(double b,int h)

{

double t=(b\*h)/2;

System.out.println("Area of triangle: "+t);

}

public void area(int a)

{

int s=a\*a;

System.out.println("Area of square: "+s);

}

public void area(double l,double w)

{

double r=l\*w;

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

}

public void area(int p1,int p2)

{

int p=p1\*p2;

System.out.println("Area of parallelogram: "+p);

}

public void area(float e1,float e2)

{

double c=3.14\*(e1\*e2);

System.out.println("Area of ellipse: "+c);

}

public static void main(String[] args)

{ int ch;

Scanner s=new Scanner(System.in);

CO3\_Shape sh =new CO3\_Shape();

System.out.println("Area of different shapes"+"\n"+" 1.Circle"+"\n"+" 2.Triangle"+"\n"+" 3.Square "+"\n"+" 4.Rectangle "+"\n"+" 5.Parallelogram "+"\n"+" 6.Ellipse");

do

{

System.out.println("Enter choice: ");

ch=s.nextInt();

switch(ch)

{

case 1:

{

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

double r=s.nextDouble();

sh.area(r);

}

break;

case 2:

{

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

double b=s.nextDouble();

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

int h=s.nextInt();

sh.area(b,h);

}

break;

case 3:

{

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

int a=s.nextInt();

sh.area(a);

}

break;

case 4:

{

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

double l=s.nextDouble();

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

double w=s.nextDouble();

sh.area(l,w);

}

break;

case 5:

{

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

int p1=s.nextInt();

System.out.println("Enter the vertical height: ");

int p2=s.nextInt();

sh.area(p1,p2);

}

break;

case 6:

{

System.out.println("Enter the minor axis: ");

float e1=s.nextFloat();

System.out.println("Enter the major axis: ");

float e2=s.nextFloat();

sh.area(e1,e2);

}

break;

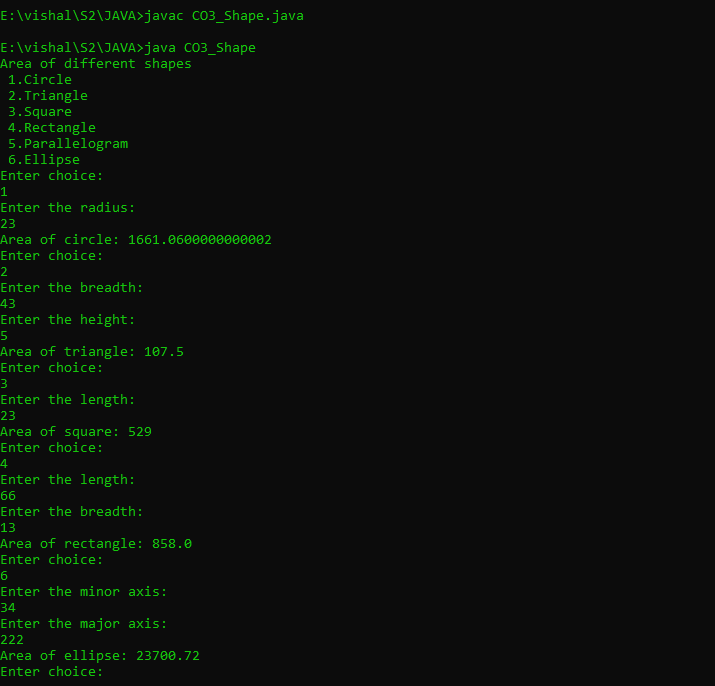
}

}

while(ch!=7);

}}

Output



Experiment : 02

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.\*;

class Employee

{

int Empid,Salary;

String Name,Address;

Scanner sc=new Scanner(System.in);

Employee()

{

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

Empid=sc.nextInt();

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

Name=sc.next();

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

Salary=sc.nextInt();

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

Address=sc.next();

}

}

class Teacher extends Employee

{

String Dep,Sub;

Teacher()

{

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

Dep=sc.next();

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

Sub=sc.next();

}

void Display()

{

System.out.println("Employee ID :"+Empid);

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

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

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

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

System.out.println("Subject: "+Sub);

}

}

class EmpTeacher{

public static void main(String[] a)

{

int n,i;

Scanner obj=new Scanner(System.in);

System.out.println("enter the number of employees :");

n=obj.nextInt();

Teacher array[]=new Teacher[n];

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

{

array[i]=new Teacher();

}

System.out.println("Employee Details ::");

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

{

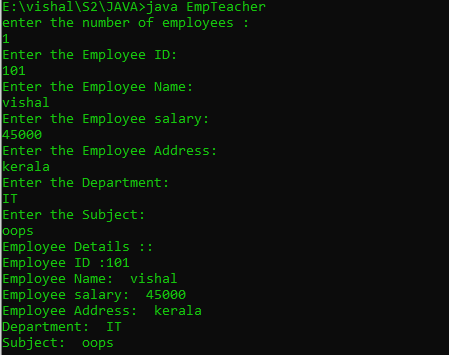
array[i].Display();

}

}

}

Output



Experiment : 03

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++)

{

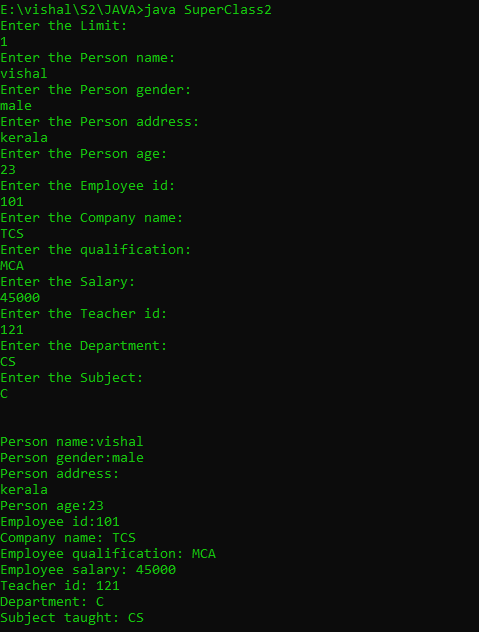
tcher[i].display();

}

}

}

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{

String pub\_name;

publisher()

{

Scanner obj=new Scanner(System.in);

System.out.println("Publisher name :");

pub\_name=obj.next();

}

}

class book extends publisher

{

String book\_name;

book()

{

Scanner obj=new Scanner(System.in);

System.out.println("book name :");

book\_name=obj.next();

}

}

class literature extends book

{

void display()

{

System.out.println("Publisher name is :"+pub\_name);

System.out.println("book name is :"+book\_name);

}

}

class fiction extends book

{

void display()

{

System.out.println("Publisher name is :"+pub\_name);

System.out.println("book name is :"+book\_name);

}

}

class library

{

public static void main(String[] args)

{

int i=0;

Scanner obj=new Scanner(System.in);

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

int size=obj.nextInt();

literature arr1[]= new literature[size];

fiction arr2[]=new fiction[size];

System.out.println("enter the details of literature books :");

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

{

arr1[i]=new literature();

}

System.out.println("enter the details of fiction books :");

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

{

arr2[i]=new fiction();

}

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

System.out.println("details of literature books :");

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

{

arr1[i].display();

}

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

System.out.println("details of fiction books :");

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

{

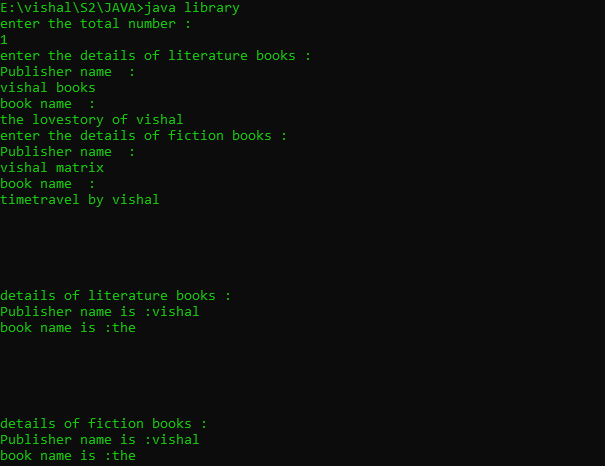
arr1[i].display();

}

}

}

Output



Experiment : 05

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.\*;

interface student{

public void getS();

public void dispS();

}

interface sports

{

public void getSp();

public void dispSp();

}

class result implements student,sports

{

Scanner obj=new Scanner(System.in);

String name,spitem;

int m1,m2,roll,rank;

double total,percentage;

public void getS()

{

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

name=obj.next();

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

roll=obj.nextInt();

System.out.println("enter the mark 1 :");

m1=obj.nextInt();

System.out.println("enter the mark2 :");

m2=obj.nextInt();

total=m1+m2;

percentage=(total\*200)/100;

}

public void getSp()

{

System.out.println("enter the sports item which the student participated in :");

spitem=obj.next();

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

rank=obj.nextInt();

}

public void dispS()

{

System.out.println("Name of the Student :"+name);

System.out.println("Roll Number of the Student :"+roll);

System.out.println("Mark 1 the Student :"+m1);

System.out.println("Mark 2 of the Student :"+m2);

System.out.println("Total Mark of the Student :"+total);

System.out.println("Percentage of the Student :"+percentage);

}

public void dispSp()

{

System.out.println("Sports item :"+spitem);

System.out.println("The Rank :"+rank);

}

}

public class multiinher{

public static void main(String[] args)

{

result obj=new result();

obj.getS();

obj.getSp();

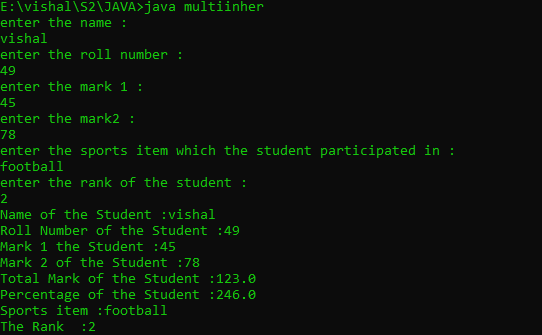
obj.dispS();

obj.dispSp();

}

}

Output



Experiment : 06

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;

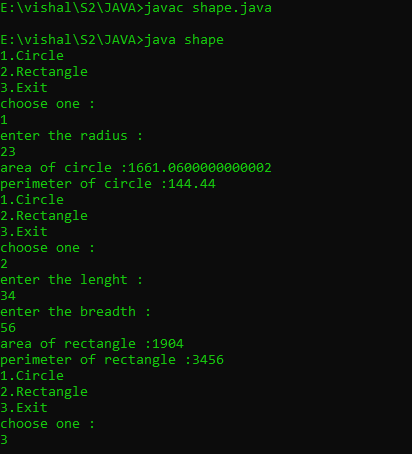
}

}

}

}

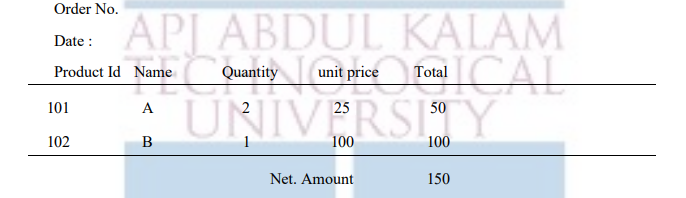
Output



Experiment : 07

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

