

Lab cycle

1) Using the concept of method overloading find the area of different shapes, rectangle, circle and square

Source code

```
import java.util.*;
```

```
class figure {
```

```
    double findArea(float r) { return r*r*3.14; }
```

```
    int findArea(String a int a) { return a*a; }
```

```
    int findArea(int l, int b) { return l*b; }
```

```
}  
class main1 {
```

```
    public static void main (String a[]) {
```

```
        float r; int l, b;
```

```
        Scanner obj = new Scanner(System.in);
```

```
        figure obj = new figure();
```

```
        System.out.println("\nEnter length of radius of circle: ");
```

```
        r = obj.nextFloat();
```

```
        System.out.println("Area of circle with radius " + r + " = " + obj.findArea(r));
```

```
        System.out.println("\nEnter length of one side of square: ");
```

```
        l = obj.nextInt();
```

```
        System.out.println("Area of square with one side length " + l + " = " +  
                             obj.findArea(l));
```

```
        System.out.println("\nEnter length of rectangle: ");
```

```
        l = obj.nextInt();
```

```
        System.out.println("\nEnter breadth of rectangle: ");
```

```
        b = obj.nextInt();
```

```
        System.out.println("Area of square rectangle with length " + l + " and breadth " + b);
```

```
+ b + " = " + obj.findArea(l, b);
```

```
}
```

```
}
```

Result :- Program successfully executed and output obtained.

Algorithm

Step 1 :- Start

Step 2 :- ~~Declare~~ Declare a class named Figure

Step 3 :- ~~Declare~~ Implemented 3 method of name findArea with different return type and no. of argument inside class Figure

Step 4 :- Create an object of Figure and call the method findArea ~~with~~ through different ways.

Step 5 :- print the area returned by each calling.

Step 6 :- Stop

Output

Enter length & radius of circle :

6

Area of circle with radius 6.0 = 112.320000

Enter length of one side of square :

6

Area of square with one side length 6 = 36

Enter length of rectangle : 3

Enter breadth of rectangle : 4

Area of rectangle with length 3 and breadth 4 = 12

2) Create a class 'Employee' with data members Empid, Name, Salary, Address and constructors to initialize the data members. Create another class 'Teacher' that inherits the properties of class employee and contain its own data members and also include display function to display all the data members. Use array of objects to display details of N teachers.

Source code

```
import java.util.*;
```

```
class Employee {  
    int empid, salary;  
    String name, addr;  
    Employee (int id, String name, int sal, String addr) {  
        empid = id ;  
        this.name = name ;  
        salary = sal ;  
        this.addr = addr ;  
    }  
}
```

```
class Teacher extends Employee {
```

```
    String dept, subj;
```

```
    Teacher (String d, String s, int id, String name, int sal, String addr) {
```

```
        super(id, name, sal, addr);
```

```
        dept = d; subj = s;
```

```
    }
```

```
    void display () {
```

```
        System.out.println "ID: " + empid + "\nName: " + name + "\nSalary : " +  
            salary + "\nAddress: " + addr + "\nDepartment : "  
            + dept + "\nSubjects taught : " + subj + "\n)";
```

```
    }
```

```
}
```



```
public class arr08 Obj {
```

```
    public static void main (String a[]) {
```

```
        Scanner s = new Scanner (System.in);
```

```
        int n, id, salary;
```

```
        String name, address, dept, sub;
```

```
        Employee e[]
```

```
        Teacher t[] = new Teacher[20];
```

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

```
        n = s.nextInt();
```

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

```
        { System.out.println ("Employee " + (i+1));
```

```
          System.out.println ("Enter the emp id & name of employee : ");
```

```
          id = s.nextInt();
```

```
          name = s.next();
```

```
          System.out.println ("Enter salary & address : ");
```

```
          salary = s.nextInt();
```

```
          address = s.next();
```

```
          System.out.println ("Teacher " + (i+1));
```

```
          System.out.println ("Enter department & subjects taught : ");
```

```
          dept = s.next();
```

```
          sub = s.next();
```

```
          t[i] = new Teacher (dept, sub, id, name, salary, address);
```

```
        }
```

```
        for (int i = 0; i < n; i++) { System.out.println ("Teacher : " + (i+1));
```

```
          t[i].display();
```

```
        }
```

```
    }
```

```
}
```

Result :- Program successfully executed and result obtained.

Output

Enter no of teachers:

2

Employee 1

Enter the emp id & name of employee:

101 Ann

Enter salary & address:

2000 Thiruvalla

Teacher 1

Enter department & subjects taught:

EEE electronics

Employee 2

Enter the emp id & name of employee:

102 Croku

Enter salary & address:

2500 Kollam

Teacher 2

Enter department & subjects taught:

MCA DS

Teacher 1

ID: 101

Name: Ann

Salary: 2000

Address: Thiruvalla

Department: EEE

Subjects taught: electronics

Teacher 2

ID: 102

Name: Croku

Salary: 2500

Address: Kollam

Department: MCA

Subjects taught: DS

3) 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 contains constructors and methods to display the data members. Use array of objects to display details of N Teachers.

Source code

```
import java.util.*;

class person {
    int age;
    String name, gender, address;
    person(String name, String gender, String address, int age) {
        this.name = name;
        this.gender = gender;
        this.address = address;
        this.age = age;
    }
}

class employee extends person {
    int empid, salary;
    String cName, qualification;
    employee(String name, String gender, String address, int age, int empid,
        String cName, String qualification, int salary) {
        super(name, gender, address, age);
        this.cName = cName; this.empid = empid; this.salary = salary;
        this.qualification = qualification;
    }
}
```



```
}  
class Teacher extends employee {
```

```
    int tId;
```

```
    String subject, department;
```

```
    Teacher (int tId, String subject, String department, String name,  
            String gender, String address, int age, int empId, String cName,  
            String qualification, int salary)
```

```
    {  
        super (name, gender, address, age, empId, cName, qualification,  
                salary);
```

```
        this.tId = tId;
```

```
        this.subject = subject;
```

```
        this.department = department;
```

```
    void display() {
```

```
        System.out.println("\tTeacher id: " + tId);
```

```
        System.out.println("\tEmployee id: " + empId);
```

```
        System.out.println("\tName : " + name);
```

```
        System.out.println("\tSubject : " + subject);
```

```
        System.out.println("\tDepartment : " + department);
```

```
        System.out.println("\tGender : " + gender);
```

```
        System.out.println("\tAddress : " + address);
```

```
        System.out.println("\tAge : " + age);
```

```
        System.out.println("\tCompany name : " + cName);
```

```
        System.out.println("\tQualification : " + qualification);
```

```
        System.out.println("\tSalary : " + salary);
```

```
    }
```

```
}
```


class main {

public static void main (String args[]) {

int n, teacherId, age, empId, salary;

String name, gender, address, cName, subject, dept, qualification;

Scanner input = new Scanner(System.in);

System.out.print("Enter the no of teachers : ");

n = input.nextInt();

Teacher t[] = new Teacher[n];

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

teacherId = i+1;

System.out.println("Enter details of Teacher " + (i+1));

System.out.print("Enter employee id : ");

empId = input.nextInt();

System.out.print("Enter name : ");

name = input.next();

System.out.print("Enter subject : "); subject = input.next();

System.out.print("Enter department : "); dept = input.next();

System.out.print("Enter gender : "); gender = input.next();

System.out.print("Enter address : "); address = input.next();

System.out.print("Enter age : "); age = input.nextInt();

System.out.print("Enter company name : "); cName = input.next();

System.out.print("Enter qualification : "); qualification = input.next();

System.out.print("Enter salary : "); salary = input.nextInt();

t[i] = new Teacher(teacherId, subject, dept, name, gender, address, age, empId, cName, qualification, salary);

System.out.println("");

System.out.println("Details of students");

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

System.out.print("\n=====");

System.out.print("Teacher " + (i+1));

t[i].display(); input.close(); }

Result:- Program successfully executed & output obtained.

Output

Enter the no of teacher : 2

Enter details of teacher 1

Enter employee id :

100

Enter name :

Anu

Enter subject :

electronics

Enter department :

EEE

Enter gender :

Female

Enter address :

Thiruvalla

Enter company name :

Rit

Enter qualification :

Mtech

Enter salary :

2500

Enter details of teacher 2

Enter employee id :

102

Enter name :

Goku

Enter subject :

DS

Enter department : MCA

Enter gender : Male

Enter address : Kollam

Enter age : 35

Enter company name : Rit

Enter qualification : MCA

Enter salary : 2400

Details of Teachers

Teacher 1

Teacher id : 1

Employee id : 100

Name : Anu

Subject : Electronics

Department : EEE

Gender : Female

Address : Thiruvella

Age : 34

Company name : Rit

Qualification : Mtech

Salary : 2500

Teacher 2

Teacher id : 2

Employee id : 102

Name : Gokul

Subject : DS

Department : MCA

Gender : Male

Address : Kollam

Age : 35

Company name : Rit

Qualification : MCA

Salary : 2400

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

Source code

```
import java.util.*;
```

```
class Publisher {
```

```
    String name;
```

```
    int date;
```

```
    Publisher (String name, int date) {
```

```
        this.date = date;
```

```
        this.name = name;
```

```
    }
```

```
}
```

```
class Book extends Publisher {
```

```
    String title, author; int no-of-pages;
```

```
    Book (String title, String author, int no-of-pages, String name, int date) {
```

```
        super (name, date);
```

```
        this.author = author; this.title = title; this.no-of-pages = no-of-pages;
```

```
    }
```

```
    void displayDetails() {
```

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

```
        System.out.println("Author: " + author);
```

```
        System.out.println("Number of pages: " + no-of-pages);
```

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

```
        System.out.println("Publish date: " + date); }
```

```
}
```

```
class Fiction extends Book {
```

```
    static String type = "Fiction";
```

```
Fiction (String title, String author, int no-of-pages, String name,
    and date) {
```

```
    super(title, author, no-of-pages, name, date); }
```

```
void display { System.out.println("Book type : " + type);
    displayDetails (); }
}
```

```
class Literature extends Book {
```

```
    static String type = "Literature";
```

```
    Literature(String author, String author, int no-of-pages, String name, int date)
    { super(title, author, no-of-pages, name, date); }
```

```
void display() { System.out.println("Book type : " + type);
    displayDetails (); }
```

```
}
```

```
public class hierarchicalInheritance {
```

```
    static String title, author, pName;
```

```
    static int no-of-pages, date;
```

```
    static void readData Scanner inp = new Scanner(System.in);
```

```
    static void readData() {
```

```
        System.out.println("Enter the book title : ");
```

```
        title = inp.next();
```

```
        System.out.println("Enter the book author : ");
```

```
        author = inp.next();
```

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

```
        pName = inp.next();
```

```
        System.out.println("Enter the number of pages in book : ");
```

```
        no-of-pages = inp.nextInt();
```

```
        System.out.println("Enter the book publish date : ");
```

```

        date = Gup.nextInt();
    }

    public static void main (String args[]) {

        System.out.println("Enter book details of fiction type");
        readData();
        Fiction f = new Fiction (title, author, no-of-pages, gName, date);
        System.out.println("Enter book details of literature type");
        readData();
        Literature l = new Literature (title, author, no-of-pages, gName, date);
        System.out.println("Book details of fiction type");
        f.display();
        System.out.println("Book details of literature type");
        l.display();
        Gup.close(); // closing scanner object
    }
}

```

Result:- Program successfully executed and output obtained.

Output

***** Enter book details of fiction type *****

Enter the book title : Alchemist

Enter the book author : Paulo

Enter the ^{number} ~~pages~~ of pages in book : 1213

Enter the book publish date : 20072000

***** Enter details of literature type *****

Class Diagram

Enter the book title : To-kill-a-mocking-bird.

Enter the book author : Harper-Lee

Enter the book publisher name : Oxford

Enter the number of pages in book : 3443

Enter the book publish date : 31081800

***** Book details of fiction type *****

Book type : Fiction

Title : Alchemist

Author : Paulo

Number of pages : 1213

Publisher name : Oxford

Publish date : 20072000

***** Book details of literature type *****

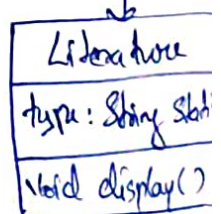
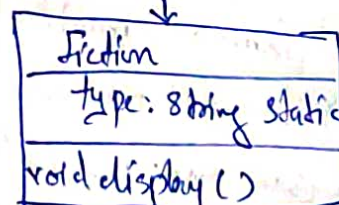
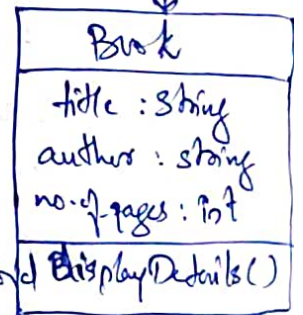
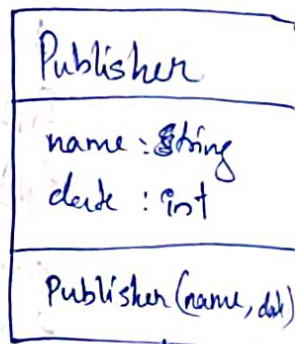
Title : To-kill-a-mocking-bird

Author : Harper-Lee

Number of pages : 3443

Publisher name : Oxford

Publish date : 31081800



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

Source Code

```
import util.*;
```

```
interface student {
```

```
    String sname = "Anandhu", gender = "Male";
```

```
    int mark = 70;
```

```
}
```

```
interface sports {
```

```
    String category = "Running";
```

```
    int place = 1, score = 20;
```

```
}
```

```
class Result implements student, sports {
```

```
    int total = score + mark;
```

```
    public void display () {
```

```
        System.out.println("Name: " + sname);
```

```
        System.out.println("Gender: " + gender);
```

```
        System.out.println("Marks recd: " + mark);
```

```
        System.out.println("Sports category: " + category);
```

```
        System.out.println("Place obtained: " + place);
```

```
        System.out.println("Marks recorded: " + score);
```

```
        System.out.println("\nTotal Marks: " + total); }
```

```
}
```

```
public class multipleInheritance {
```

```
    public static void main (String[] args) {
```

```
        Result r = new Result(); System.out.println("***** Details *****");
```

```
        r.display(); }
```

Output

***** Details *****

Name : Anandu

Gender : Male

Marks had : 70

Sports category : Running

Place obtained : 1

Mark rewarded : 20

Total Marks : 90

Result :: Program successfully
executed and o/p obtained.

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

Source Code

```
import java.util.*;

interface shape { double area();
                  double perimeter(); }

class Circle implements shape {
    int r;
    Circle(int r) { this.r = r; }
    public double area() { return 3.14 * r * r; }
    public double perimeter() { return 2 * 3.14 * r; }
}

class Rectangle implements shape {
    int l, b;
    Rectangle(int l, int b) { this.l = l; this.b = b; }
    public double area() { return l * b; }
    public double perimeter() { return 2 * (l + b); }
}

public class areaPerimeterInterface {
    public static void main(String[] args) {
        Scanner inp = new Scanner(System.in);
        boolean flag = true;
        do {
            System.out.println("***** Menu *****");
            System.out.println("1 - Circle");
            System.out.println("2 - Rectangle");
            System.out.println("0 - Exit");
        } while (flag);
    }
}
```

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

```
int c = inp.nextInt();
```

```
switch (c) {
```

```
    case 1: {
```

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

```
        Circle cr = new Circle(inp.nextInt());
```

```
        System.out.println("Area of circle is " + cr.area());
```

```
        System.out.println("Perimeter of circle is " + cr.perimeter());
```

```
        break; }
```

```
    case 2: {
```

```
        System.out.println("Enter length and breadth");
```

```
        Rectangle r = new Rectangle(inp.nextInt(), inp.nextInt());
```

```
        System.out.println("Area of rectangle is " + r.area());
```

```
        System.out.println("Perimeter of rectangle is " + r.perimeter());
```

```
        break; }
```

```
    case 0: {
```

```
        System.out.println("BYE....");
```

```
        flag = false;
```

```
        break; }
```

```
    }
```

```
    while (flag);
```

```
    inp.close();
```

```
}
```

```
}
```

Result :- Program successfully executed and output obtained.

Output

***** MENU *****

- 1 - Circle
- 2 - Rectangle
- 0 - Exit

Enter your choice : 1

Enter radius : 4

Area of circle is 50.24

Perimeter of circle is 25.12

Enter your choice : 2

Enter length and breadth : 3 4

Area of rectangle is 12.0

Perimeter of rectangle is 14.0

Enter your choice : 0

BYE.....

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

Source Code

interface amount {

void calculate();
}

class product implements amount {

static int netAmt, ordNo, date;

int productId, quantity, unit-price, total;

String name;

product(int productId, String name, int quantity, int unit-price) {

this.productId = productId;

this.name = name;

this.quantity = quantity;

this.unit-price = unit-price;

this.name = name;

}

```
public void calculate() {
```

```
    total = quantity * unit-price;
```

```
    net Amt = total;
```

```
}
```

```
void display() {
```

```
    System.out.println("Product Id : " + " " + name + " " + quantity +  
                        " " + unit-price + " " + total);
```

```
}
```

```
}
```

```
public class Bill {
```

```
    public static void main (String[] args) {
```

```
        product.date = 22092000;
```

```
        product.osdNo = 9355456;
```

```
        product p1 = new product (101, "A", 2, 25);
```

```
        product p2 = new product (102, "B", 1, 100);
```

```
        p1.calculate();
```

```
        p2.calculate();
```

```
        System.out.println("Order No: " + product.osdNo);
```

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

```
        System.out.println("In Product Id    Name    Quantity    unit-price    Total");
```

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

```
        p1.display();
```

```
        p2.display();
```

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

```
        System.out.println("||||| Net. Amount " + product.netAmt);
```

```
}
```

```
}
```

Result :- Program successfully executed and output obtained.

Output

Order No : 9355456

Date : 22/09/2000

Product Id	Name	Quantity	Unit-price	Total
101	A	2	25	50
102	B	1	100	100

Net Amount 150

8) Using the concept of method overriding, find the area of shapes, Rectangle, Circle and Square.

Source code

```
import java.util.*;
```

```
class rectangle {
```

```
    int l, b; rectangle() {}
```

```
    rectangle (int length, int breadth) { l = length; b = breadth; }
```

```
    double area() { return l * b; }
```

```
}
```

```
class circle extends rectangle {
```

```
    int r; circle() {}
```

```
    circle (int radius) { r = radius; }
```

```
    double area() { return (3.14 * r * r); }
```

```
}
```

```
class square extends circle {
```

```
    int s;
```

```
    square (int side) { s = side; }
```

```
    double area() { return (s * s); }
```

```
}
```

```
class overriding {
```

```
    public static void main (String args[]) {
```

```
        int l, b, r, side;
```

```
        Scanner s = new Scanner(System.in);
```

```
        System.out.println ("Enter length & breadth of the rectangle : ");
```

```
        l = s.nextInt();
```

```
        b = s.nextInt();
```

```
        rectangle rt = new rectangle (l, b);
```

```
        System.out.println ("Area: " + rt.area());
```

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

```
r = s.nextInt();
```

```
circle c = new circle(r);
```

```
System.out.println("Area : " + c.area());
```

```
System.out.println("Enter the side of the square : ");
```

```
side = s.nextInt();
```

```
rectangle sc = new square(side);
```

```
System.out.println("Area : " + sc.area());
```

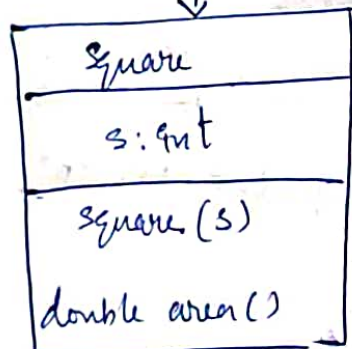
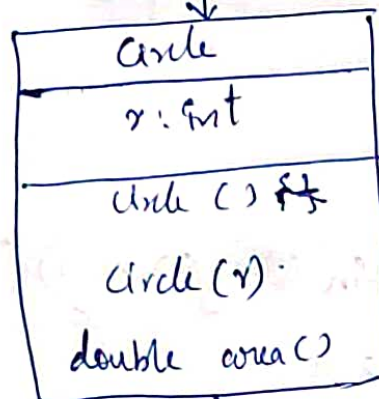
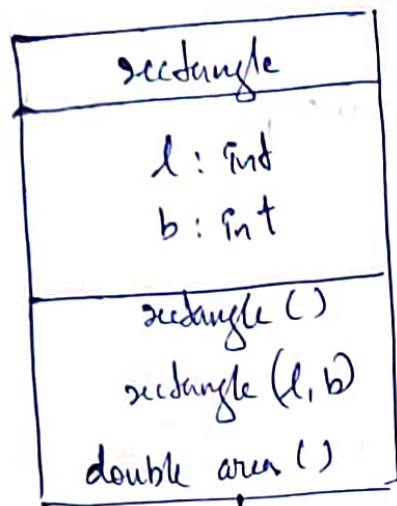
```
s.close();
```

```
}
```

```
}
```

Result :- Program successfully executed & o/p obtained.

Class diagram



Output

enter the length & breadth of the rectangle:

5 4

Area: 20.0

enter the radius:

3

Area: 28.25999999

enter the side of the square:

3

Area: 9.0

9) Create an Abstract class 'Shape' with an abstract method find Area to find the area of different shapes. Create subclasses Rectangle, Circle, and Square from Shape. Calculate and display area of Rectangle, Circle and Square

Source Code

```
abstract class Shape { abstract double area(); }
```

```
class Circle extends Shape {
```

```
    int r;
```

```
    Circle(int r) { this.r = r; }
```

```
    double area() { return 3.14 * r * r; }  
}
```

```
class Square extends Shape {
```

```
    int s;
```

```
    Square(int s) { this.s = s; }
```

```
    double area() { return s * s; }  
}
```

```
class Rectangle extends Shape { int l, b;  
    Rectangle(int l, int b) { this.l = l;  
                                this.b = b; }  
    double area() { return l * b; }  
}
```

```
public class abstractClass {
```

```
    public static void main(String[] args) {
```

```
        Circle c = new Circle(6); Square sq = new Square(6);
```

```
        Rectangle r = new Rectangle(2, 3);
```

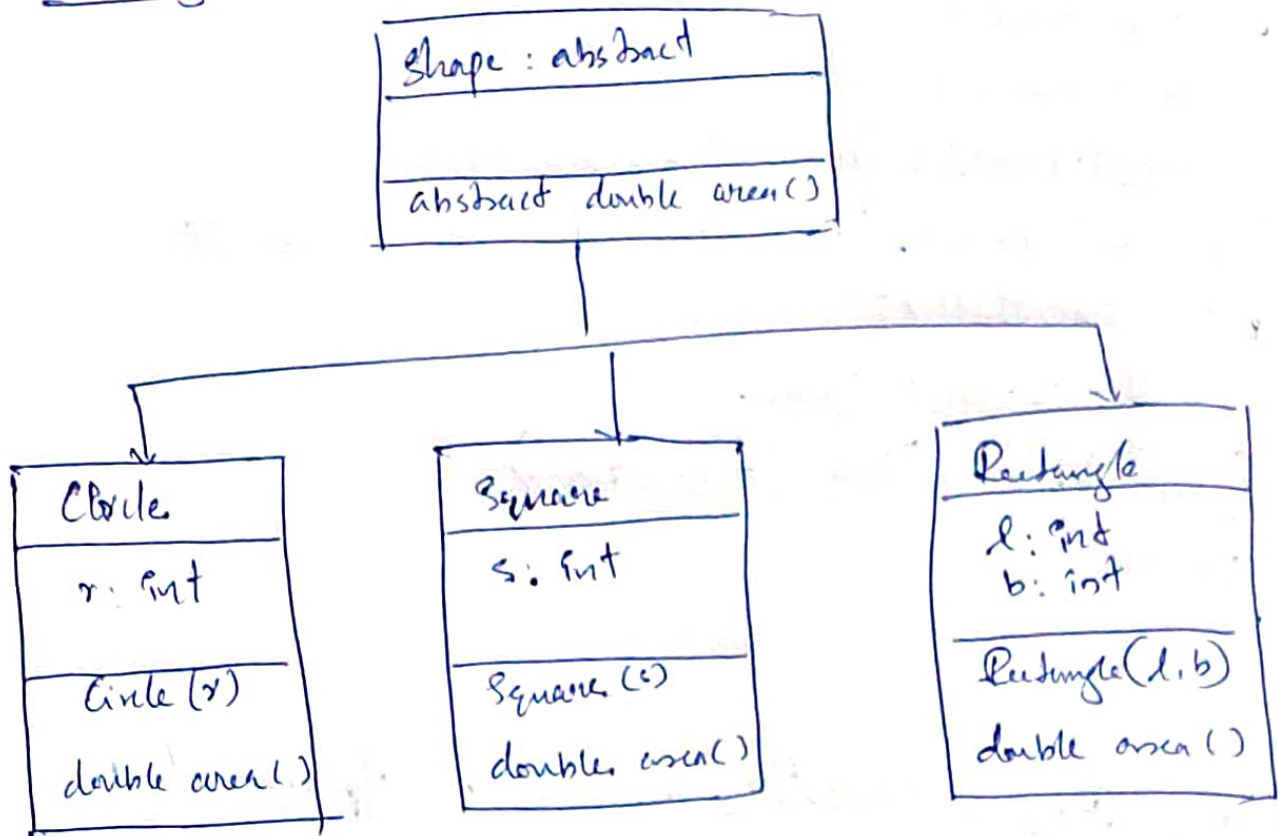
```
        System.out.println("Area of circle with radius " + c.r + " = " + c.area());
```

```
        System.out.println("Area of square with one side " + sq.s + " = " + sq.area());
```

```
        System.out.println("Area of rectangle with length " + r.l + " and breadth " + r.b + " = " + r.area());  
    }  
}
```

Result :- Program successfully executed & o/p obtained.

class diagram



Output

~~enter the length & breadth of the rectangle: 5 4~~

~~Area: 20.0~~

~~enter the radius: 3~~

~~Area: 28.2599999999~~

enter Area of circle with radius 6 = 113.039999999

Area of square with one side 6 = 36.0

Area of rectangle with length 2 and breadth 3 = 6.0