1. Create a class Area with a method findArea().
-Overload the method to calculate the areas of a square, rectangle, triangle and circle.

```
import java.util.Scanner;

public class area {

    double findArea(int side){
        return side*side;
    }

    double findArea(double l,double b){

        return l*b;

    }

    double findArea(double a,double b,double c){
        double semi=(a+b+c)/2;
        double Area=Math.sqrt(semi*(semi-a)*(semi-b)*(semi-c));
        return Area;
    }

    double findArea(double rad){
        return 3.14*rad*rad;

    }

    public static void main(String[] args) {
        area ar = new area();
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter 2 numbers:");
        double a=sc.nextDouble();
```

```
        double b=sc.nextDouble();
        int n=(int)a;
        System.out.println("Square : "+ ar.findArea(2));
        System.out.println("Rectangle : "+ ar.findArea(a,b));
        System.out.println("Triangle : "+ ar.findArea(3,5.3,7.1));
        System.out.println("Circle : "+ ar.findArea(b));
    }
}
```

2. Create a class shape with the following methods
   -draw() - to print the shape name
   -sides() - to print the number of sides
   Create 2 subclasses Square and triangle and Override the methods
   Create another class Test with the main method
   Create objects and call the methods

```
public class shape {

    shape(){
        System.out.println("From super class");
    }
    shape(int x){
        System.out.println(x);
    }
    void draw(){
        System.out.println("Name of the shape");
    }
    void sides(){
        System.out.println("Number of sides of the shape");
    }
}

class Square extends shape{
```

```
    Square(){
        super(4);
        System.out.println("From square constructer");
    }
    void draw(){

        System.out.println("Shape is Square");
    }
    void sides() {
        System.out.println("Square has 4 sides");
    }
}

class Triangle extends shape{
    void draw(){
        System.out.println("Shape is Triangle");
    }

    void sides() {
        System.out.println("Triangle has 3 sides");
    }
}

class test{
    public static void main(String[] args) {
        Square s = new Square();
        Triangle t = new Triangle();

        s.draw();
        s.sides();

        t.draw();
        t.sides();
    }
}
```

3. Create a class Product to store details of product sold by wholesaler with following details

Data members : name, amt, code

Parameterised constructer

Methods : void show()

Create another class Sales to compute total amount paid by the retailer including the fine

Data members : days, fine, totAmt

Parameterised Constructer which calls the parent constructor

void compute() : calc totAmt

void show() : displays the show() of parent class and the rest info

Using the main method call the functions

```
public class product {
    String name;
    int code;
    double amt;

    product(String nam, int c, double p){
        name = nam;
        code = c;
        amt = p;
    }

    void show(){
        System.out.println("Product details :");
```

```
        System.out.println("Name : "+name);
        System.out.println("Item code : "+code);
        System.out.println("Amount : "+amt);
    }
}

class sales extends product{

    int days,fine = 10;
    double totAmt;

    sales(String nam, int c, double p, int d) {
        super(nam, c, p);
        days = d;
    }

    void compute(){
        totAmt = amt + days*fine;
    }

    void show() {
        super.show();
        System.out.println("Days : " + days);
        System.out.println("Fine : "+fine);
        System.out.println("Total amount : "+totAmt);
    }

    public static void main(String[] args) {
        sales s = new sales("Laptop",1432,60000.00,5);
        s.compute();
        s.show();
    }
}
```