



## **TK1114 Computer Programming**

### **Lab 9**

#### **Classes**

#### **Simple Problem Solving**

29, 30 Nov & 1 Dec 2016

Name : \_\_\_\_\_

Matric Number : \_\_\_\_\_

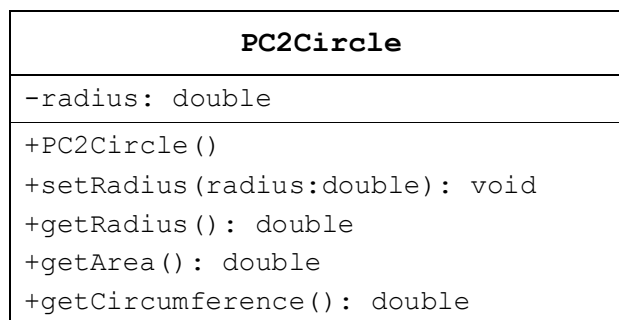
<b>9#</b>	<b>PC2 CIRCLE CLASS</b>	
	Input	Standard input
	Output	Standard output
	Topic	Classes : <b>Sample Solution</b>

### Problem Description

A class named `PC2Circle` is defined as follows.

- Private instance variable named `radius` that represents the radius of the circle.
- The constructor `PC2Circle()` that initialised `radius` to 0.0.
- `setRadius(double)` method to set a new radius value
- `getRadius()` method that returns the radius
- `getArea()` method that returns the area of the circle.
- `getCircumference()` method to returns the circumference of the circle.

The UML Class Diagram for class `PC2Circle` is as the following.



Write the code for `PC2Circle` class.

Write a program to test the `PC2Circle` class that reads the input and print the output as described below.

### Input

The first line contains an integer  $n$  ( $1 \leq n \leq 50$ ) which determines the number of test cases. Each of the following  $n$  lines contains a positive real number  $r$  ( $0.0 \leq r \leq 500.00$ ) which represents the radius of the circle.

### Output

For each test case, the output contains a line in the format "Case #x: ", where  $x$  is the case number (starting from 1) followed by the radius, area and circumference of the circle. Format the output in 4 decimal places.

**Sample Input Output**

Sample Input	Sample Output
2 9.0 5.5	Case #1: 9.000 254.4690 56.5487 Case #2: 5.500 95.0332 34.5575

Solution for this problem.

```
// File name: TestPC2Circle.java

import java.util.Scanner;
import java.text.DecimalFormat;

public class TestPC2Circle {
    public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);
        DecimalFormat df = new DecimalFormat("0.0000");
        double radius, area, circumference;

        int N = sc.nextInt();
        for (int i = 1; i <= N; i++) {
            PC2Circle myCircle = new PC2Circle(); // create PC2Circle object
            radius = sc.nextDouble();

            myCircle.setRadius(radius); // sets radius
            area = myCircle.getArea(); // returns area
            circumference = myCircle.getCircumference(); // returns

            System.out.println("Case #" + i + ": " +
                df.format(myCircle.getRadius()) + " " +
                df.format(area) + " " + df.format(circumference));
        }
    }

    class PC2Circle {                // NOT a public class
        private double radius;        // private instance variable

        public PC2Circle() {          // constructor without parameter
            radius = 0.0;
        }
        void setRadius(double rad) {
            radius = rad;
        }
        double getRadius() {
            return radius;
        }
        double getArea() {
            return radius * radius * Math.PI;
        }
        double getCircumference() {
            return 2 * Math.PI * radius;
        }
    }
}
```

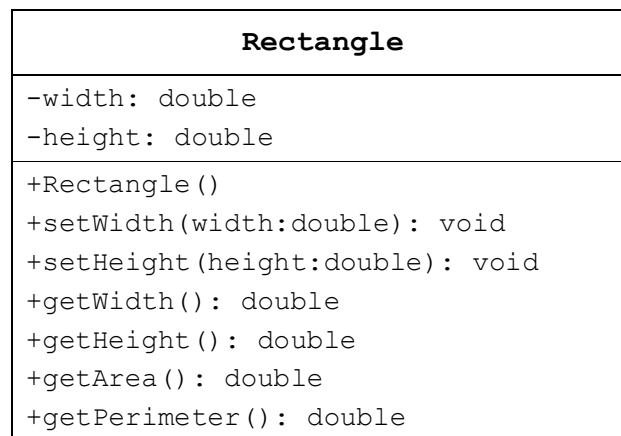
<b>9A</b>	<b>MY RECTANGLE</b>	
	Input	Standard input
	Output	Standard output
	Topic	Classes : Simple problem solving

### Problem Description

A class named `Rectangle` is defined as follows.

- Private instance variables named `width` and `height` that represents the width and height of the rectangle.
- The constructor `Rectangle()` that initialised `width` and `height` to 0.0 respectively.
- `setWidth(double)` method to set a new width
- `setHeight(double)` method to set a new height
- `getWidth()` method that returns the width
- `getHeight()` method that returns the height
- `getArea()` method that returns the area of the rectangle.
- `getPerimeter()` method to returns the perimeter of the rectangle.

The UML Class Diagram for class `Rectangle` is as the following.



Write the code for `Rectangle` class.

Write a program to test the `Rectangle` class that reads the input and print the output as described below.

**Input**

The first line contains an integer  $n$  ( $1 \leq n \leq 50$ ) which determines the number of test cases. Each of the following  $n$  lines contains two positive real number  $w$  and  $h$  ( $0.0 \leq w, h \leq 500.00$ ) which represents the width and height of the rectangle.

**Output**

For each test case, the output contains a line in the format "Case #x: ", where  $x$  is the case number (starting from 1) followed by the width, height, area and perimeter of the rectangle. (format the output in 2 decimal places).

**Sample Input Output**

Sample Input	Sample Output
2 9.0 7.33 5.5 8.88	Case #1: 9.00 7.33 65.97 32.66 Case #2: 5.50 8.88 48.84 28.76

Use the following program structure:

```
// File name: TestRectangle.java

// import statements

public class TestRectangle {
    public static void main(String [] args) {

        // code for test class
    }
}

class Rectangle {    // NOT a public class

    // code for Rectangle class
}
```

# MY EQ-TRIANGLE

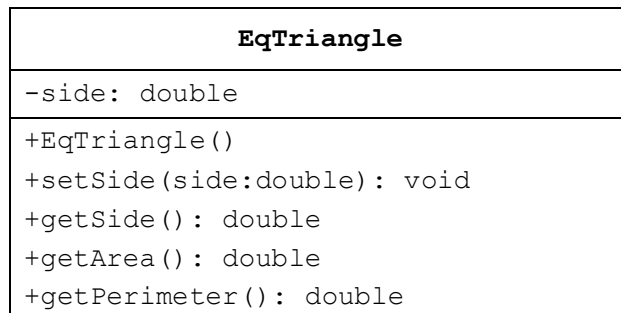
<b>9B</b>	Input	Standard input
	Output	Standard output
	Topic	Classes : Simple problem solving

### Problem Description

A class named `EqTriangle` that represents an equilateral triangle is defined as follows.

- Private instance variable named `side` that represents the sides of the rectangle.
- The constructor `EqTriangle()` that initialized the `side` to 0.0.
- `setWidth(double)` method to set a new width
- `setHeight(double)` method to set a new height
- `getWidth()` method that returns the width
- `getHeight()` method that returns the height
- `getArea()` method that returns the area of the triangle.
- `getPerimeter()` method to returns the perimeter of the triangle.

The UML Class Diagram for class `EqTriangle` is as the following.



Write the code for `EqTriangle` class.

Write a program to test the `EqTriangle` class that reads the input and print the output as described below.

### Input

The first line contains an integer  $n$  ( $1 \leq n \leq 50$ ) which determines the number of test cases. Each of the following  $n$  lines contains a positive real number `side` ( $0.0 \leq \text{side} \leq 500.00$ ) which represents the side of the equilateral triangle.

### Output

For each test case, the output contains a line in the format "Case #x: ", where x is the case number (starting from 1) followed by the side, area and perimeter of the triangle. (format the output in 4 decimal places).

**Sample Input Output**

Sample Input	Sample Output
2 9.0 5.5	Case #1: 9.0000 35.0740 18.0000 Case #2: 5.5000 13.0986 11.0000

Use the following program structure:

```
// File name: TestEqTriangle.java

// import statements

public class TestEqTriangle {
    public static void main(String [] args) {

        // code for test class
    }
}

class EqTriangle {    // NOT a public class

    // code for EqTriangle class
}
```