953232 OOAD
Software Engineering
College Of Arts, Media And Technology, Chiang Mai University
2nd Semester / Academic Year 2020

Lab Assignment 01 : Review on Class and Object

Name ....Sahachan Tippimwong................. Student ID ......622115039...... Section......701......

Class Creation

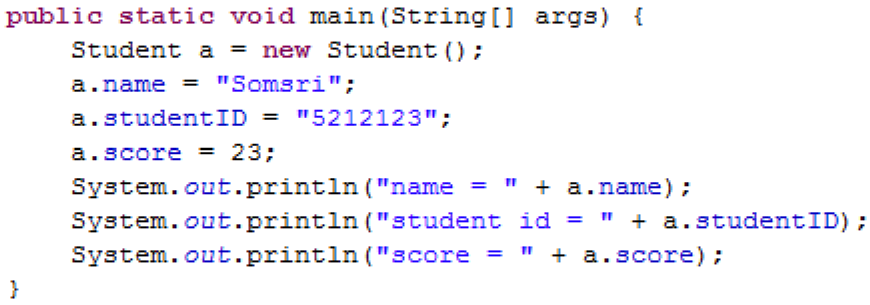
0. Class declaration

   Create a file name Student.java and insert the given source code.

```
public class Student{
   public String name;
   public String studentID;
   public int score;
   public void printScore(){
         System.out.println(“I received ”+score);
   }
}
```

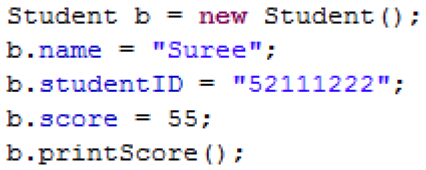
1. Object instantiation

   We can use object in the software by instantiating objects to use in software. Create a new class to run software by creating the “MainStudent1” class in the same project then adding the following code in class “MainStudent1”



```
public static void main(String[] args) {
    Student a = new Student();
    a.name = "Somsri";
    a.studentID = "5212123";
    a.score = 23;
    System.out.println("name = " + a.name);
    System.out.println("student id = " + a.studentID);
    System.out.println("score = " + a.score);
}
```

1.1. Add the following snippet at the end of the main methods



```
Student b = new Student();
b.name = "Suree";
b.studentID = "52111222";
b.score = 55;
b.printScore();
```

Then run the program, what is the additional output.

Answer……………………….55…………………………………………………………………………………….

1.2. Add the following snippet at the end of the main methods,

```
a = b;
a.printScore();
```

What is the additional input, and why it shows the same result as b not as same as the value it shown before?

Answer……………………….55 the reason that it shows the same result because the value in the object be is given to the object a…………………………………………….

2. Method declaration

A method defines the behavior of the object. A method required the return type in order to return the operation result. The return type can be any data type. *void* means there is no return data from the method. The method also required input parameter as the source information to do some operation inside the methods.

2.1. In this part, we will update the object of student class to compare the score between itself with another object. The method named "*compareScore*" is created. The method receives the other student object and compares the input student object with itself. The output is the integer value, if the return value is less than 0, the score of the object is less than the incoming object, if the return value is 0, both objects contain the same score, and if the return value is greater than 0, the current object contains greater score than new object. Adding the *compareScore* method as followed in the *Student* class

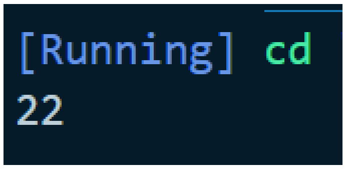
```
int compareScore(Student otherStudent){
    return otherStudent.score - this.score;
}
```

2.2. Create the new class name *MainStudent2* in *camt.oop.lab2* , and add the main method as followed

```
public static void main(String[] args) {
    Student somsak = new Student();
    somsak.name = "Somsak";
    somsak.studentID = "444111";
    somsak.score = 33;
    Student somchai = new Student();
    somchai.name = "Somchai";
    somchai.studentID = "444555";
    somchai.score = 55;
    System.out.println(somsak.compareScore(somchai));
}
```

run the program, what is the result ?

Post the code here



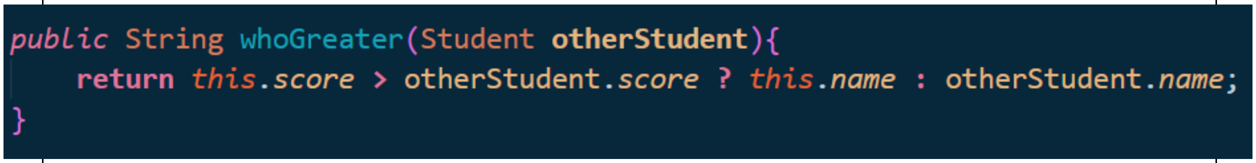
```
[Running] cd
22
```

2.3. Create new class name MainStudent3 in the same package, then create three student objects using the information shown in the table below:

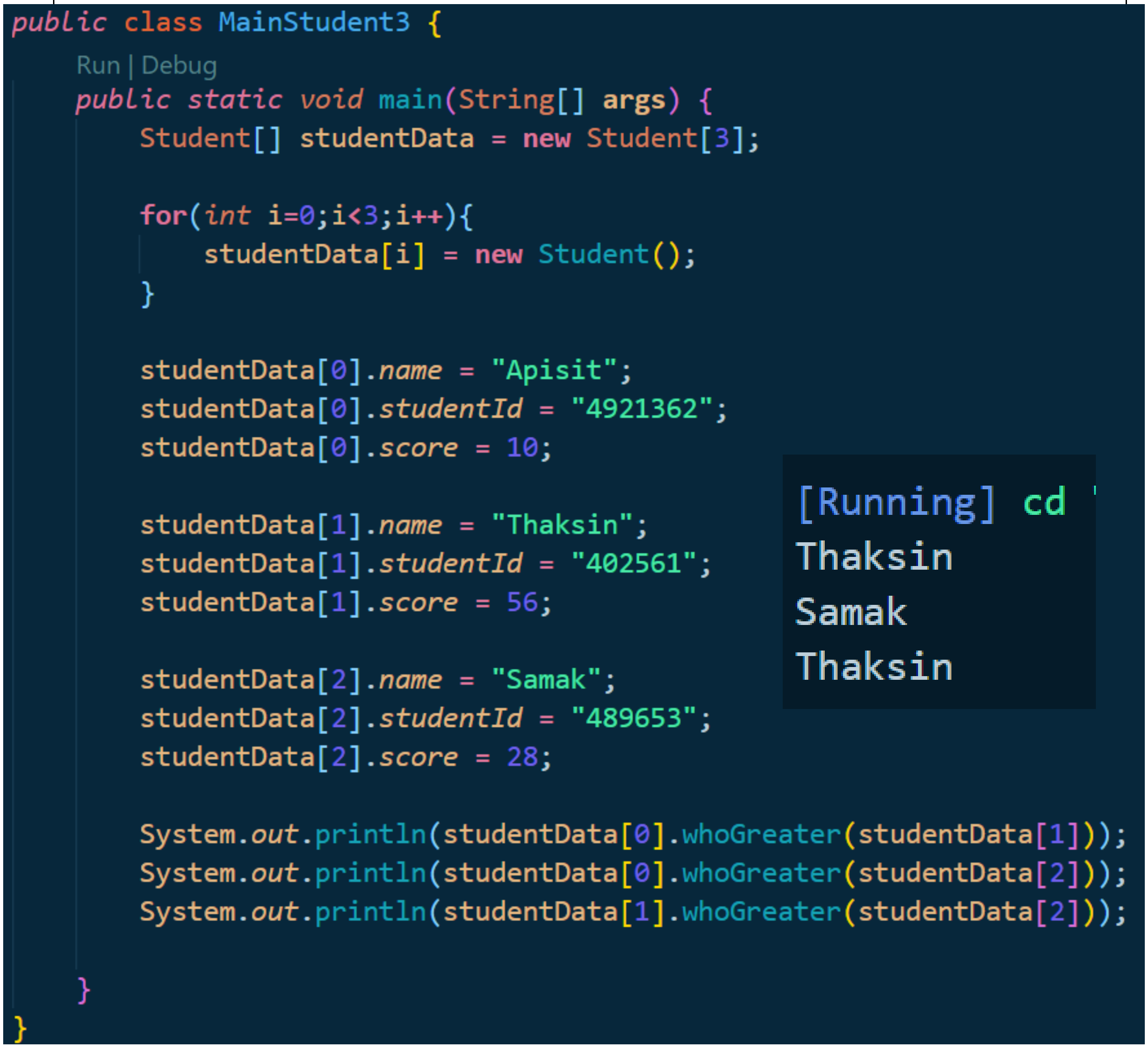
Name	StudentID	score
Apisit	“4921362”	10
Thaksin	“402561”	56
Samak	“489653”	28

Then write the code to compare any two object and print the name of object which contains greater score.

Implement the compare code



```java
public String whoGreater(Student otherStudent){
    return this.score > otherStudent.score ? this.name : otherStudent.name;
}
```



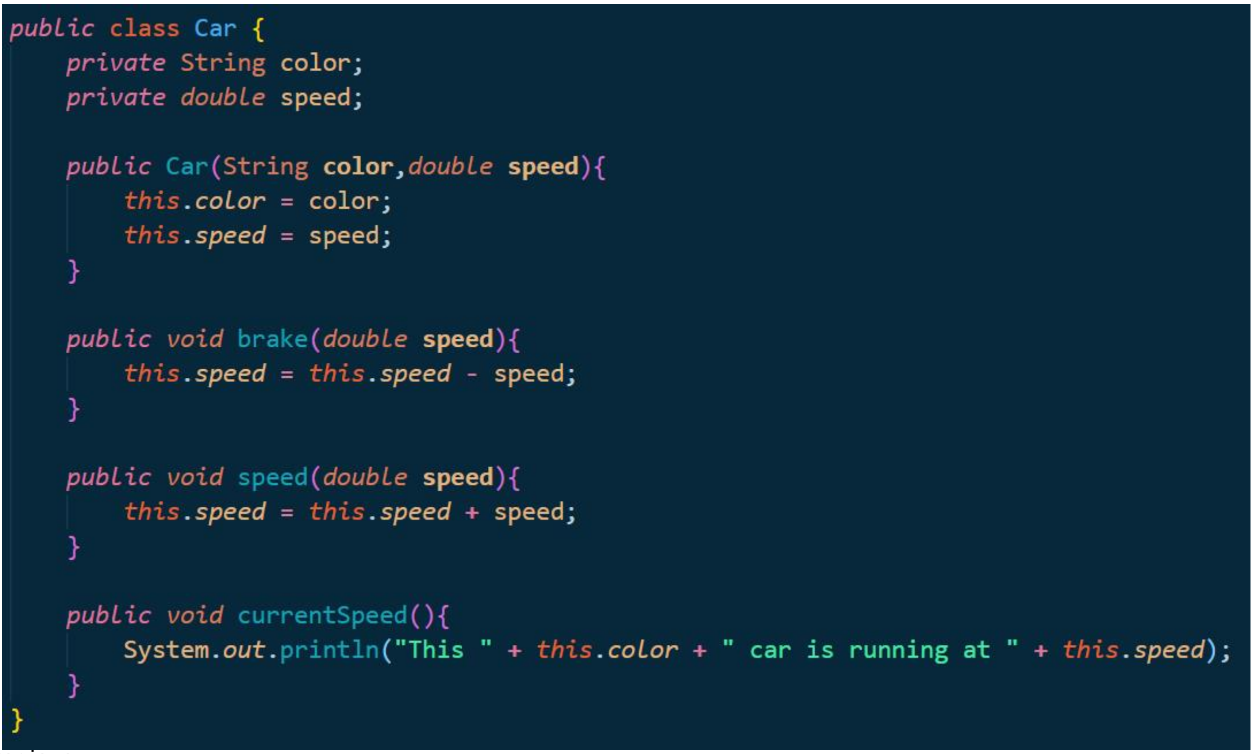
```java
public class MainStudent3 {
    Run | Debug
    public static void main(String[] args) {
        Student[] studentData = new Student[3];

        for(int i=0;i<3;i++){
            studentData[i] = new Student();
        }

        studentData[0].name = "Apisit";
        studentData[0].studentId = "4921362";
        studentData[0].score = 10;

        studentData[1].name = "Thaksin";
        studentData[1].studentId = "402561";
        studentData[1].score = 56;

        studentData[2].name = "Samak";
        studentData[2].studentId = "489653";
        studentData[2].score = 28;

        System.out.println(studentData[0].whoGreater(studentData[1]));
        System.out.println(studentData[0].whoGreater(studentData[2]));
        System.out.println(studentData[1].whoGreater(studentData[2]));

    }
}
```

```
[Running] cd '
Thaksin
Samak
Thaksin
```

3. Create a program to simulate a car. A car is characterized by the manufacturer, the color and the speed. The speed represents the current speed of the car. When user applies the brake, the car will slow down the speed. On the other hand, when user speeds up the car, the speed of the car will be increased. In this system, the user will give the speed value to control both the speed up process and application of brake. For example, given that the current speed is 60.0 km/h. If user applies the brake with 10.0 degrees, the current will go down to 50.0 (60.0-10.0). If user speeds up with 15.5 degrees, the current will go down to 75.2 (60.0+15.5).

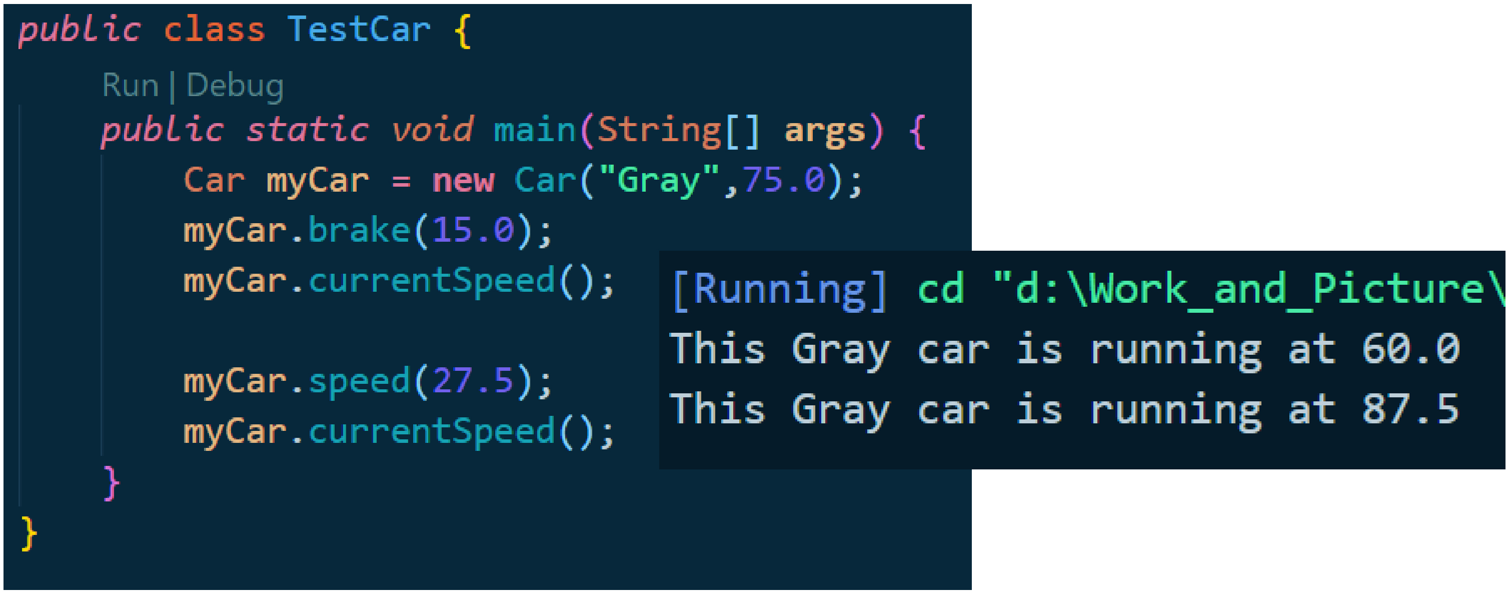
Implement the source code.

Car.java



```java
public class Car {
    private String color;
    private double speed;

    public Car(String color,double speed){
        this.color = color;
        this.speed = speed;
    }

    public void brake(double speed){
        this.speed = this.speed - speed;
    }

    public void speed(double speed){
        this.speed = this.speed + speed;
    }

    public void currentSpeed(){
        System.out.println("This " + this.color + " car is running at " + this.speed);
    }
}
```

TestCar.java



```java
public class TestCar {
    public static void main(String[] args) {
        Car myCar = new Car("Gray",75.0);
        myCar.brake(15.0);
        myCar.currentSpeed();

        myCar.speed(27.5);
        myCar.currentSpeed();
    }
}
```

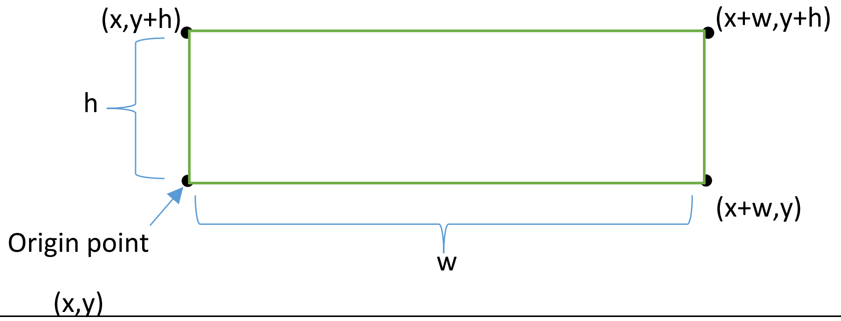
```
[Running] cd "d:\Work_and_Picture\
This Gray car is running at 60.0
This Gray car is running at 87.5
```

4. Create a program to simulate a rectangle. The rectangle is a 4 side 2D object. When user create a rectangle, the user can **either** input

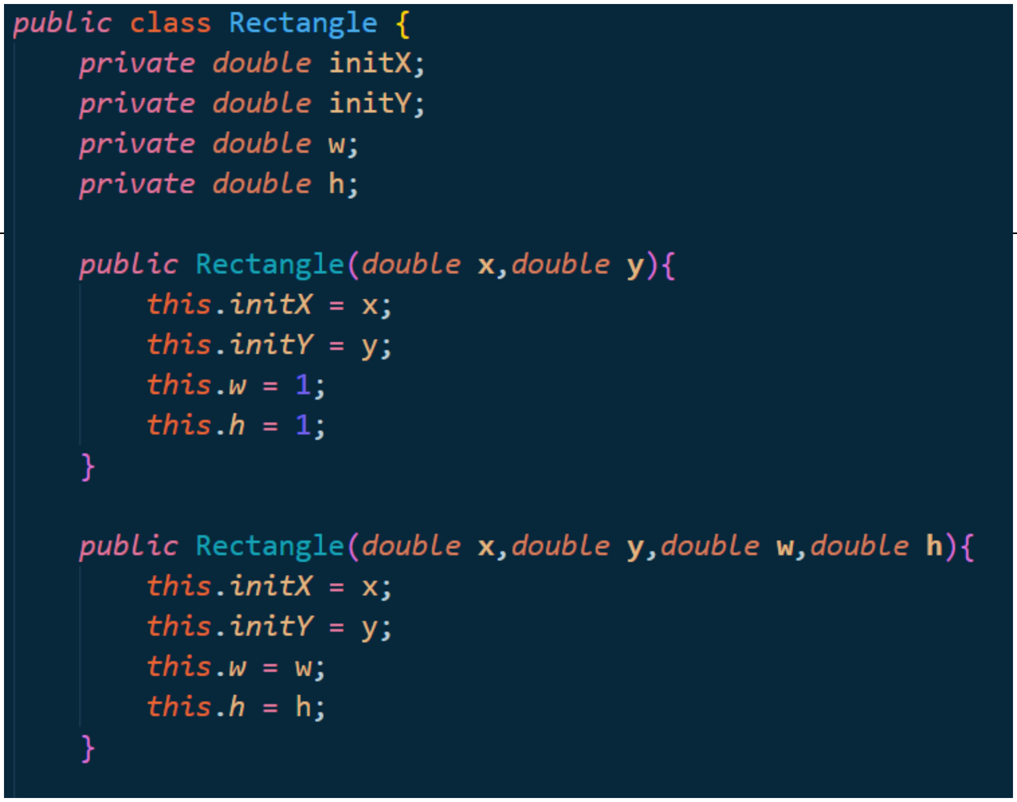
- the origin point (the width and the height will be automatically set to 1), or
- the origin point, the width (w) and the height (h).

The user can view the area of the rectangle

- get the co-ordinates of the corner points of the rectangle. and
- Get the area of the rectangle.

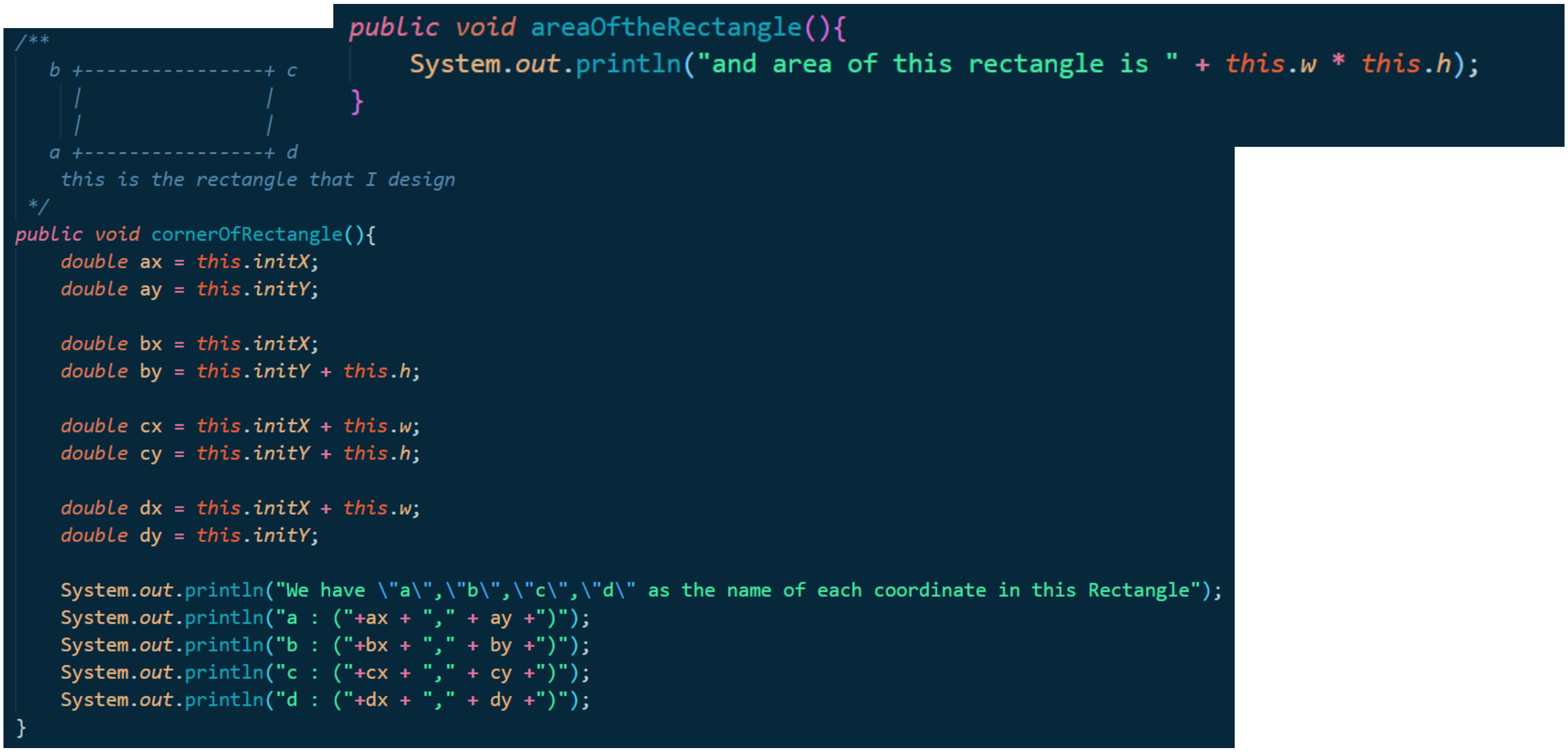


Rectangle.java (1)



```java
public class Rectangle {
    private double initX;
    private double initY;
    private double w;
    private double h;

    public Rectangle(double x,double y){
        this.initX = x;
        this.initY = y;
        this.w = 1;
        this.h = 1;
    }

    public Rectangle(double x,double y,double w,double h){
        this.initX = x;
        this.initY = y;
        this.w = w;
        this.h = h;
    }
```

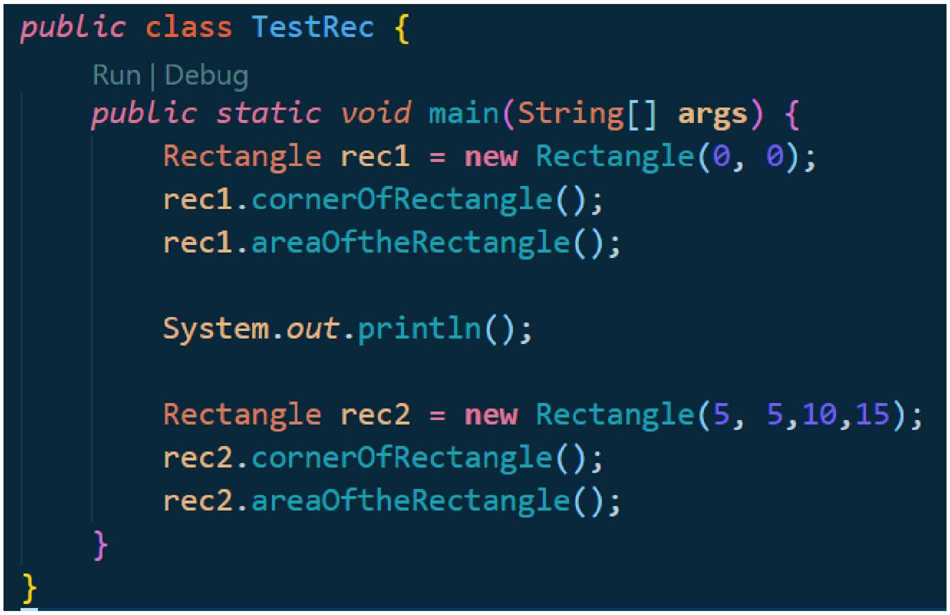
Reactangle.java(2)



```java
public void areaOftheRectangle(){
    System.out.println("and area of this rectangle is " + this.w * this.h);
}
```

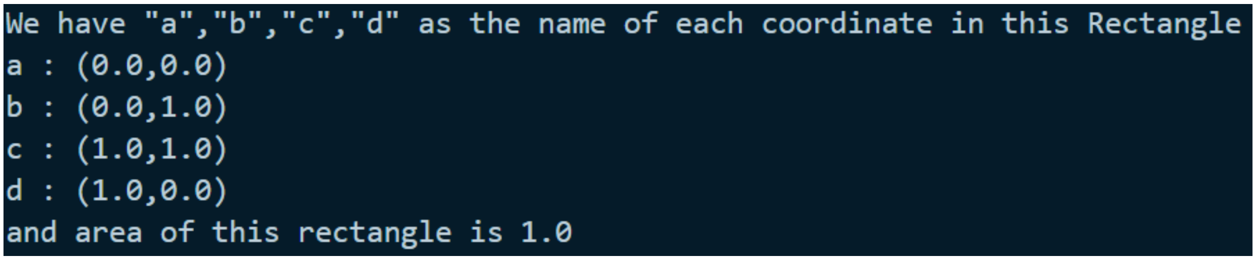
```java
/**
    b +----------------+ c
      |                |
      |                |
    a +----------------+ d
     this is the rectangle that I design
 */
public void cornerOfRectangle(){
    double ax = this.initX;
    double ay = this.initY;

    double bx = this.initX;
    double by = this.initY + this.h;

    double cx = this.initX + this.w;
    double cy = this.initY + this.h;

    double dx = this.initX + this.w;
    double dy = this.initY;

    System.out.println("We have \"a\",\"b\",\"c\",\"d\" as the name of each coordinate in this Rectangle");
    System.out.println("a : ("+ax + "," + ay +")");
    System.out.println("b : ("+bx + "," + by +")");
    System.out.println("c : ("+cx + "," + cy +")");
    System.out.println("d : ("+dx + "," + dy +")");
}
```

TestRec.java



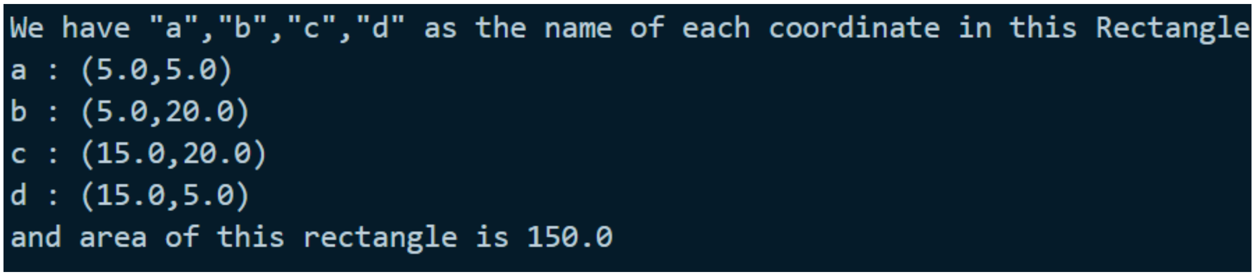
```java
public class TestRec {
    Run | Debug
    public static void main(String[] args) {
        Rectangle rec1 = new Rectangle(0, 0);
        rec1.cornerOfRectangle();
        rec1.areaOftheRectangle();

        System.out.println();

        Rectangle rec2 = new Rectangle(5, 5,10,15);
        rec2.cornerOfRectangle();
        rec2.areaOftheRectangle();
    }
}
```

Output for rec1



```
We have "a","b","c","d" as the name of each coordinate in this Rectangle
a : (0.0,0.0)
b : (0.0,1.0)
c : (1.0,1.0)
d : (1.0,0.0)
and area of this rectangle is 1.0
```

Output for rec2



```
We have "a","b","c","d" as the name of each coordinate in this Rectangle
a : (5.0,5.0)
b : (5.0,20.0)
c : (15.0,20.0)
d : (15.0,5.0)
and area of this rectangle is 150.0
```