JOBSHEET 3

Array of Object

Name
Sherly Lutfi Azkiah Sulistyawati

NIM
2341720241

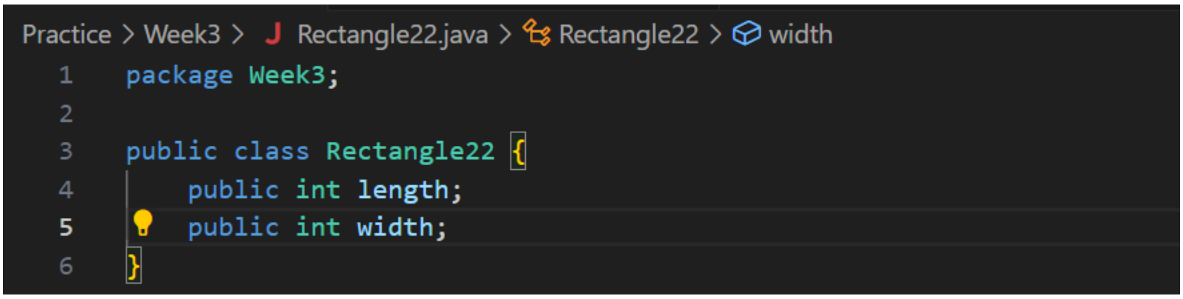
Class
1I

Major
Information Technology

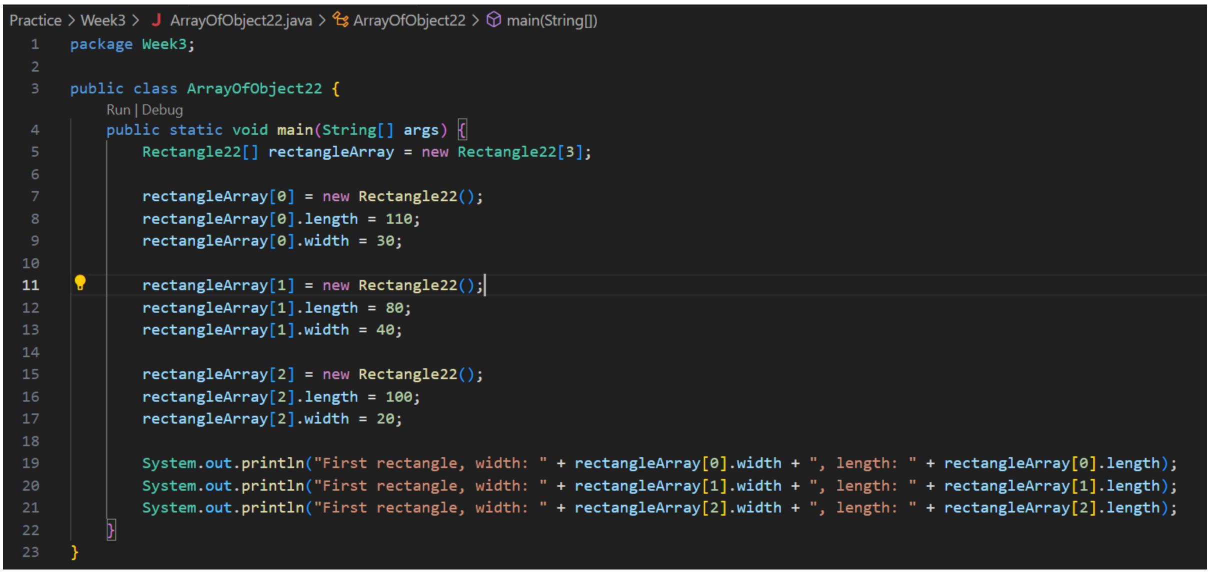
Study Program
D4 Informatics Engineering

Lab Activity 1: Create, insert, and display Array of Object

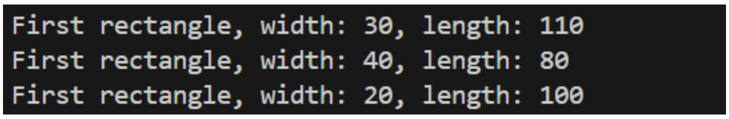
Result



```java
package Week3;

public class Rectangle22 {
    public int length;
    public int width;
}
```


Practice > Week3 > ArrayOfObject22.java > ArrayOfObject22 > main(String[])
Run | Debug

```java
package Week3;

public class ArrayOfObject22 {
    public static void main(String[] args) {
        Rectangle22[] rectangleArray = new Rectangle22[3];

        rectangleArray[0] = new Rectangle22();
        rectangleArray[0].length = 110;
        rectangleArray[0].width = 30;

        rectangleArray[1] = new Rectangle22();
        rectangleArray[1].length = 80;
        rectangleArray[1].width = 40;

        rectangleArray[2] = new Rectangle22();
        rectangleArray[2].length = 100;
        rectangleArray[2].width = 20;

        System.out.println("First rectangle, width: " + rectangleArray[0].width + ", length: " + rectangleArray[0].length);
        System.out.println("First rectangle, width: " + rectangleArray[1].width + ", length: " + rectangleArray[1].length);
        System.out.println("First rectangle, width: " + rectangleArray[2].width + ", length: " + rectangleArray[2].length);
    }
}
```



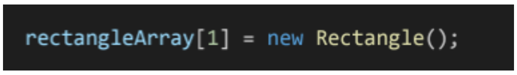
```
First rectangle, width: 30, length: 110
First rectangle, width: 40, length: 80
First rectangle, width: 20, length: 100
```

Question

1. Based on part 1.2, does the class that are going to be used as an array of object must have attributes and methods? Please explain!

   Answer: It doesn't have to, depends on the condition of the program we create. But, normally the class have attributes and methods. Attributes define the properties of the object, while methods define the behaviour or actions that object can perform.

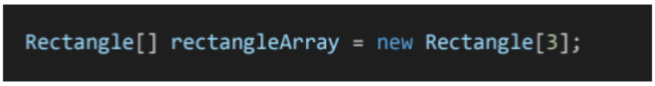
2. Does class Rectangle have constructor? If not, why we instantiate the object as follows?

   ```java
   rectangleArray[1] = new Rectangle();
   ```

Answer: The class Rectangle doesn't have explicitly define a constructor, that will be define as a default constructor. So, when we instantiate the object like the example, it calls the default constructor.

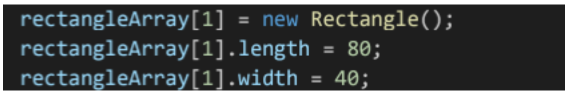
3. What's the meaning of this line of code?



```
Rectangle[] rectangleArray = new Rectangle[3];
```

Answer: That line of code Rectangle[] declares rectangleArray as an array that can hold objects of type Rectangle. rectangleArray = new Rectangle[3] initialize rectangleArray to have a length of 3.

4. What's the meaning of these lines of code?



```
rectangleArray[1] = new Rectangle();
rectangleArray[1].length = 80;
rectangleArray[1].width = 40;
```

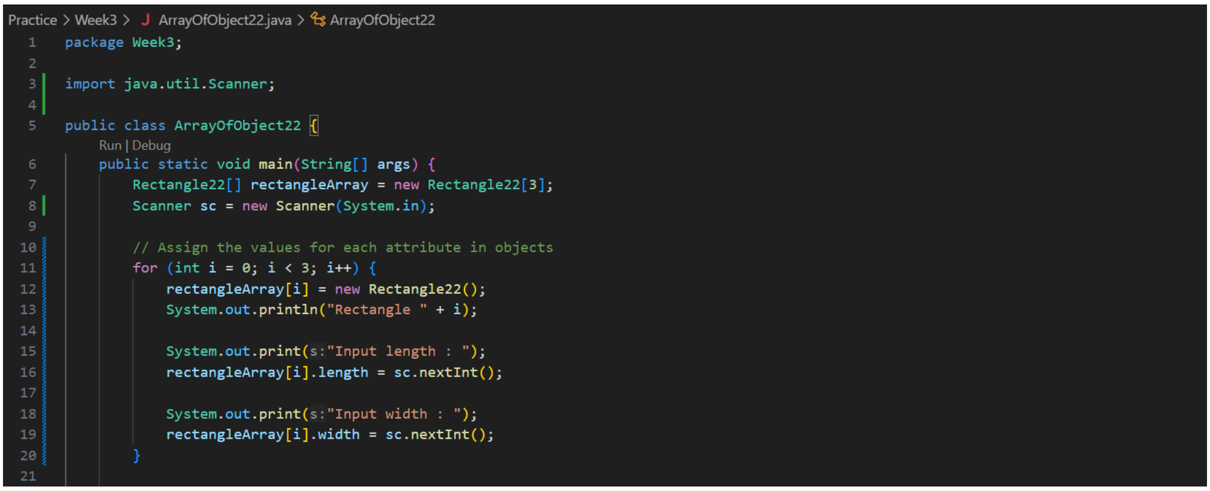
Answer: We instantiate a new Rectangle object at index 1 and set the attributes (length and width) of the Rectangle object at index 1 in rectangleArray.

5. Why ArrayOfObject class and Rectangle class should be separated?

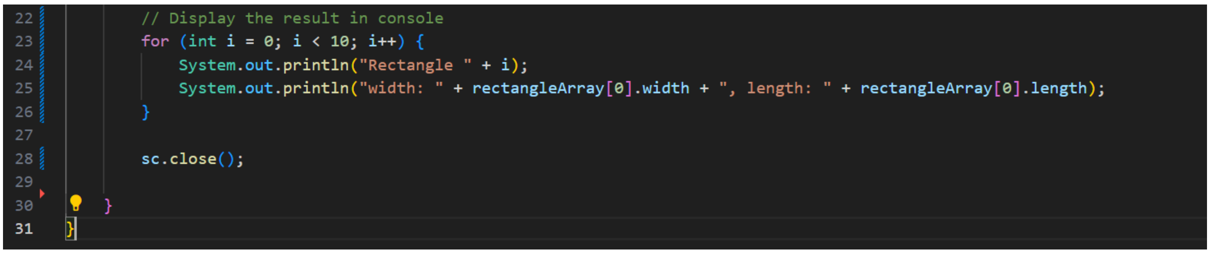
Answer: Separating classes help us to organized code. Each class should have a single responsibility. ArrayOfObject class is likely to manage the array and its manipulation, while Rectangle class represents the concept of a rectangle with its own attributes and methods.

Lab Activity 2: Input data into Array of Objects using Loops

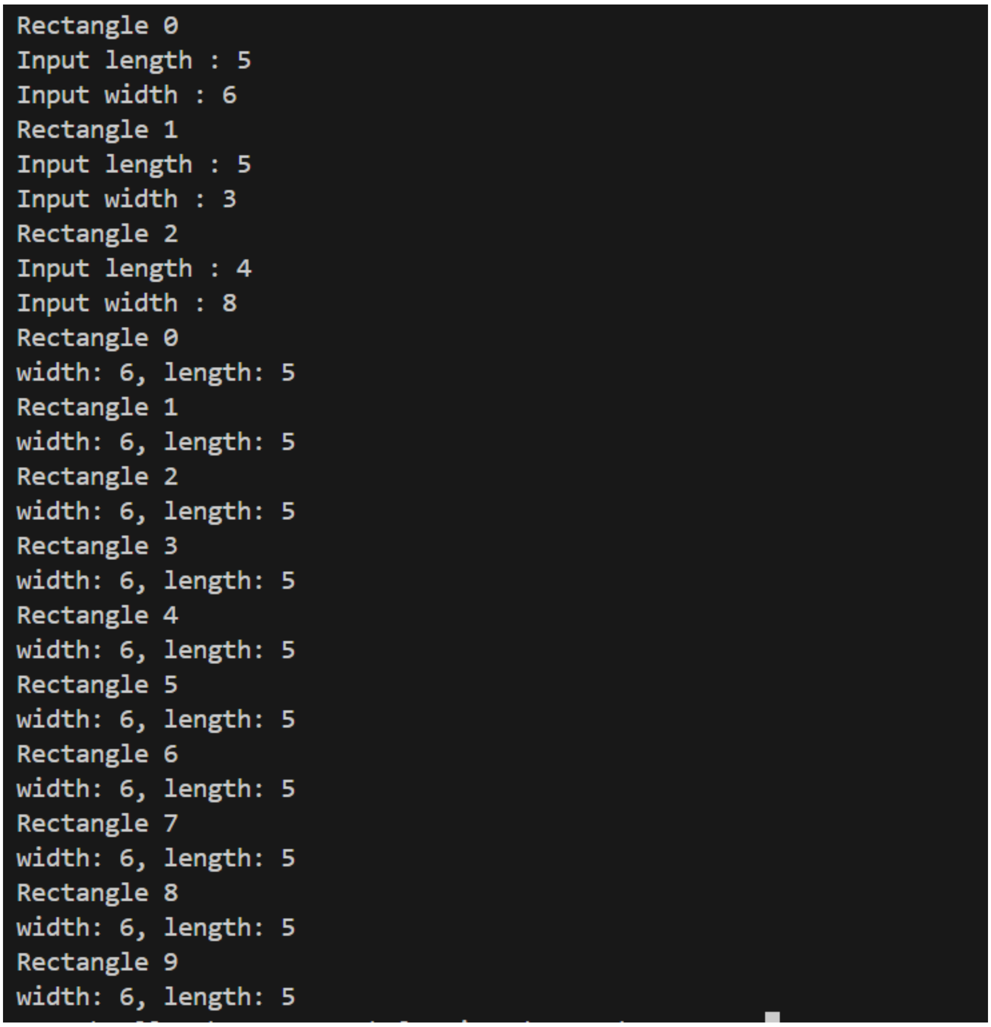
Result



```
package Week3;

import java.util.Scanner;

public class ArrayOfObject22 {
    public static void main(String[] args) {
        Rectangle22[] rectangleArray = new Rectangle22[3];
        Scanner sc = new Scanner(System.in);

        // Assign the values for each attribute in objects
        for (int i = 0; i < 3; i++) {
            rectangleArray[i] = new Rectangle22();
            System.out.println("Rectangle " + i);

            System.out.print(s:"Input length : ");
            rectangleArray[i].length = sc.nextInt();

            System.out.print(s:"Input width : ");
            rectangleArray[i].width = sc.nextInt();
        }

```



```
        // Display the result in console
        for (int i = 0; i < 10; i++) {
            System.out.println("Rectangle " + i);
            System.out.println("width: " + rectangleArray[0].width + ", length: " + rectangleArray[0].length);
        }

        sc.close();

    }
}
```



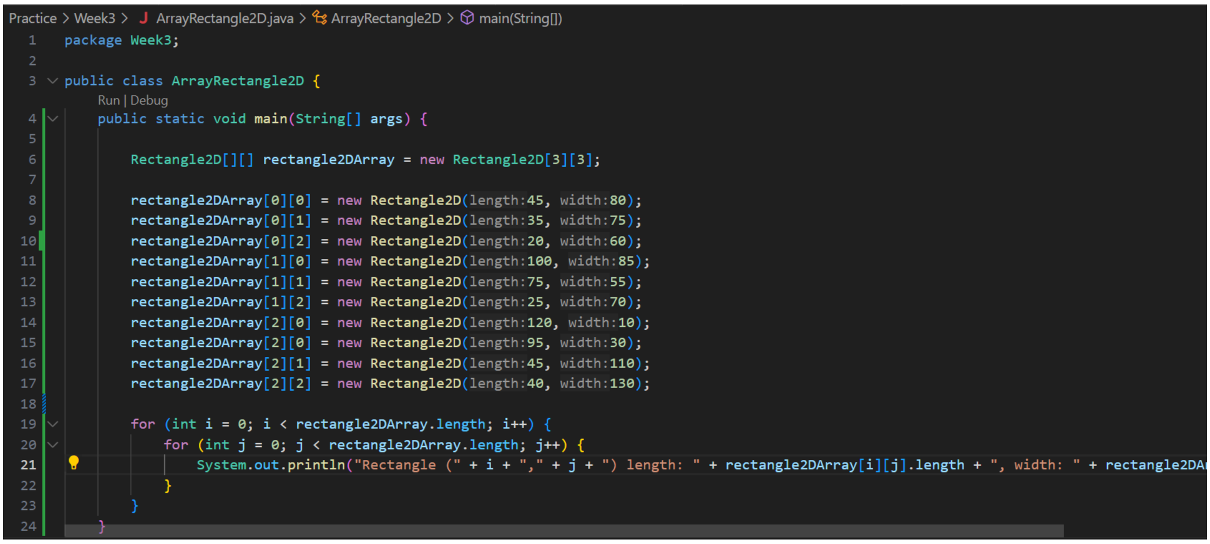
```
Rectangle 0
Input length : 5
Input width : 6
Rectangle 1
Input length : 5
Input width : 3
Rectangle 2
Input length : 4
Input width : 8
Rectangle 0
width: 6, length: 5
Rectangle 1
width: 6, length: 5
Rectangle 2
width: 6, length: 5
Rectangle 3
width: 6, length: 5
Rectangle 4
width: 6, length: 5
Rectangle 5
width: 6, length: 5
Rectangle 6
width: 6, length: 5
Rectangle 7
width: 6, length: 5
Rectangle 8
width: 6, length: 5
Rectangle 9
width: 6, length: 5
```

Question

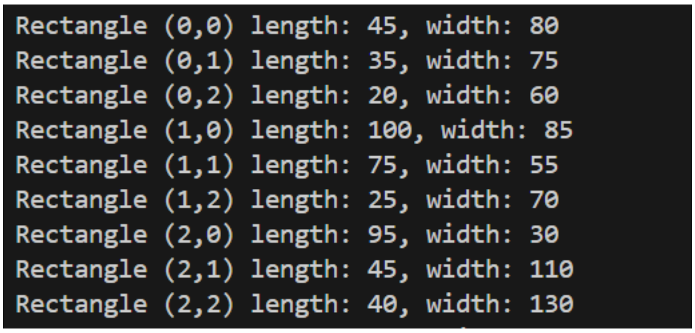
1. Does array of object can be implemented on 2D array?

   Answer: Yes, it does. Array of object can be implemented on 2D array.

2. If yes, then please give an example. Otherwise, please explain.

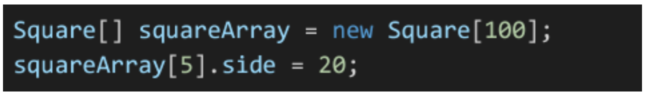

Practice > Week3 > ArrayRectangle2D.java > ArrayRectangle2D > main(String[])
Run | Debug

```java
package Week3;

public class ArrayRectangle2D {
    public static void main(String[] args) {

        Rectangle2D[][] rectangle2DArray = new Rectangle2D[3][3];

        rectangle2DArray[0][0] = new Rectangle2D(length:45, width:80);
        rectangle2DArray[0][1] = new Rectangle2D(length:35, width:75);
        rectangle2DArray[0][2] = new Rectangle2D(length:20, width:60);
        rectangle2DArray[1][0] = new Rectangle2D(length:100, width:85);
        rectangle2DArray[1][1] = new Rectangle2D(length:75, width:55);
        rectangle2DArray[1][2] = new Rectangle2D(length:25, width:70);
        rectangle2DArray[2][0] = new Rectangle2D(length:120, width:10);
        rectangle2DArray[2][0] = new Rectangle2D(length:95, width:30);
        rectangle2DArray[2][1] = new Rectangle2D(length:45, width:110);
        rectangle2DArray[2][2] = new Rectangle2D(length:40, width:130);

        for (int i = 0; i < rectangle2DArray.length; i++) {
            for (int j = 0; j < rectangle2DArray.length; j++) {
                System.out.println("Rectangle (" + i + "," + j + ") length: " + rectangle2DArray[i][j].length + ", width: " + rectangle2DA
            }
        }
    }
```



```
Rectangle (0,0) length: 45, width: 80
Rectangle (0,1) length: 35, width: 75
Rectangle (0,2) length: 20, width: 60
Rectangle (1,0) length: 100, width: 85
Rectangle (1,1) length: 75, width: 55
Rectangle (1,2) length: 25, width: 70
Rectangle (2,0) length: 95, width: 30
Rectangle (2,1) length: 45, width: 110
Rectangle (2,2) length: 40, width: 130
```

3. There is a **Square** class that has an attribute **side** with integer as its data type. There will be an error when we run this code, why?



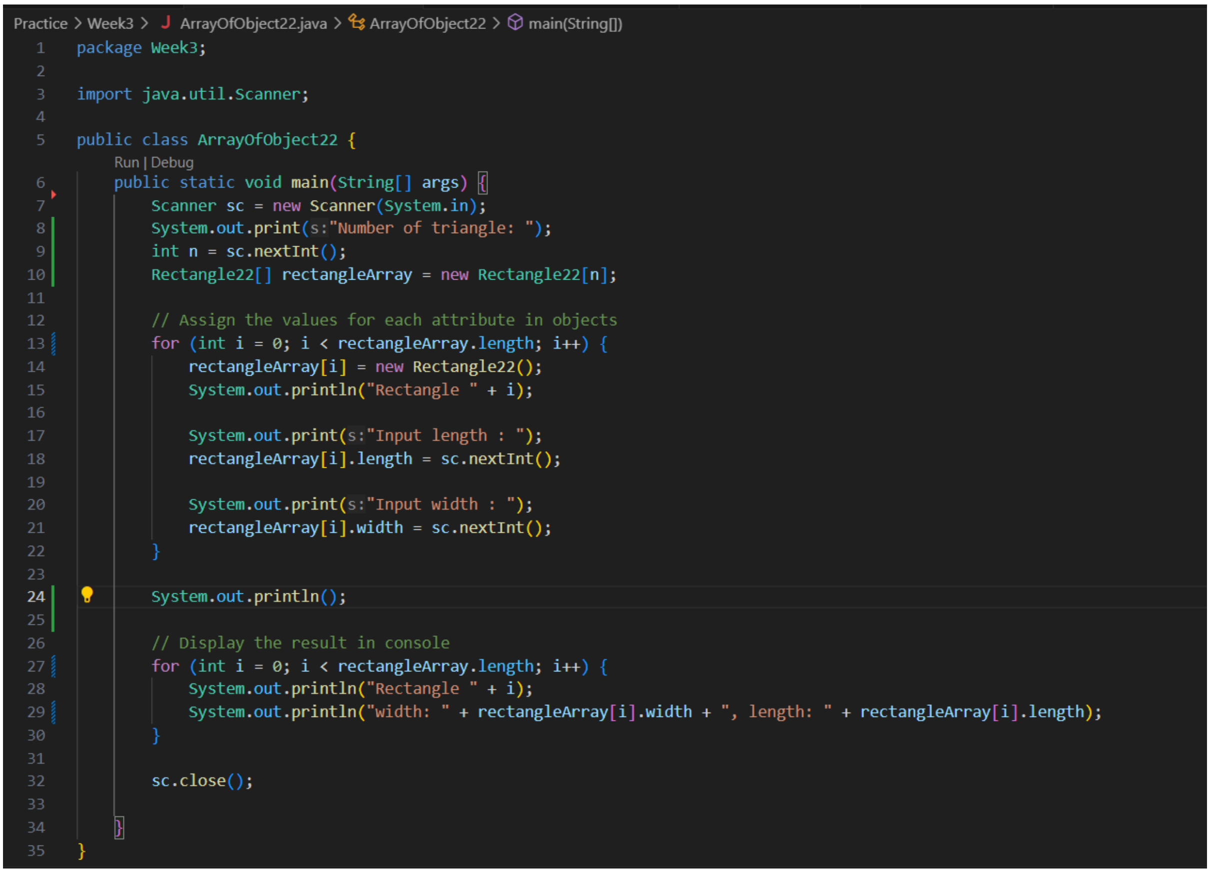
```java
Square[] squareArray = new Square[100];
squareArray[5].side = 20;
```

Answer: There will be an error when we run this code because it did not have instantiation. If we want to run the code, we must create the instantiation first.

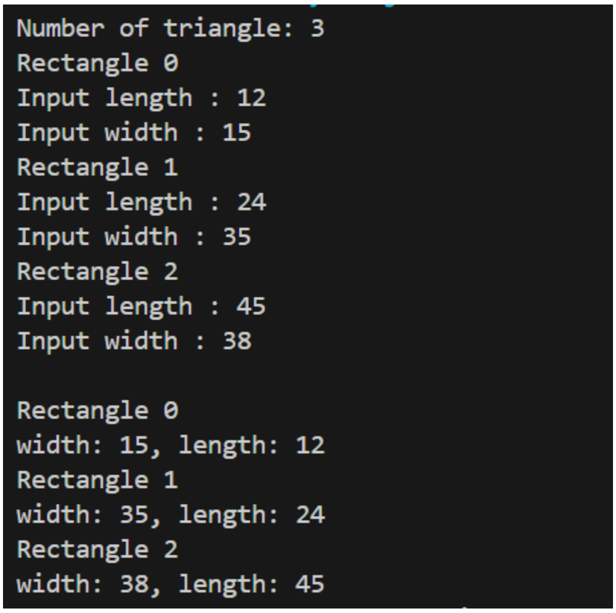
squareArray[5] = new Square();

4. Modify the code on part 1.3 so that the length of the array will be defined from user input.

Answer:


Practice > Week3 > ArrayOfObject22.java > ArrayOfObject22 > main(String[])
Run | Debug

```java
package Week3;

import java.util.Scanner;

public class ArrayOfObject22 {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print(s:"Number of triangle: ");
        int n = sc.nextInt();
        Rectangle22[] rectangleArray = new Rectangle22[n];

        // Assign the values for each attribute in objects
        for (int i = 0; i < rectangleArray.length; i++) {
            rectangleArray[i] = new Rectangle22();
            System.out.println("Rectangle " + i);

            System.out.print(s:"Input length : ");
            rectangleArray[i].length = sc.nextInt();

            System.out.print(s:"Input width : ");
            rectangleArray[i].width = sc.nextInt();
        }

        System.out.println();

        // Display the result in console
        for (int i = 0; i < rectangleArray.length; i++) {
            System.out.println("Rectangle " + i);
            System.out.println("width: " + rectangleArray[i].width + ", length: " + rectangleArray[i].length);
        }

        sc.close();

    }
}
```



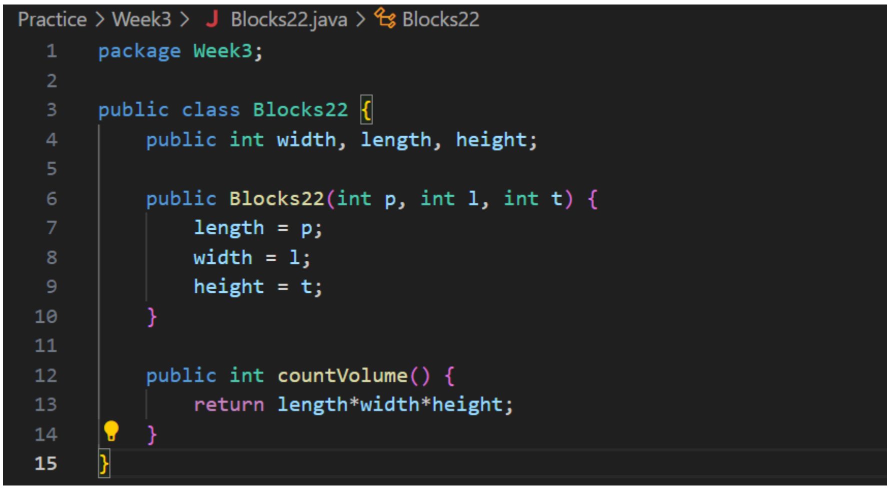
```
Number of triangle: 3
Rectangle 0
Input length : 12
Input width : 15
Rectangle 1
Input length : 24
Input width : 35
Rectangle 2
Input length : 45
Input width : 38

Rectangle 0
width: 15, length: 12
Rectangle 1
width: 35, length: 24
Rectangle 2
width: 38, length: 45
```

5. Can we duplicate the instantiation process in array of objects? For example, we assign the object in **ppArray[i]** and **ppArray[0]**, the instantiation process of **ppArray[0]** will be done twice. What's the effect of this?

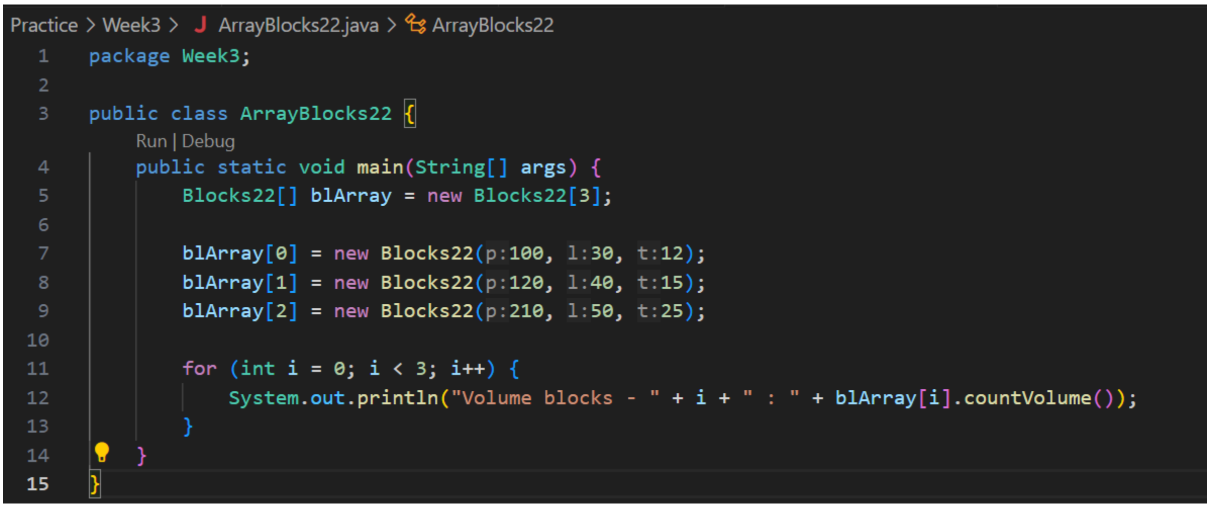
   Answer: No, we cannot duplicate it. When assign the same object to two different indices in an array, the instantiation process will not be repeated.

Lab Activity 3: Mathematical operation in array of object’s attribute

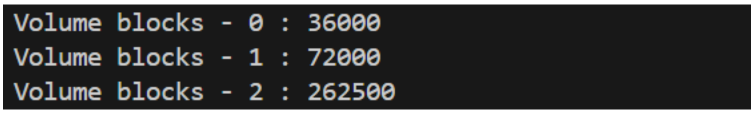
Result



```java
package Week3;

public class Blocks22 {
    public int width, length, height;

    public Blocks22(int p, int l, int t) {
        length = p;
        width = l;
        height = t;
    }

    public int countVolume() {
        return length*width*height;
    }
}
```



```java
package Week3;

public class ArrayBlocks22 {
    public static void main(String[] args) {
        Blocks22[] blArray = new Blocks22[3];

        blArray[0] = new Blocks22(p:100, l:30, t:12);
        blArray[1] = new Blocks22(p:120, l:40, t:15);
        blArray[2] = new Blocks22(p:210, l:50, t:25);

        for (int i = 0; i < 3; i++) {
            System.out.println("Volume blocks - " + i + " : " + blArray[i].countVolume());
        }
    }
}
```



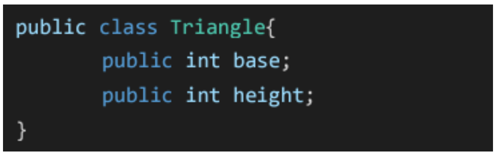
```
Volume blocks - 0 : 36000
Volume blocks - 1 : 72000
Volume blocks - 2 : 262500
```

Question

1. Can we have more than one constructor in one class? Please explain.

   Answer: Yes, we can have more than one constructor in one class, for example constructor with parameters and constructor without parameter

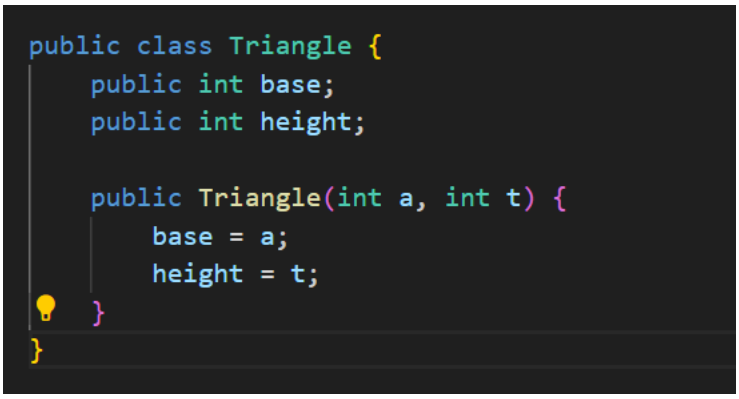
2. Create a **Triangle** class as follows.



```java
public class Triangle{
        public int base;
        public int height;
}
```

Add another constructor in this class that has parameter int a, int t. These represents its base and height.

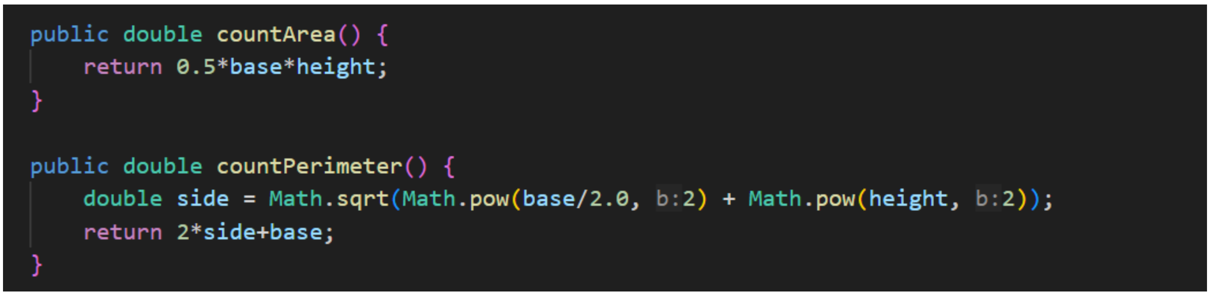
Answer:



```java
public class Triangle {
    public int base;
    public int height;

    public Triangle(int a, int t) {
        base = a;
        height = t;
    }
}
```

3. Add method **countArea()** and **countPerimeter()** in class Triangle

   Answer:

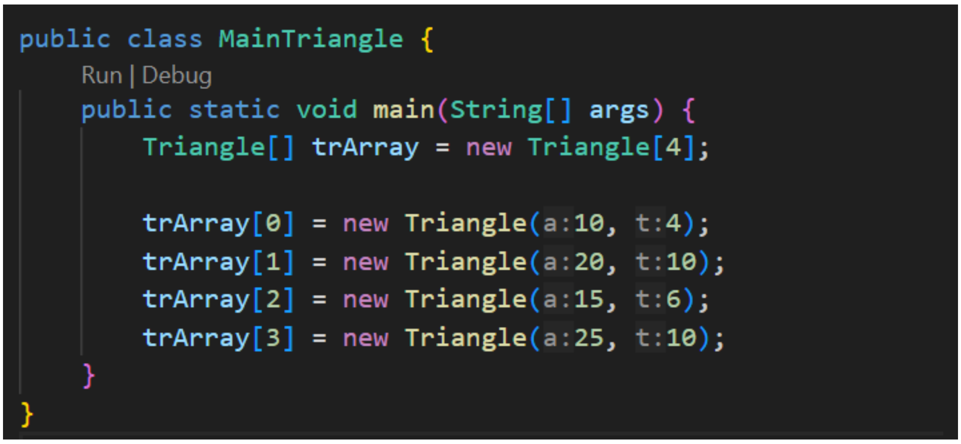


```java
public double countArea() {
    return 0.5*base*height;
}

public double countPerimeter() {
    double side = Math.sqrt(Math.pow(base/2.0, b:2) + Math.pow(height, b:2));
    return 2*side+base;
}
```

4. In main function, instantiate array of **Triangle** objects. Assign the attributes values as follows:

0th trArray	base: 10, height: 4
1st trArray	base: 20, height: 10
2nd trArray	base: 15, height: 6
3rd trArray	base: 25, height: 10

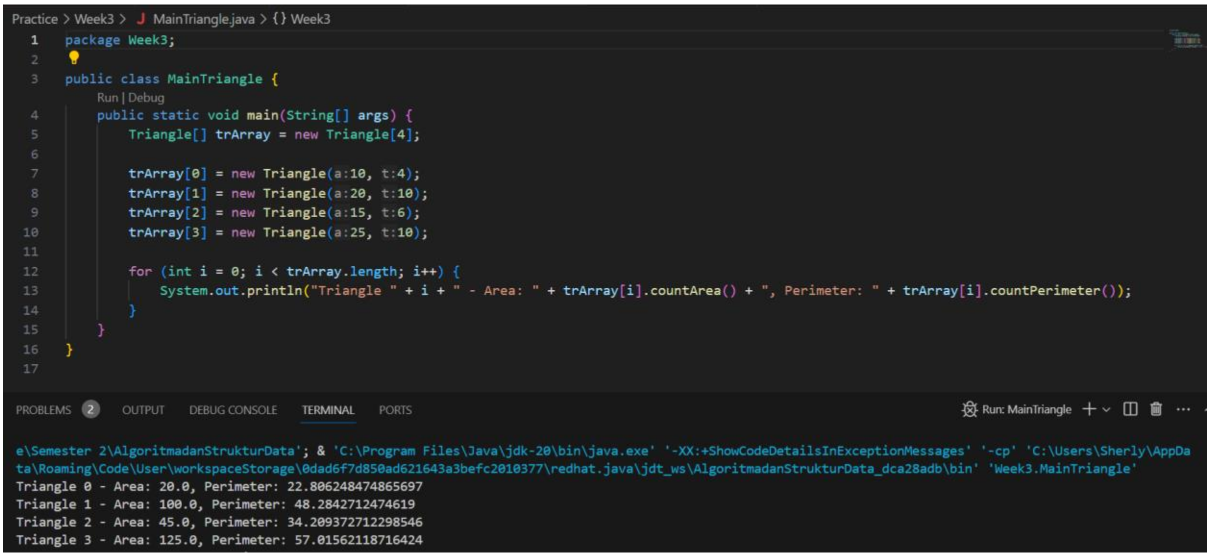
Answer:



```java
public class MainTriangle {
    public static void main(String[] args) {
        Triangle[] trArray = new Triangle[4];

        trArray[0] = new Triangle(a:10, t:4);
        trArray[1] = new Triangle(a:20, t:10);
        trArray[2] = new Triangle(a:15, t:6);
        trArray[3] = new Triangle(a:25, t:10);
    }
}
```

5. Display the result of area and perimeter for each triangle by calling the method **countArea()** and **countPerimeter()**

   Answer:



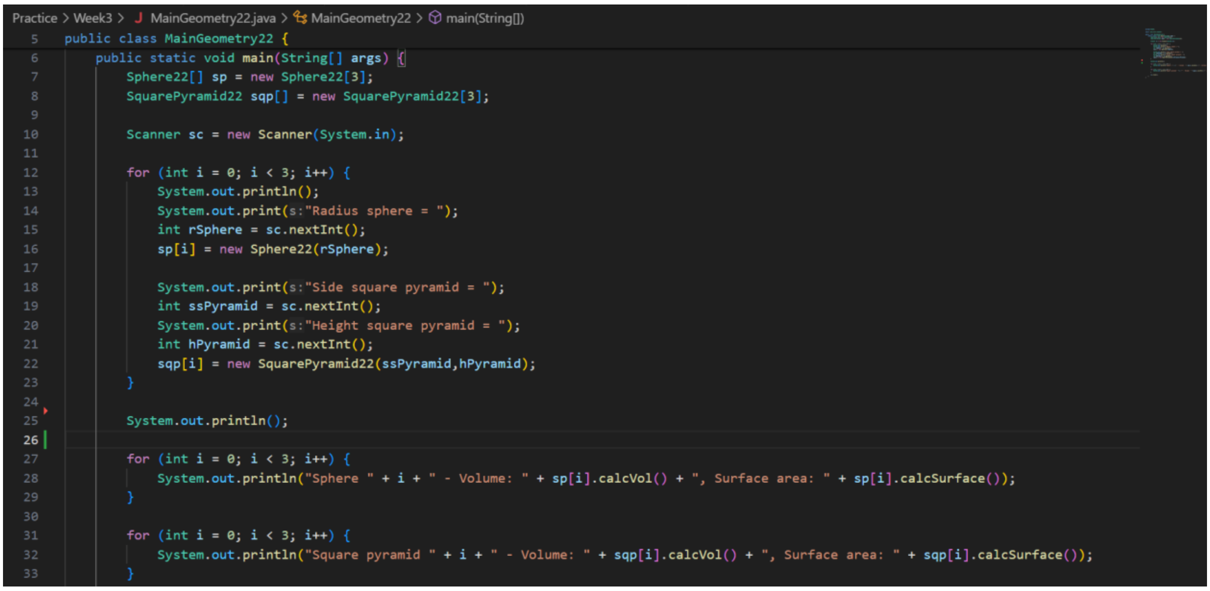
```java
package Week3;

public class MainTriangle {
    public static void main(String[] args) {
        Triangle[] trArray = new Triangle[4];

        trArray[0] = new Triangle(a:10, t:4);
        trArray[1] = new Triangle(a:20, t:10);
        trArray[2] = new Triangle(a:15, t:6);
        trArray[3] = new Triangle(a:25, t:10);

        for (int i = 0; i < trArray.length; i++) {
            System.out.println("Triangle " + i + " - Area: " + trArray[i].countArea() + ", Perimeter: " + trArray[i].countPerimeter());
        }
    }
}
```

```
e\Semester 2\AlgoritmadanStrukturData'; & 'C:\Program Files\Java\jdk-20\bin\java.exe' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'C:\Users\Sherly\AppDa
ta\Roaming\Code\User\workspaceStorage\0dad6f7d850ad621643a3befc2010377\redhat.java\jdt_ws\AlgoritmadanStrukturData_dca28adb\bin' 'Week3.MainTriangle'
Triangle 0 - Area: 20.0, Perimeter: 22.806248474865697
Triangle 1 - Area: 100.0, Perimeter: 48.2842712474619
Triangle 2 - Area: 45.0, Perimeter: 34.209372712298546
Triangle 3 - Area: 125.0, Perimeter: 57.01562118716424
```

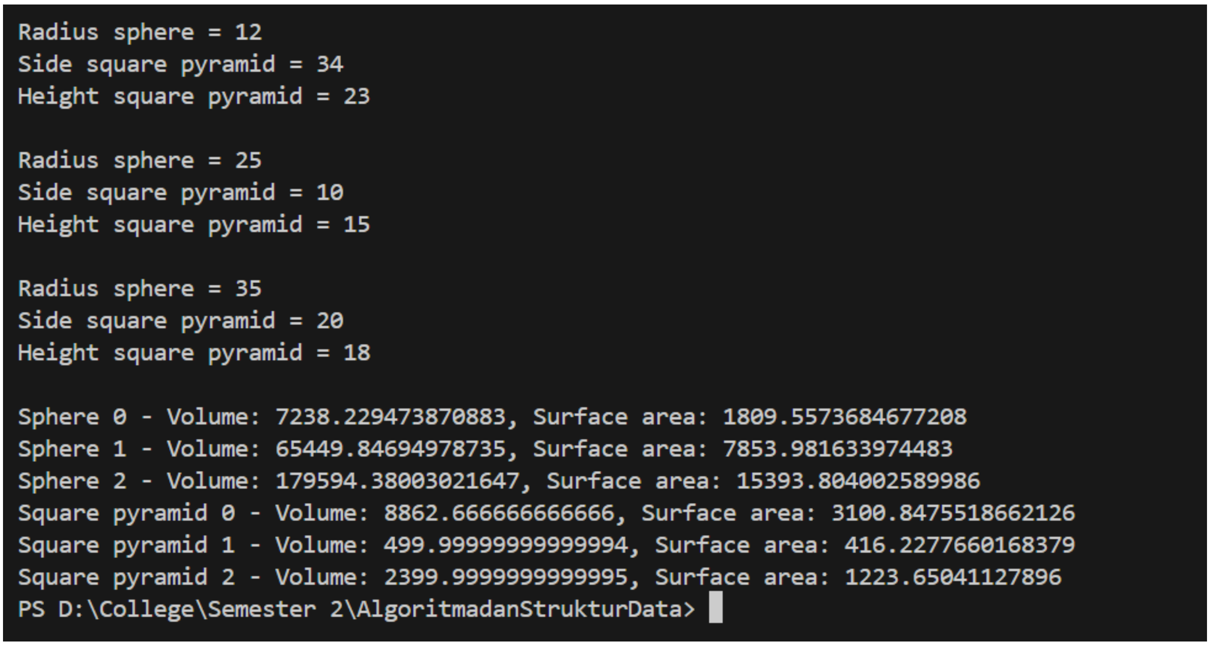
Assignment

1. Create a program that can calculate the **surface area** and **volume** of a **square pyramid**, and **sphere**. Create 3 (three) classes corresponding to each shape. Create one main class to create an array of objects that inputs the attributes using constructors for all these spatial shapes. With the following provisions:

   a. Create a loop to input each attribute, then display the surface area and volume of each type of spatial shape.

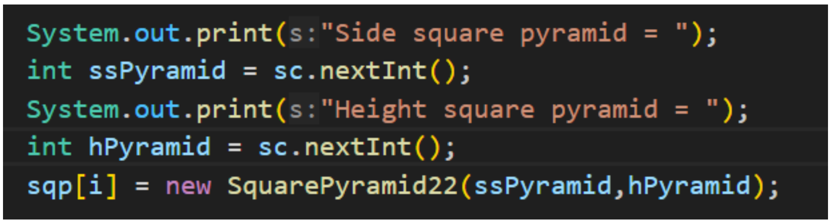

Practice > Week3 > MainGeometry22.java > MainGeometry22 > main(String[])

```java
public class MainGeometry22 {
    public static void main(String[] args) {
        Sphere22[] sp = new Sphere22[3];
        SquarePyramid22 sqp[] = new SquarePyramid22[3];

        Scanner sc = new Scanner(System.in);

        for (int i = 0; i < 3; i++) {
            System.out.println();
            System.out.print(s:"Radius sphere = ");
            int rSphere = sc.nextInt();
            sp[i] = new Sphere22(rSphere);

            System.out.print(s:"Side square pyramid = ");
            int ssPyramid = sc.nextInt();
            System.out.print(s:"Height square pyramid = ");
            int hPyramid = sc.nextInt();
            sqp[i] = new SquarePyramid22(ssPyramid,hPyramid);
        }

        System.out.println();

        for (int i = 0; i < 3; i++) {
            System.out.println("Sphere " + i + " - Volume: " + sp[i].calcVol() + ", Surface area: " + sp[i].calcSurface());
        }

        for (int i = 0; i < 3; i++) {
            System.out.println("Square pyramid " + i + " - Volume: " + sqp[i].calcVol() + ", Surface area: " + sqp[i].calcSurface());
        }
```



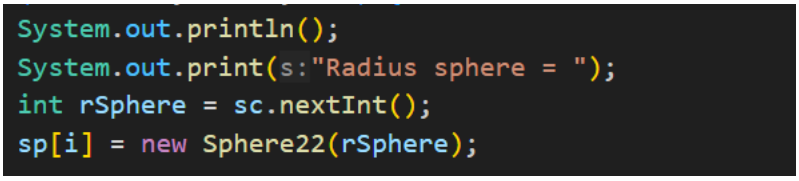
```
Radius sphere = 12
Side square pyramid = 34
Height square pyramid = 23

Radius sphere = 25
Side square pyramid = 10
Height square pyramid = 15

Radius sphere = 35
Side square pyramid = 20
Height square pyramid = 18

Sphere 0 - Volume: 7238.229473870883, Surface area: 1809.5573684677208
Sphere 1 - Volume: 65449.84694978735, Surface area: 7853.981633974483
Sphere 2 - Volume: 179594.38003021647, Surface area: 15393.804002589986
Square pyramid 0 - Volume: 8862.666666666666, Surface area: 3100.8475518662126
Square pyramid 1 - Volume: 499.99999999999994, Surface area: 416.2277660168379
Square pyramid 2 - Volume: 2399.9999999999995, Surface area: 1223.65041127896
PS D:\College\Semester 2\AlgoritmadanStrukturData>
```

b. For the square pyramid, the input for attributes is only the length of the base side and the height of the pyramid.



```java
System.out.print(s:"Side square pyramid = ");
int ssPyramid = sc.nextInt();
System.out.print(s:"Height square pyramid = ");
int hPyramid = sc.nextInt();
sqp[i] = new SquarePyramid22(ssPyramid,hPyramid);
```

c. For the sphere, the input for attributes is only the radius.



```java
System.out.println();
System.out.print(s:"Radius sphere = ");
int rSphere = sc.nextInt();
sp[i] = new Sphere22(rSphere);
```

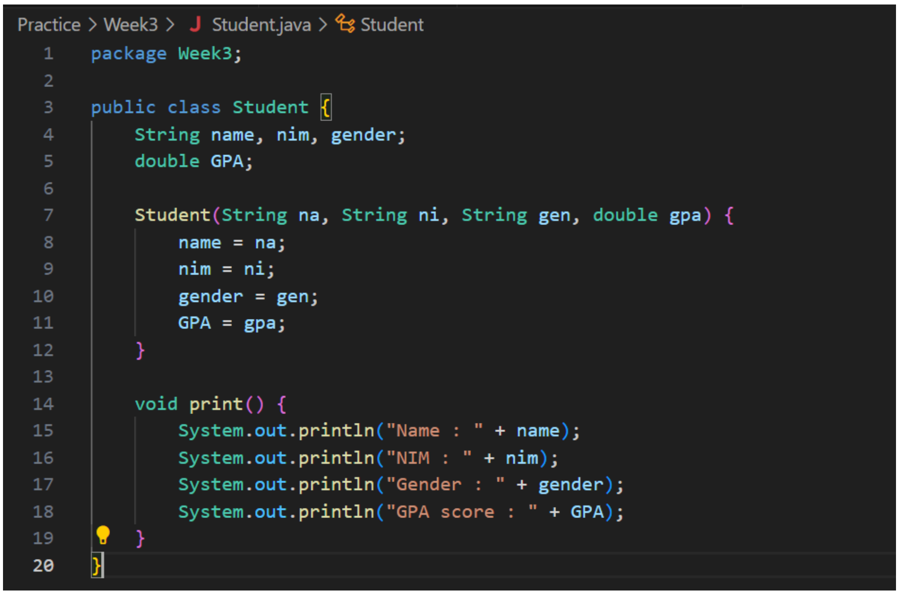
2. A university needs a program to display student’s information such as name, nim, gender, and GPA. This program should be able to receive input from all of those informations and display it to the user. Implement the program if there is 3 data sample, here is a reference of how you do it:

```
Insert 1st student data
Insert name :Rina
Insert nim :1234567
Insert gender :P
Insert GPA :3.5

Insert 2nd student data
Insert name :Rio
Insert nim :7654321
Insert gender:L
Insert GPA :4.0

Insert 3rd student data
Insert name :Reza
Insert nim :8765398
Insert gender:L
Insert GPA :3.8
```

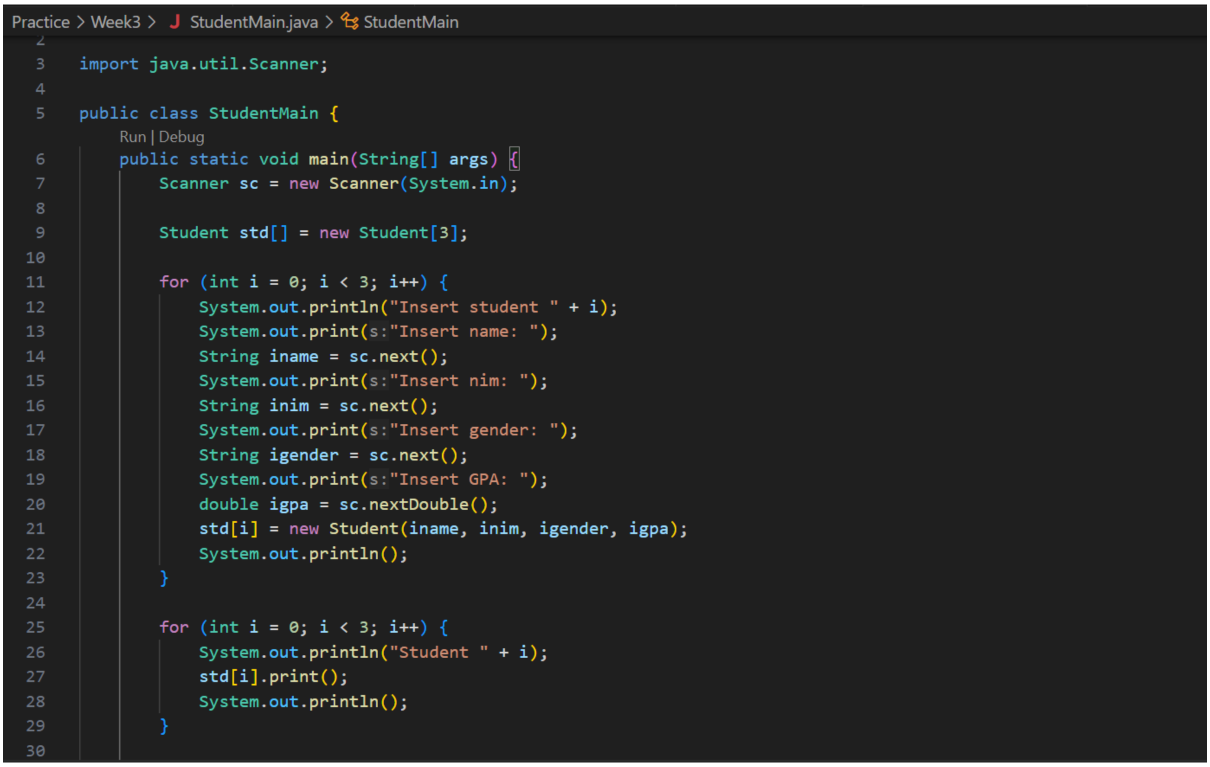
Result:

```
1st Student Data
name : Rina
nim : 1234567
gender: P
GPA score: 3.5

2nd Student Data
name : Rio
nim : 7654321
gender: L
GPA score: 4.0

3rd student Data
name : Reza
nim : 8765398
gender: L
GPA score: 3.8
```

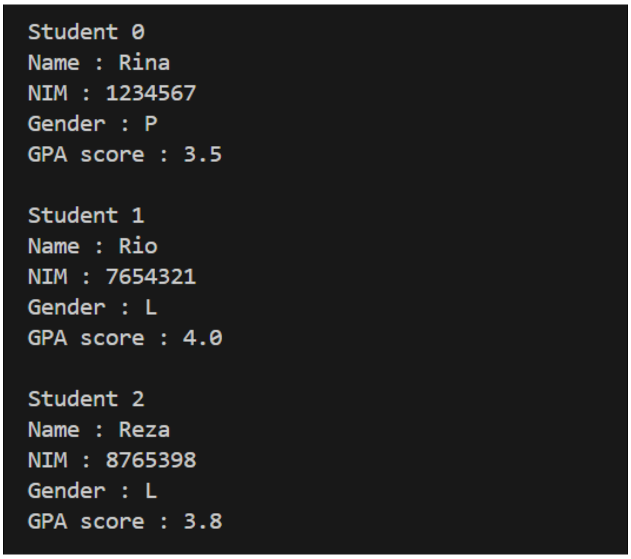
Answer:



```java
package Week3;

public class Student {
    String name, nim, gender;
    double GPA;

    Student(String na, String ni, String gen, double gpa) {
        name = na;
        nim = ni;
        gender = gen;
        GPA = gpa;
    }

    void print() {
        System.out.println("Name : " + name);
        System.out.println("NIM : " + nim);
        System.out.println("Gender : " + gender);
        System.out.println("GPA score : " + GPA);
    }
}
```

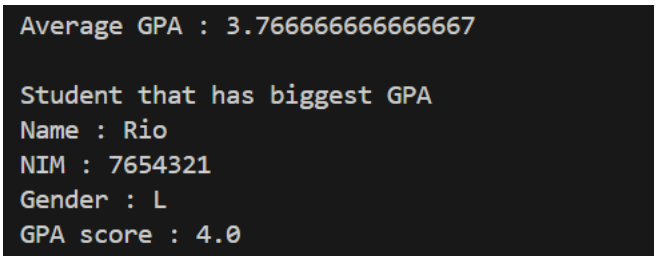


```java
import java.util.Scanner;

public class StudentMain {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);

        Student std[] = new Student[3];

        for (int i = 0; i < 3; i++) {
            System.out.println("Insert student " + i);
            System.out.print(s:"Insert name: ");
            String iname = sc.next();
            System.out.print(s:"Insert nim: ");
            String inim = sc.next();
            System.out.print(s:"Insert gender: ");
            String igender = sc.next();
            System.out.print(s:"Insert GPA: ");
            double igpa = sc.nextDouble();
            std[i] = new Student(iname, inim, igender, igpa);
            System.out.println();
        }

        for (int i = 0; i < 3; i++) {
            System.out.println("Student " + i);
            std[i].print();
            System.out.println();
        }

```

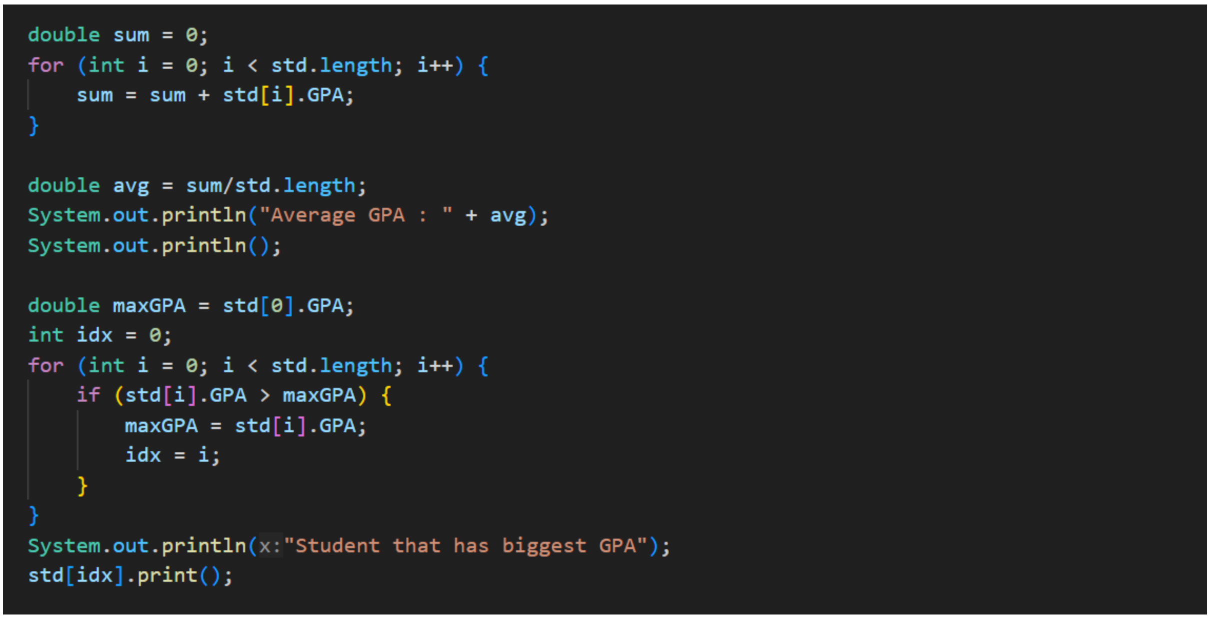


```
Student 0
Name : Rina
NIM : 1234567
Gender : P
GPA score : 3.5

Student 1
Name : Rio
NIM : 7654321
Gender : L
GPA score : 4.0

Student 2
Name : Reza
NIM : 8765398
Gender : L
GPA score : 3.8
```

3. Modify the resulting program at no.2, so that it could be used to calculate Average GPA, as well as to display student information that has biggest GPA! (use method to implement each process)



```
Average GPA : 3.766666666666667

Student that has biggest GPA
Name : Rio
NIM : 7654321
Gender : L
GPA score : 4.0
```



```
double sum = 0;
for (int i = 0; i < std.length; i++) {
    sum = sum + std[i].GPA;
}

double avg = sum/std.length;
System.out.println("Average GPA : " + avg);
System.out.println();

double maxGPA = std[0].GPA;
int idx = 0;
for (int i = 0; i < std.length; i++) {
    if (std[i].GPA > maxGPA) {
        maxGPA = std[i].GPA;
        idx = i;
    }
}
System.out.println(x:"Student that has biggest GPA");
std[idx].print();
```