Data Structure and Algorithm Practicum
Array of Objects

Name

Muhammad Baihaqi Aulia Asy'ari

NIM

2241720145

Class

1I

Department

Information Technology

Study Program

D4 Informatics Engineering

1.2 Create, insert, and display Array of Object

1.2.1 Steps

1.
```
package ArrayOfObjects;

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

2.
```
package ArrayOfObjects;

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

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

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

    rectangleArray[2] = new Rectangle();
    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);
    }
}
```

1.2.2 Result

```
"C:\Program Files\Java\jdk-18.0.2.1\bin\java.exe" -
↪  javaagent:C:\Users\ASUS\AppData\Local\JetBrains\Toolbox\apps\
↪  IDEA-C\ch-0\223.8617.56\lib\idea_rt.jar=53329:C:\Users\ASUS\
↪  AppData\Local\JetBrains\Toolbox\apps\IDEA-C\ch-0\223.8617.56\bin
↪  -Dfile.encoding=UTF-8 -Dsun.stdout.encoding=UTF-8
↪  -Dsun.stderr.encoding=UTF-8 -classpath "D:\Kuliah Smt 2\Algoritma
↪  dan Struktur Data\Praktikum Week
↪  3\codes\ArrayOfObjects\target\classes"
↪  ArrayOfObjects.ArrayOfObjects
First Rectangle, width: 30, length: 110
Second Rectangle, width: 40, length: 80
Third Rectangle, width: 20, length: 100

Process finished with exit code 0

```

1.2.3 Questions

1. Based on practicum 1.2, does the class that are going to be used as an array of object must have attributes and methods? Please explain
   Answer:
   Yes, each element of the array is an object of its own, because of that attributes and methods are needed. without them, they wouldn't have behaviour or interaction and data to interact with.

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

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

   Answer:
   Yes, by default. But no, we didn't declared the constructor. But by default the class already have constructor despite not being declared.

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

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

   Answer:
   it means an array of Objects by the name of rectangleArray is instantiate with 3 element from class **Rectangle**

4. Whats the meaning of these lines of code?

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

   `Answer:`
   it's declaring the second element of rectangleArray array as an object of class **Rectangle** and then declaring its attributes of length and width to 80 and 40.

5. Why **ArrayOfObject** class and **Rectangle** class should be differentiated?
   `Answer:`
   So that the class **ArrayOfObject** can make multiple independent objects of **Rectangle** that can be manipulated and would be individually different.

1.3 Get input in Array of Objects using Loops

1.3.1 Steps

```
package ArrayOfObjects;

import java.util.Scanner;

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

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

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

            System.out.print("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);
        }
    }
}
```

1.3.2 Result

```
"C:\Program Files\Java\jdk-18.0.2.1\bin\java.exe" -
↪  javaagent:C:\Users\ASUS\AppData\Local\JetBrains\Toolbox\apps\
↪  IDEA-C\ch-0\223.8617.56\lib\idea_rt.jar=53329:C:\Users\ASUS\
↪  AppData\Local\JetBrains\Toolbox\apps\IDEA-C\ch-0\223.8617.56\bin
↪  -Dfile.encoding=UTF-8 -Dsun.stdout.encoding=UTF-8
↪  -Dsun.stderr.encoding=UTF-8 -classpath "D:\Kuliah Smt 2\Algoritma
↪  dan Struktur Data\Praktikum Week
↪  3\codes\ArrayOfObjects\target\classes"
↪  ArrayOfObjects.ArrayOfObjects
Rectangle 0
Input length : 5
Input width : 6
Rectangle 1
Input length : 5
Input width : 6
Rectangle 2
Input length : 5
Input width : 6
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

Process finished with exit code 0

```

1.3.3 Questions

1. Does array of object can be implemented on 2D array?
   `Answer:`
   Yes we can

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

```java
public class Example {
    public int x;
    public int y;
}

public class ArrayOfExample {
    public static void main(String[] args) {
        int length = 3, width = 3;
        Example[][] exampleArray = new Example[length][width];
        for (int arrayLength = 0; arrayLength < length; arrayLength++) {
            for (int arrayWidth = 0; arrayWidth < width; arrayWidth++) {
                exampleArray[arrayLength][arrayWidth] = new Example();
                exampleArray[arrayLength][arrayWidth].x = arrayLength;
                exampleArray[arrayLength][arrayWidth].y = arrayWidth;
            }

        }

    }

}
```

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?

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

   Answer:
   The error occure because not every object in the array has their attribute declared

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

   ```
   package ArrayOfObjects;

   import java.util.Scanner;

   public class ArrayOfObjects {
       public static void main(String[] args) {
           Scanner sc = new Scanner(System.in);
           System.out.print("Enter length of the Array: ")
           int arrayLength = sc.nextInt();
           Rectangle[] rectangleArray = new Rectangle[arrayLength];
           // Assign the values for each attributes in objects
           for (int i = 0; i < arrayLength; i++) {
               rectangleArray[i] = new Rectangle();
               System.out.println("Rectangle " + i);

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

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

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:`
   it will create a seperate instance of an object. the instantiation from the loop would be different than the seperate duplicate instantiation.

1.4 Mathematical operation in array of object's attribute

1.4.1 Steps

```
package ArrayBlock;

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

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

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

```
package ArrayBlock;

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

        blArray[0] = new Blocks(100, 30, 12);
        blArray[1] = new Blocks(128, 40, 15);
        blArray[2] = new Blocks(210, 50, 25);

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

1.4.2 Result

```
"C:\Program Files\Java\jdk-18.0.2.1\bin\java.exe" -
↪  javaagent:C:\Users\ASUS\AppData\Local\JetBrains\Toolbox\apps\
↪  IDEA-C\ch-0\223.8617.56\lib\idea_rt.jar=53329:C:\Users\ASUS\
↪  AppData\Local\JetBrains\Toolbox\apps\IDEA-C\ch-0\223.8617.56\bin
↪  -Dfile.encoding=UTF-8 -Dsun.stdout.encoding=UTF-8
↪  -Dsun.stderr.encoding=UTF-8 -classpath "D:\Kuliah Smt 2\Algoritma
↪  dan Struktur Data\Praktikum Week
↪  3\codes\ArrayOfObjects\target\classes" ArrayOfBlock.ArrayOfBlock
Volume blocks - 0 : 36000
Volume blocks - 1 : 76800
Volume blocks - 2 : 262500

Process finished with exit code 0

```

1.4.3 Questions

1. Can we have more than one constructor in one class? Please explain
   Answer:
   We can, but if we add more than 1 constructor we would need to add a default constructor.

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

   ```
   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:

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

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

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

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

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

       public void countArea() {
           int area = (int) 0.5*this.base*this.height;
       }

       public void countPerimiter() {
           int perimiter = this.base*3;
       }
   }
   ```

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

   ```
   package Triangle;

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

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

       public void countArea() {
           double area = 0.5*this.base*this.height;
           System.out.print(area);
       }
   ```

```
    public void countPerimiter() {
        int perimiter = this.base*3;
        System.out.print(perimiter);
    }

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

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

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

```
package Triangle;

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

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

    public void countArea() {
        double area = 0.5*this.base*this.height;
        System.out.print(area);
    }

    public void countPerimiter() {
        int perimiter = this.base*3;
        System.out.print(perimiter);
    }

    public static void main(String[] args) {
        Triangle[] triangleArray = new Triangle[4];
```

```
            triangleArray[0] = new Triangle(10, 4);
            triangleArray[1] = new Triangle(20, 10);
            triangleArray[2] = new Triangle(15, 6);
            triangleArray[3] = new Triangle(25, 10);

            for (int i = 0; i < triangleArray.length; i++) {
                System.out.printf("Triangle %d has an area of ", i +
                ↪  1);
                triangleArray[i].countArea();
                System.out.print(" and a perimiter of ");
                triangleArray[i].countPerimiter();
                System.out.println();
            }
        }
    }
```

```
PS D:\Kuliah>  d:; cd 'd:\Kuliah'; & 'C:\Program
↪  Files\Java\jdk-18.0.2.1\bin\java.exe'
↪  '-XX:+ShowCodeDetailsInExceptionMessages' '-cp'
↪  'C:\Users\ASUS\AppData\Roaming\Code\User\workspaceStorage\
↪  ce3fcb236261368a6cbd019dc8ddda8b\redhat.java\jdt_ws\
↪  Kuliah_28156aa7\bin' 'Triangle.Triangle'
Triangle 1 has an area of 20.0 and a perimiter of 30
Triangle 2 has an area of 100.0 and a perimiter of 60
Triangle 3 has an area of 45.0 and a perimiter of 45
Triangle 4 has an area of 125.0 and a perimiter of 75
```

1.5 Practice

1. Create a program that can count surface area and volume of some 3D Geometry object (Cube, blocks, cylinder, etc). Then, create one more class to instantiate the array of objects with its constructor to assign values of its attributes.

   Note: Create loop to get user input and assign it to the attributes of the objects, then display the surface area and volume of each 3rd geometry object in console

   ```
   Answer:

   package Practice;

   public class Geometry {
       int length;
       int width;
   ```

```
    int height;
    public Geometry(int l, int w, int h) {
        this.length = l;
        this.width = w;
        this.height = h;
    }
    public void calculateSurfaceArea() {
        int surfaceArea =
         ↪  2*((length*width)+(width*height)+(height*length));
        System.out.print(surfaceArea);
    }
    public void calculateVolume() {
        int volume = length*width*height;
        System.out.print(volume);
    }
}

package Practice;

import java.util.Scanner;

public class GeometryMain {
    public static void main(String[] args) {
        Scanner input = new Scanner(System.in);
        System.out.print("Enter the length of the Array: ");
        int limit = input.nextInt();

        Geometry[] geometries = new Geometry[limit];
        for (int i = 0; i < geometries.length; i++) {
            System.out.print("Enter Block " + (i+1) + " length:
             ↪  ");
            int l = input.nextInt();
            System.out.print("Enter Block " + (i+1) + " width:
             ↪  ");
            int w = input.nextInt();
            System.out.print("Enter Block " + (i+1) + " height:
             ↪  ");
            int h = input.nextInt();
            geometries[i] = new Geometry(l, w, h);
        }
        for (int i = 0; i < geometries.length; i++) {
            System.out.printf("The surface area of block %d is ",
             ↪  i+1);
            geometries[i].calculateSurfaceArea();
            System.out.print(" and volume of ");
            geometries[i].calculateVolume();
            System.out.println();
        }
        input.close();
    }
}
```

2. A company that handles land transaction needs a program to calculate land area. This program must receive user input to assign values of these:

   - How many lands?
   - Length and width of the land

   This program calculates the area of inputted land information as its output. Check this following program:

   ```
   How many lands: 3

   Land 1
   Length: 100
   Width : 40

   Land 2
   Length: 250
   Width : 100

   Land 3
   Length: 120
   Width : 100

   Land Area 1: 4000
   Land Area 2: 25000
   Land Area 3: 12000
   ```

   ```
   Answer:
   ```

   ```
   package Practice;

   public class Land {
       int length;
   ```

```
    int width;
    public Land(int l, int w) {
        this.length = l;
        this.width = w;
    }
    public int area() {
        return length * width;
    }
}

package Practice;

import java.util.Scanner;

public class LandMain {
    public static void main(String[] args) {
        Scanner input = new Scanner(System.in);
        System.out.print("How many lands: ");
        int unit = input.nextInt();
        System.out.println();

        Land[] lands = new Land[unit];
        for (int i = 0; i < lands.length; i++) {
            System.out.printf("Land %d\n", i+1);
            System.out.print("length: ");
            int l = input.nextInt();
            System.out.print("width : ");
            int w = input.nextInt();
            lands[i] = new Land(l, w);
            System.out.println();
        }
        for (int i = 0; i < lands.length; i++) {
            System.out.printf("Land Area %d: %,d\n", i+1,
             ↪  lands[i].area());
        }
        input.close();
    }
}
```

3. Modify the program above so that it can display the widest area. (Additional note: create a different function to get the widest area)

```
Land 1
Length: 100
Width : 40

Land 2
Length: 250
Width : 100

Land 3
Length: 120
Width : 100

Land Area 1: 4000
Land Area 2: 25000
Land Area 3: 12000

The widest land is Land 2
```

```java
package Practice;

import java.util.Scanner;

public class LandMain {
    public static void main(String[] args) {
        Scanner input = new Scanner(System.in);
        System.out.print("How many lands: ");
        int unit = input.nextInt();
        System.out.println();

        Land[] lands = new Land[unit];
        for (int i = 0; i < lands.length; i++) {
            System.out.printf("Land %d\n", i+1);
            System.out.print("length: ");
            int l = input.nextInt();
            System.out.print("width : ");
            int w = input.nextInt();
            lands[i] = new Land(l, w);
            System.out.println();
        }
        for (int i = 0; i < lands.length; i++) {
            System.out.printf("Land Area %d: %,d\n", i+1,
             ↪  lands[i].area());
        }
        int wide = 0;
        int widest = 0;
        for (int i = 0; i < lands.length; i++) {
            if (lands[i].area() < wide) continue;
            wide = lands[i].area();
            widest = i + 1;
        }
        System.out.println();
        System.out.printf("The widest land is Land %d", widest);
        input.close();
    }
}
```

4. 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 IPK :3.5

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

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

   `Result`

   1st student data

name :Rina
nim :1234567
gender :P
IPK :3.5

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

3rd student data
name :Reza
nim :8765398
gender:L
IPK :3.8

```java
package Practice;

public class StudentData {
    public String name;
    public String NIM;
    public String gender;
    public double IPK;
    public void printData() {

    }
}

package Practice;

import java.util.Scanner;

public class StudentDataMain {
    public static void main(String[] args) {
        Scanner input = new Scanner(System.in);
        int limit = 3;

        StudentData[] studentDatas = new StudentData[limit];
        for (int i = 0; i < limit; i++) {
            studentDatas[i] = new StudentData();
            System.out.printf("Insert %d student data\n", i+1);
            System.out.print("Insert name: ");
            studentDatas[i].name = input.next();
            input.nextLine();
            System.out.print("Insert NIM: ");
            studentDatas[i].NIM = input.next();
            input.nextLine();
            System.out.print("Insert gender: ");
            studentDatas[i].gender = input.next();
            input.nextLine();
            System.out.print("Insert IPK: ");
            studentDatas[i].IPK = input.nextDouble();
            System.out.println();
        }
        for (int i = 0; i < limit; i++) {
            System.out.printf("%d student data\n", i+1);
            System.out.printf("name: %s\n",
            ↪   studentDatas[i].name);
            System.out.printf("nim: %s\n", studentDatas[i].NIM);
            System.out.printf("gender: %s\n",
            ↪   studentDatas[i].gender);
            System.out.printf("IPK: %.1f\n",
            ↪   studentDatas[i].IPK);
            System.out.println();
        }
        input.close();
    }
}
```

5. Modify the program above so that it can receive the average of IPK score from all students.
   (Note: create a new function to calculate the average of IPK Score in class **Students**)

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

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

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

`Result`

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

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

3rd student data
name :Reza
nim :8765398
gender:L
IPK :3.8

Avergae IPK of all students : 3.7666667
```

```java
package Practice;

import java.util.Scanner;

public class StudentDataMain {
    public static void main(String[] args) {
        Scanner input = new Scanner(System.in);
        int limit = 3;

        StudentData[] studentDatas = new StudentData[limit];
        for (int i = 0; i < limit; i++) {
            studentDatas[i] = new StudentData();
            System.out.printf("Insert %d student data\n", i+1);
            System.out.print("Insert name: ");
            studentDatas[i].name = input.next();
            input.nextLine();
            System.out.print("Insert NIM: ");
            studentDatas[i].NIM = input.next();
            input.nextLine();
            System.out.print("Insert gender: ");
            studentDatas[i].gender = input.next();
            input.nextLine();
            System.out.print("Insert IPK: ");
            studentDatas[i].IPK = input.nextDouble();
            System.out.println();
        }
        for (int i = 0; i < limit; i++) {
            System.out.printf("%d student data\n", i+1);
            System.out.printf("name: %s\n",
             ↪  studentDatas[i].name);
            System.out.printf("nim: %s\n", studentDatas[i].NIM);
            System.out.printf("gender: %s\n",
             ↪  studentDatas[i].gender);
            System.out.printf("IPK: %.1f\n",
             ↪  studentDatas[i].IPK);
            System.out.println();
        }
        double sumIPK = 0;
        for (int i = 0; i < limit; i++) {
            sumIPK += studentDatas[i].IPK;
        }
        System.out.printf("Average IPK of all students: %.6f",
         ↪  sumIPK/limit);
        input.close();
    }
}
```