



Name : Azaria Cindy Sahasika
 Number Id : 2341760169 / 06
 Class : 1G – Business Information System
 Lesson : Algorithm and Data Structure
 Material : Jobsheet 3
 Github Link : <https://github.com/azariacindy/algorithm-ds>

JOBSHEET III

ARRAY OF OBJECTS

3.1 Tujuan Praktikum

Setelah melakukan materi praktikum ini, mahasiswa mampu:

1. Memahami dan menjelaskan fungsi array yang berisikan variabel object.
2. Mahasiswa mampu menerapkan instansiasi array of objects dalam Java
3. Mahasiswa mampu melakukan operasi terhadap elemen dalam array of objects

3.2 Membuat Array dari Object, Mengisi dan Menampilkan

Didalam praktikum ini, kita akan mempraktekkan bagaimana membuat array of objects, kemudian mengisi dan menampilkan array tersebut.

3.2.1 Langkah-langkah Percobaan

1. Buat folder baru dengan nama Praktikum03.
2. Buat class **PersegiPanjang**:

```
public class PersegiPanjang {
    public int panjang;
    public int lebar;
}
```

3. Buat class **PersegiPanjangDemo** kemudian tambahkan fungsi main sebagai berikut

```
public static void main(String[] args) {
    PersegiPanjang[] arrayOfPersegiPanjang = new PersegiPanjang[3];
}
```

4. Kemudian isikan masing-masing atributnya:

```
public static void main(String[] args) {
    PersegiPanjang[] arrayOfPersegiPanjang = new PersegiPanjang[3];

    arrayOfPersegiPanjang[0] = new PersegiPanjang();
    arrayOfPersegiPanjang[0].panjang = 110;
    arrayOfPersegiPanjang[0].lebar = 30;

    arrayOfPersegiPanjang[1] = new PersegiPanjang();
    arrayOfPersegiPanjang[1].panjang = 80;
    arrayOfPersegiPanjang[1].lebar = 40;

    arrayOfPersegiPanjang[2] = new PersegiPanjang();
    arrayOfPersegiPanjang[2].panjang = 100;
    arrayOfPersegiPanjang[2].lebar = 20;
}
```

5. Cetak ke layar semua atribut dari objek **ppArray**:

```
System.out.println("Persegi panjang ke-1, panjang: " + arrayOfPersegiPanjang[0].panjang +
    ", lebar: " + arrayOfPersegiPanjang[0].lebar);
System.out.println("Persegi panjang ke-2, panjang: " + arrayOfPersegiPanjang[1].panjang +
    ", lebar: " + arrayOfPersegiPanjang[1].lebar);
System.out.println("Persegi panjang ke-3, panjang: " + arrayOfPersegiPanjang[2].panjang +
    ", lebar: " + arrayOfPersegiPanjang[2].lebar);
```

6. Run program dan amati hasilnya.

```
PS D:\cooleyah\smstr2\algorithm and data structure\assignment\jobsheet3\code> java persegiPanjangDemo
Persegi panjang ke-1, panjang: 110, lebar30
Persegi panjang ke-2, panjang: 80, lebar40
Persegi panjang ke-2, panjang: 100, lebar20
```

```
jobsheet3 > code > J persegiPanjangDemo.java > ...
Click here to ask Blackbox to help you code faster | Comment Code |
1 public class persegiPanjangDemo {
2     Run | Debug
3     public static void main(String[] args) {
4         persegiPanjang[] arrayOfPersegiPanjang = new persegiPanjang[3];
5
6         // Mengisi masing-masing objek persegiPanjang dalam array
7         arrayOfPersegiPanjang[0] = new persegiPanjang();
8         arrayOfPersegiPanjang[0].panjang = 110;
9         arrayOfPersegiPanjang[0].lebar = 30;
10
11         arrayOfPersegiPanjang[1] = new persegiPanjang();
12         arrayOfPersegiPanjang[1].panjang = 80;
13         arrayOfPersegiPanjang[1].lebar = 40;
14
15         arrayOfPersegiPanjang[2] = new persegiPanjang();
16         arrayOfPersegiPanjang[2].panjang = 100;
17         arrayOfPersegiPanjang[2].lebar = 20;
18
19         System.out.println("Persegi panjang ke-1, panjang: " + arrayOfPersegiPanjang[0].panjang + ", lebar" + arrayOfPersegiPanjang[0].lebar);
20         System.out.println("Persegi panjang ke-2, panjang: " + arrayOfPersegiPanjang[1].panjang + ", lebar" + arrayOfPersegiPanjang[1].lebar);
21         System.out.println("Persegi panjang ke-2, panjang: " + arrayOfPersegiPanjang[2].panjang + ", lebar" + arrayOfPersegiPanjang[2].lebar);
22     }
23 }
```

3.2.2 Verifikasi Hasil Percobaan

Cocokkan hasil compile kode program anda dengan gambar berikut ini.

```
Persegi panjang ke-1, panjang: 110, lebar: 30
Persegi panjang ke-2, panjang: 80, lebar: 40
Persegi panjang ke-3, panjang: 100, lebar: 20
```



3.2.3 Pertanyaan

1. Berdasarkan uji coba 3.2, apakah class yang akan dibuat array of object harus selalu memiliki atribut dan sekaligus method? Jelaskan!

➔ No, a class to be created as an array of objects does not have to have attributes and methods simultaneously. Classes can be attribute-only or method-only, or both, depending on the needs of the program.

2. Apa yang dilakukan oleh kode program berikut?

```
PersegiPanjang[] arrayOfPersegiPanjang = new PersegiPanjang[3];
```

➔ Declare and instantiate the array object arrayOfLongSquare.

3. Apakah class PersegiPanjang memiliki konstruktor? Jika tidak, kenapa bisa dilakukan pemanggilan konstruktor pada baris program berikut?

```
arrayOfPersegiPanjang[0] = new PersegiPanjang();
```

➔ if the Rectangle class doesn't provide its own constructor, then at least there will be a default constructor that can be called. That default constructor will create an object with default values for the corresponding data type.

4. Apa yang dilakukan oleh kode program berikut?

```
arrayOfPersegiPanjang[0].panjang = 110;
arrayOfPersegiPanjang[0].lebar = 30;
```

➔ Initializes the panjang dan lebar attribute of the object at index 0 of the array arrayOfPersegiPanjang array.

5. Mengapa class PersegiPanjang dan PersegiPanjangDemo dipisahkan pada uji coba 3.2?

➔ The separation of classes into different files is a practice of Object Oriented Programming (OOP).

3.3 Menerima Input Isian Array Menggunakan Looping

Pada praktikum ini kita akan mengubah hasil program dari praktikum 3.2 sehingga program dapat menerima input dan menggunakan looping untuk mengisi atribut dari semua persegi panjang

3.3.1 Langkah-langkah Percobaan

1. Import scanner pada class **PersegiPanjangDemo**.

```
import java.util.Scanner;
```

Note: Letakkan kode import dibawah kode package (jika ada).

2. Pada praktikum 3.2 poin nomor 4, modifikasi kode program sebagai berikut. Buat objek **Scanner** untuk menerima input, kemudian lakukan looping untuk menerima informasi panjang dan lebar:

```
Scanner sc = new Scanner(System.in);
PersegiPanjang[] arrayOfPersegiPanjang = new PersegiPanjang[3];

for (int i = 0; i < 3; i++) {
    arrayOfPersegiPanjang[i] = new PersegiPanjang();

    System.out.println("Persegi panjang ke-" + (i + 1));
    System.out.print("Masukkan panjang: ");
    arrayOfPersegiPanjang[i].panjang = sc.nextInt();
    System.out.print("Masukkan lebar: ");
    arrayOfPersegiPanjang[i].lebar = sc.nextInt();
}
```

3. Pada praktikum 3.2 poin nomor 5, modifikasi kode program sebagai berikut. Lakukan looping untuk mengakses elemen array dan menampilkan informasinya ke layar:

```
for (int i = 0; i < 3; i++) {
    System.out.println("Persegi panjang ke-" + (i + 1) + ", panjang: "
        + arrayOfPersegiPanjang[i].panjang + ", lebar: " + arrayOfPersegiPanjang[i].lebar);
}
```

4. Run program dan amati hasilnya.



```

jobsheet3 > code > J persegiPanjangDemo.java > ...
  Click here to ask Blackbox to help you code faster |
1  import java.util.Scanner;
2
3  Comment Code
4  public class persegiPanjangDemo {
5
6      Run | Debug
7      public static void main(String[] args) {
8          Scanner sc = new Scanner(System.in);
9          persegiPanjang[] arrayOfPersegiPanjang = new persegiPanjang[3];
10
11          for (int i = 0; i < 3; i++) {
12              arrayOfPersegiPanjang[i] = new persegiPanjang();
13              System.out.println("Persegi panjang ke-" + (i + 1));
14              System.out.println(x:"Masukkan panjang: ");
15              arrayOfPersegiPanjang[i].panjang = sc.nextInt();
16              System.out.println(x:"Masukkan lebar: ");
17              arrayOfPersegiPanjang[i].lebar = sc.nextInt();
18          }
19
20          for (int i = 0; i < 3; i++) {
21              System.out.println("Pesegi panjang ke-" + (i + 1) + ", panjang: "
22                  + arrayOfPersegiPanjang[i].panjang + ", lebar: " + arrayOfPersegiPanjang[i].lebar);
23          }
24      }
25  }

```

```

PS D:\cooleyah\smstr2\algorithm and data structure\assignment\jobsheet3\code> java persegiPanjangDemo
Persegi panjang ke-1
Masukkan panjang:
2
Masukkan lebar:
3
Persegi panjang ke-2
Masukkan panjang:
4
Masukkan lebar:
5
Persegi panjang ke-3
Masukkan panjang:
2
Masukkan lebar:
4
Pesegi panjang ke-1, panjang: 2, lebar: 3
Pesegi panjang ke-2, panjang: 4, lebar: 5
Pesegi panjang ke-3, panjang: 2, lebar: 4

```

3.3.2 Verifikasi Hasil Percobaan

Contoh verifikasi hasil percobaan ini.

```

Persegi panjang ke-1
Masukkan panjang: 2
Masukkan lebar: 3
Persegi panjang ke-2
Masukkan panjang: 4
Masukkan lebar: 5
Persegi panjang ke-3
Masukkan panjang: 2
Masukkan lebar: 4
Persegi panjang ke-1, panjang: 2, lebar: 3
Persegi panjang ke-2, panjang: 4, lebar: 5
Persegi panjang ke-3, panjang: 2, lebar: 4

```



3.3.3 Pertanyaan

1. Tambahkan method cetakInfo() pada class PersegiPanjang kemudian modifikasi kode program pada langkah no 3.

```
for (int i = 0; i < 3; i++) {
    System.out.print("Persegi panjang ke-" + (i + 1) + ": ");
    arrayOfPersegiPanjang[i].cetakInfo();
}
```

```
public class persegiPanjang {
    public int panjang;
    public int lebar;

    public void cetakInfo() {
        System.out.println("Panjang: " + panjang + ", Lebar: " + lebar);
    }
}
```

2. Misalkan Anda punya **array baru** bertipe array of PersegiPanjang dengan nama **myArrayOfPersegiPanjang**. Mengapa kode berikut menyebabkan error?

```
PersegiPanjang[] myArrayOfPersegiPanjang = new PersegiPanjang[100];
myArrayOfPersegiPanjang[5].panjang = 20;
```

➔ Because the array element (object 'persegiPanjang') has not been instantiated yet. The object at index 5 has not been created, so accessing the 'panjang' attribute of the object will cause a 'NullPointerException'. The solution is to instantiate or create the object using 'new' before accessing the array element.

```
Exception in thread "main" java.lang.NullPointerException: Cannot assign field "panjang" because "<local3>[5]" is null
at persegiPanjangDemo.main(persegiPanjangDemo.java:24)
```

```
23     persegiPanjang[] myArrayOfPersegiPanjang = new persegiPanjang[100];
24     for (int i = 0; i < myArrayOfPersegiPanjang.length; i++) {
25         myArrayOfPersegiPanjang[i] = new persegiPanjang();
26     }
27
28     myArrayOfPersegiPanjang[5].panjang = 20; // No more NullPointerException
29 }
30 }
```



3.4 Constructor Berparameter

Pada praktikum ini kita akan melakukan pengoperasian matematika beberapa atribut pada masing-masing anggota array.

3.4.1 Langkah-langkah Percobaan

1. Pada class PersegiPanjang, tambahkan constructor berparameter

```
public PersegiPanjang(int p, int l) {
    panjang = p;
    lebar = l;
}
```

2. Run program kemudian amati hasilnya.

```
persegiPanjangDemo.java:10: error: constructor persegiPanjang in class persegiPanjang cannot be applied to given types;
    arrayOfPersegiPanjang[i] = new persegiPanjang();
                                ^
    required: int,int
    found:    no arguments
    reason: actual and formal argument lists differ in length
persegiPanjangDemo.java:25: error: constructor persegiPanjang in class persegiPanjang cannot be applied to given types;
    myArrayOfPersegiPanjang[i] = new persegiPanjang();
                                ^
    required: int,int
    found:    no arguments
    reason: actual and formal argument lists differ in length
2 errors
```

3. Modifikasi class PersegiPanjangDemo sehingga instansiasi dilakukan menggunakan constructor berparameter

```
Scanner sc = new Scanner(System.in);
PersegiPanjang[] arrayOfPersegiPanjang = new PersegiPanjang[3];
int panjang, lebar;

for (int i = 0; i < 3; i++) {
    System.out.println("Persegi panjang ke-" + (i + 1));
    System.out.print("Masukkan panjang: ");
    panjang = sc.nextInt();
    System.out.print("Masukkan lebar: ");
    lebar = sc.nextInt();

    arrayOfPersegiPanjang[i] = new PersegiPanjang(panjang, lebar);
}
```

4. Run program dan amati hasilnya.

```
public static void main(String[] args) {
    Scanner sc = new Scanner(System.in);
    persegiPanjang[] arrayOfPersegiPanjang = new persegiPanjang[3];
    int panjang, lebar;

    for (int i = 0; i < 3; i++) {
        System.out.println("Persegi panjang ke-" + (i + 1));
        System.out.println(x:"Masukkan panjang: ");
        panjang = sc.nextInt();
        System.out.println(x:"Masukkan lebar: ");
        lebar = sc.nextInt();

        arrayOfPersegiPanjang[i] = new persegiPanjang(panjang, lebar);
    }

    for (int i = 0; i < 3; i++) {
        System.out.print("Persegi panjang ke-" + (i + 1) + ": ");
        arrayOfPersegiPanjang[i].cetakInfo();
    }
}
```

```
Persegi panjang ke-1
Masukkan panjang:
2
Masukkan lebar:
3
Persegi panjang ke-2
Masukkan panjang:
4
Masukkan lebar:
5
Persegi panjang ke-3
Masukkan panjang:
2
Masukkan lebar:
4
Persegi panjang ke-1: Panjang: 2, Lebar: 3
Persegi panjang ke-2: Panjang: 4, Lebar: 5
Persegi panjang ke-3: Panjang: 2, Lebar: 4
```

3.4.2 Verifikasi Hasil Percobaan

Cocokkan hasil compile kode program anda dengan gambar berikut ini.

```
Persegi panjang ke-1
Masukkan panjang: 2
Masukkan lebar: 3
Persegi panjang ke-2
Masukkan panjang: 4
Masukkan lebar: 5
Persegi panjang ke-3
Masukkan panjang: 2
Masukkan lebar: 4
Persegi panjang ke-1, panjang: 2, lebar: 3
Persegi panjang ke-2, panjang: 4, lebar: 5
Persegi panjang ke-3, panjang: 2, lebar: 4
```

3.4.3 Pertanyaan

1. Mengapa terjadi compile error pada langkah no 2?
 - Error, because it adds a parameterized constructor while persegiPanjangDemo doesn't use parameters.

2. Apakah suatu class dapat memiliki lebih dari 1 constructor? Jika iya, berikan contohnya

➤ Yes, a class can have more than one constructor. here's an example:

```

jobsheet3 > code > J persegiPanjangDemo.java > ...
  Click here to ask Blackbox to help you code faster |
1  import java.util.Scanner;
2
3  Comment Code
4  public class persegiPanjangDemo {
5
6      Run | Debug
7      public static void main(String[] args) {
8          Scanner sc = new Scanner(System.in);
9
10         // objek dengan konstruktor satu parameter
11         persegiPanjang[] arrayOfPersegiPanjang = new persegiPanjang[3];
12         int panjang, lebar;
13         for (int i = 0; i < 3; i++) {
14             System.out.println("Persegi panjang ke-" + (i + 1));
15             System.out.println(x:"Masukkan panjang: ");
16             panjang = sc.nextInt();
17             System.out.println(x:"Masukkan lebar: ");
18             lebar = sc.nextInt();
19
20             arrayOfPersegiPanjang[i] = new persegiPanjang(panjang, lebar);
21         }
22
23         for (int i = 0; i < 3; i++) {
24             System.out.print("Persegi panjang ke-" + (i + 1) + ": ");
25             arrayOfPersegiPanjang[i].cetakInfo();
26         }
27
28         // objek dengan konstruktor satu parameter
29         persegiPanjang[] myArrayOfPersegiPanjang = new persegiPanjang[100];
30         for (int i = 0; i < myArrayOfPersegiPanjang.length; i++) {
31             myArrayOfPersegiPanjang[i] = new persegiPanjang(s:5); // Panjang dan lebar menjadi 5
32         }
33
34         for (int i = 0; i < myArrayOfPersegiPanjang.length; i++) {
35             System.out.print("Persegi panjang ke-" + (i + 1) + ": ");
36             myArrayOfPersegiPanjang[i].cetakInfo();
37         }
38     }
39 }
  
```

3. Tambahkan method hitungLuas() dan hitungKeliling()

```
public int hitungLuas() {
    return panjang * lebar;
}

public int hitungKeliling() {
    return 2 * (panjang + lebar);
}
```

```
Persegi panjang ke-1
Masukkan panjang:
2
Masukkan lebar:
4
Persegi panjang ke-2
Masukkan panjang:
10
Masukkan lebar:
20
Persegi panjang ke-3
Masukkan panjang:
5
Masukkan lebar:
7
Persegi panjang ke-1: Panjang: 2, Lebar: 4, Luas: 8, Keliling: 12
Persegi panjang ke-2: Panjang: 10, Lebar: 20, Luas: 200, Keliling: 60
Persegi panjang ke-3: Panjang: 5, Lebar: 7, Luas: 35, Keliling: 24
Persegi panjang ke-1: Panjang: 5, Lebar: 5, Luas: 25, Keliling: 20
```

4. Lakukan modifikasi kode program untuk menampilkan luas dan keliling dengan contoh output sebagai berikut:

```
Persegi panjang ke-1
Masukkan panjang: 2
Masukkan lebar: 4
Persegi panjang ke-2
Masukkan panjang: 10
Masukkan lebar: 20
Persegi panjang ke-3
Masukkan panjang: 5
Masukkan lebar: 7
Persegi panjang ke-1, panjang: 2, lebar: 4, luas: 8, keliling: 12
Persegi panjang ke-2, panjang: 10, lebar: 20, luas: 200, keliling: 60
Persegi panjang ke-3, panjang: 5, lebar: 7, luas: 35, keliling: 24
```

5. Modifikasi kode program pada praktikum 3.3 agar length array ditentukan dari user melalui input dengan Scanner



```

jobsheet3 > code > J persegiPanjangDemo.java > ...
  Click here to ask Blackbox to help you code faster |
1  import java.util.Scanner;
2
3  Comment Code
4  public class persegiPanjangDemo {
5
6      Run | Debug
7      public static void main(String[] args) {
8          Scanner sc = new Scanner(System.in);
9
10         // masukkan panjang array
11         System.out.print(s:"Masukkan jumlah persegi panjang: ");
12         int jumlahPersegiPanjang = sc.nextInt();
13
14         persegiPanjang[] arrayOfPersegiPanjang = new persegiPanjang[jumlahPersegiPanjang];
15         int panjang, lebar;
16
17         for (int i = 0; i < jumlahPersegiPanjang; i++) {
18             System.out.println("Persegi panjang ke-" + (i + 1));
19             System.out.println(x:"Masukkan panjang: ");
20             panjang = sc.nextInt();
21             System.out.println(x:"Masukkan lebar: ");
22             lebar = sc.nextInt();
23
24             arrayOfPersegiPanjang[i] = new persegiPanjang(panjang, lebar);
25         }
26
27         for (int i = 0; i < jumlahPersegiPanjang; i++) {
28             System.out.print("Persegi panjang ke-" + (i + 1) + ": ");
29             arrayOfPersegiPanjang[i].cetakInfo();
30         }
31
32         sc.close();
33     }
34 }
  
```

```

Masukkan jumlah persegi panjang: 3
Persegi panjang ke-1
Masukkan panjang:
2
Masukkan lebar:
4
Persegi panjang ke-2
Masukkan panjang:
10
Masukkan lebar:
20
Persegi panjang ke-3
Masukkan panjang:
5
Masukkan lebar:
7
Persegi panjang ke-1: Panjang: 2, Lebar: 4, Luas: 8, Keliling: 12
Persegi panjang ke-2: Panjang: 10, Lebar: 20, Luas: 200, Keliling: 60
Persegi panjang ke-3: Panjang: 5, Lebar: 7, Luas: 35, Keliling: 24
  
```



3.5 Tugas

1. Sebuah kampus membutuhkan program untuk menampilkan informasi mahasiswa berupa nama, NIM, jenis kelamin dan juga IPK mahasiswa. Program dapat menerima input semua informasi tersebut, kemudian menampilkanya kembali ke user. Tambahkan informasi rata-rata IPK pada bagian akhir. Gunakan looping dengan **FOR** untuk pembuatan object. Gunakan looping dengan **FOREACH** untuk pencetakan data ke layar.

Contoh output sebagai berikut:

```
Masukkan data mahasiswa ke- 1
Masukkan nama:Rina
Masukkan NIM:1234567
Masukkan jenis kelamin:P
Masukkan IPK:3.6

Masukkan data mahasiswa ke- 2
Masukkan nama:Rio
Masukkan NIM:7654321
Masukkan jenis kelamin:L
Masukkan IPK:4.0

Masukkan data mahasiswa ke- 3
Masukkan nama:Reza
Masukkan NIM:8765398
Masukkan jenis kelamin:L
Masukkan IPK:3.8

Data Mahasiswa ke-1
Nama: Rina
Nim: 1234567
Jenis kelamin: P
Nilai IPK: 3.6
Data Mahasiswa ke-2
nama: Rio
NIM: 7654321
Jenis kelamin: L
Nilai IPK: 4.0
Data Mahasiswa ke-3
nama: Reza
NIM: 8765398
Jenis kelamin: L
Nilai IPK: 3.8

Rata-rata IPK: 3.8
```

```

jobsheet3 > code > J mahasiswaMain06.java > ...
  Click here to ask Blackbox to help you code faster |
import java.util.Scanner;

2

Comment Code
3 public class mahasiswaMain06 {
4
5     Run | Debug
6     public static void main(String[] args) {
7         Scanner scanner = new Scanner(System.in);
8
9         // Meminta pengguna untuk memasukkan jumlah mahasiswa
10        System.out.print(s:"Enter the number of students: ");
11        int jumlahMahasiswa = scanner.nextInt();
12
13        // Membuat array untuk menyimpan objek Mahasiswa
14        mahasiswa06[] mahasiswaArray = new mahasiswa06[jumlahMahasiswa];
15
16        // Input data mahasiswa
17        for (int i = 0; i < jumlahMahasiswa; i++) {
18            System.out.println("\nEnter student data to-" + (i + 1));
19
20            System.out.print(s:"Enter a name: ");
21            String nama = scanner.next();
22
23            System.out.print(s:"Enter the NIM: ");
24            int nim = scanner.nextInt();
25
26            System.out.print(s:"Enter the gender (L/P): ");
27            char jenisKelamin = scanner.next().charAt(index:0);
28
29            System.out.print(s:"Enter the IPK: ");
30            double ipk = scanner.nextDouble();
31
32            // Membuat objek Mahasiswa dan menyimpannya dalam array
33            mahasiswaArray[i] = new mahasiswa06(nama, nim, jenisKelamin, ipk);
34        }
35
36        // Output data mahasiswa
37        for (int i = 0; i < jumlahMahasiswa; i++) {
38            System.out.println("\nEnter student data to-" + (i + 1));
39            mahasiswaArray[i].cetakInfo();
40        }
41
42        // Menghitung rata-rata IPK
43        double totalIpk = 0;
44        for (mahasiswa06 mahasiswa : mahasiswaArray) { // foreach
45            totalIpk += mahasiswa.getIpk();
46        }
47        double rataRataIpk = totalIpk / jumlahMahasiswa;
48
49        System.out.println("\nAverage IPK: " + rataRataIpk);
50
51        scanner.close();
52    }
53 }

```

Masukkan jumlah mahasiswa: 3
 Masukkan data mahasiswa ke-1
 Masukkan nama: Rina
 Masukkan NIM: 1234567
 Masukkan jenis kelamin (L/P): P
 Masukkan IPK: 3.6
 Masukkan data mahasiswa ke-2
 Masukkan nama: Rio
 Masukkan NIM: 7654321
 Masukkan jenis kelamin (L/P): L
 Masukkan IPK: 4.0
 Masukkan data mahasiswa ke-3
 Masukkan nama: Reza
 Masukkan NIM: 8765398
 Masukkan jenis kelamin (L/P): L
 Masukkan IPK: 3.8
 Data Mahasiswa ke-1
 Nama: Rina
 NIM: 1234567
 Jenis kelamin: P
 Nilai IPK: 3.6
 Data Mahasiswa ke-2
 Nama: Rio
 NIM: 7654321
 Jenis kelamin: L
 Nilai IPK: 4.0
 Data Mahasiswa ke-3
 Nama: Reza
 NIM: 8765398
 Jenis kelamin: L
 Nilai IPK: 3.8
 Rata-rata IPK: 3.7999999999999994

```

jobsheet3 > code > J mahasiswa06.java > ...
  Click here to ask Blackbox to help you code faster | Comment Code |
public class mahasiswa06 {
1
2     private String nama;
3     private long nim;
4     private char jenisKelamin;
5     private double ipk;
6
7     // Constructor untuk membuat objek Mahasiswa
8     public mahasiswa06(String nama, int nim, char jenisKelamin, double ipk) {
9         this.nama = nama;
10        this.nim = nim;
11        this.jenisKelamin = jenisKelamin;
12        this.ipk = ipk;
13    }
14
15    // Metode untuk mencetak informasi mahasiswa
16    public void cetakInfo() {
17        System.out.println("Name: " + nama);
18        System.out.println("NIM: " + nim);
19        System.out.println("Gender: " + jenisKelamin);
20        System.out.println("IPK: " + ipk);
21    }
22
23    // Getter untuk mendapatkan nilai IPK
24    public double getIpk() {
25        return ipk;
26    }
27 }

```

2. Implementasikan tugas ASD Teori no 1 ke dalam kode program. Buatlah array of objects berdasarkan input dari user dan tampilkan data seperti pada Tugas no 1.

```

jobsheet3 > code > J product06.java > ...
  Click here to ask Blackbox to help you code faster | Comment Code |
1 + public class product06 {
2     public String productName;
3     public double price;
4     public int quantity;
5
6     public product06(String productName, double price, int quantity) {
7         this.productName = productName;
8         this.price = price;
9         this.quantity = quantity;
10    }
11
12    public void cetakInfo() {
13        System.out.println("Name of product: " + productName);
14        System.out.println("Price: " + price);
15        System.out.println("Quantity: " + quantity);
16    }
17 }

jobsheet3 > code > J transaction06.java > ...
  Click here to ask Blackbox to help you code faster | Comment Code |
1 class transaction06 {
2     product06[] products;
3     int productCount;
4
5     public transaction06(int maxSize) {
6         products = new product06[maxSize];
7         productCount = 0;
8     }
9
10    public void addProduct(product06 product) {
11        products[productCount++] = product;
12    }
13
14    public double calculateTotal() {
15        double total = 0;
16        for (int i = 0; i < productCount; i++) {
17            total += products[i].price * products[i].quantity;
18        }
19        return total;
20    }
21 }
  
```

```

jobsheet3 > code > J productMain06.java > ...
  Click here to ask Blackbox to help you code faster |
import java.util.Scanner;

1
2
Comment Code
3 public class productMain06 {
  Run | Debug
4     public static void main(String[] args) {
5         Scanner sc = new Scanner(System.in);
6
7         System.out.print(s:"Enter the number of products: ");
8         int productAmount = sc.nextInt();
9
10        // Membuat array untuk menyimpan objek Product06
11        product06[] productArray = new product06(productAmount);
12
13        for (int i = 0; i < productAmount; i++) {
14            sc.nextLine(); // Consume the newline character
15            System.out.println("\nEnter details for product " + (i + 1));
16            System.out.print(s:"Product Name: ");
17            String productName = sc.nextLine();
18            System.out.print(s:"Price: ");
19            double price = sc.nextDouble();
20            System.out.print(s:"Quantity: ");
21            int quantity = sc.nextInt();
22
23            productArray[i] = new product06(productName, price, quantity);
24        }
25
26        // Membuat objek Transaction dan menambahkan produk ke dalamnya
27        transaction06 transaction = new transaction06(productAmount);
28
29        for (int i = 0; i < productAmount; i++) {
30            transaction.addProduct(productArray[i]);
31        }
32
33        // Menampilkan informasi produk dalam transaksi
34        System.out.println(x:"\nTransaction details:");
35        for (int i = 0; i < productAmount; i++) {
36            System.out.println("\nProduct " + (i + 1) + " details:");
37            productArray[i].cetakInfo();
38        }
39
40        // Menghitung dan menampilkan total harga transaksi
41        double totalTransaction = transaction.calculateTotal();
42        System.out.println("\nTotal Transaction Amount: Rp." + String.format(format:"%.2f", totalTransaction));
43
44        sc.close();
45    }
46 }
  
```



```
Enter the number of products: 2

Enter details for product 1
Product Name: Milk
Price: 7000
Quantity: 3

Enter details for product 2
Product Name: Soap
Price: 3000
Quantity: 2

Transaction details:

Product 1 details:
Name of product: Milk
Price: 7000.0
Quantity: 3

Product 2 details:
Name of product: Soap
Price: 3000.0
Quantity: 2

Total Transaction Amount: Rp.27000.00
```