

Nama : Elsa Marthalinda

Prodi : D-IV Teknik Informatika

NIM : 254107020204

### LAPORAN JOBSHEET 3

#### 3.2 Percobaan 1: Membuat Array dari Object, Mengisi dan Menampilkan

```
'  
NIM      : 244107060033  
Nama     : AGNES TITANIA KINANTI  
Kelas    : SIB-1E  
IPK      : 3.75  
-----  
NIM      : 2341720172  
Nama     : ACHMAD MAULANA HAMZAH  
Kelas    : TI-2A  
IPK      : 3.36  
-----  
NIM      : 244107023006  
Nama     : DIRHAMAWAN PUTRANTO  
Kelas    : TI-2E  
IPK      : 3.8  
-----'
```

Kode Program:

```
package Jobsheet3;  
public class Mahasiswa7 {  
    public String nim;  
    public String nama;  
    public String kelas;  
    public float ipk;  
}
```

```
package Jobsheet3;  
public class MahasiswaDemo7 {  
    public static void main(String[] args) {  
  
        Mahasiswa7[] arrayOfMahasiswa = new Mahasiswa7[3];  
        arrayOfMahasiswa[0] = new Mahasiswa7();
```

```
arrayOfMahasiswa[0].nim = "244107060033";
arrayOfMahasiswa[0].nama = "AGNES TITANIA KINANTI";
arrayOfMahasiswa[0].kelas = "SIB-1E";
arrayOfMahasiswa[0].ipk = (float) 3.75;

arrayOfMahasiswa[1] = new Mahasiswa7();
arrayOfMahasiswa[1].nim = "2341720172";
arrayOfMahasiswa[1].nama = "ACHMAD MAULANA HAMZAH";
arrayOfMahasiswa[1].kelas = "TI-2A";
arrayOfMahasiswa[1].ipk = (float) 3.36;

arrayOfMahasiswa[2] = new Mahasiswa7();
arrayOfMahasiswa[2].nim = "244107023006";
arrayOfMahasiswa[2].nama = "DIRHAMAWAN PUTRANTO";
arrayOfMahasiswa[2].kelas = "TI-2E";
arrayOfMahasiswa[2].ipk = (float) 3.80;

System.out.println("NIM      : "+ arrayOfMahasiswa[0].nim);
System.out.println("Nama    : "+ arrayOfMahasiswa[0].nama);
System.out.println("Kelas   : "+ arrayOfMahasiswa[0].kelas);
System.out.println("IPK     : "+ arrayOfMahasiswa[0].ipk);
System.out.println("-----");
System.out.println("NIM      : "+ arrayOfMahasiswa[1].nim);
System.out.println("Nama    : "+ arrayOfMahasiswa[1].nama);
System.out.println("Kelas   : "+ arrayOfMahasiswa[1].kelas);
System.out.println("IPK     : "+ arrayOfMahasiswa[1].ipk);
System.out.println("-----");
System.out.println("NIM      : "+ arrayOfMahasiswa[2].nim);
System.out.println("Nama    : "+ arrayOfMahasiswa[2].nama);
System.out.println("Kelas   : "+ arrayOfMahasiswa[2].kelas);
System.out.println("IPK     : "+ arrayOfMahasiswa[2].ipk);
System.out.println("-----");
}

}
```

### 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!

➤ Tidak. Karena isi dari sebuah class tersebut sangat bergantung pada kebutuhan desain program itu sendiri. Sebuah class bisa didesain hanya berisi atribut jika perannya murni sebagai struktur penyimpan data (sebagai *model*).

2. Apa yang dilakukan oleh kode program berikut?

```
Mahasiswa[] arrayOfMahasiswa = new Mahasiswa[3];
```

➤ Baris kode tersebut melakukan deklarasi sekaligus pembuatan (instansiasi) wadah array bernama arrayOfMahasiswa dengan tipe class Mahasiswa. Panjang array tersebut sebanyak 3 elemen. Namun, objek Mahasiswa di dalam masing-masing indeks array tersebut belum dibuat (masih bernilai null)

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

```
arrayOfMahasiswa[0] = new Mahasiswa();
```

➤ Tidak memiliki knstruktor. Tetapi pemanggilan new Mahasiswa() tetap bisa dilakukan karena Java akan otomatis membuatkan dan menyediakan default constructor (konstruktor kosong) untuk setiap class yang tidak mendefinisikan konstruktor sama sekali.

4. Apa yang dilakukan oleh kode program berikut?

```
arrayOfMahasiswa[0] = new Mahasiswa();
arrayOfMahasiswa[0].nim = "244107060033";
arrayOfMahasiswa[0].nama = "AGNES TITANIA KINANTI";
arrayOfMahasiswa[0].kelas = "SIB-1E";
arrayOfMahasiswa[0].ipk = (float) 3.75;
```

➤ arrayOfMahasiswa[0] = new Mahasiswa(); : Menciptakan (instansiasi) satu objek Mahasiswa baru secara spesifik dan meletakkannya di dalam array indeks ke-0.  
➤ Baris selanjutnya: Mengakses objek yang baru dibuat di indeks ke-0 tersebut untuk mengisi nilai pada atribut-atributnya (nim, nama, kelas, ipk).

5. Mengapa class **Mahasiswa** dan **MahasiswaDemo** dipisahkan pada uji coba 3.2?

- Karena class **Mahasiswa** bertindak murni sebagai Blueprint/Cetakan yang mendefinisikan kerangka data (atribut), sedangkan class **MahasiswaDemo** bertindak sebagai Driver Class/Main Class yang khusus digunakan untuk mengeksekusi program, menjalankan main method, instansiasi objek, dan berinteraksi dengan alur logika.

### 3.3 Percobaan 2: Meneripa Input Isian Array Menggunakan Looping

```

Masukkan Data Mahasiswa ke-1
NIM      : 244107060033
Nama     : AGNES TITANIA KINANTI
Kelas    : SIB-1E
IPK      : 3.75

-----
Masukkan Data Mahasiswa ke-2
NIM      : 2341720172
Nama     : ACHMAD MAULANA HAMZAH
Kelas    : TI-2A
IPK      : 3.36

-----
Masukkan Data Mahasiswa ke-3
NIM      : 244107023006
Nama     : DIRHAMAWAN PUTRANTO
Kelas    : TI-2E
IPK      : 3.80

-----
Data Mahasiswa ke-1
NIM      : 244107060033
Nama     : AGNES TITANIA KINANTI
Kelas    : SIB-1E
IPK      : 3.75

-----
Data Mahasiswa ke-2
NIM      : 2341720172
Nama     : ACHMAD MAULANA HAMZAH
Kelas    : TI-2A
IPK      : 3.36

-----
Data Mahasiswa ke-3
NIM      : 244107023006
Nama     : DIRHAMAWAN PUTRANTO
Kelas    : TI-2E
IPK      : 3.8
-----
```

Kode program:

```

package Jobsheet3;
import java.util.Scanner;
public class MahasiswaDemo7 {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
```

```
Mahasiswa7[] arrayOfMahasiswa = new Mahasiswa7[3];
arrayOfMahasiswa[0] = new Mahasiswa7();
arrayOfMahasiswa[0].nim = "244107060033";
arrayOfMahasiswa[0].nama = "AGNES TITANIA KINANTI";
arrayOfMahasiswa[0].kelas = "SIB-1E";
arrayOfMahasiswa[0].ipk = (float) 3.75;
String dummy;

arrayOfMahasiswa[1] = new Mahasiswa7();
arrayOfMahasiswa[1].nim = "2341720172";
arrayOfMahasiswa[1].nama = "ACHMAD MAULANA HAMZAH";
arrayOfMahasiswa[1].kelas = "TI-2A";
arrayOfMahasiswa[1].ipk = (float) 3.36;
for(int i = 0; i < 3; i++) {
    arrayOfMahasiswa[i] = new Mahasiswa7();

arrayOfMahasiswa[2] = new Mahasiswa7();
arrayOfMahasiswa[2].nim = "244107023006";
arrayOfMahasiswa[2].nama = "DIRHAMAWAN PUTRANTO";
arrayOfMahasiswa[2].kelas = "TI-2E";
arrayOfMahasiswa[2].ipk = (float) 3.80;

System.out.println("Masukkan Data Mahasiswa ke-" + (i + 1));
System.out.print("NIM      : ");
arrayOfMahasiswa[i].nim = sc.nextLine();
System.out.print("Nama      : ");
arrayOfMahasiswa[i].nama = sc.nextLine();
System.out.print("Kelas     : ");
arrayOfMahasiswa[i].kelas = sc.nextLine();
System.out.print("IPK       : ");
dummy = sc.nextLine();
arrayOfMahasiswa[i].ipk = Float.parseFloat(dummy);
System.out.println("-----");
}
```

```

System.out.println("NIM      : "+ arrayOfMahasiswa[0].nim);
System.out.println("Nama     : "+ arrayOfMahasiswa[0].nama);
System.out.println("Kelas    : "+ arrayOfMahasiswa[0].kelas);
System.out.println("IPK      : "+ arrayOfMahasiswa[0].ipk);
System.out.println("-----");
System.out.println("NIM      : "+ arrayOfMahasiswa[1].nim);
System.out.println("Nama     : "+ arrayOfMahasiswa[1].nama);
System.out.println("Kelas    : "+ arrayOfMahasiswa[1].kelas);
System.out.println("IPK      : "+ arrayOfMahasiswa[1].ipk);
System.out.println("-----");
System.out.println("NIM      : "+ arrayOfMahasiswa[2].nim);
System.out.println("Nama     : "+ arrayOfMahasiswa[2].nama);
System.out.println("Kelas    : "+ arrayOfMahasiswa[2].kelas);
System.out.println("IPK      : "+ arrayOfMahasiswa[2].ipk);
System.out.println("-----");
for(int i=0; i < 3; i++) {
    System.out.println("Data Mahasiswa ke- " + (i + 1));
    System.out.println("NIM      : "+ arrayOfMahasiswa[i].nim);
    System.out.println("Nama     : "+ arrayOfMahasiswa[i].nama);
    System.out.println("Kelas    : "+ arrayOfMahasiswa[i].kelas);
    System.out.println("IPK      : "+ arrayOfMahasiswa[i].ipk);
    System.out.println("-----");
}
}
}

```

### 3.3.3 Pertanyaan

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

```

public void cetakInfo() {
    System.out.println("NIM      : " + nim);
    System.out.println("Nama     : " + nama);
    System.out.println("Kelas    : " + kelas);
    System.out.println("IPK      : " + ipk);
    System.out.println("-----");
}

```

```
> arrayOfMahasiswa[i].cetakInfo();
```

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

```
Mahasiswa[] myArrayOfMahasiswa = new Mahasiswa[3];
myArrayOfMahasiswa[0].nim = "244107060033";
myArrayOfMahasiswa[0].nama = "AGNES TITANIA KINANTI";
myArrayOfMahasiswa[0].kelas = "SIB-1E";
myArrayOfMahasiswa[0].ipk = (float) 3.75;
```

- Karena jumlah elemen array sudah dibuat dengan new Mahasiswa[3] dan belum melakukan instansiasi objek untuk ruang indeks ke-0 tersebut. Jadi kita tidak bisa mengisikan atribut nim, nama, dll. sebelum objeknya benar-benar ada.

### 3.4 Percobaan 3: Constructor Berparameter

```
Masukkan Data Matakuliah ke-1
Kode      : 12345
Nama      : Algoritma & Struktur Data
SkS       : 2
Jumlah Jam : 6

-----
Masukkan Data Matakuliah ke-2
Kode      : 54321
Nama      : Sistem Basis Data
SkS       : 2
Jumlah Jam : 4

-----
Masukkan Data Matakuliah ke-3
Kode      : 83652
Nama      : Dasar Pemrograman
SkS       : 2
Jumlah Jam : 4

-----
Data Matakuliah ke-1
Kode      : 12345
Nama      : Algoritma & Struktur Data
SkS       : 2
Jumlah Jam : 6

-----
Data Matakuliah ke-2
Kode      : 54321
Nama      : Sistem Basis Data
SkS       : 2
Jumlah Jam : 4

-----
Data Matakuliah ke-3
Kode      : 83652
Nama      : Dasar Pemrograman
SkS       : 2
Jumlah Jam : 4
```

Kode program:

```
package Jobsheet3;

public class Matakuliah7 {
    public String kode;
    public String nama;
    public int sks;
    public int jumlahJam;

    public Matakuliah7(String kode, String nama, int sks, int jumlahJam) {
        this.kode = kode;
        this.nama = nama;
        this.sks = sks;
        this.jumlahJam = jumlahJam;
    }
}
```

```
package Jobsheet3;
import java.util.Scanner;
public class MatakuliahDemo7 {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        Matakuliah7[] arrayOfMatakuliah = new Matakuliah7[3];
        String kode, nama, dummy;
        int sks, jumlahJam;

        for(int i = 0; i < 3; i++) {
            System.out.println("Masukkan Data Matakuliah ke-" + (i + 1));
            System.out.print("Kode : ");
            kode = sc.nextLine();
            System.out.print("Nama : ");
            nama = sc.nextLine();
            System.out.print("Sks : ");

```

```

        dummy = sc.nextLine();
        sks = Integer.parseInt(dummy);
        System.out.print("Jumlah Jam : ");
        dummy = sc.nextLine();
        jumlahJam = Integer.parseInt(dummy);
        System.out.println("-----");

        arrayOfMatakuliah[i] = new Matakuliah7(kode, nama, sks, jumlahJam);
    }

    for(int i=0; i < 3; i++) {
        System.out.println("Data Matakuliah ke-" + (i + 1));
        System.out.println("Kode      : " + arrayOfMatakuliah[i].kode);
        System.out.println("Nama      : " + arrayOfMatakuliah[i].nama);
        System.out.println("Sks       : " + arrayOfMatakuliah[i].sks);
        System.out.println("Jumlah Jam : " + arrayOfMatakuliah[i].jumlahJam);
        System.out.println("-----");
    }
}
}

```

### 3.4.3 Pertanyaan

- Apakah suatu class dapat memiliki lebih dari 1 constructor? Jika iya, berikan contohnya
  - Bisa. Tetapi konstrutor harus memiliki susunan atau jumlah parameter yang berbeda.

```

package Jobsheet3;
public class Matakuliah {
    public String kode;
    public String nama;

    // Konstruktor 1 (Tanpa Parameter / Kosong)
    public Matakuliah() {
    }

    // Konstruktor 2 (Dengan Parameter)
    public Matakuliah(String kode, String nama) {
        this.kode = kode;
        this.nama = nama;
    }
}

```

Contoh: }

2. Tambahkan method **tambahData()** pada class **Matakuliah**, kemudian gunakan method tersebut di class **MatakuliahDemo** untuk menambahkan data Matakuliah

```
-----  
== Tambah Data Matakuliah Baru ==  
Kode      : 20014  
Nama      : Agama  
Sks       : 2  
Jumlah Jam : 2  
Data ke-1 telah ditambah:  
Kode      : 20014  
Nama      : Agama  
Sks       : 2  
Jumlah Jam : 2  
-----
```



3. Tambahkan method **cetakInfo()** pada class **Matakuliah**, kemudian gunakan method tersebut di class **MatakuliahDemo** untuk menampilkan data hasil inputan di layer



4. Modifikasi kode program pada class **MatakuliahDemo** agar panjang (jumlah elemen) dari **array of object Matakuliah** ditentukan oleh user melalui input dengan Scanner

```
File Edit Selection View ... ← → | Algoritma dan Struktur Data | Run: MatakuliahDemo | + - ×
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS C:\Users\elfif\Pictures\Algoritma dan Struktur Data> & 'C:\Program Files\Java\jdk-24\bin\java.exe' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'C:\Users\elfif\AppData\Roaming\Code\User\worksapce\cestorage\962710f700ea1df9e\aa97397cc9216\vedhat.java;jdt_ws\Algoritma dan Struktur Data_7513a321\b1n' 'Jobsheet3.MatakuliahDemo'
Masukkan jumlah matakuliah : 2
Masukkan data Matakuliah ke-1
Kode      : 12345
Nama      : Agama
Sks       : 2
Jumlah Jam : 2
-----
Masukkan data Matakuliah ke-2
Kode      : 54321
Nama      : Sistem Basis Data
Sks       : 2
Jumlah Jam : 4
-----
Data Matakuliah ke-1
Kode      : 12345
Nama      : Agama
Sks       : 2
Jumlah Jam : 2
-----
Data Matakuliah ke-2
Kode      : 54321
Nama      : Sistem Basis Data
Sks       : 2
Jumlah Jam : 4
-----
== Tambah Data Matakuliah Baru ==
Kode      : 20014
Nama      : Matdas
Sks       : 2
Jumlah Jam : 4
Data ke-1 telah ditambah:
Kode      : 20014
Nama      : Matdas
Sks       : 2
Jumlah Jam : 4
-----
PS C:\Users\elfif\Pictures\Algoritma dan Struktur Data>
```

