



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.

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!
2. Apa yang dilakukan oleh kode program berikut?

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

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

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

4. Apa yang dilakukan oleh kode program berikut?

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

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

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.

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.
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;
```

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.
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.



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?
2. Apakah suatu class dapat memiliki lebih dari 1 constructor? Jika iya, berikan contohnya
3. Tambahkan method hitungLuas() dan hitungKeliling()
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

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
```

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.