JOBSHEET III

**ARRAY OF OBJECTS**

# Tujuan Praktikum

Setelah melakukan materi praktikum ini, mahasiswa mampu:

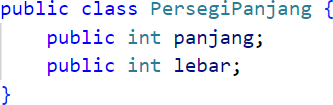
* + 1. Memahami dan menjelaskan fungsi array yang berisikan variabel object.
    2. Mahasiswa mampu menerapkan instansiasi array of objects dalam Java
    3. Mahasiswa mampu melakukan operasi terhadap elemen dalam array of objects

# Membuat Array dari Object, Mengisi dan Menampilkan

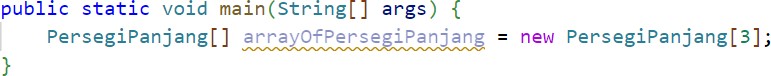
Didalam praktikum ini, kita akan mempraktekkan bagaimana membuat array of objects, kemudian mengisi dan menampilkan array tersebut.

## Langkah-langkah Percobaan

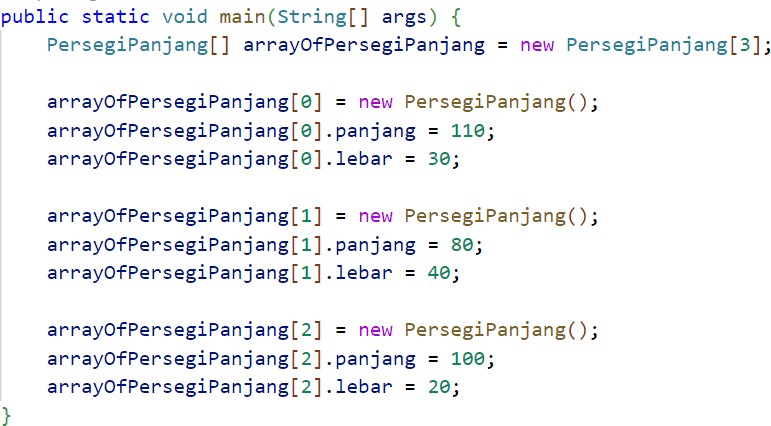
1. Buat folder baru dengan nama Praktikum03.
2. Buat class **PersegiPanjang**:



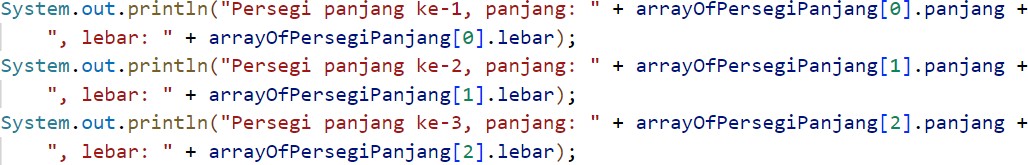
1. Buat class PersegiPanjangDemo kemudian tambahkan fungsi main sebagai berikut



1. Kemudian isikan masing-masing atributnya:



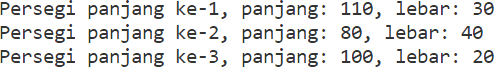
1. Cetak ke layar semua atribut dari objek **ppArray**:



1. Run program dan amati hasilnya.

## Verifikasi Hasil Percobaan

Cocokkan hasil compile kode program anda dengan gambar berikut ini.



## 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?



1. Apakah class PersegiPanjang memiliki konstruktor? Jika tidak, kenapa bisa dilakukan pemanggilan

konstruktur pada baris program berikut?



1. Apa yang dilakukan oleh kode program berikut?



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

# 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 mengisikan atribut dari semua persegi panjang

## Langkah-langkah Percobaan

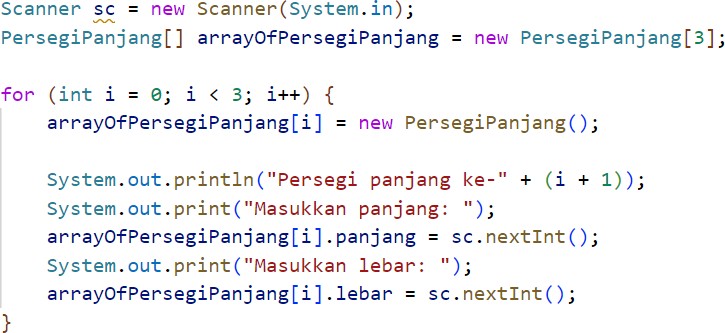
1. Import scanner pada class **PersegiPanjangDemo**.



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

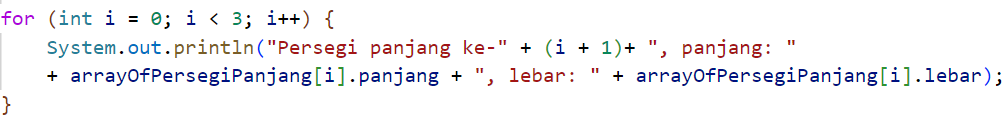
1. Pada praktikum 3.2 poin nomor 4, modifikasi kode program sebagai berikut. Buat objek **Scanner**

untuk menerima input, kemudian lakukan looping untuk menerima informasi panjang dan lebar:



1. Pada praktikum 3.2 poin nomor 5, modifikasi kode program sebagai berikut. Lakukan looping

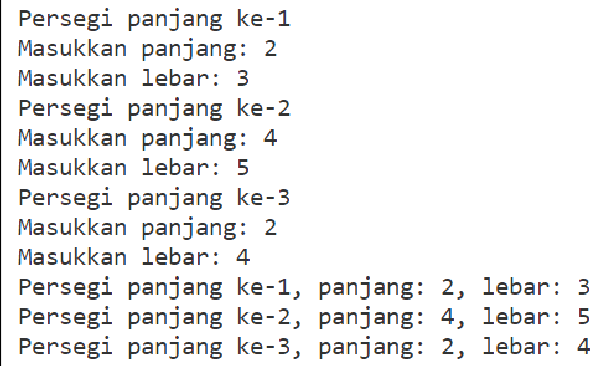
untuk mengakses elemen array dan menampilkan informasinya ke layar:



1. Run program dan amati hasilnya.

## Verifikasi Hasil Percobaan

Contoh verifikasi hasil percobaan ini.



## 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?

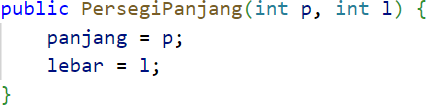
PersegiPanjang[] myArrayOfPersegiPanjang = new PersegiPanjang[100]; myArrayOfPersegiPanjang[5].panjang = 20;

# Constructor Berparameter

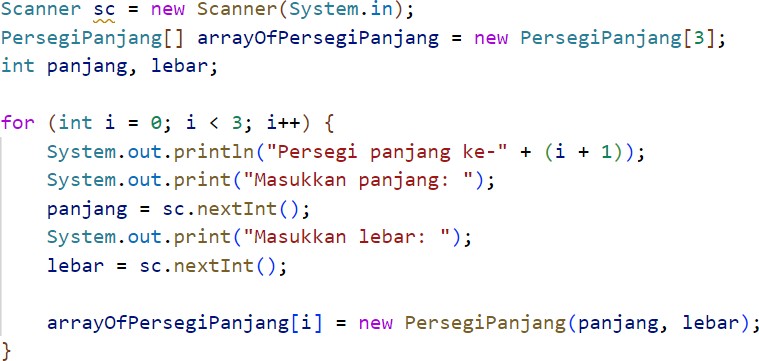
Pada praktikum ini kita akan melakukan pengoperasian matematika beberapa atribut pada masing-masing anggota array.

## Langkah-langkah Percobaan

1. Pada class PersegiPanjang, tambahkan constructor berparameter



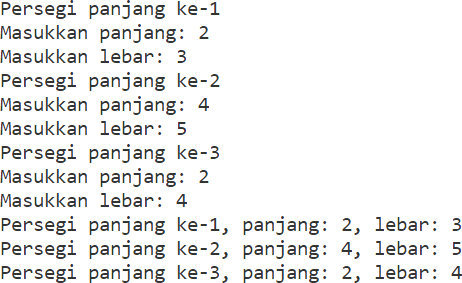
1. Run program kemudian amati hasilnya.
2. Modifikasi class PersegiPanjangDemo sehingga instansiasi dilakukan menggunakan constructor berparameter



1. Run progam dan amati hasilnya.

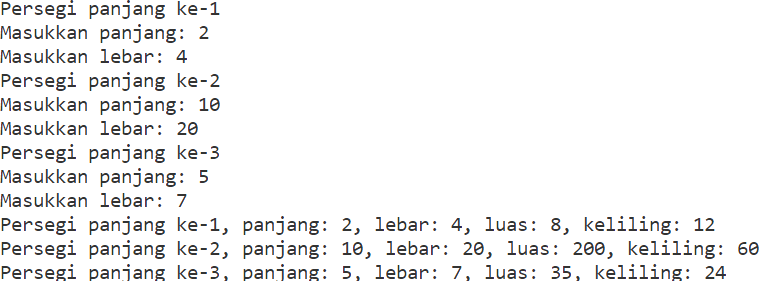
## Verifikasi Hasil Percobaan

Cocokkan hasil compile kode program anda dengan gambar berikut ini.



## 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:



* + - 1. Modifikasi kode program pada praktikum 3.3 agar length array ditentukan dari user melalui

input dengan Scanner

# 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

1. 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.