# LAPORAN PRAKTIKUM ALGORITMA DAN STRUKTUR DATA

Dosen Pengajar : Triana Fatmawati, S.T, M.T

# Jobsheet-3 Array of Object



Nama : Surya Rahmat Fatahillah

NIM : 2341760020

Prodi : Sistem Informasi Bisnis

# JURUSAN TEKNOLOGI INFORMASI POLITEKNIK NEGERI MALANG 2023/2024

# JOBSHEET III ARRAY OF OBJECTS

# 3.2 Membuat Array dari Object, Mengisi dan Menampilkan

# 3.2.1 Langkah-langkah Percobaan

# 1. Buat folder baru dengan nama Praktikum03.

# 2. Buat class PersegiPanjang:

# 

# 3. Buat class PersegiPanjangDemo kemudian tambahkan fungsi main sebagai berikut:

# 

# 4. Kemudian isikan masing-masing atributnya:

# 

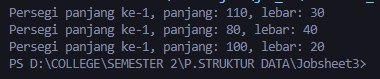
# 5. Cetak ke layar semua atribut dari objek ppArray:

# 

# 6. Run program dan amati hasilnya.

**3.2.2 Verifikasi Hasil Percobaan**

Cocokkan hasil compile kode program anda dengan gambar berikut ini.



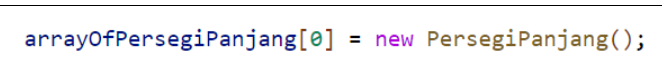
**3.2.3 Pertanyaan**

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

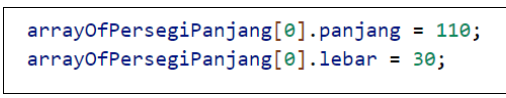
2. Apa yang dilakukan oleh kode program berikut?



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



4. Apa yang dilakukan oleh kode program berikut?



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

**JAWABAN!**

1. Tidak, class yang akan dibuat array of object tidak harus selalu memiliki atribut dan sekaligus method. Seperti yang ditunjukkan dalam contoh kode diatas, class PersegiPanjang27 hanya memiliki atribut panjang dan lebar, dan tidak memiliki method. Hal ini tetap valid dan memungkinkan untuk membuat array of object dari class tersebut.
2. Kode program ini mendeklarasikan variabel array bernama arrayoOfPersegiPanjang yang dapat menampung 3 atau dengan Panjang 3 objek dari class PersegiPanjang27. Kode ini juga membuat instance array baru dengan panjang 3 dan menginisialisasi elemen-elemen array dengan nilai default.
3. Tidak, class PersegiPanjang27 tidak memiliki konstruktor. Konstruktor default akan otomatis dibuat oleh Java jika tidak ada konstruktor yang didefinisikan secara eksplisit dalam class. Konstruktor default tersebut tidak mengambil argumen dan hanya memberikan nilai default untuk atribut-atribut class.

Pada baris program arrayoOfPersegiPanjang[0] = new PersegiPanjang27();, dilakukan pemanggilan konstruktor default yang dibuat oleh Java untuk membuat objek dari class PersegiPanjang27. Meskipun tidak ada konstruktor yang didefinisikan dalam class tersebut, Java akan membuat konstruktor default secara otomatis. Pemanggilan konstruktor ini akan membuat objek baru dari class PersegiPanjang27 dan kemudian objek tersebut disimpan di dalam elemen array arrayoOfPersegiPanjang pada indeks ke-0.

1. Kode program tersebut menginisialisasi nilai dari instance variables panjang dan lebar dari objek yang pertama (arrayoOfPersegiPanjang[0]) dari array arrayoOfPersegiPanjang.

arrayoOfPersegiPanjang[0].panjang = 110; mengatur nilai panjang dari objek pertama dalam array menjadi 110.

arrayoOfPersegiPanjang[0].lebar = 30; mengatur nilai lebar dari objek pertama dalam array menjadi 30.

1. Pemisahan class PersegiPanjang27 dan PersegiPanjangDemo dalam uji coba 3.2 bertujuan untuk memisahkan operasi yang terkait dengan objek PersegiPanjang itu sendiri (seperti pengaturan panjang dan lebar) dari logika aplikasi utama atau main class. Selain itu tujuan lain pemisahan ini bertujuan supaya kode lebih terstruktur, dan supaya tidak ada duplikasi class dalam satu file.

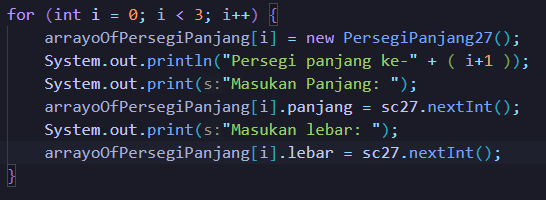
**3.3 Menerima Input Isian Array Menggunakan Looping**

**3.3.1 Langkah-langkah Percobaan**

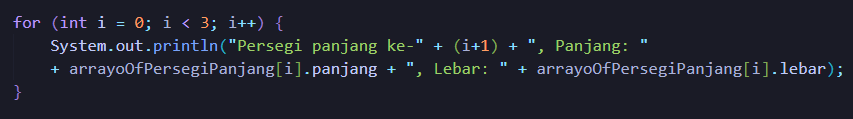
1. Import scanner pada class PersegiPanjangDemo.

****

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:



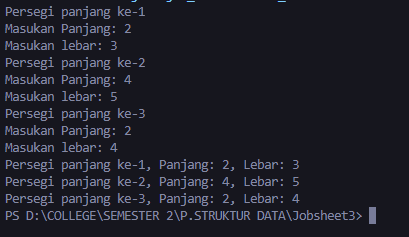
1. Pada praktikum 3.2 poin nomor 5, modifikasi kode program sebagai berikut. Lakukan looping untuk mengakses elemen array dan menampilkan informasinya ke layar:



1. Run program dan amati hasilnya.

**3.3.2 Verifikasi Hasil Percobaan**

Contoh verifikasi hasil percobaan ini



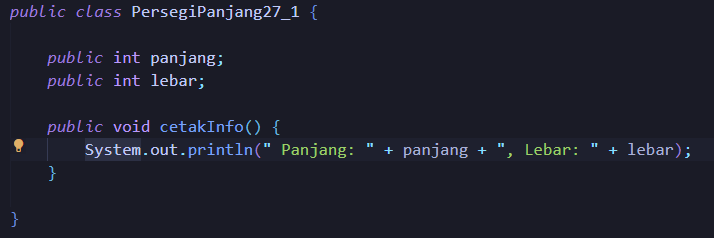
**3.3.3 Pertanyaan**

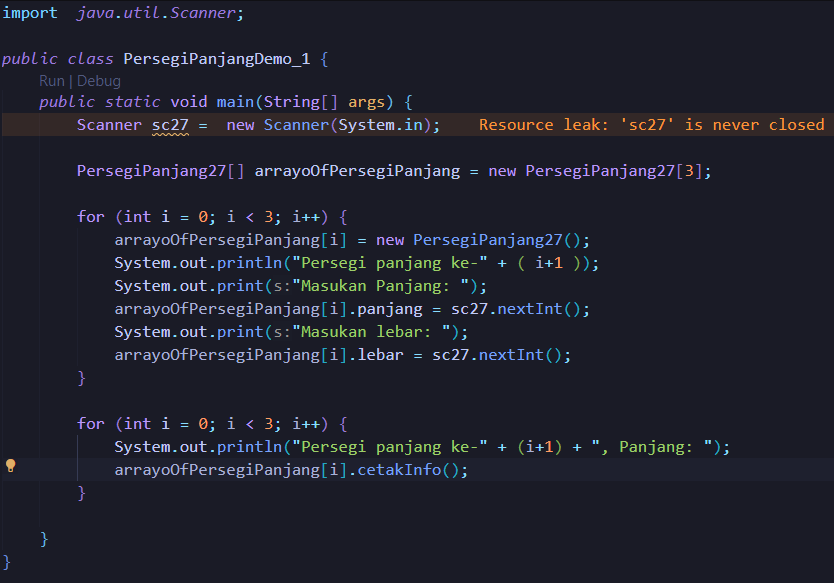
1. Tambahkan method cetakInfo() pada class PersegiPanjang kemudian modifikasi kode program pada langkah no 3.

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

**Jawaban!**

1. Berikut modfiikasi kode dengan menambahkan method cetakInfo() pada class PersegiPanjang



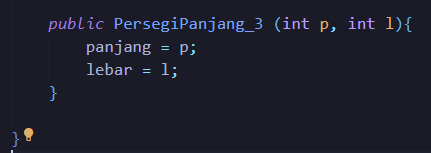


1. Kode tersebut error dikarenakan saat membuat array baru **myArrayOfPersegiPanjang** dari tipe **PersegiPanjang**, elemen-elemen array tersebut secara otomatis diinisialisasi dengan nilai default untuk tipe data atau null yang menyebabkan nilai pada index tersebut bernilai null. Oleh karena itu, ketika mencoba mengakses **myArrayOfPersegiPanjang[5].panjang**, sebenarnya kita mencoba mengakses atribut **panjang** dari objek yang bernilai null.

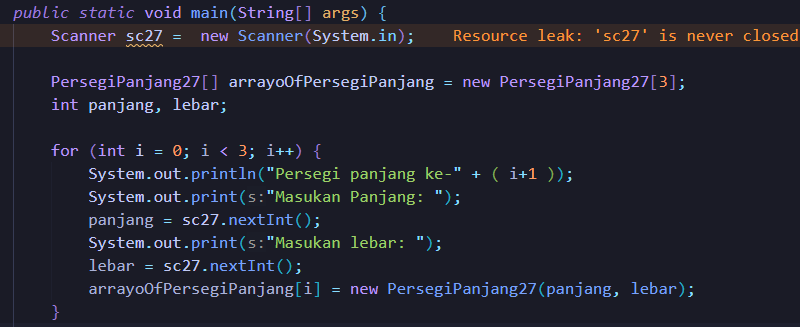
**3.4 Constructor Berparameter**

**3.4.1 Langkah-langkah Percobaan**

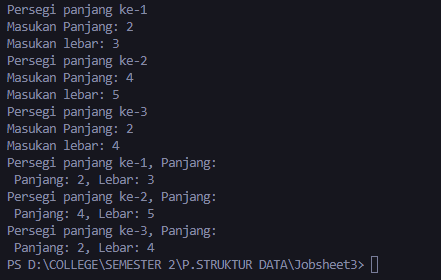
1. Pada class PersegiPanjang, tambahkan constructor berparameter

****

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

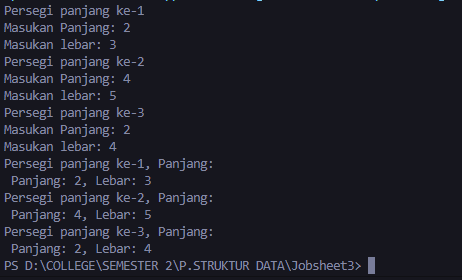


1. Run progam dan amati hasilnya.



**3.4.2 Verifikasi Hasil Percobaan**

Cocokkan hasil compile kode program anda dengan gambar berikut ini.



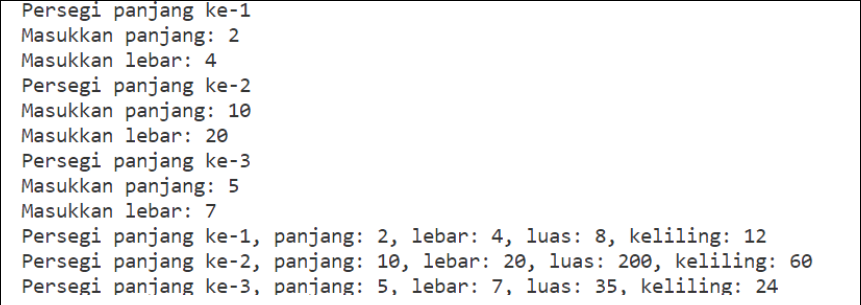
**3.4.3 Pertanyaan**

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:



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

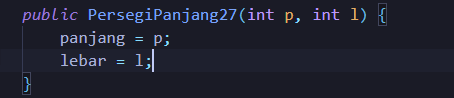
**JAWABAN!**

1. Karena ketika didalam suatu class ditambahkan constructor berparameter maka harus ditambahkan juga constructor yang tidak berparameter. Kode pada langkah ke 2 error karena program hanya menambahkan constructor berparameter tanpa adanya contstructor tidak berparameter.

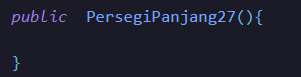


1. Bisa, suatu class bisa memiliki lebih dari 1 constructor. Beberapa constructor dalam suatu class dapat berbeda dalam jumlah dan jenis parameter yang diterimanya. Ada constructor berparameter dan ada constructor tidak berparameter. Berikut merupakan contoh constructor berparameter dan tidak berparameter:

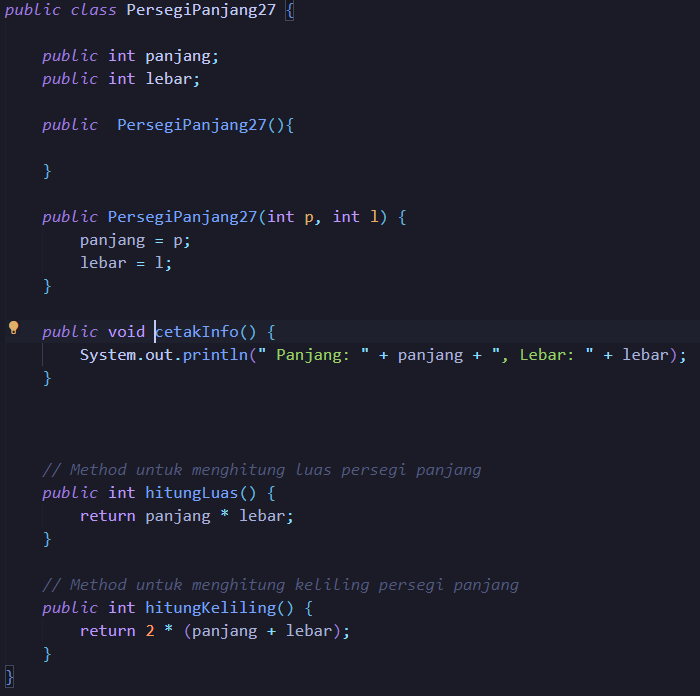
- Berparameter:



* Tidak Berparameter:



1. Berikut modifikasi kode program dengan menambahkan method hitungLuas() dan hitungKeliling()

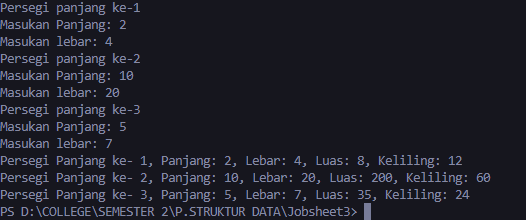


1. Berikut merupakan hasil modifikasi kode program untuk menampilkan luas dan keliling:

Kode program:

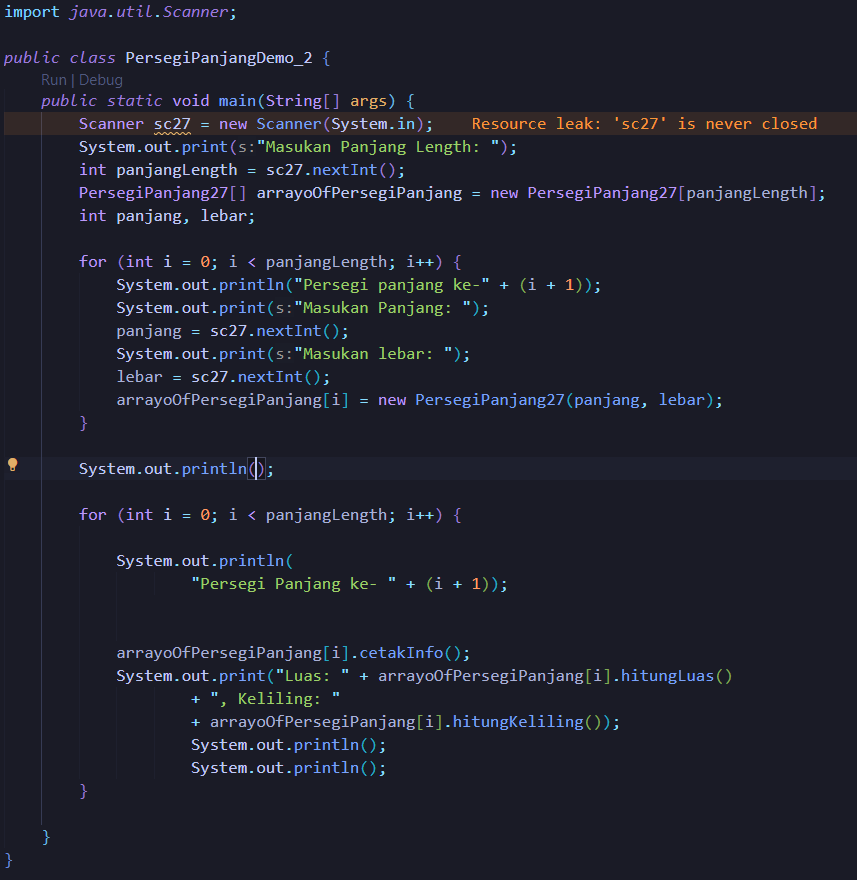


Output:

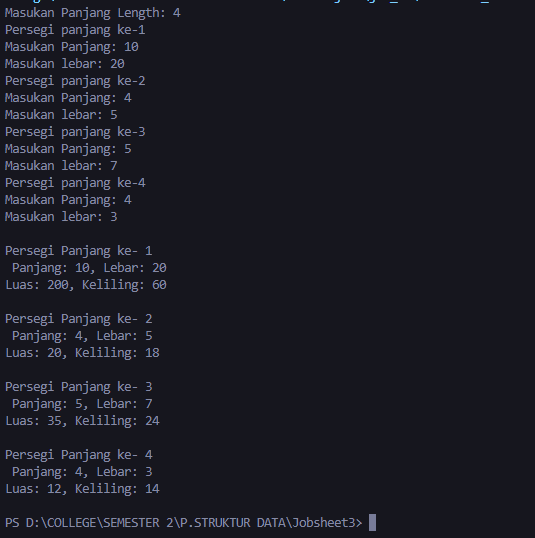


1. Berikut modifikasi kode program pada praktikum 3.3 agar length array ditentukan dari user melalui input dengan Scanner.

Kode program:



Output:



**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.

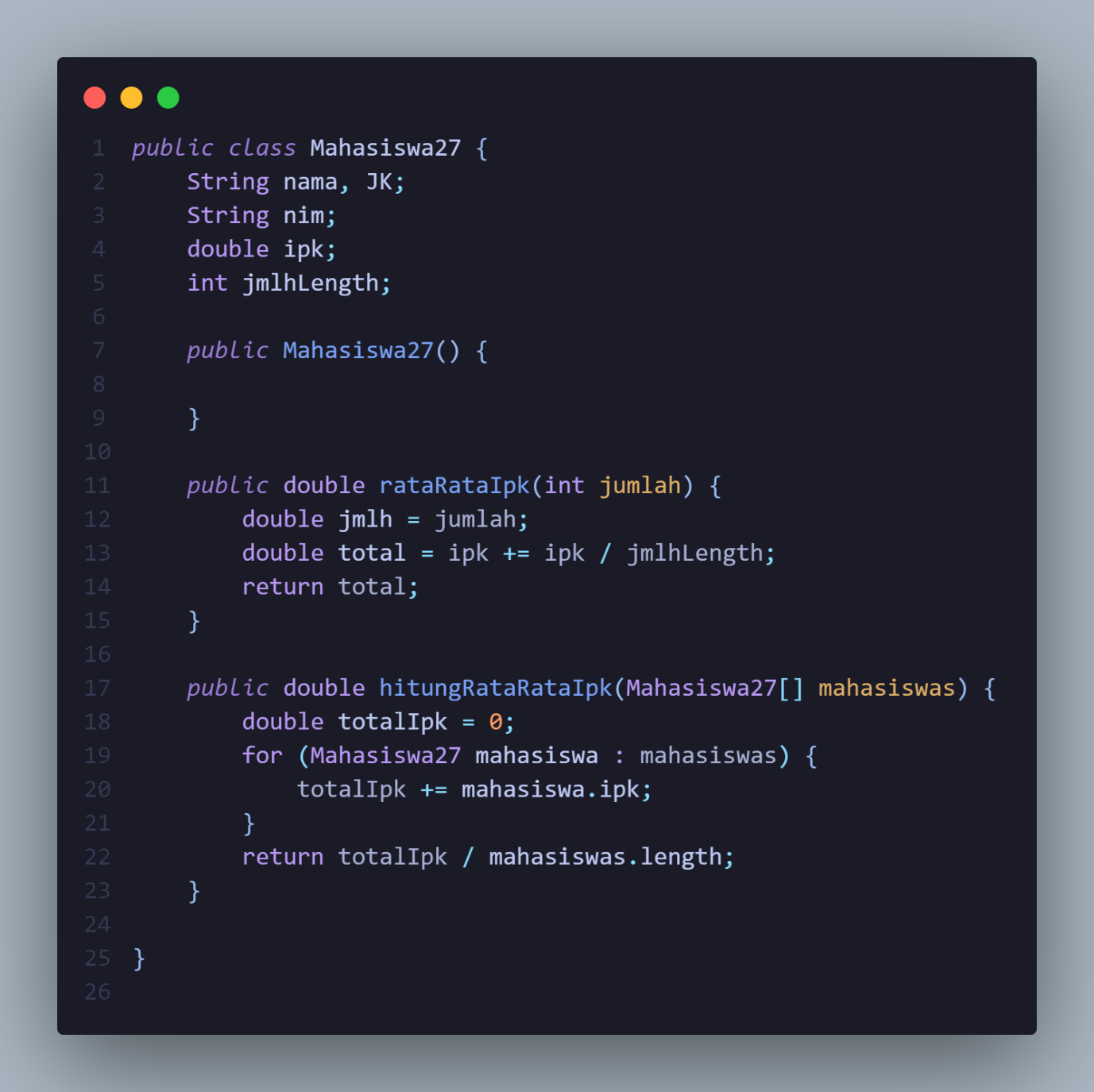
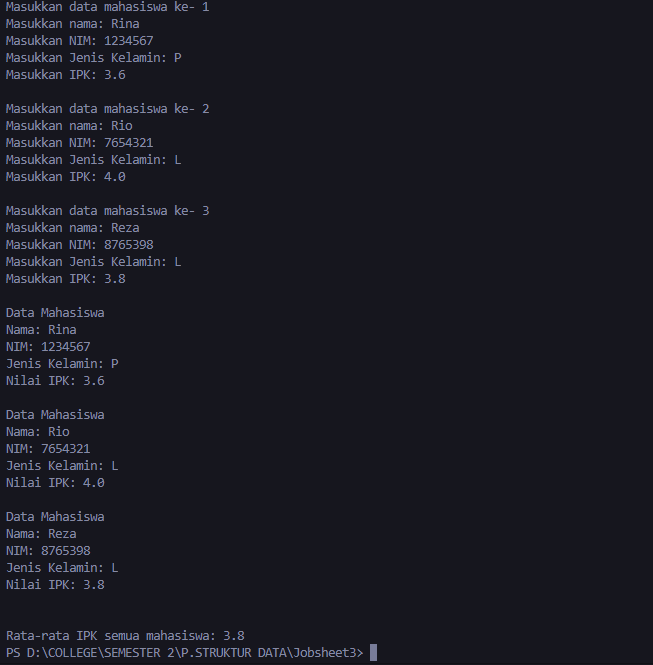
2. Implementasikan tugas ASD Teori no 1 ke dalam kode program. Buatlah array of objects berdasarkan input dari user dan tampilkan data seperti pada Tugas no 1

**JAWABAN!**

1. Berikut merupakan kode 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 dengan menggunakan FOREACH untuk pencetakan data ke layar:

* **Class MahasiswaMain27**



* **Class Mahasiswa27**
* **Hasil Output Program**

1. Berikut Implementasi tugas ASD Teori no 1 ke dalam kode program, terdapat beberapa class antara lain karyawan, karyawanMain, Admin dan AdminMain

* **Class Admin**



* **Class AdminMain**



* **Output Program**



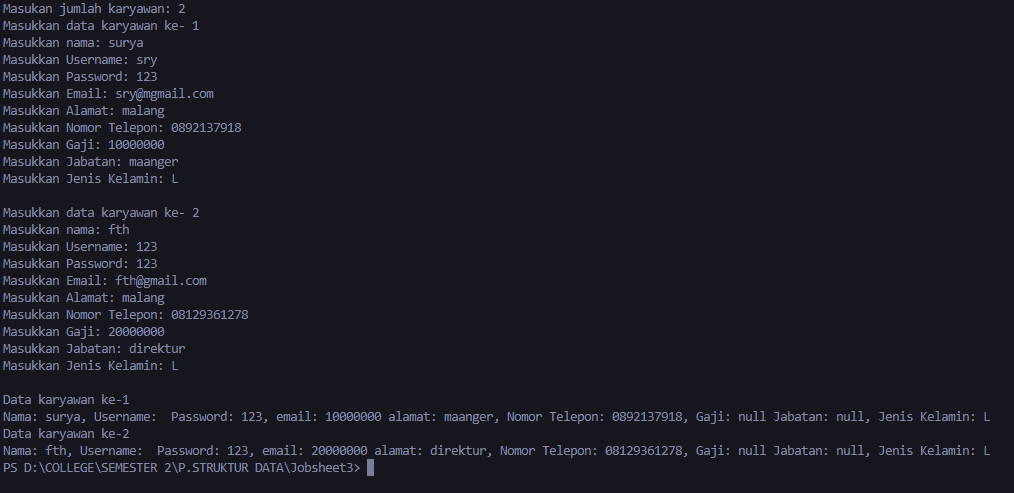
* **Class Karyawan**



* **Class KaryawanMain**



* **Ouput Program**



* **Class Gaji**



* **Class GajiAdmin**



* **Output Program**

