

Tugas Pendahuluan: Modul 5

NIM: 105222033

Nama: Mohamad Dandung Sadat

Soal

1. Cakupan variabel (Variable Scope):

- a. Jelaskan perbedaan antara variabel lokal, instance, kelas (static), dan parameter dalam Java!

Jawaban:

Variabel yang dideklarasikan di dalam metode, konstruktor, atau blok tertentu, dan hanya dapat diakses di dalamnya. Tidak disimpan sebagai bagian dari objek.

Instance: Variabel non-static yang dideklarasikan di dalam kelas tapi di luar metode. Setiap objek memiliki salinan sendiri dari variabel ini.

Kelas (Static) Variabel yang dideklarasikan dengan keyword static. Nilainya dimiliki bersama oleh semua objek dari kelas tersebut.

Parameter: Variabel yang dideklarasikan dalam tanda kurung () saat mendefinisikan metode atau konstruktor. Digunakan untuk menerima input saat metode/konstruktor dipanggil.

- b. Mengapa variabel static memiliki nilai yang sama untuk semua objek dari kelas yang sama?

Jawaban: Karena variabel static disimpan dalam memori kelas, bukan pada masing-masing objek. Artinya, hanya ada **satu salinan variabel static** yang dibagikan oleh **semua objek** dari kelas tersebut. Ketika satu objek mengubah nilainya, maka perubahan itu terlihat oleh semua objek lainnya.

2. Java packages:

- a. Apa tujuan utama penggunaan package dalam Java?

Jawaban:

Tujuan utama penggunaan package adalah:

- Mengorganisir kelas dan interface ke dalam grup yang terstruktur.
- Menghindari konflik nama antar kelas dengan nama sama di proyek besar.

- Meningkatkan keamanan dengan membatasi akses ke kelas.
 - Memudahkan pemeliharaan kode dan kolaborasi antar tim.
- b. Jelaskan perbedaan antara built-in package dan user-defined package dalam Java!

Jawaban:

Built-in Package: Paket yang sudah disediakan oleh Java, seperti java.util, java.io, java.lang. Tidak perlu membuatnya, tinggal diimpor dan digunakan.

User-defined Package: Paket yang dibuat sendiri oleh programmer untuk mengelompokkan kelas-kelas sesuai kebutuhan. Dibuat menggunakan package nama_paket; di awal file Java.

3. Access Modifiers:

- a. Sebutkan dan jelaskan empat jenis access modifier dalam Java!

Jawaban:

public: Dapat diakses dari mana saja, baik dalam paket yang sama maupun berbeda.

private: Hanya dapat diakses di dalam kelas tempat deklarasi.

protected: Dapat diakses dalam paket yang sama atau oleh subclass di luar paket.

default (tanpa modifier): Hanya bisa diakses dalam paket yang sama.

- b. Mengapa access modifier private sering digunakan dalam konsep enkapsulasi?

Jawaban: Karena private menyembunyikan data dari akses langsung oleh luar kelas. Hal ini menjaga **integritas data**, dan hanya memungkinkan perubahan melalui **getter dan setter** yang bisa dikontrol. Ini merupakan prinsip dasar dari **enkapsulasi** untuk menjaga keamanan dan keterbatasan akses terhadap data internal objek.

4. Enkapsulasi:

- a. Apa yang dimaksud dengan enkapsulasi dalam pemrograman berorientasi objek?

Jawaban: Enkapsulasi adalah konsep menyembunyikan detail internal suatu kelas dari akses langsung oleh luar kelas dan hanya memperbolehkan akses melalui metode publik. Tujuannya adalah untuk menjaga integritas data dan mengontrol bagaimana data diakses atau dimodifikasi.

- b. Mengapa penggunaan getter dan setter penting dalam implementasi enkapsulasi?

Jawaban: Getter dan setter memungkinkan:

- Akses terkontrol ke atribut private.
- Validasi atau logika tambahan saat data diakses atau dimodifikasi.
- Mencegah perubahan langsung terhadap atribut, menjaga keamanan data.

5. Array of object:

- a. Apa perbedaan antara array biasa dan array of object dalam Java?

Jawaban:

Array Biasa: Menyimpan tipe data primitif seperti int[], double[].

Array of Object: Menyimpan referensi ke objek dari sebuah kelas. Misalnya:

Mahasiswa[] daftar = new Mahasiswa[5];

- b. Sebutkan dan jelaskan dua cara untuk menginisialisasi array of object dalam Java!

Dalam Loop:

=====

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

```
for (int i = 0; i < daftar.length; i++) {  
    daftar[i] = new Mahasiswa();  
}
```

Saat Deklarasi:

```
Mahasiswa[] daftar = {  
    new Mahasiswa("A", "123"),  
    new Mahasiswa("B", "456"),  
    new Mahasiswa("C", "789")  
};
```
