

Tugas Pendahuluan: Modul 5

NIM: 105223002

Nama: Gerald Eberhard

Instruksi: Jawablah pertanyaan-pertanyaan berikut dengan jelas dan rinci. Gunakan contoh kode jika diperlukan untuk memperjelas jawaban Anda. Kumpulkan dengan format **TP5_NIM_PrakPBO.pdf**. Deadline, Kamis 27 Maret 2025, pukul 13:00 (sebelum praktikum). Link pengumpulan: <https://forms.gle/xhGM2XDmCeKpwrZP7>

Soal

1. Cakupan variabel (Variable Scope):

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

Jawab:

- Variabel lokal adalah variabel yang terdapat di dalam method
 - Variabel instance adalah variabel yang dideklarasikan di dalam kelas tetapi berada di luar method
 - Variabel kelas/static itu adalah variabel yang dideklarasikan dengan kata kunci static di dalam kelas dan berada di luar method. Perbedaan utama yang terdapat pada variabel ini adalah variabel ini itu bisa diubah tanpa harus memiliki objek, sedangkan variabel instance itu harus melalui objek. Jadi bisa dikatakan variabel static adalah variabel global yang terdapat di kelas tersebut. Artinya variabel ini akan dimiliki dan akan selalu bernilai sama di semua variabel oleh karena itu kalau ada satu object yang melakukan perubahan pada variabel ini maka variabel ini juga akan ikut diubah pada seluruh object yang lain.
 - Paramateter dalam java adalah variabel yang dideklarasikan di dalam tanda kurung (). Entrah itu pada method ataupun constructor.
- b. Mengapa variabel static memiliki nilai yang sama untuk semua objek dari kelas yang sama?

Jawab:

Hal ini dapat terjadi karena variabel static tidak memiliki keterikatan terhadap satupun object. Variabel ini adalah variabel yang terikat dengan kelas tersebut. Jadi setiap object yang dihasilkan oleh kelas yang mempunyai nilai static pasti akan dapat berbagi nilai tersebut.

2. Java packages:

- a. Apa tujuan utama penggunaan package dalam Java?

Jawab:

- Mengorganisasi kode agar lebih terstruktur dan mudah dikelola.
- Mencegah konflik nama antara kelas dengan menyediakan namespace unik.

- Mengatur hak akses antar kelas melalui access modifier.
- Memudahkan penggunaan ulang kode (reusability).

b. Jelaskan perbedaan antara built-in package dan user-defined package dalam Java!

Jawab:

Dari namanya saja sudah kelihatan bedanya yang satu built-in package artinya ini adalah package yang telah disediakan langsung dari java contohnya adalah java.util. Sedangkan user-defined package adalah package yang dibuat sendiri oleh user untuk mengorganisir kelas dan antarmuka yang telah mereka buat sendiri.

3. Access Modifiers:

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

Jawab:

- Public: label yang berfungsi memberikan sifat dapat diakses dari manapun dan oleh siapapun.
- Private: label yang berfungsi memberikan sifat akses hanya pada class itu sendiri
- Protected: label yang berfungsi memberikan sifat akses hanya pada class yang masih berada dalam satu package yang sama atau subclass atau kelas turunannya.
- Default/no modifier: label yang muncul apabila pada saat kita membuat suatu kelas kita tidak memberikan keterangan public/private/protected. Dengan demikian hal ini akan mengakibatkan kelas hanya dapat diakses oleh pada classnya sendiri dan class yang berada di dalam satu package dengannya.

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

Jawab:

Untuk menjawab ini kita harus paham apa itu enkapsulasi. Enkapsulasi adalah proses membuat codingan itu menjadi modular. Apa yang dimaksud dengan codingan tersebut menjadi modular? Yaitu adanya method-method yang menyembunyikan informasi dari program tersebut. Jadi intinya enkapsulasi itu adalah bagaimana menjaga agar data ini tetap utuh dan terhindar dari akses yang tidak diinginkan oleh karena perlu adanya information hiding. Nah salah satu opsi terbaik untuk melakukan ini adalah dengan menggunakan private access modifier.

4. Enkapsulasi:

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

Jawab:

Enkapsulasi adalah prinsip OOP yang menggabungkan data (variabel) dan metode yang memanipulasi data tersebut dalam satu unit (kelas), sambil menyembunyikan detail implementasi dari luar. Hal ini dicapai dengan menggunakan access modifier seperti private untuk variabel dan menyediakan akses melalui metode public.

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

Jawab:

Sebenarnya ini adalah sebuah pertanyaan yang istilah klise banget. Hal ini tentunya sudah pasti harus demikian karena kalau tidak maka bagaimana caranya agar kita bisa mengakses variabel yang di private tersebut? Atau bagaimana cara kita untuk mengubah nilai dari data yang telah di buat private? Tanpa konsep getter dan setter hal ini dapat dikatakan mustahil.

5. Array of object:

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

Jawab:

Sebenarnya konsepnya sama aja, sama-sama merupakan array. Bedanya satunya array yang lebih kecil sedangkan yang satu lagi adalah array yang berukuran lebih besar. Secara teknis array biasa hanya menyimpan tipe data primitif dan string. Sedangkan array of object itu akan menyimpan object dimana object itu memiliki tipe data yang lebih kompleks karena terdiri dari beberapa tipe data primitive atau dapat kita sebut juga sebagai tipe data komposit.

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

Jawab:

- Inisialisasi array kemudian isi array of object tersebut
Hal ini itu sederhananya kita buat dulu array-nya baru kemudian pada line codingan berikutnya kita isi codingannya satu persatu.
- Inisialisasi langsung pada saat membuat array of objects
Kalo di atas tadi kita buat secara terpisah yaitu satu persatu. Ada cara lain juga yang dapat kita gunakan yaitu kita bisa saja buat bersamaan sekaligus. Jadi kita buat array langsung beserta dengan isinya.

Referensi

- [1]. [Scope of a variable - GeeksforGeeks](#)
- [2]. Materi Pelajaran dari pak Randi
- [3]. [Pengertian, Fungsi Dan Kegunaan Paket \(Package\) di Java](#)
- [4]. [User-Defined Packages in Java - GeeksforGeeks](#)
- [5]. [Macam – Macam Access Modifier di Java - Josi Kie](#)
- [6]. [Penjelasan dan Macam-macam Access Modifier \(Akses pengubah\) - Belajar C++](#)
- [7]. [Enkapsulasi \(Pembungkusan data, Modifier, Akses data\) – StayUP](#)
- [8]. [Array Objek di Java](#)