

Tugas Pendahuluan: Modul 7

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 **TP7_NIM_PrakPBO.pdf**. Deadline, Kamis 08 Mei 2025, pukul 13:00 (sebelum praktikum). Link pengumpulan: <https://forms.gle/AWxTPKBmXt2BhuCD9>

Soal

1. Jelaskan dengan kata-kata Anda sendiri apa yang dimaksud dengan konsep inheritance (pewarisan) dalam pemrograman berorientasi objek (OOP) di Java. Mengapa konsep ini penting dalam pengembangan perangkat lunak?

Jawab:

Inheritance adalah konsep pewarisan. Konsep pewarisan yang dimaksud disini adalah jika kita mempunyai suatu kelas A maka seluruh atribut kecuali private atribut dan setiap method yang dimiliki oleh kelas A akan dapat kita salin dan kita modifikasi ke dalam sebuah kelas B. Jadi intinya kita punya sebuah kelas. kemudian kelas ini itu akan kita gunakan lagi codingannya untuk menghasilkan fungsi yang baru tapi lebih spesifik. contohnya kita punya kelas mamalia. Kelas monyet adalah kelas turunan dari kelas mamalia. Karena monyet adalah mamalia hanya saja monyet memiliki deskripsi yang lebih spesifik dari mamalia. Inheritance adalah konsep yang penting karena dengan adanya konsep ini maka kita akan dapat mengembangkan suatu codingan yang efisien karena tidak akan terjadi perulangan baris codingan. Jadi dengan ada konsep ini codingan akan lebih ringkas dan tentu saja akan lebih cepat.

2. Sebutkan dan jelaskan minimal 3 manfaat utama dari penggunaan inheritance dalam Java. Berikan masing-masing analogi nyata yang relevan.

Jawab:

1. Reuseability, dengan menggunakan konsep ini codingan yang sudah diketik sebelumnya dapat digunakan kembali dengan mudah. Hanya perlu untuk dipanggil dan dimodifikasi sesuka hati. Analoginya seperti resep kangkung warisan nenek. Resep ini bisa digunakan sama persis namun ditambahkan sedikit garam karena kita ingin masakannya sedikit lebih asin.
2. Maintenance yang lebih mudah, dengan menggunakan konsep ini maka ketika programmer ingin untuk melakukan pemeliharaan maka pemeliharaan akan dapat dilakukan dengan jauh lebih mudah karena ketika kita melakukan pembaharuan di Superclass/Kelas Parent maka seluruh perubahan tersebut akan otomatis terapkan pada setiap subclass atau child classnya. Hal ini jelas akan sangat mempermudah ketimbang kita tidak menggunakan konsep inheritance dan harus memperbaharui kode satu persatu di setiap bagian fungsi tertentu hanya karena satu perubahan pada codingan untuk fungsi program lainnya. Analoginya itu seperti peraturan

umum dan ketentuan yang ditetapkan oleh perusahaan tentunya harus dipatuhi oleh setiap departemen yang ada di dalam Perusahaan tersebut. Kemudian apapun perubahan yang dilakukan pada peraturan umum di Perusahaan tersebut maka seluruh departemen yang ada di dalam Perusahaan tersebut pasti harus taat dan mengikutinya.

3. Codingan yang lebih terstruktur, Hal ini tentu sangat jelas dengan ada konsep inheritance maka setiap modifikasi dan setiap penambahan atau setiap modifikasi yang dilakukan pada subclass akan lebih terstruktur dan terlihat dengan mudah ketimbang kita menggunakan codingan yang sama dan di copy paste kemudian baru di modifikasi. Tentu kita semua sudah tahulahya mana yang akan lebih melelahkan. Analoginya itu seperti Kita ingin mengkalsifikasikan Anjing ini hewan mamalia. Kita dapat mengetahuinya karena anjing ini mewarisi sifat-sifat dari kelas Mamalia.

3. Apa perbedaan antara superclass dan subclass? Jelaskan juga apa yang dimaksud dengan keyword extends dalam konteks pewarisan.

Jawab:

Superclass itu adalah kelas yang menjadi acuan untuk membuat kelas baru. Jadi kelas inilah yang nanti-nya atributnya dan methodnya akan dimodifikasi untuk menghasilkan suatu spesifikasi baru pada Subclass. Sedangkan Subclass adalah kelas baru yang memanfaatkan informasi dari Superclass untuk mendapatkan spesifikasi fungsi tertentu. Jadi Superclass itu adalah ayahnya sedangkan subclass itu adalah anaknya. Keywords “extends” itu digunakan dalam deklarasi kelas untuk menunjukkan bahwa sebuah kelas itu mewarisi atribut dan method dari kelas lain. Contoh: “class Mobil extends Kendaraan”. Nah kalimat extend ini adalah syntaks untuk membuat kita tahu bahwa kelas mobil itu mewarisi atribut dan method dari kelas kendaraan.

4. Bagaimana peran keyword super dalam pewarisan di Java? Jelaskan dua kegunaan utamanya dan berikan ilustrasi dengan kode sederhana.

Jawab:

```
src > Main.java > Java > Kucing > Kucing(String nama)
Windsurf: Refactor | Explain
1  class Hewan {
2      String nama_super = "antoniooooo";
3
4      // Hewan(String nama) {
5      //     this.nama = nama;
6      // }
7
8      Windsurf: Refactor | Explain | Generate Javadoc | X
9      void suara() {
10         System.out.println(x:"Hewan bersuara");
11     }
12
13     Windsurf: Refactor | Explain
14     class Kucing extends Hewan {
15         String nama;
16         Kucing(String nama) {
17             this.nama = nama; //memberikan nama objek kucing
18             System.out.println("Nama kucing: " + super.nama_super); // Memanggil variabel nama dari s
19         }
20
21         Windsurf: Refactor | Explain | Generate Javadoc | X
22         @Override
23         void suara() {
24             super.suara(); // Memanggil metode suara dari superclass
25             System.out.println(x:"Kucing: Meow Rawrwww!");
26         }
27     }
28
29     Windsurf: Refactor | Explain
30     public class Main {
31         Run | Debug | Run main | Debug main | Windsurf: Refactor | Explain | Generate Javadoc | X
32         public static void main(String[] args) {
33             Kucing kucing = new Kucing(nama:"Ucok");
34             kucing.suara();
35         }
36     }
```

Nah keyword super ini itu adalah syntaks yang berfungsi untuk mengakses anggota dari kelas superclass seperti atribut, konstruktor, atau method.

Seperti pada contoh kita bisa lihat kalo kita sedang construct sebuah object kucing Bernama ucok. Tapi ternyata yang muncul adalah objek Bernama antoniooo yang berasal dari kelas hewan. ini itu adalah salah satu kegunaan super. yaitu untuk memanggil atribut dari Superclass melalui subclass. Kemudian ada juga kegunaan super yang kedua yaitu untuk memanggil method yang ada di Superclass melalui subclass.

5. Jelaskan mengapa konstruktor tidak diwarisi oleh subclass dalam Java. Lalu bagaimana cara subclass bisa tetap menggunakan konstruktor dari superclass?

Jawab:

Konstruktor bukan anggota kelas seperti atribut atau metode, melainkan metode khusus untuk menginisialisasi objek. Setiap kelas memiliki kebutuhan inisialisasi yang spesifik, sehingga konstruktor superclass tidak otomatis diwarisi oleh subclass. Selain itu, konstruktor terkait erat dengan nama kelas, dan subclass memiliki nama kelas yang berbeda. Oleh karena itu sebuah cara agar subclass dapat memanggil konstruktor dari Superclass adalah menggunakan keyword `super` di dalam konstruktornya. Hal ini akan menginisialisasi atribut yang diwarisi dari superclass. Namun dalam hal ini, subclass tidak sedang membuat objek subclass melainkan sedang membuat objek dari superclass.

Referensi

- [1]. [Inheritance – School of Information Systems](#)
- [2]. [Kata kunci dari inheritance adalah manfaat warisan yang memudahkan hidup jutaan programmer! - Tambah Pinter](#)
- [3]. [Belajar Java OOP: Apa fungsi 'this' dan 'super' pada Java?](#)
- [4]. Dst..