

Modul 3

Polymorphysm dan Overriding

Tujuan :

Memahami konsep polymorphism dan dapat menerapkannya dalam program

Materi :

Polymorphism, Overriding atribut & method

Referensi :

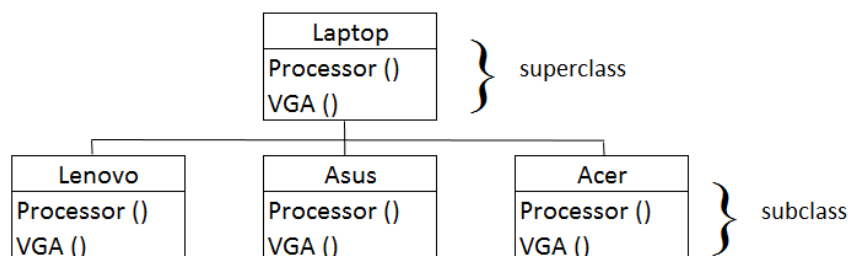
- Fatchurrahman, M. Kom. 2008. Pemrograman Berorientasi Objek dengan Bahasa Java. Malang : Uin-Press
- Modul Praktikum Pemrograman Berorientasi Objek Universitas Islam Negeri Maulana Malik Ibrahim Malang

Polymorphysm

Polymorphysm adalah salah satu karakteristik dalam pemrograman berorientasi objek dimana sebuah method dapat memiliki perilaku (behavior) yang berbeda bergantung jenis objeknya. Polymorphysm juga dapat dikatakan sebagai konsep sederhana dalam bahasa pemrograman berorientasi objek yang berarti kemampuan dari suatu variabel referensi objek untuk memiliki aksi berbeda bila method yang sama dipanggil, dimana aksi method tergantung dari tipe objeknya. Kondisi yang harus dipenuhi supaya Polymorphysm dapat diimplementasikan adalah :

- Method yang dipanggil harus melalui variabel dari superclass.
- Method yang dipanggil harus juga menjadi method dari superclass.
- Method access attribute pada subclass tidak boleh lebih terbatas dari superclass

Perhatikan analogi contoh berikut



Class Laptop yang merupakan class induk (superclass) dari class Lenovo, Asus dan Acer mempunyai method processor() dan VGA(). Class-class anak (subclass) juga mempunyai method processor() dan VGA(). Meskipun keempat class tersebut mempunyai nama method yang sama,

tetapi isi (source code/yang dilakukan/output) dari masing-masing method tersebut **berbeda**. Jika kita menginginkan sebuah objek yang dapat memanggil setiap method (yaitu method processor & VGA) yang ada pada setiap class (pada superclass maupun subclass), maka digunakan teknik Polymorphysm. Polymorphysm hanya berlaku pada method dan tidak berlaku untuk atribut.

Untuk mendapatkan operasi Polymorphysm dari suatu method, maka method tersebut haruslah merupakan method yang ada di class induk (lihat diagram diatas bahwa method processor() dan VGA(), selain terdapat di class-class turunan class Laptop, juga terdapat di class Laptop).

```
class Laptop
{
    public void Processor() {
        System.out.println("Jenis Processor");
    }
    public void VGA() {
        System.out.println("Jenis VGA");
    }
}

class Lenovo extends Laptop
{
    public void Processor(){
        System.out.println("core i7");
    }
    public void VGA() {
        System.out.println("NVIDIA GEFORCE");
    }
}

class Asus extends Laptop
{
    public void Processor() {
        System.out.println("core i5");
    }
    public void VGA() {
        System.out.println("Radeon");
    }
}

class Acer extends Laptop
{
    public void Processor() {
        System.out.println("core i3");
    }
    public void VGA() {
        System.out.println("Intel HD Graphics");
    }
}
```

Overriding Method

Overriding method adalah kemampuan dari subclass untuk memodifikasi method dari superclassnya, yaitu dengan cara menumpuk (mendefinisikan kembali) method superclass-nya. Jadi apabila class anak memiliki method yang sama dengan class induk, maka pemanggilan method pada class anak akan menyebabkan nilai-nilai yang ada pada method induk tertimpa oleh nilai-nilai pada method anak tersebut

Latihan

Dalam suatu survey klasifikasi berat badan ideal seseorang dapat ditentukan melalui tingkatan umur dan tinggi berat badan seseorang tersebut. Sehingga diharapkan untuk mengetahui suatu berat badan ideal seseorang cukup dengan menginputkan berapa umur orang tersebut dan tinggi badan orang tersebut. Setelah diketahui berapa berat badan orang tersebut, terdapat anjuran untuk mengkonsumsi beberapa makanan dengan kadar tertentu. Untuk itu buatlah sebuah aplikasi sebagai implementasi dari kasus tersebut dengan ketentuan sebagai berikut :

- Terdapat tiga kelas utama yakni : manusia sebagai kelas induk, lakilaki dan perempuan sebagai kelas anak(turunann). Buat juga satu kelas tambahan untuk memanggil aplikasi tersebut.
- Pada kelas manusia, terdapat :
 - Tinggibadan : sebagai tinggi badan seseorang (cm)
 - BB : berat badan ideal seseorang (kg)
 - Kadar : kadar makanan yang seharusnya dikonsumsi (satuan berat)
 - Makananideal : jenis makanan yang seharusnya dikonsumsi
 - Tinggi() : tinggi orang
 - HitungBB() : menghitung berat badan
 - Kadmakanan() : kadar dari makanan
 - Makananideal() : jenis makanan
- Pada kelas turunan/anak terdapat method-method yang ada pada kelas induk
- Ketentuan perhitungan :
 - Laki-laki : Berat badan ideal=TinggiBadan dikurangkan 100 kemudian dikalikan 90%
 - Perempuan : Berat badan ideal =TinggiBadan dikurangkan 100 kemudian dikalikan 80%
- Jika berat badan yang dihasilkan < 45 kg, seseorang tersebut harus mengkonsumsi karbohidrat sebanyak 15% dari berat badan ideal yang dihasilkan
- Jika berat badan yang dihasilkan ≥ 45 kg dan ≤ 55 kg, seseorang tersebut harus mengkonsumsi protein sebanyak 10% dari berat badan ideal yang dihasilkan

- Jika berat badan yang dihasilkan > 55 kg, seseorang tersebut harus mengkonsumsi vitamin sebanyak 5% dari berat badan ideal yang dihasilkan
- User menginputkan nama, usia, tinggi badan (JOptionPane)
- Output berupa standart console
- Kerjakan tugas ini dengan menerapkan konsep Polymorphysm dan overriding