

Nama : Stevanus Andika Galih Setiawan .

Kelas : RK231 .

NIM : 20230310008 .

Link github :

Tugas Review Materi 1-6

Soal : <https://stevanusandika.github.io/OOP/>

Latihan Soal dan Studi Kasus

1. Apa perbedaan utama antara enkapsulasi dan abstraksi?
2. Bagaimana perbedaan antara pewarisan (inheritance) dan polimorfisme?
3. Mengapa kita membutuhkan interface jika sudah memiliki class abstrak?

Soal Implementasi Kode :

1. Buatlah class Manusia yang memiliki atribut nama dan method perkenalan(). Lalu buat class Mahasiswa yang mewarisi Manusia dan menambahkan atribut nim.
2. Buatlah sebuah interface bernama Kendaraan yang memiliki method jalan(). Lalu buatlah dua class Mobil dan Sepeda yang mengimplementasikan method tersebut dengan cara berbeda

Jawaban :

1. Perbedaan utama antara enkapsulasi dan abstraksi

Enkapsulasi	Abstraksi
Menyembunyikan data internal dan hanya menampilkan bagian yang diperlukan melalui metode (getter/setter).	Menyembunyikan kompleksitas dan hanya menunjukkan fungsionalitas penting.
Perlindungan data dari akses luar yang tidak sah.	Penyederhanaan dan penyembunyian detail implementasi.
Menggunakan access modifier (private, public, protected).	Menggunakan class abstrak atau interface.
Bertujuan untuk Keamanan data.	Bertujuan untuk Kemudahan penggunaan dan pemahaman program.

2. Perbedaan antara pewarisan (inheritance) dan polimorfisme:

Pewarisan (Inheritance)	Polimorfisme (Polymorphism)
Kemampuan sebuah class untuk mewarisi properti dan method dari class lain.	Kemampuan objek untuk mengambil banyak bentuk (perilaku berbeda walaupun punya antarmuka yang sama).
Reusabilitas kode dan hirarki class.	Fleksibilitas dan dinamis dalam pemanggilan method.

3. Alasan mengapa interface dan abstrak berbeda :
- Interface digunakan untuk membuat kontrak murni — hanya berisi nama method, tanpa isi. Cocok kalau kita ingin class lain wajib punya method tertentu.
 - Class abstrak bisa mempunyai isi method dan atribut, cocok kalau programmer ingin membuat class dasar yang bisa diwarisi dan digunakan ulang.
 - Interface bisa diimplementasikan banyak sekaligus, tapi class abstrak hanya bisa diwarisi satu.

4. Link Github : <https://stevanusandika.github.io/OOP/pertemuan%207/>
Link git repo :
<https://github.com/StevanusAndika/OOP/tree/main/pertemuan%207>

5. Link Github :
<https://stevanusandika.github.io/OOP/pertemuan%207/Kendaraan.html>

Link git repo :
<https://github.com/StevanusAndika/OOP/blob/main/pertemuan%207/Kendaraan.html>