



unindra
universitas indraprasta PGRI

Konsep Dasar Pemrograman Berorientasi Objek / PBO

(Object Oriented Programming / OOP)

PERTEMUAN 1

OOP



Object

Inheritance

Encapsulation

Abstraction

Polymorphism

Attribute

Method

Class

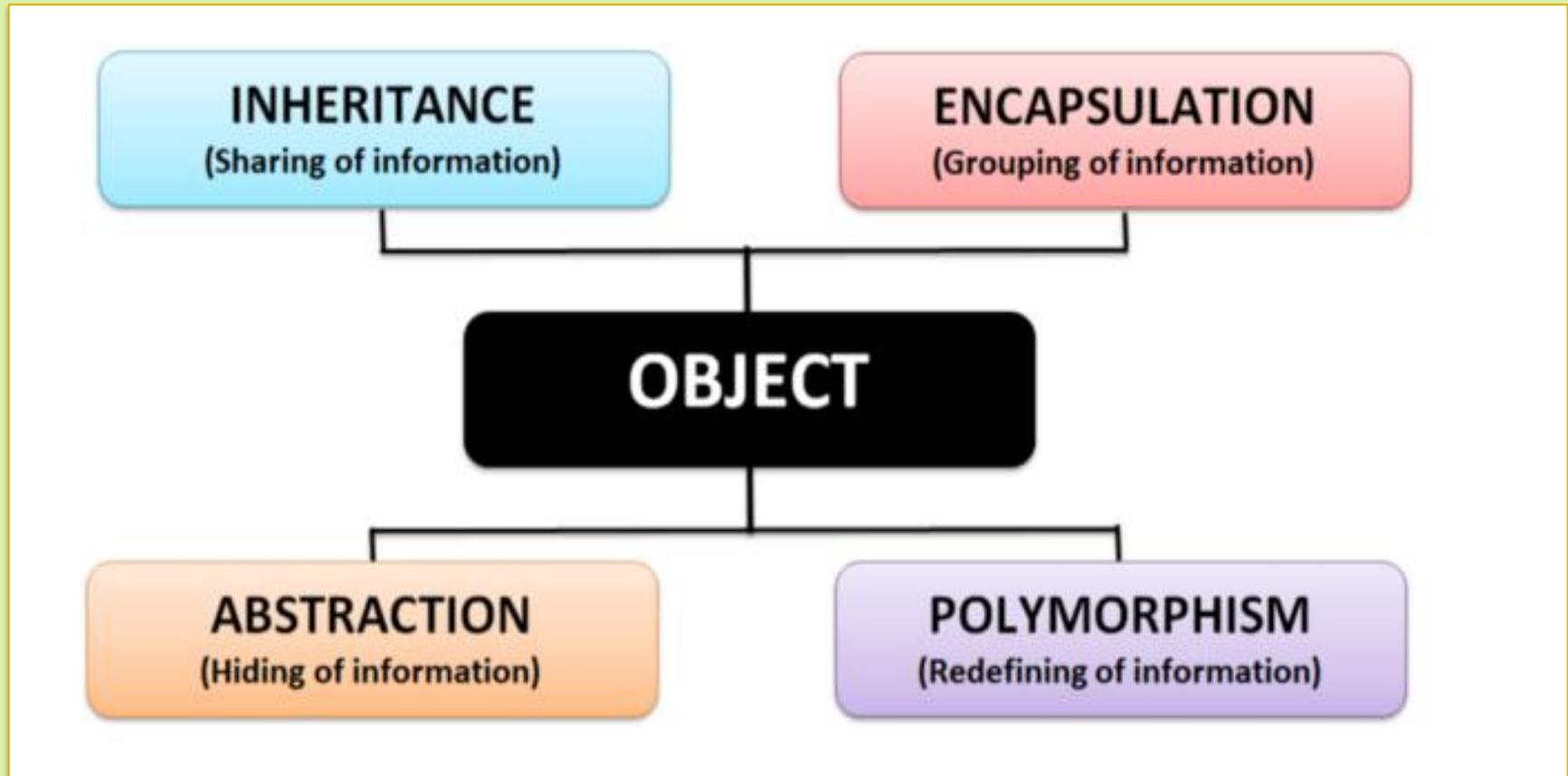
Pengertian Pemrograman Berorientasi Objek

Pemrograman berorientasi objek merupakan paradigma (cara berpikir) pemrograman berdasarkan konsep "objek", yang dapat berisi data, dalam bentuk field atau dikenal juga sebagai atribut.

Serta kode, dalam bentuk fungsi/prosedur atau dikenal juga sebagai method.

Sehingga saling berkaitan dan disusun kedalam satu kelompok atau yang disebut dengan *class*.

4 Pilar dalam Pemrograman Berorientasi Objek



Tujuan Dasar Pemrograman Berorientasi Objek

1. Mengetahui dan menggunakan konsep dasar beorientasi object.
 - Class
 - Object
 - Atribut
 - Method
 - Konstruktor

2. Mengetahui dengan jelas tentang konsep lanjutan berorientasi object dan menggunakannya dengan baik

- Package
- Pewarisan
- Enkapsulasi
- Polimorfisme
- Interface
- Abstraksi

3. Mengetahui dengan jelas penggunaan kata kunci

- *This*
- *Super*
- *Final*
- *Static*

4. Membedakan antara method *overloading* dan method *overriding*

Kenapa Pemrograman Berorientasi Objek

- Memudahkan dalam menganalisa program sesuai dengan kebutuhan, fungsi, apalagi kalau project besar
- Kode-kode yang kita buat menjadi lebih rapi dan terstruktur
- Konsepnya per modul/per class artinya yang dibaca oleh sistem sesuai fungsi yang kita panggil, terutama jika ada error tidak merubah dari awal program
- Konsepnya dapat di lakukan secara tim / team work jadi lebih efisien dalam waktu

Kelebihan dan Kelemahan paradigma Pemrograman Berorientasi Objek:

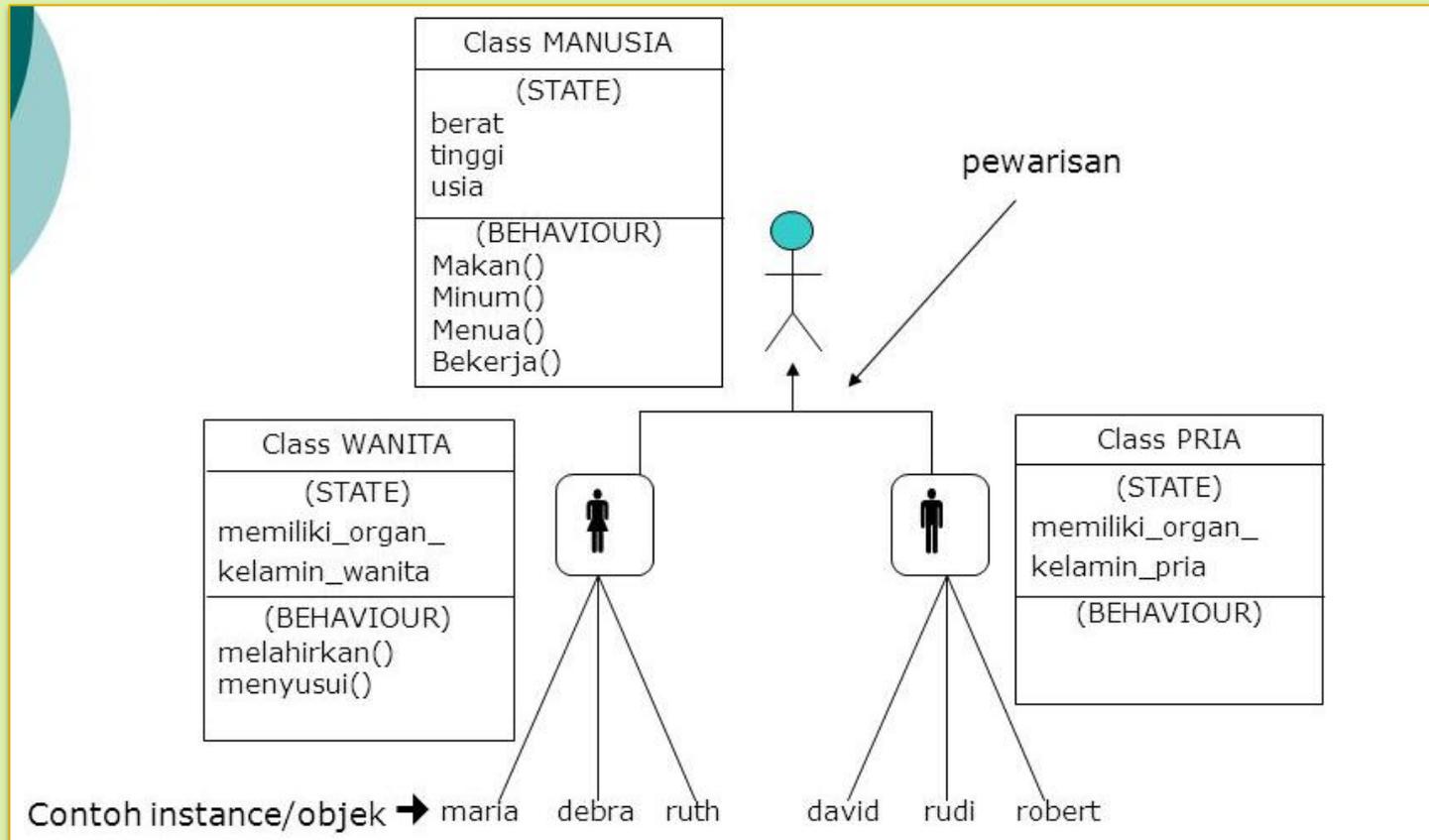
Kelebihan:

- Multiplatform adalah sistem operasi yang dapat di jalankan secara bersamaan. Dapat di jalankan di segala program,
- Dalam mendefinisikan sebuah class cukup sekali tidak perlu ditulis berulang kali namun dapat di turunkan,
- Dapat menambahkan fitur pada class tanpa harus mengedit class asal, Metode ini didukung oleh banyak *library* objek, sehingga mempercepat penyelesaian program
- Data dapat diatur secara private sehingga tidak berpengaruh pada fungsi lainnya,
- Serta dapat membangun *library-library* yang dapat memudahkan programmer dalam membangun sebuah sistem informasi.

Kelemahan :

- Membutuhkan ruang memori yang lebih besar dibandingkan dengan pemrograman terstruktur,
- Runtime pada program berjalan lebih lambat.

Contoh dari Konsep Pemrograman Berorientasi Objek



Latihan

Konsep Pemrograman Berorientasi Objek

1. Berikan simpulan menurut penjelasan dari masing – masing mahasiswa/i apa itu Konsep Pemrograman Berorientasi Objek !
2. Berikan contoh Konsep Pemrograman Berorientasi Objek dari kehidupan sehari – hari !