

Nama : Satia Bili

Nim : 2301010289

Ujian Tengah Semester OOP

Buatlah UML class yang akan dibuat diatas kertas putih/buku tulis dengan tulis tangan terapkan juga prinsip access modifier, constructor, overloading dan overriding pada class tersebut.

Jawab :

Tema : Sistem Kendaraan

1. Class : Kendaraan (super class)

Kendaraan

- merk : String
- tahun : int

+ Kendaraan ()

+ Kendaraan (merk : String,
tahun : int)

+ Info() : Void

2. Class : Mobil (Extends kendaraan)

Mobil

- Jumlah Pintu : int

+ Mobil ()

+ Mobil (merk : String,
tahun : int,

Jumlahpintu : int)
+ Info () : void // override dari kendaraan

Hubungan :

Mobil - -> kendaraan
(inheritance / pewarisan)

Catatan :

- Constructor overloading : kendaraan() dan kendaraan (string, int)
- Method overriding : info() di class Mobil menerima info() di class kendaraan
- Gunakan simbol :
 - + Untuk public
 - - Untuk private
 - # Untuk protected (jika dibutuhkan)

Penjelasan UML class Diagram diatas :

1. Class : kendaraan

- Merupakan superclass atau induk dari semua jenis kendaraan
- Memiliki atribut :
 - merk (merk kendaraan) bertipe string
 - tahun (tahun keluaran) bertipe int
- Memiliki dua constructor :
 - Constructor tanpa parameter (kendaraan())
 - Constructor dengan parameter (kendaraan (merk, tahun)) - ini contoh overloading

- Method `info()` bertipe `void` → memberikan informasi kendaraan.

2. Class : Mobil

- Merupakan subclass dari kendaraan, artinya mewarisi semua atribut dan method dari kendaraan.
- Menambahkan atribut sendiri yaitu jumlah Pintu (Jumlah pintu mobil).
- Memiliki constructor tanpa parameter dan constructor dengan parameter lengkap (termasuk warisan dari superclass).
- Method `info()` disini mang-overwrite method `info()` milik kendaraan → ini disebut overriding.

Simbol Access Modifier :

- `+` (public) : Bisa diakses dari mana saja
- `-` (private) : Hanya bisa diakses dari dalam kelas itu sendiri
- `#` (protected) : Bisa diakses dari kelas itu sendiri dan subclass-nya

Kesimpulan :

UML ini menggambarkan relasi inheritance (pewarisan), penggunaan access modification, constructor overloading, dan method overloading dalam OOP. Diagram ini membantu programmer atau tim pengembang memahami struktur program sebelum menulis code.