

Ujian Tengah Semester Pemrograman Berorientasi Obyek - Buku Terbuka, Internet dan Perangkat Elektronik tertutup

Dosen: Bayu Adhi Nugroho Oktober 2023

1. Jelaskan perbedaan mendasar antara Interface dan Class Abstract !
2. Jelaskan perbedaan mendasar antara override dan overload!
3. Jelaskan perbedaan mendasar antara keyword public, protected dan private!
4. Jelaskan arti keyword static!
5. Jelaskan perbedaan antara Double dan double dalam bahasa Java!
6. Buatlah CRUD dengan Java Swing lengkap dengan laporannya untuk entitas Buku dengan atribut: ISBN, Judul Buku, Tahun Terbit, Penerbit!

JAWABAN

1. **Abstract class** adalah sebuah class setengah jadi (abstrak) yang memuat/memiliki method dan atribut. Abstract class sebenarnya adalah sebuah class, sehingga memiliki semua sifat dari class biasa (punya konstruktor). Hanya saja sifatnya masih abstrak, karena itu biasanya method kosong/belum di implementasikan. Namun, Abstract class dapat mengimplementasikan method tersebut. Abstract class akan selalu menjadi superclass / hirarki tertinggi dari subclass-subclass-nya. Jadi, abstract class merupakan class yang tidak dibuat instan sebelum diturunkan.

Contoh:

```
public abstract class Janin () {
```

Janin b = new Janin [ini salah], yang benar adalah menggunakan extends.

Interface merupakan sebuah definisi method yang umum yang dapat menghubungkan class-class yang berbeda. Dengan kata lain, interface memungkinkan kita mengimplementasikan method yang sama terhadap class yang tidak ada hubungan sama sekali (tidak dalam satu hirarki). Oleh karena itu interface bukanlah sebuah class, walaupun memiliki ciri yang serupa dengan abstract class.

Contoh: membuat hewan > ular > menggigit > kering (tidak mengeluarkan bisa) / basah (mengeluarkan bisa)

2. **Override:** Override merujuk pada penggunaan kembali (menggantikan) metode yang sudah didefinisikan dalam kelas induk (kelas basis) dalam kelas turunannya. Ini digunakan untuk mengganti implementasi metode yang sudah ada dalam kelas dasar. Nah, biasanya override terjadi dalam hubungan warisan antara kelas dasar (superclass) dan kelas turunan (subclass).

Contoh: dalam kelas dasar ada metode `hitungLuas()` yang di override dalam kelas turunan dengan implementasi yang berbeda untuk menghitung luas berdasarkan jenis bentuk yang berbeda (misalnya, persegi dan lingkaran).

Overload: Overload merujuk pada definisi ulang metode dengan nama yang sama dalam sebuah kelas, tetapi dengan parameter yang berbeda. Ini digunakan untuk memiliki beberapa versi metode dengan nama yang sama tetapi berbeda dalam jumlah atau jenis parameter. Biasanya overload terjadi dalam satu kelas yang sama, dimana metode dengan nama yang sama didefinisikan dengan parameter yang berbeda.

Contoh: dalam kelas yang sama, ada beberapa metode dengan nama jumlah (`int a, int b`) yang menerima parameter berbeda, seperti `jumlah (int a)` dan `jumlah(int a, int b, int c)`.

3. **Public:** dapat diakses/dipanggil dari mana saja, baik dari dalam kelas itu sendiri, atau kelas lain dalam paket yang sama. Namun, jika terletak pada package yang berbeda, maka wajib menggunakan import agar bisa di run.

Protected: hanya dapat diakses dari kelas itu sendiri dan juga dari kelas turunan (subclass) yang ada dalam paket yang sama atau di paket yang berbeda.

Private: hanya dapat diakses dari dalam kelas itu sendiri. Mereka tidak dapat diakses dari kelas lain, bahkan kelas turunan sekalipun.

4. Static mengacu pada variabel dan method yang terkait dengan kelas itu sendiri, bukan dengan instance (objek) dari kelas tersebut. Jadi, nilai variabel statis akan bersifat bersama di seluruh objek yang dibuat dari kelas tersebut. Sedangkan, Metode statis tidak dapat mengakses atau memodifikasi variabel atau metode non-statis dalam kelas. Nah jika berbicara dalam bahasa pemrograman Java, metode statis khusus yang disebut "main" digunakan sebagai titik masuk dalam program. Program dimulai dengan menjalankan metode main yang didefinisikan sebagai `public static void main (String[] args)` dalam kelas.

5. **double** merupakan tipe data primitif, sehingga variabel dengan tipe data ini tidak bisa menjadi objek dan tidak memiliki metode atau operasi khusus seperti tipe data yang berupa kelas. Sedangkan, **Double** adalah reserved keyword yang dalam bahasa Java digunakan untuk membuat objek yang mengenkapsulasi nilai double. Ini adalah bagian dari tipe data referensi (non-primitif) dan berperilaku sebagai objek.

6.



