

# **Object Oriented Programming**

## **Quiz 1**



BY :

**D4 INFORMATICS ENGINEERING (1I)**  
**Shofa Yasmin Fauziah (21)**

**State Polytechnic of Malang**  
**Soekarno Hatta street No.9, Malang, East Java**  
**65141 2022/2023**

# Quiz 1

## 1. Class dan Object:

- Apa yang dimaksud dengan "class" dalam pemrograman berorientasi objek?

Class adalah cetak biru atau blueprint dari object. Class digunakan hanya untuk membuat kerangka dasar. Yang akan kita pakai nantinya adalah hasil cetakan dari class, yakni object.

- Bagaimana Anda mendefinisikan objek dari suatu class dalam bahasa pemrograman Java?

Pertama, mendefinisikan class

```
<modifier> class <nama_class> {  
    //deklarasi atribut dan method  
}
```

Kedua, mendefinisikan object

```
NamaClass namaObject = new NamaClass();
```

- Misalkan Anda memiliki class "Barang" dalam sistem informasi inventaris. Bagaimana Anda akan membuat objek "laptop" dari class tersebut?

```
Barang laptop = new Barang( nama: "Laptop Asus",  
                             idBrg: "001",  
                             harga: 6000000 );
```

## 2. Encapsulation:

- Jelaskan konsep encapsulation dalam pemrograman berorientasi objek dan mengapa hal ini penting dalam pengembangan sistem informasi inventaris barang.

Konsep "encapsulation" adalah salah satu prinsip utama dalam pemrograman berorientasi objek (OOP). Ini mengacu pada pengemasan data bersama dengan metode yang mengoperasikannya dalam satu unit tunggal yang disebut "class." Memungkinkan untuk menyembunyikan implementasi dari luar dunia, dan hanya mengizinkan akses terbatas ke data dan operasi yang relevan melalui metode-metode yang ditentukan.

Alasan mengapa encapsulation sangat penting:

- Dapat membatasi akses langsung ke data
- Mengubah cara data atau operasi tertentu tanpa mempengaruhi bagian lain
- Berinteraksi dengan objek tanpa perlu tahu tentang detail implementasinya

Contoh modifier akses seperti private, protected, dan public untuk mengendalikan tingkat akses ke atribut dan metode dalam class

- Dalam konteks sistem informasi inventaris, sebutkan contoh atribut (variabel) yang harus di-encapsulate dan mengapa.

- Harga barang: memastikan bahwa harga selalu positif
- Stok: memastikan bahwa nilai stok tidak pernah negatif atau melampaui batas maksimum
- Nomor Seri: memastikan bahwa nomor seri selalu unik dan valid.

### 3. Relasi Kelas:

- Apa yang dimaksud dengan relasi antara kelas dalam pemrograman berorientasi objek?

Relasi antara kelas merupakan hubungan atau interaksi antar kelas yang memiliki berbagai jenis, seperti inheritance, asosiasi, komposisi, dan agregasi.

- Dalam sistem informasi inventaris barang, bagaimana Anda akan menggambarkan relasi antara kelas "Barang" dan kelas "Kategori"?

Menggunakan has-a atau aggregation dengan menggunakan atribut 'kategori' dalam kelas "Barang" untuk menggambarkan hubungan asosiasi dengan kelas "Kategori."

Kelas "Kategori" memiliki atribut "namaKategori" untuk menyimpan nama kategori.

Kelas "Barang" memiliki atribut "nama" untuk menyimpan nama barang dan atribut "kategori" yang merupakan objek dari kelas "Kategori."

### 4. PBL:

- Berdasarkan kasus sistem informasi inventaris barang, coba buat sebuah class sederhana beserta atribut dan metodenya yang menggambarkan suatu entitas dalam sistem tersebut (misalnya, class "Barang").

#### Attributes

```
1 package QuizNo4PBL;
2 public class Barang {
3     private String kodeBarang;
4     private String namaBarang;
```

#### Method

```

6  public Barang(String kodeBarang, String namaBarang) {
7      this.kodeBarang = kodeBarang;
8      this.namaBarang = namaBarang;
9  }
10
11 public String getKodeBarang() {
12     return kodeBarang;
13 }
14
15 public String getNamaBarang() {
16     return namaBarang;
17 }
18
19 public void setKodeBarang(String kodeBarang) {
20     this.kodeBarang = kodeBarang;
21 }
22
23 public void setNamaBarang(String namaBarang) {
24     this.namaBarang = namaBarang;
25 }
26
27 public void tampilkanInfo() {
28     System.out.println("Kode Barang: " + kodeBarang);
29     System.out.println("Nama Barang: " + namaBarang);
30 }
31 }

```

- Bagaimana Anda akan menggunakan encapsulation untuk melindungi atribut-atribut dalam class tersebut?

```

10
11 //getter
12 public String getKodeBarang() {
13     return kodeBarang;
14 }
15
16 public String getNamaBarang() {
17     return namaBarang;
18 }
19
20 //setter
21 public void setKodeBarang(String kodeBarang) {
22     this.kodeBarang = kodeBarang;
23 }
24
25 public void setNamaBarang(String namaBarang) {
26     this.namaBarang = namaBarang;
27 }
28

```

- Gambarkan hierarki class atau hubungan antar class yang mungkin ada dalam

sistem informasi inventaris barang di jurusan Teknologi Informasi. Berikan contoh relasi antar class (misalnya, inheritance atau association) dalam konteks tersebut.

Class "Barang"

Atribut: kodeBarang, namaBarang, harga, stok

Metode: getKodeBarang(), getNamaBarang(), getHarga(), getStok(),  
tambahStok(), kurangiStok(), tampilkanInfo()

Class "Kategori"

Atribut: namaKategori

Metode: getNamaKategori()

Relasi kelas: Aggregation (Agregasi)

Class "Barang" memiliki hubungan Aggregasi dengan Class "Kategori." Setiap barang termasuk dalam satu kategori tertentu.