

# Object Oriented Programming

## Quiz 1



Name: Azahra Salsabila

NIM: 2241720073

Class: 2I

No: 05

Major: Information Technology

Study Program: Informatic Engineering

### 1. Class dan Object:

- Apa yang dimaksud dengan "class" dalam pemrograman berorientasi objek?
  - "Class" dalam pemrograman berorientasi objek adalah blueprint atau template untuk menciptakan objek. Ini mendefinisikan atribut (variabel) dan metode (fungsi) yang akan dimiliki oleh objek dari class tersebut.
- Bagaimana Anda mendefinisikan objek dari suatu class dalam bahasa pemrograman Java?
  - Untuk membuat objek dari suatu class dalam bahasa pemrograman Java, harus melakukan langkah berikut:  
`NamaClass objek = new NamaClass();`
- Misalkan Anda memiliki class "Barang" dalam sistem informasi inventaris. Bagaimana Anda akan membuat objek "laptop" dari class tersebut?
  - `Barang laptop = new Barang();`

### 2. Encapsulation:

- Jelaskan konsep encapsulation dalam pemrograman berorientasi objek dan mengapa hal ini penting dalam pengembangan sistem informasi inventaris barang.
  - Konsep encapsulation adalah mekanisme di mana atribut dan metode yang berkaitan dengan suatu objek dikemas bersama sebagai satu unit yang disebut class. Akses ke atribut class tersebut biasanya diatur dengan hak akses (public, private, protected) dan metode get dan set.
- Dalam konteks sistem informasi inventaris, sebutkan contoh atribut (variabel) yang harus di-encapsulate dan mengapa.
  - Penting dalam sistem informasi inventaris barang karena melindungi data dari modifikasi yang tidak sah. Contoh atribut yang harus di-encapsulate dalam kelas "Barang" bisa termasuk harga, jumlah, tanggal pembelian, dll. Encapsulation digunakan untuk mengontrol bagaimana data ini dapat diakses dan dimodifikasi, serta memberikan validasi dan keamanan.

### 3. Relasi Kelas:

- Apa yang dimaksud dengan relasi antara kelas dalam pemrograman berorientasi objek?
  - Relasi antara kelas dalam pemrograman berorientasi objek menggambarkan bagaimana kelas satu berinteraksi dengan kelas lainnya. Ini mencakup inheritance (pewarisan), association (asosiasi), aggregation (agregasi), dan lainnya.
- Dalam sistem informasi inventaris barang, bagaimana Anda akan menggambarkan relasi antara kelas "Barang" dan kelas "Kategori"?
  - Dalam sistem informasi inventaris barang, relasi antara kelas "Barang" dan "Kategori" bisa dijelaskan dengan association, di mana "Barang" memiliki relasi dengan "Kategori" karena setiap barang terkait dengan suatu kategori tertentu. Contoh: suatu objek barang memiliki atribut yang merujuk ke objek kategori yang menjelaskan jenis barang tersebut.

#### 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").

```
1 package semester3.quiz1;
2 public class Barang {
3     private String nama;
4     private double harga;
5     private int jumlah;
6     private String tanggalPembelian;
7
8     public Barang(String nama, double harga, int jumlah, String tanggalPembelian) {
9         this.nama = nama;
10        this.harga = harga;
11        this.jumlah = jumlah;
12        this.tanggalPembelian = tanggalPembelian;
13    }
14
15    public void tampilInfo() {
16        System.out.println("Nama: " + nama);
17        System.out.println("Harga: " + harga);
18        System.out.println("Jumlah: " + jumlah);
19        System.out.println("Tanggal Pembelian: " + tanggalPembelian);
20    }
21 }
```

- Bagaimana Anda akan menggunakan encapsulation untuk melindungi atribut atribut dalam class tersebut?
  - Untuk menggunakan encapsulation dalam class "Barang," atributnya telah di-deklarasikan sebagai private. Ini berarti harus menggunakan metode getter (get) dan setter (set) untuk mengakses atau memodifikasi atribut tersebut.

```
1 package semester3.quiz1;
2 public class Barang {
3     private String nama;
4     private double harga;
5     private int jumlah;
6     private String tanggalPembelian;
7
8     public Barang(String nama, double harga, int jumlah, String tanggalPembelian) {
9         this.nama = nama;
10        this.harga = harga;
11        this.jumlah = jumlah;
12        this.tanggalPembelian = tanggalPembelian;
13    }
14
15    public String getNama() {
16        return nama;
17    }
18
19    public void setNama(String nama) {
20        this.nama = nama;
21    }
22
23    public double getHarga() {
24        return harga;
25    }
26
27    public void setHarga(double harga) {
28        this.harga = harga;
29    }
30
31    public int getJumlah() {
32        return jumlah;
33    }
34
35    public void setJumlah(int jumlah) {
36        this.jumlah = jumlah;
37    }
38
39    public String getTanggalPembelian() {
40        return tanggalPembelian;
41    }
42
43    public void setTanggalPembelian(String tanggalPembelian) {
44        this.tanggalPembelian = tanggalPembelian;
45    }
46
47    public void tampilInfo() {
48        System.out.println("Nama: " + nama);
49        System.out.println("Harga: " + harga);
50        System.out.println("Jumlah: " + jumlah);
51        System.out.println("Tanggal Pembelian: " + tanggalPembelian);
52    }
53 }
54 }
```

- 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
  - Hierarki class dan hubungan antar class yang mungkin ada dalam sistem informasi inventaris barang di jurusan Teknologi Informasi dapat terdiri dari beberapa class, seperti "Barang," "Kategori," "Supplier," dan lainnya. Contoh relasi antar class dalam konteks ini adalah:
    - Class "Barang" dapat memiliki class "Kategori" sebagai atribut, yang menunjukkan kategori barangnya.
    - Class "Barang" juga dapat memiliki class "Supplier" sebagai atribut, yang menunjukkan supplier barang tersebut.
    - Class "Kategori" dapat memiliki beberapa class "Barang" sebagai atribut, menunjukkan bahwa banyak barang termasuk dalam kategori yang sama.
    - Class "Supplier" dapat memiliki beberapa class "Barang" sebagai atribut, menunjukkan bahwa supplier menyediakan berbagai barang.
    - Selain itu, class "Barang" mungkin memiliki subclass seperti "Elektronik," "Peralatan," dsb., jika atribut dan metodenya berlaku secara umum untuk semua jenis barang.