



Jurusan Teknologi Informasi Politeknik Negeri Malang

## Week 5 : Quiz

Mata Kuliah **Object Oriented Programming**

Pengampu: Tim Ajar Object Oriented Programming

November 2023

---

<b>Nama</b>	:	MAULIA BALQIS ANSYA AULIA
<b>Nim</b>	:	2241720246
<b>Kelas</b>	:	TI – 2I

### SOAL

#### 1. Class dan Object:

- Apa yang dimaksud dengan "class" dalam pemrograman berorientasi objek?
- Bagaimana Anda mendefinisikan objek dari suatu class dalam Bahasa pemrograman Java?
- Misalkan Anda memiliki class "Barang" dalam sistem informasi inventaris. Bagaimana Anda akan membuat objek "laptop" dari class tersebut?

#### 2. Encapsulation:

- Jelaskan konsep encapsulation dalam pemrograman berorientasi objek dan mengapa hal ini penting dalam pengembangan sistem informasi inventaris barang.
- Dalam konteks sistem informasi inventaris, sebutkan contoh atribut (variabel) yang harus di-encapsulate dan mengapa.

#### 3. Relasi Kelas:

- Apa yang dimaksud dengan relasi antara kelas dalam pemrograman berorientasi objek?
- Dalam sistem informasi inventaris barang, bagaimana Anda akan menggambarkan relasi antara kelas "Barang" dan 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").
- Bagaimana Anda akan menggunakan encapsulation untuk melindungi atribut-atribut dalam class tersebut?
- 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.



Jurusan Teknologi Informasi Politeknik Negeri Malang

## Week 5 : Quiz

Mata Kuliah **Object Oriented Programming**

Pengampu: Tim Ajar Object Oriented Programming

November 2023

Soal No	Jawaban
1	<ul style="list-style-type: none"><li>• <b>Pengertian Class</b><ul style="list-style-type: none"><li>○ Dalam pemrograman berorientasi objek, class merupakan sebuah cetak biru atau template yang digunakan untuk membuat objek. Class menentukan atribut dan metode yang dimiliki oleh objek.</li></ul></li><li>• <b>Definisi Objek dari Class</b><ul style="list-style-type: none"><li>○ Objek adalah sebuah instance dari class. Objek memiliki atribut dan metode yang ditentukan oleh classnya.</li></ul></li><li>• <b>Contoh</b><pre>Barang laptop = new Barang("Laptop", "Elektronik");</pre><ul style="list-style-type: none"><li>○ Kode tersebut akan membuat objek "laptop" dengan atribut dan metode yang ditentukan oleh class "Barang".</li></ul></li></ul>
2	<ul style="list-style-type: none"><li>• <b>Konsep Encapsulation dalam OOP:</b><ul style="list-style-type: none"><li>○ Encapsulation melibatkan menyembunyikan detail implementasi internal suatu objek dan hanya mengekspos apa yang diperlukan. Ini dilakukan dengan membuat atribut private dan menyediakan metode public untuk mengakses atau memodifikasi atribut tersebut. Ini penting untuk meningkatkan keamanan dan memudahkan pemeliharaan kode.</li></ul></li><li>• <b>Contoh Atribut yang Harus Di-Encapsulate</b><ul style="list-style-type: none"><li>○ Dalam konteks sistem informasi inventaris, contoh atribut yang harus di-encapsulate adalah:<ul style="list-style-type: none"><li>- jumlah: Jumlah barang yang tersedia</li><li>- harga: Harga barang</li><li>- status: Status barang (aktif, tidak aktif, dll.)</li></ul></li></ul></li></ul>



Jurusan Teknologi Informasi Politeknik Negeri Malang

**Week 5 : Quiz**

Mata Kuliah **Object Oriented Programming**

Pengampu: Tim Ajar Object Oriented Programming

November 2023

---

	<ul style="list-style-type: none"><li>○ Atribut-atribut tersebut harus dilindungi dari akses yang tidak diinginkan untuk menjaga keamanan dan keakuratan data inventaris.</li></ul>
3	<ul style="list-style-type: none"><li>• <b>Pengertian Relasi Kelas</b><ul style="list-style-type: none"><li>○ Relasi kelas adalah hubungan antara dua atau lebih kelas. Relasi kelas dapat digambarkan menggunakan notasi UML.</li><li>○ Relasi antara kelas mencerminkan bagaimana kelas satu berinteraksi dengan kelas lain. Ada beberapa jenis relasi, termasuk inheritance (pewarisan), association (hubungan), dan lainnya.</li></ul></li><li>• <b>Relasi antara Kelas "Barang" dan "Kategori":</b><ul style="list-style-type: none"><li>○ Dalam konteks ini, kelas "Barang" dan "Kategori" dapat memiliki relasi association, di mana satu barang terkait dengan satu kategori atau banyak barang terkait dengan satu kategori. Ini dapat diimplementasikan melalui atribut kategori dalam class "Barang."</li></ul></li></ul>



4

```
public class Barang {  
    private String nama;  
    private int harga;  
    private int stok;  
  
    // Constructor  
    public Barang(String nama, int harga, int stok) {  
        this.nama = nama;  
        this.harga = harga;  
        this.stok = stok;  
    }  
  
    // Metode-metode public  
    public String getNama() {  
        return nama;  
    }  
  
    public int getHarga() {  
        return harga;  
    }  
  
    public int getStok() {  
        return stok;  
    }  
  
    public void tambahStok(int jumlah) {  
        this.stok += jumlah;  
    }  
  
    // Metode lainnya sesuai kebutuhan  
}
```

- **Penggunaan Encapsulation:**

- Dengan membuat atribut private dan menyediakan metode public, atribut dalam class "Barang" di-encapsulate. Contohnya, penggunaan metode **getNama()**, **getHarga()**, dan **getStok()**.



Jurusan Teknologi Informasi Politeknik Negeri Malang

## Week 5 : Quiz

Mata Kuliah **Object Oriented Programming**

Pengampu: Tim Ajar Object Oriented Programming

November 2023

---

- **Hierarki Class dan Relasi antar Class:**

- Hierarki class dapat mencakup class "Elektronik" yang merupakan subclass dari "Barang." Relasi antar class dapat berupa inheritance jika "Elektronik" mewarisi atribut dan metode dari "Barang."

```
public class Elektronik extends Barang {  
    private String merek;  
  
    public Elektronik(String nama, int harga, int stok, String merek) {  
        super(nama, harga, stok);  
        this.merek = merek;  
    }  
  
    // Metode-metode public khusus Elektronik  
}
```