

Nama : Allya Saffira
Nim : 20220040262

Sesi 2

Pemrograman Berbasis Object

1. Jelaskan apa yang dimaksud dengan class dan object!

Class : Class adalah sebuah template atau blueprint yang digunakan untuk mendefinisikan atribut dan perilaku yang akan dimiliki oleh objek yang dibuat dari class tersebut. Class mendefinisikan struktur data dan fungsi yang akan digunakan oleh objek. Class juga dapat berisi variabel data (attributes), metode (methods).

Object : Object adalah instance dari sebuah class. Setiap objek adalah representasi dari class dan dapat menggunakan atribut dan metode yang didefinisikan dalam class tersebut. Objek memiliki state (representasi oleh atribut), behavior (representasi oleh metode), dan identity (memberikan nama unik kepada objek). Objek dibuat menggunakan keyword new dalam bahasa pemrograman seperti Java, dan setiap objek menggunakan memori yang dialokasikan untuk menyimpan state-nya

2. Sebutkan dan jelaskan jenis-jenis method !

- **Metode Instance**: Metode yang terikat pada objek tertentu dan memiliki akses ke data objek tersebut. Metode ini biasanya digunakan untuk melakukan operasi atau manipulasi pada data objek itu sendiri.
- **Metode Kelas (Class Method)**: Metode yang terikat pada kelas (class) dan bukan pada objek. Metode ini biasanya digunakan untuk melakukan operasi atau manipulasi yang terkait dengan kelas itu sendiri, bukan dengan objek individu.
- **Method static**: Method yang didefinisikan di dalam kelas tetapi tidak terkait dengan instance objek tertentu. Method ini dapat dipanggil langsung dari kelas tanpa perlu membuat instance terlebih dahulu.
- **Metode Getter(Accessor)**: Metode yang digunakan untuk mengakses dan mengembalikan nilai dari atribut atau properti objek. Metode ini sering disebut sebagai "getter" karena mereka digunakan untuk mendapatkan nilai.
- **Metode Setter (Mutator)**: Metode yang digunakan untuk mengatur atau mengubah nilai dari atribut atau properti objek.
- **Metode Abstract**: Metode ini dideklarasikan dalam kelas abstrak dan tidak memiliki implementasi di dalam kelas itu sendiri. Metode abstrak dapat dipanggil melalui referensi objek yang merujuk ke kelas abstrak atau kelas turunannya. Metode abstrak sering digunakan dalam konsep pewarisan dan polimorfisme dalam pemrograman berorientasi objek

3. Berdasarkan gambar berikut ini, jelaskan masing-masing bagian sesuai dengan nomor yang ada!

Jawab:

```
public class Komputer {
```

(1) Pada baris satu pada kode tersebut ialah mendeklarasikan kelas Komputer sebagai public, yang berarti kelas ini dapat diakses dari kelas lain dalam aplikasi.

```
String jenis_komputer;  
private String merk;
```

(2) Pada nomor 2 itu ada kode yang mendeklarasikan `jenis_komputer` dan `merk` sebagai variabel ber tipe String, tetapi pada `merk` terdapat penggunaan modifier `private`, yang berarti variabel ini hanya dapat diakses dari dalam kelas `Komputer` itu sendiri.

```
public void setDataKomputer(String jenis, String merk) {  
    this.jenis_komputer = jenis;  
    this.merk = merk;
```

(3) Pada nomor 3 Mendefinisikan metode `setDataKomputer` yang memungkinkan pengaturan nilai untuk `jenis_komputer` dan `merk`. Metode ini menerima dua parameter, `jenis` dan `merk`, yang keduanya bertipe String.

Menggunakan kata kunci `this` untuk merujuk ke variabel instance `jenis_komputer` dan `merk` dari kelas `Komputer`, dan menetapkan nilai dari parameter `jenis` ke variabel tersebut.

```
public String getJenis() {  
    return jenis_komputer;
```

(4) Mendefinisikan metode `getJenis` yang mengembalikan nilai dari variabel instance `jenis_komputer` dan di bawah nya ada `return` yang berfungsi mengembalikan nilai dari variabel `jenis_komputer`.

```
public String getMerk() {  
    return merk;
```

(5) Mendefinisikan metode `getMerk` yang mengembalikan nilai dari variabel instance `merk` dan di bawah nya ada `return` yang berfungsi mengembalikan nilai dari variabel `merk`.

```
Komputer mykom = new Komputer();
```

(6) Membuat objek `mykom` dari kelas `Komputer` menggunakan konstruktor default (yang dianggap ada oleh Java jika tidak ada konstruktor lainnya yang ditentukan).

```
mykom.setDataKomputer("LAPTOP", "MACBOOK");
```

(7) Memanggil metode `setDataKomputer` pada objek `mykom` untuk mengatur `jenis_komputer` menjadi "LAPTOP" dan `merk` menjadi "MACBOOK".

```
System.out.println(mykom.getJenis());  
System.out.println(mykom.getMerk());
```

(8) Mencetak jenis komputer yang telah disetel sebelumnya ke konsol.
Mencetak merk komputer yang telah disetel sebelumnya ke konsol.