

Nama : Selpia Meilani
NIM : 20220040148
Kelas : TI22A
Matkul : Pemograman Berbasis Objek

Tugas Sesi 2

1. Class dan object adalah konsep dasar dalam Pemrograman Berorientasi Objek (PBO) yang digunakan untuk merepresentasikan konsep dan entitas dunia nyata.

❖ Class:

- a) Class adalah blueprint atau prototipe yang digunakan untuk membuat objek.
- b) Class mewakili kelompok objek yang memiliki sifat dan perilaku serupa.
- c) Contohnya, kelas “Hewan” mewakili semua hewan, dan kelas “Anjing” mewakili semua anjing.
- d) Class hanya merupakan template dan tidak mengambil memori.
- e) Class terdiri dari variabel (data member) dan metode (fungsi).
- f) Deklarasi kelas dalam Java memiliki komponen berikut:
 - Kata kunci class: Digunakan untuk membuat kelas.
 - Nama kelas: Dimulai dengan huruf awal (biasanya huruf kapital).
 - Superclass (jika ada): Nama kelas induk, jika ada, diikuti oleh kata kunci extends.
 - Interface (jika ada): Daftar antarmuka yang diimplementasikan oleh kelas, jika ada, diikuti oleh kata kunci implements.
 - Tubuh kelas: Dikelilingi oleh kurung kurawal { }.
- g) Contoh deklarasi kelas dalam Java:

```
public class Mahasiswa {  
    int id;  
    String nama;  
  
    public static void main(String[] args) {  
        Mahasiswa mhs = new Mahasiswa();  
        System.out.println(mhs.id); // Output: 0  
        System.out.println(mhs.nama); // Output: null  
    }  
}
```

❖ Object:

- a) Sebuah "object" adalah instansi konkret dari suatu class.
- b) Objek merupakan representasi nyata dari konsep atau entitas yang didefinisikan oleh class.
- c) Setiap objek yang dibuat dari class memiliki atribut dan metode yang sama, tetapi nilai atribut mungkin berbeda antar objek.
- d) Objek adalah unit dasar dari pemrograman berorientasi objek, dan mereka digunakan untuk merepresentasikan data dan perilaku dalam suatu sistem.

2. Jenis-jenis method dalam pemrograman berorientasi objek dapat dibagi menjadi beberapa kategori:

a) Method Instance:

- Terkait dengan instance atau objek dari suatu class.
- Mereka dapat mengakses dan memanipulasi data yang dimiliki oleh objek tersebut.
- Dipanggil pada objek dan biasanya digunakan untuk berinteraksi dengan atribut objek.
- Contoh: objek.metode_instance().

b) Class Method:

- Terkait dengan class secara keseluruhan, bukan instance tunggal.
- Dapat mengakses dan memanipulasi data yang terkait dengan class, bukan objek spesifik.
- Dideklarasikan menggunakan decorator @classmethod.
- Contoh: Kelas.metode_kelas().

c) Static Method:

- Terkait dengan class, tetapi tidak memiliki akses ke atribut kelas atau objek.
- Tidak memerlukan referensi objek atau class untuk dipanggil.
- Dideklarasikan menggunakan decorator @staticmethod.
- Sering digunakan untuk fungsi yang terkait dengan class tetapi tidak bergantung pada state objek.
- Contoh: Kelas.metode_statis().

d) Getter dan Setter Method:

- Digunakan untuk mengakses (getter) dan mengubah (setter) nilai atribut objek dengan cara yang terkendali.
- Getter mengembalikan nilai atribut, sedangkan setter mengatur nilai atribut.
- Mencegah akses langsung ke atribut untuk menjaga integritas data.
- Contoh: objek.get_atribut() dan objek.set_atribut(nilai).

3. Berikut adalah penjelasan untuk masing-masing bagian sesuai dengan nomor yang ada pada gambar:

A. Deklarasi Kelas (Class):

- Pada bagian ini, kita melihat deklarasi kelas dengan nama “Komputer”.
- Kelas adalah blueprint atau prototipe yang digunakan untuk membuat objek.
- Dalam contoh ini, kelas “Komputer” akan digunakan untuk membuat objek- objek yang merepresentasikan komputer.
- Deklarasi kelas dimulai dengan kata kunci public class diikuti oleh nama kelas (“Komputer” dalam hal ini).
- Setelah deklarasi kelas, kita akan menentukan variabel dan metode yang dimiliki oleh kelas ini.

B. Variabel (Data Member):

- Terdapat dua variabel yang dideklarasikan dalam kelas “Komputer”:
 - jenis_komputer: Variabel ini memiliki tipe data String dan akan digunakan untuk menyimpan informasi tentang jenis komputer (misalnya “LAPTOP”, “DESKTOP”, dll.).
 - merk: Variabel ini juga memiliki tipe data String dan dideklarasikan sebagai private. Ini berarti variabel ini hanya dapat diakses dari dalam kelas “Komputer” dan tidak dapat diakses langsung dari luar kelas.

C. Metode (Method):

- Terdapat dua metode yang didefinisikan dalam kelas “Komputer”:
 - setDataKomputer(String jenis, String merk): Metode ini digunakan untuk mengatur nilai dari variabel jenis_komputer dan merk. Nilai yang diberikan sebagai argumen akan disimpan dalam variabel yang sesuai.
 - getJenis(): Metode ini bertugas mengembalikan nilai dari variabel jenis_komputer.
 - getMerk(): Metode ini juga bertugas mengembalikan nilai dari variabel merk.

D. Objek (Instance):

- Dalam metode main, kita melihat pembuatan objek dari kelas “Komputer” dengan nama mykom.
- Objek ini akan memiliki atribut dan metode yang sama seperti yang telah didefinisikan dalam kelas “Komputer”.
- Nilai atribut objek mykom diatur menggunakan metode setDataKomputer. Sebagai contoh, jenis komputer diatur ke “LAPTOP” dan merk diatur ke “MACBOOK”.
- Nilai jenis dan merk komputer kemudian dicetak ke konsol.

4. Berikut adalah perbaikan pada kode program yang diberikan:

```
public class HandPhone {
    String jenis_hp;
    int tahun_pembuatan;

    public void setDataHP(String jenis, int tahun) {
        jenis_hp = jenis;
        tahun_pembuatan = tahun;
    }

    public String getJenisHP() {
        return jenis_hp;
    }

    public int getTahunPembuatan() {
        return tahun_pembuatan;
    }

    public static void main(String[] args) {
        HandPhone hp = new HandPhone();
        hp.setDataHP("Smartphone", 2022);
        System.out.println("Jenis HP: " + hp.getJenisHP());
        System.out.println("Tahun Pembuatan: " + hp.getTahunPembuatan());
    }
}
```