

Mata Kuliah	Pemrograman Berorientasi Objek	NIM	20230040010
Sesi Pertemuan	2(Dua)	Nama Mahasiswa	Eneng Sapitri
Materi	Tugas Class & Object	Kelas	TI23C
Dosen	Yulhan Wahyudin, M.Kom.		

1. Jelaskan apa yang dimaksud dengan class dan object!
2. Sebutkan dan jelaskan jenis-jenis method!
3. Berdasarkan gambar berikut ini, jelaskan masing-masing bagian sesuai dengan nomor yang ada!

```

public class Komputer {
    String jenis_komputer;
    private String merk;

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

    public String getJenis(){
        return jenis_komputer;
    }

    public String getMerk(){
        return merk;
    }

    public static void main(String[] args){
        Komputer mykom = new Komputer();
        mykom.setDataKomputer("LAPTOP", "MACBOOK");
        System.out.println(mykom.getJenis());
        System.out.println(mykom.getMerk());
    }
}

```

Diagram showing annotations for the code above:

- 1: `public class Komputer {`
- 2: `String jenis_komputer;` and `private String merk;`
- 3: `public void setDataKomputer(String jenis, String merk){`
- 4: `return jenis_komputer;`
- 5: `return merk;`
- 6: `Komputer mykom = new Komputer();`
- 7: `mykom.setDataKomputer("LAPTOP", "MACBOOK");`
- 8: `System.out.println(mykom.getJenis());` and `System.out.println(mykom.getMerk());`

4. Tuliskan perbaikan pada kode program berikut ini, agar kode program tersebut dapat *dicompile* sehingga program akan berjalan dengan benar!

```

2 public class HandPhone {
3     String jenis_hp;
4     int tahun_pembuatan;
5
6     String setDataHP(String jenis_hp, int tahun_pembuatan){
7         jenis_hp = jenis_hp;
8         tahun_pembuatan = tahun_pembuatan;
9     }
10
11     String getJenisHP(){
12
13     }
14
15     String getTahunPembuatan(){
16
17     }
18
19     public static main void(String args[]){
20         HandPhone hp = new HandPhone();
21         hp.setDataHP(jenis_hp, tahun_pembuatan);
22         hp.getJenisHP();
23         hp.getTahunPembuatan();
24     }
25 }

```

Jawab:

1. Class merupakan tempat untuk membuat object (blue print), sedangkan object yaitu hasil instansiasi dari class, object disebut juga dengan instance.
2. Jenis-jenis method:
 - Konstruktor: Method khusus yang digunakan untuk membuat dan menginisialisasi objek dari class.
 - Accessor (Getter): Method yang digunakan untuk mengambil nilai atribut objek.
 - Mutator (Setter): Method yang digunakan untuk mengubah nilai atribut objek.
 - Method Static: Method yang dimiliki oleh class itu sendiri, bukan oleh objek dari class.
3. Penjelasan:

1) Mendeklarasikan sebuah *class* bernama Komputer.

```
public class Komputer { - 1
```

2) Bagian ini adalah deklarasi atribut (variabel anggota) dari class Komputer.

```
String jenis_komputer;  
private String merk; - 2
```

- jenis_komputer adalah atribut publik.

- merk adalah atribut privat yang hanya dapat diakses melalui metode di dalam class.

3) Method setter yang berfungsi untuk memberikan nilai pada atribut jenis_komputer dan merk.

- this.merk = merk; digunakan untuk membedakan antara parameter merk dan atribut merk.

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

4) Method getter yang mengembalikan nilai dari atribut jenis_komputer.

```
public String getJenis(){  
    return jenis_komputer;  
} - 4
```

5) Method getter yang mengembalikan nilai dari atribut merk. Diperlukan karena merk bersifat private.

```
public String getMerk(){  
    return merk;  
} - 5
```

6) Pembuatan objek dari class Komputer dengan nama mykom.

```
Komputer mykom = new Komputer(); - 6
```

7) Memanggil method setDataKomputer untuk mengisi data jenis dan merk komputer ke objek mykom.

```
mykom.setDataKomputer("LAPTOP", "MACBOOK"); - 7
```

8) Bagian ini mencetak ke layar hasil dari pemanggilan method getJenis() dan getMerk() dari objek mykom.

```
System.out.println(mykom.getJenis());  
System.out.println(mykom.getMerk()); - 8
```

4. Kode program terlampir.

TUGAS LANJUTAN SESI 2

1. Buat program sederhana seperti gambar berikut.

```
class Mobil {
    String merk;

    Mobil(String merk) {
        this.merk = merk; // "this.merk" adalah variabel instance
    }

    void tampilkanMerk() {
        System.out.println("Merk mobil: " + this.merk); // Menampilkan merk mobil
    }
}

public class Main {
    public static void main(String[] args) {
        Mobil m = new Mobil("Toyota");
        m.tampilkanMerk(); // Output: Merk mobil: Toyota
    }
}
```

```
class Mahasiswa {
    String nama;

    Mahasiswa(String nama) {
        this.nama = nama;
    }

    void perkenalkanDiri() {
        System.out.println("Halo, nama saya " + this.nama);
    }
}

public class Main {
    public static void main(String[] args) {
        Mahasiswa mhs = new Mahasiswa("Budi");
        mhs.perkenalkanDiri(); // Output: Halo, nama saya Budi
    }
}
```

2. Jelaskan maksud program tersebut.
3. Jelaskan sintaks program dari tiap baris kode nya.

Jawab:

1. Kode program terlampir.
2. Maksud program:
 - a. Program pertama membuat sebuah kelas bernama Mobil yang menyimpan informasi tentang merk mobil, lalu menampilkannya ke layar melalui method tampilkanMerk(). Di dalam main, objek Mobil dibuat dengan merk "Toyota", lalu merk tersebut ditampilkan.
 - b. Program kedua mendefinisikan kelas Mahasiswa yang memiliki atribut nama. Program ini menampilkan sapaan dari seorang mahasiswa menggunakan method perkenalkanDiri(). Objek Mahasiswa dibuat dengan nama "Budi" dan memperkenalkan dirinya melalui output.
3. Program pertama:
 - `class Mobil {` = Mendeklarasikan kelas Mobil
 - `String merk;` = Mendeklarasikan variabel instance merk bertipe String untuk menyimpan merk mobil.
 - `Mobil(String merk) {` = Konstruktor mobil menerima parameter merk
 - `this.merk = merk;` = Menyimpan nilai parameter ke variabel instance
 `}` = this digunakan agar tidak rancu antara parameter dan variabel instance.
 - `void tampilkanMerk() {`
 `System.out.println("Merk mobil: " + this.merk);`
= Method tampilkanMerk() menampilkan nilai dari merk ke layar.
= this.merk mengambil nilai merk dari objek saat ini.

Program kedua:

- *class Mahasiswa* { = Mendeklarasikan kelas Mahasiswa
- *string nama;* = Variabel instance nama bertipe String untuk menyimpan nama mahasiswa.
- *Mahasiswa(String nama)* { = Konstruktor kelas menerima parameter nama.
 this.nama = nama; = menyimpan nilai parameter ke variabel instance.
 } = this digunakan agar tidak rancu antara parameter dan variabel instance.
- *void perkenalkanDiri()* {
 System.out.println("Halo, nama saya " + this.nama);
 }
= Method perkenalkanDiri() menampilkan sapaan beserta nama mahasiswa yang diambil dari variabel instance nama.

Public class main:

- *public class main* {
 public static void main(String[] args) {
 = Class main berisi method main, titik awal eksekusi program
- *Mobil m = new Mobil("Toyota");* = Membuat objek Mobil dengan merk "Toyota" dan menyimpannya ke dalam variabel m
- *m.tampilkanMerk();* = Memanggil method tampilkanMerk() dari objek m, yang akan mencetak: Merk mobil: Toyota
- *Mahasiswa mhs = new Mahasiswa("Budi");* = Membuat objek Mahasiswa bernama mhs dengan nilai "Budi" sebagai nama.
- *mhs.perkenalkanDiri();* = Memanggil method perkenalkanDiri() dari objek mhs, sehingga output-nya adalah: Halo, nama saya Budi