

## **JOBSHEET 3**

### **CLASS DAN OBJEK**

#### **1. Kompetensi**

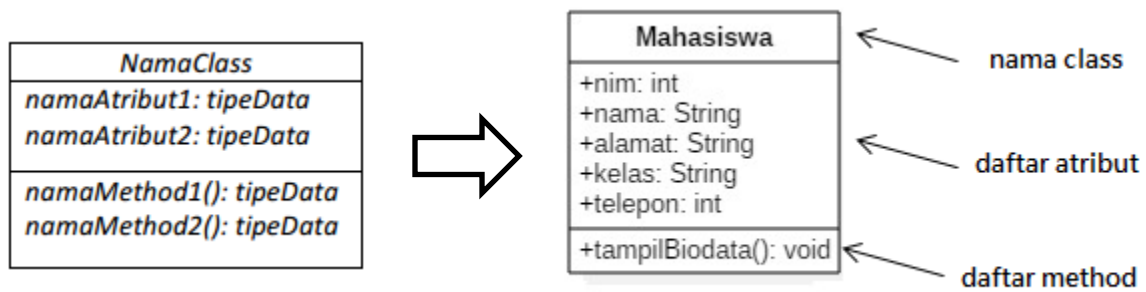
- Mahasiswa dapat memahami Class Diagram
- Mahasiswa memahami cara membuat class
- Mahasiswa mampu memahami cara membuat atribut
- Mahasiswa mampu memahami cara membuat method
- Mahasiswa mampu memahami cara membuat objek/instansiasi

#### **2. Pendahuluan**

##### **2.1 Unified Modeling Language ( UML )**

*Unified Modeling Language ( UML )* adalah tujuan umum, perkembangan, bahasa pemodelan di bidang rekayasa perangkat lunak , yang dimaksudkan untuk menyediakan cara standar untuk memvisualisasikan desain sistem. UML menyediakan sembilan jenis diagram yaitu Diagram kelas (*Class Diagram*), Diagram paket (*Package Diagram*), Diagram use-case (*Usecase Diagram*), Diagram interaksi dan sequence (*Sequence Diagram*), Diagram komunikasi (*Communication Diagram*), Diagram statechart (*Statechart Diagram*), Diagram aktivitas (*Activity Diagram*), Diagram komponen (*Component Diagram*), dan Diagram deployment (*deployment diagram*). Pada materi ini yang akan dipelajari adalah diagram class (*class diagram*).

*Class diagram* adalah sebuah class yang menggambarkan struktur dan penjelasan class, paket, dan objek serta hubungan satu sama lain seperti pewarisan, asosiasi, dan lain-lain. *Class diagram* juga menjelaskan hubungan antar class dalam sebuah sistem yang sedang dibuat dan bagaimana caranya agar mereka saling berkolaborasi untuk mencapai sebuah tujuan. Class juga memiliki 3 area pokok (utama) yaitu : nama, atribut, dan operasi. Nama berfungsi untuk member identitas pada sebuah kelas, atribut fungsinya adalah untuk member karakteristik pada data yang dimiliki suatu objek di dalam kelas, sedangkan operasi fungsinya adalah memberikan sebuah fungsi ke sebuah objek. Berikut ini merupakan contoh dari class diagram:



Gambar 1. Contoh *class diagram* Mahasiswa

## 2.2 Percobaan 1: Membuat Class Diagram

### Studi Kasus 1:

Dalam suatu perusahaan salah satu data yang diolah adalah data karyawan. Setiap karyawan memiliki id, nama, jenis kelamin, jabatan, jabatan, dan gaji. Setiap mahasiswa juga bisa menampilkan data diri pribadi dan melihat gajinya.

1. Gambarkan desain class diagram dari studi kasus 1!
2. Sebutkan Class apa saja yang bisa dibuat dari studi kasus 1!
3. Sebutkan atribut beserta tipe datanya yang dapat diidentifikasi dari masing-masing class dari studi kasus 1!
4. Sebutkan method-method yang sudah anda buat dari masing-masing class pada studi kasus 1!

## 2.3 Class dan Objek

### 2.3.1 Class

Class adalah prototype atau template dari objek. Kita bisa membuat banyak objek dari satu macam class. Di dalam class kita dapat mendeklarasikan variabel dan menciptakan objek (instansiasi). Sebuah class mempunyai anggota yang terdiri dari **atribut dan method**. Atribut adalah semua field identitas yang kita berikan pada suatu class, misal class manusia memiliki field atribut berupa nama dan umur. Method dapat kita artikan sebagai semua fungsi ataupun prosedur yang merupakan perilaku (*behaviour*) dari suatu class.

Untuk membuat suatu Class, digunakan kata kunci class dan diikuti dengan nama Class yang akan dibuat. Deklarasi class dapat dilakukan dengan sintaks berikut:

```
<modifier> class <nama_class> {  
  
    //deklarasi atribut dan method  
  
}
```

Aturan penulisan class adalah sebagai berikut:

1. Berupa kata benda
2. Diawali dengan **HURUF BESAR**
3. Jika terdiri dari lebih dari 1 kata, maka antar kata satu dengan kata yang lain digandeng, dan tiap huruf awal dari tiap kata menggunakan **HURUF BESAR**.

Untuk Access Modifier tidak dibahas pada jobsheet ini, melainkan akan dibahas pada jobsheet berikutnya.

Contoh deklarasi class:

```
public class Mahasiswa{  
  
    }
```

### 2.3.2 Atribut

Untuk melakukan pendeklarasian atribut dapat dilakukan dengan sintaks sebagai berikut:

```
<modifier> <tipe> <nama_atribut> ;
```

Aturan penulisan atribut adalah sebagai berikut:

1. Berupa kata benda
2. Diawali dengan **HURUF KECIL**
3. Jika terdiri dari 2 atau lebih kata, kata pertama diawali **HURUF KECIL** sedangkan kata selanjutnya diawali **HURUF BESAR**. Dan antar kata disambung tidak (dipisah)

### Contoh deklarasi atribut:

```
public int nim;  
public String nama;  
public String alamat;  
public float luas;
```

### 2.3.3 Method

Deklarasi method dapat dilakukan dengan sintaks sebagai berikut:

```
<modifier> <tipe_data> <nama_metode> ([daftar_argumen])  
  
{  
    //statement  
}
```

Method adalah suatu blok dari program yang berisi kode program nama dan properti yang dapat digunakan kembali. Method dapat mempunyai nilai balik atau tidak. Method yang tidak mempunyai nilai balik dipanggil dalam pernyataan yang akan dikerjakan, sedangkan method yang mempunyai nilai balik dipanggil dari suatu ekspresi. Kata kunci untuk mengembalikan/mengeluarkan suatu nilai adalah **return**

Method dengan **tipe data void**, berarti **tidak memiliki nilai balik**, berarti tidak memerlukan kata kunci return di dalamnya. Method dengan **tipe data bukan void**, berarti memerlukan suatu nilai balik, yaitu harus memerlukan **return** di dalamnya

Contoh:

```
public void sayHello(){  
    System.out.println("Hello World!!");  
}  
  
public int tambah (int a, int b){  
    int hasil = a+b;  
    return hasil;  
}
```

TIDAK PERLU RETURN / TIDAK ADA NILAI KEMBALIAN

- TIPE DATA METHOD INT, BERARTI METHOD TSB HARUS MENGEMBALIKAN NILAI INT
- HARUS ADA RETURN

Aturan penulisan method adalah sebagai berikut:

1. Berupa kata kerja
2. Diawali dengan **HURUF KECIL**
3. Jika terdiri dari 2 atau lebih kata, kata pertama diawali **HURUF KECIL** sedangkan kata selanjutnya diawali **HURUF BESAR**. Dan antar kata disambung tidak (dipisah)

Contoh deklarasi method:

```
public void tampil(){  
    System.out.println ("Hallo PBO!!");  
}
```

```
public int tambah(int a, int b){  
    return a+b;  
}
```

### 2.3.4 Objek

Setelah Class dibuat, langkah selanjutnya adalah membuat Object. Proses pembuatan Object dari suatu Class disebut instansiasi. Format dasar instansiasi adalah sebagai berikut:

```
NamaClass namaObject = new NamaClass();
```

Proses membuat objek dari suatu class adalah INSTANSIASI, dan ditandai kata kunci **new**. Aturan penulisan objek adalah **sama** seperti penulisan atribut.

Contoh :

```
Random r = new Random();
```

```
Pegawai p2 = new Pegawai();
```

```
Mahasiswa mhs1 = new Mahasiswa();
```

## 2.3 Percobaan 2: Membuat dan mengakses anggota suatu class

### Studi Kasus 2:

Perhatikan class diagram dibawah ini. Buatlah program berdasarkan class diagram tersebut!



### Langkah kerja:

1. Bukalah text editor, misalnya Notepad ++.
2. Ketikkan kode program berikut ini:

```
1 public class Mahasiswa {
2     public int nim;
3     public String nama;
4     public String alamat;
5     public String kelas;
6
7     public void tampilBiodata () {
8         System.out.println ("Nim      : "+nim);
9         System.out.println ("Nama      : "+nama);
10        System.out.println ("Alamat   : "+alamat);
11        System.out.println ("Kelas   : "+kelas);
12    }
13 }
```

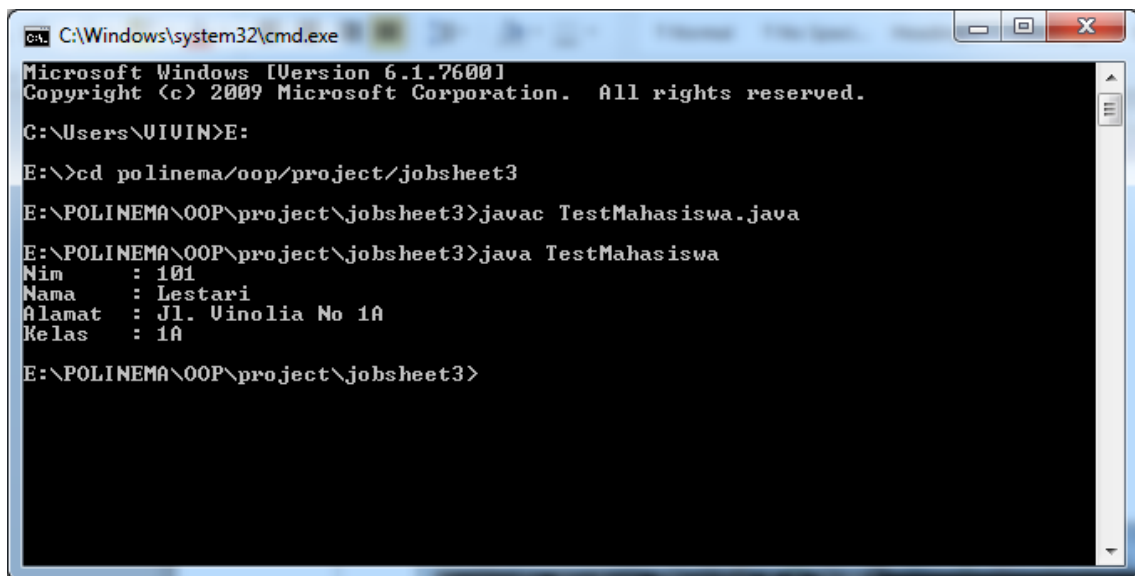
3. Simpan dengan nama file Mahasiswa.java.
4. Untuk dapat mengakses anggota-anggota dari suatu obyek, maka harus dibuat instance dari class tersebut terlebih dahulu. Berikut ini adalah cara pengaksesan anggota-anggota dari class Mahasiswa dengan membuka file baru kemudian ketikkan kode program berikut:

```

1 public class TestMahasiswa {
2     public static void main (String args[]) {
3         Mahasiswa mhs1=new Mahasiswa();
4         mhs1.nim=101;
5         mhs1.nama="Lestari";
6         mhs1.alamat="Jl. Vinolia No 1A";
7         mhs1.kelas="1A";
8         mhs1.tampilBiodata();
9     }
10 }

```

5. Simpan file dengan TestMahasiswa.java
6. Jalankan class TestMahasiswa dengan perintah berikut:



```

C:\Windows\system32\cmd.exe
Microsoft Windows [Version 6.1.7600]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

C:\Users\UIUIN>E:
E:\>cd polinema/oop/project/jobsheet3
E:\POLINEMA\OOP\project\jobsheet3>javac TestMahasiswa.java
E:\POLINEMA\OOP\project\jobsheet3>java TestMahasiswa
Nim      : 101
Nama     : Lestari
Alamat   : Jl. Vinolia No 1A
Kelas   : 1A
E:\POLINEMA\OOP\project\jobsheet3>

```

7. Jelaskan pada bagian mana proses pendeklarasian atribut pada program diatas!
8. Jelaskan pada bagian mana proses pendeklarasian method pada program diatas!
9. Berapa banyak objek yang di instansiasi pada program diatas!
10. Apakah yang sebenarnya dilakukan pada sintaks program “mhs1.nim=101” ?
11. Apakah yang sebenarnya dilakukan pada sintaks program “mhs1.tampilBiodata()” ?
12. Instansiasi 2 objek lagi pada program diatas!

## 2.4 Percobaan 3: Menulis method yang memiliki argument/parameter dan memiliki return

### Langkah kerja:

1. Bukalah text editor baru, misalnya notepad ++.

2. Ketikkan kode program berikut ini:

```
1 public class Barang {
2     public String namaBrg;
3     public String jenisBrg;
4     public int stok;
5
6     public void tampilBarang() {
7         System.out.println ("Nama Barang      : "+namaBrg);
8         System.out.println ("Jenis Barang   : "+jenisBrg);
9         System.out.println ("Stok        : "+stok);
10    }
11
12    //method dengan argumen dan nilai balik (return)
13    public int tambahStok(int brgMasuk) {
14        int stokBaru=brgMasuk+stok;
15        return stokBaru;
16    }
17 }
```

3. Simpan dengan nama file Barang.java

4. Untuk dapat mengakses anggota-anggota dari suatu obyek, maka harus dibuat instance dari class tersebut terlebih dahulu. Berikut ini adalah cara pengaksesan anggota-anggota dari class Barang dengan membuka file baru kemudian ketikkan kode program berikut:

```
1 public class TestBarang{
2     public static void main (String args[]){
3         Barang brg1=new Barang();
4         brg1.namaBrg="Pensil";
5         brg1.jenisBrg="ATK";
6         brg1.stok=10;
7         brg1.tampilBarang();
8         // menampilkan dan mengisi argumen untuk menambahkan stok barang
9         System.out.println ("Stok Baru adalah " +brg1.tambahStok(20));
10    }
11 }
```

5. Simpan dengan nama file TestBarang.java

6. Jalankan program tersebut!

```
E:\POLINEMA\OOP\project\jobsheet3>javac TestBarang.java
E:\POLINEMA\OOP\project\jobsheet3>java TestBarang
Nama Barang      : Pensil
Jenis Barang     : ATK
Stok             : 10
Stok Baru adalah 30
```

7. Apakah fungsi argumen dalam suatu method?



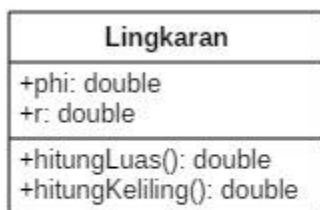
8. Ambil kesimpulan tentang kegunaan dari kata kunci return , dan kapan suatu method harus memiliki return!

## 2.5 Tugas

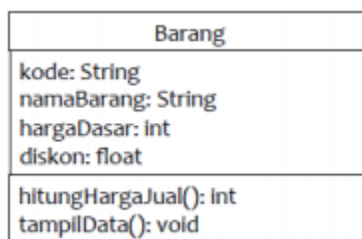
1. Suatu toko persewaan video game salah satu yang diolah adalah peminjaman, dimana data yang dicatat ketika ada orang yang melakukan peminjaman adalah id, nama member, nama game, dan harga yang harus dibayar. Setiap peminjaman bisa menampilkan data hasil peminjaman dan harga yang harus dibayar. Buatlah class diagram pada studi kasus diatas!

Penjelasan:

- Harga yang harus dibayar diperoleh dari lama sewa x harga.
  - Diasumsikan 1x transaksi peminjaman game yang dipinjam hanya 1 game saja.
2. Buatlah program dari class diagram yang sudah anda buat di no 1!
  3. Buatlah program sesuai dengan class diagram berikut ini:



4. Buatlah program sesuai dengan class diagram berikut ini:



Deskripsi:

- Nilai atribut hargaDasar dalam Rupiah dan atribut diskon dalam %
- Method hitungHargaJual() digunakan untuk menghitung harga jual dengan perhitungan berikut ini:

$$\text{harga jual} = \text{harga dasar} - (\text{diskon} \times \text{harga dasar})$$

- Method `tampilData()` digunakan untuk menampilkan nilai dari kode, namaBarang, hargaDasar, diskon dan harga jual.