



JOB SHEET 2

Object Oriented Program



AYU JOVITA WIDYADHARI
2241720219 / 04/ 21

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, dan gaji. Setiap karyawan juga bisa menampilkan data diri pribadi dan melihat gajinya.

1. Gambarkan desain class diagram dari studi kasus 1!,

Jawab:

Karyawan
<ul style="list-style-type: none">• ID : int• Nama: String• jenis kelamin: String• jabatan: String• gaji: int
<ul style="list-style-type: none">❖ tampilDataPribadi(): void❖ tampilGajiKaryawan: int

2. Class apa saja yang bisa dibuat dari studi kasus 1?

Jawab:

Class yang bisa dibuat adalah kelas Karyawan yang mana kelas ini merupakan salah satu data yang diolah karena memiliki attribute seperti id, nama, jenis kelamin, jabatan, dan gaji. Dan kelas ini juga memiliki method yakni tampilDataPribadi() dan tampilGajiKaryawan().

3. Sebutkan atribut beserta tipe datanya yang dapat diidentifikasi dari masing-masing class dari studi kasus 1!

Jawab:

Atribut merupakan karakteristik yang dimiliki oleh entitas yang berperan memberikan penjelasan secara mendetail mengenai entitas tersebut. Jadi, attribute dari kelas Karyawan adalah

- ID dengan tipe data int karena id merupakan sekumpulan angka,
- Nama dengan tipe data String,
- Jenis kelamin dengan tipe data String
- Gaji dengan tipe data double

4. Sebutkan method-method yang sudah anda buat dari masing masing class pada studi kasus 1!

Jawab:

Method adalah bagian dari sebuah class yang berisi serangkaian instruksi atau operasi yang dapat dijalankan atau dipanggil untuk melakukan tugas-tugas tertentu. Metode digunakan untuk mengatur perilaku atau tindakan yang terkait dengan objek dari class tersebut.

Method yang dibuat dari kelas karyawan yakni:

- tampilkanDataDiri(): method ini akan menampilkan biodata dari karyawan mulai dari id, nama, hingga gaji
- tampilkanGajiKaryawn(): method yang akan menampilkan rician gaji yang didapat oleh seorang karyawan.

Percobaan 2: Membuat dan mengakses anggota suatu class

1. Bukalah text editor atau IDE, misalnya Notepad ++ / netbeans.

```
10 public class Mahasiswa2 {
11     public int nim;
12     public String nama;
13     public String alamat;
14     public String kelas;
15
16     public void tampilanBiodata() {
17         System.out.println("ID Student: "+nim);
18         System.out.println("Name      : "+nama);
19         System.out.println("Address   : "+alamat);
20         System.out.println("Class    : "+kelas);
21     }
22 }
23
```

- 2.
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 anggotaanggota dari class Mahasiswa dengan membuka file baru kemudian ketikkan kode program berikut:

```
public class TestMahasiswa {
    public static void main(String[] args) {
        Mahasiswa2 mhs1 = new Mahasiswa2();
        mhs1.nim = 101;
        mhs1.nama="Lestari";
        mhs1.alamat="St.Vinolia 1A";
        mhs1.kelas="1A";
        mhs1.tampilanBiodata();
    }
}
```

5. Simpan file dengan TestMahasiswa.java
6. Jalankan class TestMahasiswa

```
Output - Jobsheet_1 (run)

run:
ID Student: 101
Name      : Lestari
Address   : St.Vinolia 1A
Class     : 1A
BUILD SUCCESSFUL (total time: 0 seconds)
```

7. Jelaskan pada bagian mana proses pendeklarasian atribut pada program diatas!

Jawab:

Deklarasi attribute ada di dalam class yakni pada langkah ke-13 sampai langkah ke-16.

```
10 public class Mahasiswa2 {
11
12     //deklarasi attribute
13     public int nim;
14     public String nama;
15     public String alamat;
16     public String kelas;
17 }
```

8. Jelaskan pada bagian mana proses pendeklarasian method pada program diatas!

Jawab:

Setelah mendeklarasikan attribute langkah selanjutnya adalah mendeklarasikan method tetap didalam sebuah class

```
18     //deklarasi method
19     public void tampilBiodata() {
20         System.out.println("ID Student: "+nim);
21         System.out.println("Name      : "+nama);
22         System.out.println("Address   : "+alamat);
23         System.out.println("Class    : "+kelas);
24     }
25 }
26 }
```

9. Berapa banyak objek yang di instansiasi pada program diatas!

Jawab:

Instansiasi objek adalah proses membuat objek dari suatu class

Pada class tersebut hanya ada satu objek yang diinstansiasi yakni mhs1.

10. Apakah yang sebenarnya dilakukan pada sintaks program “mhs1.nim=101” ?

Jawab:

Sintaks program “mhs1.nim=101” merupakan sebuah proses memberikan nilai pada attribute. Jadi, attribute nim memiliki nilai 101 dari objek mhs1.

11. Apakah yang sebenarnya dilakukan pada sintaks program “mhs1.tampilBiodata()” ?

Jawab:

sintaks program “mhs1.tampilBiodata()” merupakan proses menampilkan informasi biodata dari karyawan. Sintaks program **mhs1.tampilBiodata()** adalah pemanggilan metode **tampilBiodata()** pada objek **mhs1**. Objek **mhs1** adalah sebuah objek yang merupakan instance dari suatu class yang memiliki metode **tampilBiodata()**.

12. Instansiasi 2 objek lagi pada program diatas!

```

10 public class TestMahasiswa {
11     public static void main(String[] args){
12         Mahasiswa2 mhs1 = new Mahasiswa2();
13         mhs1.nim = 101;
14         mhs1.nama="Lestari";
15         mhs1.alamat="St.Vinolia 1A";
16         mhs1.kelas="1A";
17         mhs1.tampilanBiodata();
18         System.out.println();
19
20         Mahasiswa2 mhs2 = new Mahasiswa2();
21         mhs2.nim = 102;
22         mhs2.nama="Dewi";
23         mhs2.alamat="St.Vinolia 1A";
24         mhs2.kelas="1A";
25         mhs2.tampilanBiodata();
26         System.out.println();
27
28         Mahasiswa2 mhs3 = new Mahasiswa2();
29         mhs3.nim = 103;
30         mhs3.nama="Ayu";
31         mhs3.alamat="St.Vinolia 1A";
32         mhs3.kelas="1A";
33         mhs3.tampilanBiodata();
34     }
35 }

```

Output - Jobsheet_1 (run)

```

>> run:
>> ID Student: 101
  Name      : Lestari
  Address    : St.Vinolia 1A
  Class      : 1A

  ID Student: 102
  Name      : Dewi
  Address    : St.Vinolia 1A
  Class      : 1A

  ID Student: 103
  Name      : Ayu
  Address    : St.Vinolia 1A
  Class      : 1A
BUILD SUCCESSFUL (total time: 0 seconds)

```

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

1. Bukalah text editor atau IDE, misalnya Notepad ++ / netbeans.
2. Ketikkan kode program berikut ini
3. Simpan dengan nama file Barang.java

```

10 public class Barang {
11     public String namaBrg;
12     public String jenisBrg;
13     public int stok;
14
15     public void tampilBarang() {
16         System.out.println("Nama Barang : " + namaBrg);
17         System.out.println("Jenis Barang : " + jenisBrg);
18         System.out.println("Stok : " + stok);
19     }
20     //method dengan argumen dan nila balik (return)
21     public int tambahStok(int brgMasuk) {
22         int stokBaru = brgMasuk + stok;
23         return stokBaru;
24     }
25 }
26

```

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:

```

10 public class TestBarang {
11     public static void main(String[] args) {
12         Barang brg1 = new Barang();
13         brg1.namaBrg = "Pensil";
14         brg1.jenisBrg = "ATK";
15         brg1.stok = 10;
16         brg1.tampilBarang();
17         //menampilkan dan mengisi untuk menambahkan stok barang
18         System.out.println("Stok Baru adalah " + brg1.tambahStok(brgMasuk: 20));
19     }
20 }
21

```

5. Simpan dengan nama file TestBarang.java
6. Jalankan program tersebut!

```

Output - Jobsheet_1 (run)

run:
Nama Barang : Pensil
Jenis Barang : ATK
Stok : 10
Stok Baru adalah 30
BUILD SUCCESSFUL (total time: 0 seconds)

```

7. Apakah fungsi argumen dalam suatu method?

Jawaban:

Argumen dalam suatu metode (atau sering disebut juga sebagai parameter) adalah nilai-nilai yang diterima oleh metode tersebut saat dipanggil. Fungsi argumen dalam suatu metode

digunakan untuk memberikan data input ke dalam metode. Ini memungkinkan metode untuk bekerja dengan nilai-nilai yang berbeda setiap kali dipanggil, sehingga metode dapat digunakan secara fleksibel untuk berbagai kebutuhan. Argumen memungkinkan metode berinteraksi dengan kode lain dalam program. Misalnya, Anda dapat mengirimkan objek dari satu metode ke metode lain untuk memprosesnya atau mengambil hasil dari satu metode dan menggunakannya dalam metode lain.

8. Ambil kesimpulan tentang kegunaan dari kata kunci `return`, dan kapan suatu method harus memiliki `return`!
 - Method dengan tipe data `void` tidak perlu mengembalikan data (tidak perlu `return`), sedangkan method dengan tipe data bukan `void` (bias `int`, `String`, dll) mempunyai nilai balik dengan menambahkan kata kunci '`return`'.
 - **`return`** digunakan untuk mengembalikan nilai dari sebuah metode kepada pemanggilnya. Ini memungkinkan metode untuk memberikan hasil atau output yang dapat digunakan dalam kode pemanggil.
 - Setelah sebuah pernyataan **`return`** dieksekusi dalam sebuah metode, eksekusi metode tersebut akan berakhir, dan kontrol akan kembali ke pemanggil. Ini berarti kode di dalam metode setelah pernyataan **`return`** tidak akan dijalankan.

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.

```
10 public class Game {
11     private int gameId;
12     private String nama;
13     private double harga;
14
15     public Game(int game, String nama, double harga) {
16         this.gameID = gameId;
17         this.nama = nama;
18         this.harga = harga;
19     }
20     public int getGameID() {
21         return gameId;
22     }
23     public String getNama() {
24         return nama;
25     }
26     public double getHarga() {
27         return harga;
28     }
29 }
```

```

10 public class Peminjaman {
11     private int id;
12     private String namaMember;
13     private String namaGame;
14     private double hargaTotal;
15
16     public Peminjaman(int id, String namaMember, String namaGame, double lamaSewa, double hargaPerHari){
17         this.id = id;
18         this.namaMember = namaMember;
19         this.namaGame = namaGame;
20         this.hargaTotal = lamaSewa*hargaPerHari;
21     }
22
23     public void tampilPeminjaman(){
24         System.out.println("ID Peminajamn: " + id);
25         System.out.println("Nama Member: " + namaMember);
26         System.out.println("Nama Game: " + namaGame);
27         System.out.println("Total Harga: " + hargaTotal);
28     }
29 }
30

```

```

10 public class PeminjamanMain {
11     public static void main(String[] args){
12         Game gml = new Game(game:1, nama: "Game A", harga: 5.000);
13         Peminjaman pmj1 = new Peminjaman(id: 101, namaMember:"Clara", namaGame: "Game A", lamaSewa: 3, hargaPerHari: gml.getHarga());
14
15         pmj1.tampilPeminjaman();
16     }
17 }
18

```

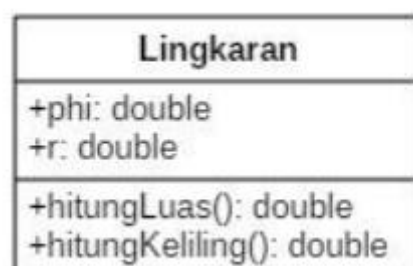
Output - Jobsheet_1 (run)

```

run:
ID Peminajamn: 101
Nama Member: Clara
Nama Game: Game A
Total Harga: 15.0
BUILD SUCCESSFUL (total time: 0 seconds)

```

2. Buatlah program dari class diagram yang sudah anda buat di no 1!
3. Buatlah program sesuai dengan class diagram berikut ini:




```

10 public class Lingkaran {
11     double phi = 3.14;
12     double r;
13
14     public double hitungLuas() {
15         double L = phi*r*r;
16         return L;
17     }
18     public double hitungKeliling() {
19         double K = 2*phi*r;
20         return K;
21     }
22 }
23

```

```

10 public class LingkaranTest {
11     public static void main(String[] args) {
12         Lingkaran l1 = new Lingkaran();
13         l1.r = 5;
14         System.out.println("Luas lingkaran dengan r = " + l1.r + ": ");
15         System.out.println("Luas: " + l1.hitungLuas());
16         System.out.println("Keliling: " + l1.hitungKeliling());
17     }
18 }
19

```

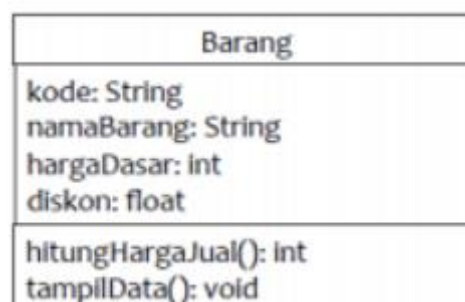
Output - Jobsheet_1 (run)

```

run:
Luas lingkaran dengan r = 5.0:
Luas: 78.5
Keliling: 31.400000000000002
BUILD SUCCESSFUL (total time: 0 seconds)

```

4. Buatlah program sesuai dengan class diagram berikut ini:



Deskripsi / Penjelasan :

- Nilai atribut hargaDasar dalam Rupiah dan atribut diskon dalam %
- Method hitungHargaJual() digunakan untuk menghitung harga jual dengan perhitungan berikut ini:
harga jual = harga dasar – (diskon x harga dasar)
- Method tampilData() digunakan untuk menampilkan nilai dari kode, namaBarang, hargaDasar, diskon dan harga jual.

```
10 public class Barang2 {
11     String kode;
12     String namaBarang;
13     int hargaDasar;
14     float diskon;
15
16     public int hitungHargaJual() {
17         int hargaJual= (int) (hargaDasar-(diskon/100*hargaDasar));
18         return hargaJual;
19     }
20     public void tampilData() {
21         System.out.println("Kode: " +kode);
22         System.out.println("Nama Barang: " +namaBarang);
23         System.out.println("Harga: Rp " +hargaDasar + ",00");
24         System.out.println("Diskon: " +diskon+"%");
25         System.out.println("Total: Rp "+hitungHargaJual()+",00");
26     }
27 }
28
```

```
10 public class TestBarang2 {
11     public static void main(String[] args) {
12         Barang2 br2 = new Barang2();
13
14         br2.kode = "01";
15         br2.namaBarang = "Roti";
16         br2.hargaDasar = 15000;
17         br2.diskon = 5;
18         br2.hitungHargaJual();
19         br2.tampilData();
20     }
21 }
22
```

Output - Jobsheet_1 (run)

```
>> run:
>> Kode: 01
Nama Barang: Roti
Harga: Rp 15000,00
Diskon: 5.0%
Total: Rp 14250,00
BUILD SUCCESSFUL (total time: 0 seconds)
```