

Jobsheet 2: Class dan Object

Object Oriented Programming



**Arranged by :
Shofwah Kanaka Ebsa Anargya
2241720254 / 22
2I**

**INFORMATION TECHNOLOGY
D-IV INFORMATICS ENGINEERING
MALANG STATE POLYTECHNIC**

2023

Percobaan

4.1 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!,

Karyawan	Perusahaan
<div>+ id: String + nama: String + jenisKelamin: String + jabatan: String + gaji: Double</div>	<div>+ idKaryawan: String + dataKaryawan:String + dataGajiKaryawan: Double</div>
<div>+ tampilDataDiri(): void + melihatGaji(): void</div>	<div>+ tambahKaryawan(): void + hapusKaryawan(): void + hitungGaji(): Double + tampilGaji(): Double + tampilDataDiriKaryawan(): void</div>

2. Sebutkan Class apa saja yang bisa dibuat dari studi kasus 1!,
 - Karyawan
 - Perusahaan
3. Sebutkan atribut beserta tipe datanya yang dapat diidentifikasi dari masing-masing class dari studi kasus 1!
 - a. Karyawan
 - + id: String
 - + nama: String
 - + jenisKelamin: String
 - + jabatan: String
 - + gaji: Double
 - b. Perusahaan
 - + idKaryawan(): String
 - + dataKaryawan():String
 - + dataGajiKaryawan(): Double
4. Sebutkan method-method yang sudah anda buat dari masing-masing class pada studi kasus 1!
 - a. Karyawan

+ tampilDataDiri(): void

+ melihatGaji(): void

b. Perusahaan

+ tambahKaryawan(): void

+ hapusKaryawan(): void

+ hitungGaji(): Double

+ tampilGaji(): Double

+ tampilDataDiriKaryawan(): void

4.2 Percobaan 2: Membuat dan mengakses anggota suatu class

Studi Kasus 2:

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



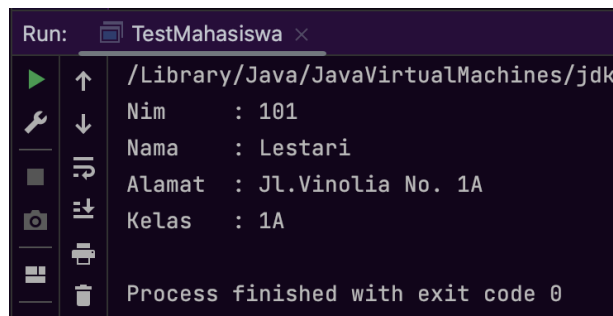
Langkah kerja:

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

```
1 package js2;
2
3 public class Mahasiswa {
4     2 usages
5     public int nim;
6     2 usages
7     public String nama;
8     2 usages
9     public String alamat;
10    2 usages
11    public String kelas;
12
13    1 usage
14    public void tampilBiodata(){
15        System.out.println("Nim\t\t: "+ nim);
16        System.out.println("Nama\t\t: "+ nama);
17        System.out.println("Alamat\t\t: "+alamat);
18        System.out.println("Kelas\t\t: "+kelas);
19    }
20 }
```

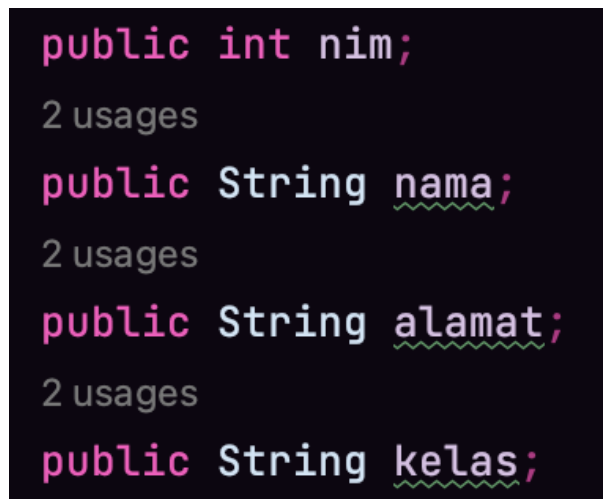
3. Simpan dengan nama file Mahasiswa.java.

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



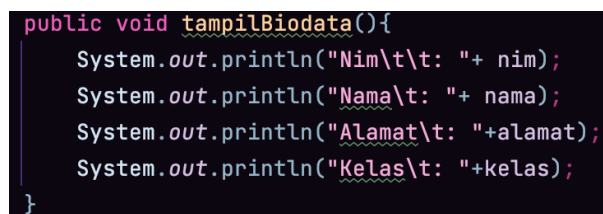
```
Run: TestMahasiswa x
/Library/Java/JavaVirtualMachines/jdk
Nim    : 101
Nama   : Lestari
Alamat : Jl.Vinolial No. 1A
Kelas : 1A
Process finished with exit code 0
```

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



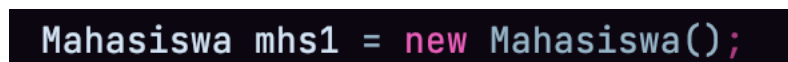
```
public int nim;
2 usages
public String nama;
2 usages
public String alamat;
2 usages
public String kelas;
```

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



```
public void tampilBiodata(){
    System.out.println("Nim\t\t: " + nim);
    System.out.println("Nama\t: " + nama);
    System.out.println("Alamat\t: " + alamat);
    System.out.println("Kelas\t: " + kelas);
}
```

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



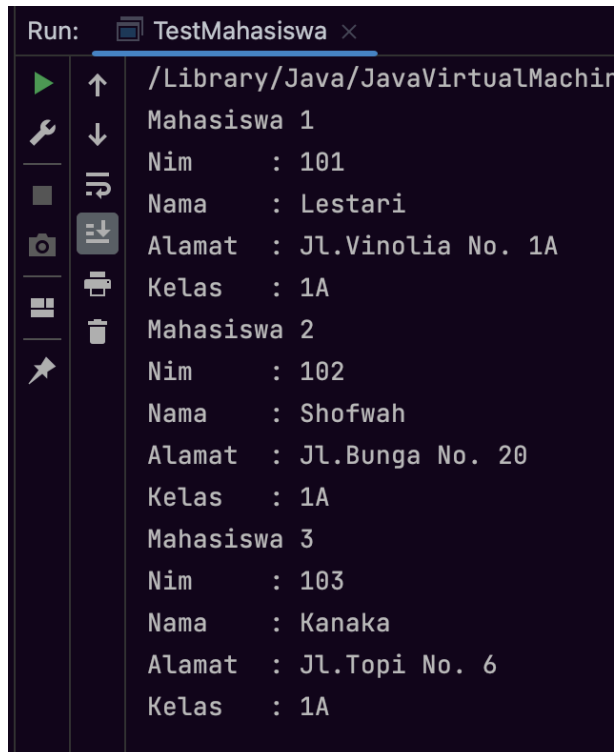
```
Mahasiswa mhs1 = new Mahasiswa();
```

10. Apakah yang sebenarnya dilakukan pada sintaks program “mhs1.nim=101” ?
melakukan inisialisasi atau pengisian nilai atribut nim dari objek mhs1 dengan nilai 101
11. Apakah yang sebenarnya dilakukan pada sintaks program “mhs1.tampilBiodata()” ?

memanggil metode tampilBiodata() dari objek mhs1. Menjalankan kode yang ada dalam metode tampilBiodata() yang mungkin digunakan untuk menampilkan biodata mahasiswa yang terkait dengan objek mhs1.

12. Instansiasi 2 objek lagi pada program diatas!

```
12      System.out.println("Mahasiswa 2");
13      Mahasiswa mhs2 = new Mahasiswa();
14      mhs2.nim=102;
15      mhs2.nama="Shofwah";
16      mhs2.alamat="Jl.Bunga No. 20";
17      mhs2.kelas="1A";
18      mhs2.tampilBiodata();
19      System.out.println("Mahasiswa 3");
20      Mahasiswa mhs3 = new Mahasiswa();
21      mhs3.nim=103;
22      mhs3.nama="Kanaka";
23      mhs3.alamat="Jl.Topi No. 6";
24      mhs3.kelas="1A";
25      mhs3.tampilBiodata();
```



```
Run: TestMahasiswa x
/Library/Java/JavaVirtualMachin
Mahasiswa 1
Nim      : 101
Nama     : Lestari
Alamat  : Jl.Vinolia No. 1A
Kelas   : 1A
Mahasiswa 2
Nim      : 102
Nama     : Shofwah
Alamat  : Jl.Bunga No. 20
Kelas   : 1A
Mahasiswa 3
Nim      : 103
Nama     : Kanaka
Alamat  : Jl.Topi No. 6
Kelas   : 1A
```

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

Langkah kerja:

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

```

1 package js2;
2
3 public class Barang {
4     public String namaBrg;
5     public String jenisBrg;
6     public int stock;
7
8     public void tampilBarang(){
9         System.out.println("Nama Barang\t\t: "+namaBrg);
10        System.out.println("Jenis Barang\t\t: "+jenisBrg);
11        System.out.println("Stock\t\t\t\t: "+stock);
12    }
13
14    //method dengan argumen dan nilai balik (return)
15    public int tambahStock(int brgMasuk){
16        int stockBaru = brgMasuk + stock;
17        return stockBaru;
18    }
19 }

```

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 package js2;
2
3 public class Barang {
4     public String namaBrg;
5     public String jenisBrg;
6     public int stock;
7
8     public void tampilBarang(){
9         System.out.println("Nama Barang\t\t: "+namaBrg);
10        System.out.println("Jenis Barang\t\t: "+jenisBrg);
11        System.out.println("Stock\t\t\t\t: "+stock);
12    }
13
14    //method dengan argumen dan nilai balik (return)
15    public int tambahStock(int brgMasuk){
16        int stockBaru = brgMasuk + stock;
17        return stockBaru;
18    }
19 }

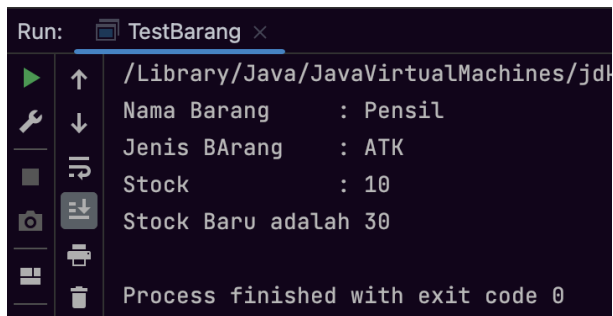
```

```

1 package js2;
2
3 no usages
4 public class TestBarang {
5     no usages
6     public static void main(String[] args) {
7         Barang brg1 = new Barang();
8         brg1.namaBrg = "Pensil";
9         brg1.jenisBrg = "ATK";
10        brg1.stock = 10;
11        brg1.tampilBarang();
12        //menampilkan dan mengisi argumen untuk menambahkan stock barang
13        System.out.println("Stock Baru adalah " + brg1.tambahStock( brgMasuk: 20));
14    }
15 }

```

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



```

Run: TestBarang x
/Library/Java/JavaVirtualMachines/jdk-11.0.2/bin/java
Nama Barang      : Pensil
Jenis BARang     : ATK
Stock            : 10
Stock Baru adalah 30
Process finished with exit code 0

```

7. Apakah fungsi argumen dalam suatu method?
 Nilai atau data yang diberikan sebagai input ke dalam method.
8. Ambil kesimpulan tentang kegunaan dari kata kunci return , dan kapan suatu method harus memiliki return!

Suatu method harus memiliki pernyataan return jika ia memiliki tipe data pengembalian (return type) selain void. Kata kunci return digunakan untuk menghentikan eksekusi method dan mengembalikan nilai yang sesuai dengan tipe data pengembalian method.

4.4 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.

```
package js2;

public class Game {
    private int id;
    private String nama;
    private double harga;

    public Game(int id, String nama, double harga) {
        this.id = id;
        this.nama = nama;
        this.harga = harga;
    }

    // Getter dan Setter

    public int getId() {
        return id;
    }

    public void setId(int id) {
        this.id = id;
    }

    public String getNama() {
        return nama;
    }

    public void setNama(String nama) {
        this.nama = nama;
    }

    public double getHarga() {
        return harga;
    }

    public void setHarga(double harga) {
        this.harga = harga;
    }
}
```



```

package js2;

public class Member {
    private int id;
    private String nama;

    public Member(int id, String nama) {
        this.id = id;
        this.nama = nama;
    }

    // Getter dan Setter

    public int getId() {
        return id;
    }

    public void setId(int id) {
        this.id = id;
    }

    public String getNama() {
        return nama;
    }

    public void setName(String nama) {
        this.nama = nama;
    }
}

```

```

package js2;
import java.util.*;
public class MainGame {
    public static void main(String[] args) {
        // Membuat objek game
        Game game1 = new Game(1, "Game A", 20000);

        // Membuat objek member
        Member member1 = new Member(001, "Shofwah");

        // Membuat objek pinjaman
        Peminjaman peminjaman1 = new Peminjaman(1001, member1, game1,
3);

        // Menampilkan data peminjaman
        peminjaman1.tampilkanData();
    }
}

```

```

package js2;

public class Peminjaman {
    private int id;
    private Member member;
    private Game game;
    private int lamaSewa;

    public Peminjaman(int id, Member member, Game game, int lamaSewa) {
        this.id = id;
        this.member = member;
        this.game = game;
        this.lamaSewa = lamaSewa;
    }

    // Getter dan Setter

    public int getId() {
        return id;
    }

    public void setId(int id) {
        this.id = id;
    }

    public Member getMember() {
        return member;
    }

    public void setMember(Member member) {
        this.member = member;
    }

    public Game getGame() {
        return game;
    }

    public void setGame(Game game) {
        this.game = game;
    }

    public int getLamaSewa() {
        return lamaSewa;
    }

    public void setLamaSewa(int lamaSewa) {
        this.lamaSewa = lamaSewa;
    }

    public double hitungHarga() {
        return game.getHarga() * lamaSewa;
    }

    public void tampilkanData() {
        System.out.println("ID Peminjaman: " + id);
        System.out.println("Nama Member: " + member.getNama());
        System.out.println("Nama Game: " + game.getNama());
        System.out.println("Lama Sewa: " + lamaSewa + " hari");
        System.out.println("Harga yang Harus Dibayar: " +
            hitungHarga());
    }
}

```

```

Run: [MainGame x]
/Library/Java/JavaVirtualMachines/jdk-
ID Peminjaman: 1001
Nama Member: Shofwah
Nama Game: Game A
Lama Sewa: 3 hari
Harga yang Harus Dibayar: 60000.0
Process finished with exit code 0

```

2. Buatlah program dari class diagram yang sudah anda buat di no 1!

Member	Karyawan	Peminjaman
<div>- id: int - nama: String</div>	<div>- id: int - nama: String - harga: double</div>	<div>- id: int - Member: Member - Game: Game - lamaSewa: Int</div>
<div>+ getId(): void + setId(): void + getNama(): void + setNama(): void</div>	<div>+ getId(): void + setId(): void + getNama(): void + setNama(): void + getHarga(): void + setHarga(): void</div>	<div>+ getId(): void + setId(): void + getMember(): void + setMember(): void + getGame(): void + setGame(): void + getLamaSewa(): void + setLamaSewa(): void + hitungHarga(): double + tampilkanData(): void</div>

3. Buatlah program sesuai dengan class diagram berikut ini:

Lingkaran
<div>+phi: double +r: double</div>
<div>+hitungLuas(): double +hitungKeliling(): double</div>

```

package js2;

public class Lingkaran {
    private double phi;
    private double r;

    public Lingkaran(double phi, double r) {
        this.phi = phi;
        this.r = r;
    }

    // Getter dan Setter untuk phi dan r
    public double getPhi() {
        return phi;
    }

    public void setPhi(double phi) {
        this.phi = phi;
    }

    public double getR() {
        return r;
    }

    public void setR(double r) {
        this.r = r;
    }

    // Metode untuk menghitung luas lingkaran
    public double hitungLuas() {
        return phi * r * r;
    }

    // Metode untuk menghitung keliling lingkaran
    public double hitungKeliling() {
        return 2 * phi * r;
    }

    public static void main(String[] args) {
        // Membuat objek Lingkaran
        Lingkaran lingkaran = new Lingkaran(3.14, 12);

        // Menghitung dan menampilkan luas lingkaran
        double luas = lingkaran.hitungLuas();
        System.out.println("Luas Lingkaran: " + luas);

        // Menghitung dan menampilkan keliling lingkaran
        double keliling = lingkaran.hitungKeliling();
        System.out.println("Keliling Lingkaran: " + keliling);
    }
}

```

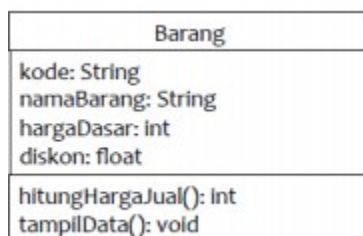
Run: Lingkaran ×

```

/Library/Java/JavaVirtualMachines/jdk-
Luas Lingkaran: 452.15999999999997
Keliling Lingkaran: 75.36
Process finished with exit code 0

```

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:

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

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

```
package js2;

public class Barang2 {
    private String kode;
    private String namaBarang;
    private int hargaDasar;
    private float diskon;

    public Barang2(String kode, String namaBarang, int hargaDasar, float
diskon) {
        this.kode = kode;
        this.namaBarang = namaBarang;
        this.hargaDasar = hargaDasar;
        this.diskon = diskon;
    }

    // Getter dan Setter untuk semua atribut
    public String getKode() {
        return kode;
    }

    public void setKode(String kode) {
        this.kode = kode;
    }

    public String getNamaBarang() {
        return namaBarang;
    }

    public void setNamaBarang(String namaBarang) {
        this.namaBarang = namaBarang;
    }

    public int getHargaDasar() {
        return hargaDasar;
    }

    public void setHargaDasar(int hargaDasar) {
        this.hargaDasar = hargaDasar;
    }

    public float getDiskon() {
        return diskon;
    }

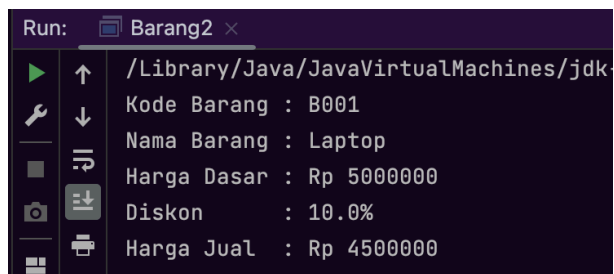
    public void setDiskon(float diskon) {
        this.diskon = diskon;
    }

    // Metode untuk menghitung harga jual
    public int hitungHargaJual() {
        float diskonPersen = diskon / 100;
        float potonganDiskon = hargaDasar * diskonPersen;
        int hargaJual = hargaDasar - Math.round(potonganDiskon);
        return hargaJual;
    }

    // Metode untuk menampilkan data barang
    public void tampilData() {
        System.out.println("Kode Barang\t: " + kode);
        System.out.println("Nama Barang\t: " + namaBarang);
        System.out.println("Harga Dasar\t: Rp " + hargaDasar);
        System.out.println("Diskon\t\t: " + diskon + "%");
        System.out.println("Harga Jual\t: Rp " + hitungHargaJual());
    }

    public static void main(String[] args) {
        // Membuat objek Barang
        Barang2 barang1 = new Barang2("B001", "Laptop", 5000000, 10.0f);

        // Menampilkan data barang
        barang1.tampilData();
    }
}
```



```
Run: Barang2 ×
/Library/Java/JavaVirtualMachines/jdk-
Kode Barang : B001
Nama Barang : Laptop
Harga Dasar : Rp 5000000
Diskon      : 10.0%
Harga Jual  : Rp 4500000
```