

# PRAKTIKUM 7

---

## PENGENALAN PEMROGRAMAN BERBASIS OBYEK

---

### A. TUJUAN PEMBELAJARAN

1. Mendeklarasikan suatu class
2. Mendeklarasikan suatu atribut
3. Mendeklarasikan suatu metod
4. Mengakses anggota suatu obyek

### B. DASAR TEORI

Deklarasi class dapat dilakukan dengan sintaks sebagai berikut:

```
<modifier> class <nama_class> {  
    [deklarasi_atribut]  
    [deklarasi_konstruktor]  
    [deklarasi_metode]  
}
```

Contoh:

```
public class Siswa {  
    ...  
}
```

Deklarasi atribut dapat dilakukan dengan sintaks sebagai berikut:

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

Contoh:

```
public class Siswa {  
    public int nrp;
```

```
        public String nama;
    }
```

Deklarasi metode dapat dilakukan dengan sintaks sebagai berikut:

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

Contoh:

```
public class Siswa {
    public int nrp;
    public String nama;
    public void info() {
        System.out.println("Ini siswa PENS");
    }
}
```

Untuk dapat mengakses anggota-anggota dari suatu obyek, maka harus dibuat instance dari class tersebut terlebih dahulu. Berikut ini adalah contoh pengaksesan anggota-anggota dari class Siswa:

```
public class Siswa {
    public static void main(String args[]) {
        Siswa it=new Siswa();
        it.nrp=5;
        it.nama="Andi";
        it.info();
    }
}
```

### **C. TUGAS PENDAHULUAN**

1. Apakah yang dimaksud dengan kelas, method, atribut dan obyek?
2. Buatlah contoh suatu kelas dan definisikan atribut dan methodnya!
3. Buatlah kode program soal no. 2 diatas!

4. Buatlah kelas yang berisi main method yang membuat obyek dari kelas yang telah dibuat di soal no. 3. Selanjutnya obyek tersebut mengakses atribut dan methodnya.

#### **D. PERCOBAAN**

##### **Percobaan 1 : Mengakses anggota suatu class**

Amati program dibawah ini:

```
public class Siswa {
    int nrp;
    public void setNrp(int i) {
        nrp=i;
    }
}

public class Test {
    public static void main(String args[]) {
        Siswa anak=new Siswa();
        anak.setNrp(5);
        System.out.println(anak.nrp);
    }
}
```

##### **Percobaan 2 : Mengakses anggota suatu class**

Amati program dibawah ini:

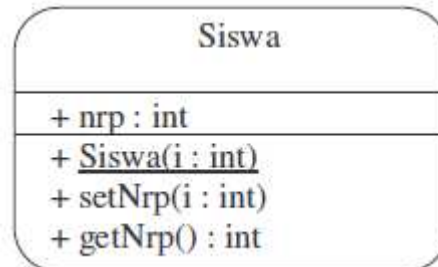
```
public class Siswa {
    int nrp;
    String nama;

    public void setNrp(int i) {
        nrp=i;
    }

    public void setNama(String i) {
        nama=i;
    }
}
```

### Percobaan 3 : Mengimplementasikan UML class diagram dalam program

Berikut adalah sebuah UML class diagram dari suatu kasus:



Dari class diagram tersebut, dapat diimplementasikan ke dalam program sebagai berikut:

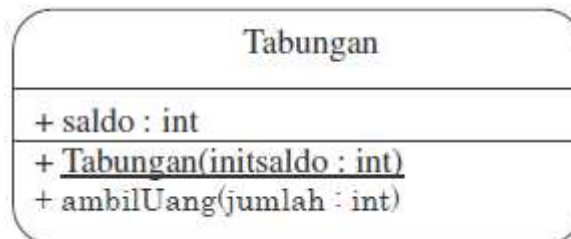
```
public class Siswa {
    public int nrp;

    public Siswa(int i) {
        nrp=i;
    }

    public void setNrp(int i) {
        nrp=i;
    }
    public int getNrp() {
        return nrp;
    }
}
```

## E. LATIHAN

**Latihan 1 : Mengimplementasikan UML class diagram dalam program untuk class Tabungan.**



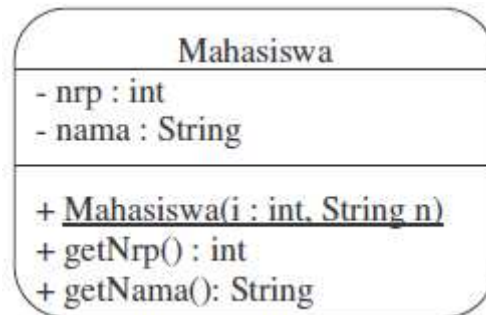
Transformasikan class diagram diatas ke dalam bentuk program. Tulislah listing program berikut ini sebagai pengetesan.

```
public class TesLatihan1{
    public static void main(String args[]){
        Tabungan tabungan = new Tabungan(5000);
        System.out.println("Saldo awal : " + tabungan.saldo);
        tabungan.ambilUang(2300);
        System.out.println("Jumlah uang yang diambil : 2300");
        System.out.println("Saldo sekarang : " + tabungan.saldo);
    }
}
```

Lakukan kompilasi pada program diatas dan jalankan. Jika tampilan di layar tampak seperti dibawah ini, maka program anda sudah benar. Jika tidak sama, benahi kembali program anda dan lakukan hal yang sama seperti diatas.

```
Saldo awal : 5000
Jumlah uang yang diambil : 2300
Saldo sekarang : 2700
```

**Latihan 2 : Mengimplementasikan UML class diagram dalam program untuk class Mahasiswa**



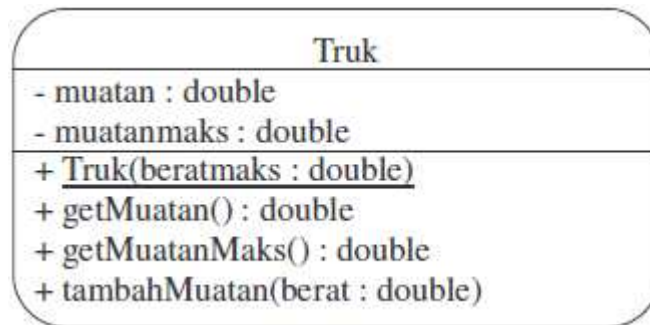
Transformasikan class diagram diatas ke dalam bentuk program! Tulislah listing program berikut ini sebagai pengetesan.

```
public class TesLatihan2{
    public static void main(String args[]){
        Mahasiswa mhs = new Mahasiswa(12345, "Jono");
        System.out.println("NRP : " + mhs.getNrp());
        System.out.println("Nama : " + mhs.getnama());
    }
}
```

Lakukan kompilasi pada program diatas dan jalankan. Jika tampilan di layar tampak seperti dibawah ini, maka program anda sudah benar. Jika tidak sama, benahi kembali program anda dan lakukan hal yang sama seperti diatas.

```
NRP : 12345
Nama : Jono
```

**Latihan 3 : Mengimplementasikan UML class diagram dalam program untuk class Truk**



Transformasikan class diagram diatas ke dalam bentuk program! Tulislah listing program berikut ini sebagai pengetesan.

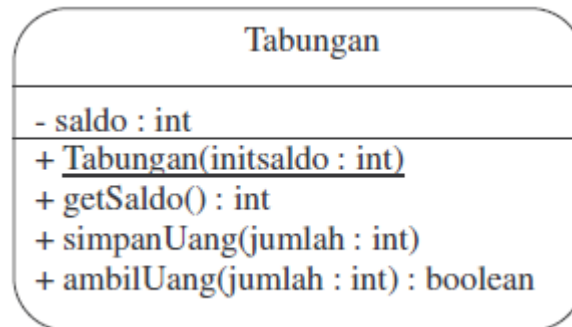
```
public class TesLatihan3{
    public static void main(String args[]){
        Truk truk = new Truk(1000);
        System.out.println("Muatan maksimal = "+truk.getMuatanMaks());
        truk.tambahMuatan(500.0);
        System.out.println("Tambah muatan : 500 ");
        truk.tambahMuatan(350.0);
        System.out.println("Tambah muatan : 350 ");
        truk.tambahMuatan(100.0);
        System.out.println("Tambah muatan : 100 ");
        truk.tambahMuatan(150.0);
        System.out.println("Tambah muatan : 150 ");
        System.out.println("Muatan sekarang = " + truk.getMuatan());
    }
}
```

Lakukan kompilasi pada program diatas dan jalankan. Jika tampilan di layar tampak seperti dibawah ini, maka program anda sudah benar. Jika tidak sama, benahi kembali program anda dan lakukan hal yang sama seperti diatas.

```
Muatan maksimal : 1000.0
Tambah muatan : 500
Tambah muatan : 350
Tambah muatan : 100
Tambah muatan : 150
Muatan sekarang = 950.0
```

## F. TUGAS

### Tugas 1 : Mengimplementasikan UML class diagram dalam program untuk class Tabungan



Transformasikan class diagram diatas ke dalam bentuk program! Tulislah listing program berikut ini sebagai pengetesan.

```
public class TesTugas1 {
    public static void main (String srt[]){
        boolean status;
        Tabungan tabungan = new Tabungan(5000);
        System.out.println("Saldo awal :"+tabungan.getSaldo());

        tabungan.simpanUang(3000);
        System.out.println("Jumlah uang yang disimpan : 3000");

        status = tabungan.ambilUang(6000);
        System.out.println("Jumlah uang yang diambil : 6000");
        if (status)
            System.out.println("Ok");
        else
            System.out.println("Gagal");

        tabungan.simpanUang(3500);
        System.out.println("Jumlah uang yang disimpan :3500");

        status = tabungan.ambilUang(4000);
        System.out.println("Jumlah uang yang diambil : 4000");
    }
}
```



```

    if (status)
        System.out.println("Ok");
    else
        System.out.println("Gagal");

    status = tabungan.ambilUang(1600);
    System.out.println("Jumlah uang yang diambil : 1600");
    if (status)
        System.out.println("Ok");
    else
        System.out.println("Gagal");

    tabungan.simpanUang(2000);
    System.out.println("Jumlah uang yang disimpan :2000");
    System.out.println("Saldo sekarang = " + tabungan.getSaldo());
}
}

```

Lakukan kompilasi pada program diatas dan jalankan. Jika tampilan di layar tampak seperti dibawah ini, maka program anda sudah benar. Jika tidak sama, benahi kembali program anda dan lakukan hal yang sama seperti diatas.

```

Saldo awal : 5000
Jumlah uang yang disimpan : 3000
Jumlah uang yang diambil : 6000    ok
Jumlah uang yang disimpan : 3500
Jumlah uang yang diambil : 4000    ok
Jumlah uang yang diambil : 1600    gagal
Jumlah uang yang disimpan : 2000
Saldo sekarang = 3500

```

## **Tugas 2 : Menganalisa, membuat UML class diagram dan implementasi program**

Seorang pengusaha rental mobil kesulitan mengingat armada kendaraan yang dimilikinya. Oleh karena itu pengusaha tersebut menugaskan pegawainya untuk mengidentifikasi tersebut. Hasil identifikasi dicatat dalam suatu table sebagaimana bisa

dilihat pada Tabel 7.1. Sayangnya karena merupakan pegawai baru maka ia tidak memahami nama hal yang diidentifikasi (A,B,C, D, dan E).

- a. Bantulah pegawai tersebut dalam menentukan nama hal yang diidentifikasi (A,B,C, D, dan E).
- b. Bantulah pengusaha tersebut dalam membuat UML class diagram Mobil. Tambahkan method infoMobil() yang bertujuan untuk menampilkan semua karakteristik mobil (A,B,C, D, dan E).
- c. Buatlah kelas Mobil.java yang mengimplementasikan desain UML class diagram anda!
- d. Buatlah kelas TesMobil.java yang berisi pembuatan 4 (empat) buah obyek bernama mobil1, mobil2, mobil3, mobil4. Mengeset karakteristik masing-masing dan menampilkan info karakteristik mobil.

Tabel 7.1. Data karakteristik mobil

Obyek	A	B	C	D	E
mobil1	Toyota	Biru	minibus	2000	7
mobil2	Daihatsu	Hitam	pick up	1500	2
mobil3	Suzuki	Silver	suv	1800	5
mobil4	Honda	Merah	sedan	1300	5

## H. LAPORAN RESMI

Kumpulkan hasil latihan dan tugas di atas. Tambahkan analisa dalam laporan resmi.