

Praktikum

Pengenalan pemrograman berbasis obyek

Pokok Bahasan

- Deklarasi class
- Deklarasi atribut
- Deklarasi metode
- Pengaksesan anggota obyek

Tujuan Belajar

Dengan praktikum ini mahasiswa diharapkan dapat:

- Mendeklarasikan suatu class
- Mendeklarasikan suatu atribut
- Mendeklarasikan suatu metode
- Mengakses anggota suatu obyek

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

```

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;
    }

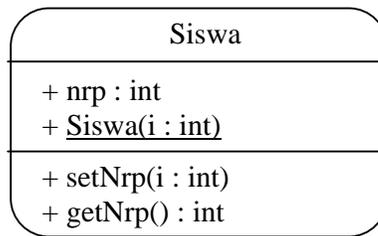
    public void setNama(String i) {
        nama=i;
    }
}

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

```

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:

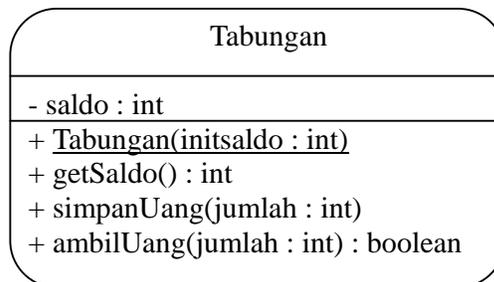
```
public class Siswa {
    public int nrp;

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

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

Tugas

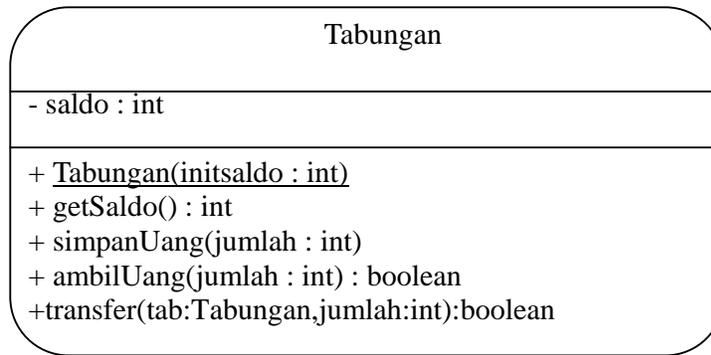
Tugas 1 : Mengimplementasikan UML class diagram dalam program untuk class Tabungan



Transformasikan class diagram diatas ke dalam bentuk program? Jalankan file TesTugas1.class. 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 : Menambahkan pada class Tabungan fungsi transfer().



***** Selamat Mengerjakan*****