

**Jurusan Teknik Informatika**

# **[Praktikum PBO]**

**[Modul 3]**

# Pengaksesan Member dan Access Modifier

## Membuat instance dari suatu class (object)

Untuk dapat membuat sebuah object dari suatu class harus ditentukan terlebih dahulu variable yang akan menampungnya. Pembuatan object ini hanya dapat dilakukan dalam sebuah method. Syntax untuk membuat objek adalah sbb:

```
namaClass namaVariable = new namaClass();
```

Contoh:

```
Nasabah x = new Nasabah();
```

## Pengaksesan Member

Setelah objek dibuat, barulah member (field/atribut dan method) dari objek tersebut dapat diakses. Syntax pengaksesan member adalah sebagai berikut:

```
// mengakses nilai suatu field
namaVariable.namaField

// memberi nilai suatu field
namaVariable.namaField = value;

// memanggil method tanpa parameter
namaVariable.namaMethod()

// memanggil method dengan parameter
namaVariable.namaMethod(param1, param2)
```

Contoh:

```
// Membuat objek
Nasabah a = new Nasabah();
Rekening b = new Rekening();
Kalkulator c = new Kalkulator();

// Memberi nilai pada field
a.nama = "Budi";

// Mengakses field
System.out.println("Nama: " + a.nama);

// Memanggil method tanpa parameter
String status = b.status();
System.out.println(status);
System.out.println(b.status());

// Memanggil method dengan parameter
int hasil = c.tambah(3, 2);
System.out.println("Hasil: " + hasil);
System.out.println("Hasil: " + c.tambah(4, 3));
```

## Access Modifier

Access modifier menentukan apakah class lain dapat menggunakan field atau memanggil method tertentu dari suatu class. Ada dua level access modifier:

- Level class (public atau package-private)
  - public
  - package-private
- Level member
  - public
  - protected
  - private
  - package-private

Contoh level class

```
// class dengan access modifier public
public class Nasabah {
}

// class dengan access modifier package-private
class Nasabah {
}
```

Contoh level member

```
public class Contoh {

    public String nama;
    protected int umur;
    private boolean menikah;
    String tempatLahir;
}
```

#### Deskripsi access modifier

- **private** hanya dapat diakses dari class-nya sendiri
- **protected** dapat diakses dari:
  - Class-nya sendiri
  - Class yang berada dalam satu package
  - Class turunannya
- **package-private** (tanpa modifier) dapat diakses dari:
  - Class-nya sendiri
  - Class yang berada dalam satu package
- **public** dapat diakses dari manapun

Modifier	Notasi UML	Class	Package	Subclass	World
public	+	Y	Y	Y	Y
protected	#	Y	Y	Y	N
<i>no modifier</i>	~	Y	Y	N	N
private	-	Y	N	N	N

## this

Dalam setiap class Java menyediakan variable khusus yaitu **this**. Variable ini digunakan untuk mengakses member yang dimiliki oleh objek itu sendiri. Pengaksesan melalui variable **this** tidak akan terkena aturan access modifier (karena mengakses member miliknya sendiri)

```
package contoh;

public class Satu {

    public String nama;

    public void dummy() {
        // mengakses member sendiri
        this.nama = "hello";

        // mengakses member dari objek lain
        // namun dari class yang sama
        Satu x = new Satu();
        x.nama = "world";

        System.out.println("Nama saya: " + this.nama);
        System.out.println("Nama dia: " + x.nama);
    }
}
```

```
package contoh;

public class Main {

    public static void main(String[] args) {
        Satu a = new Satu();
        a.dummy();
    }
}
```

# Tugas Praktikum

1. Buat class kucing dengan ketentuan
  - a) memiliki variabel private nama kucing, dan tanggal lahir (tgl - bulan - tahun).
  - b) untuk men-set nilai variabel private, harus di lakukan dari method.
  - c) Buat method untuk menghitung usia kucing
2. Buat class taxi untuk menghitung tarif taxi, dengan ketentuan
  - a) Dari object yang dibuat, tidak boleh akses langsung ke variabel, harus melalui metod.
  - b) Di catat nama penumpang, nama driver, tgl, jarak tempuh
  - c) Biaya
    - i. Start : 5000
    - ii. 0 - 10km, tarif 3500/km
    - iii. 10 - 20, 35000 + 2500/km
    - iv. >20 , (35000 + 25000) + 2000/km

## Referensi

- Objects First with Java - A Practical Introduction using BlueJ - , David Barnes & Michael Kolling, Prentice Hall, 2002
- Head First JAVA, 2nd Edition, Kathy Sierra & Bert Bates, O'Reilly
- Java 6 in 21 Days, Rogers Cadenhead & Laura Lemay, SAMS, 2007