



Class dan Object

PEMROGRAMAN BERORIENTASI OBJEK

JURUSAN TEKNOLOGI INFORMASI

POLITEKNIK NEGERI MALANG

Class

- **Class** adalah prototype atau template dari objek.
- Sebuah **class** mempunyai anggota yang terdiri dari **atribut** dan **method**.
- **Atribut** adalah semua field identitas yang kita berikan pada suatu class, exp :
 - Class Mahasiswa : - Nim , nama , alamat.
- **Method** dapat kita artikan sebagai semua fungsi ataupun prosedur yang merupakan perilaku (behaviour) dari suatu class.

<i>NamaClass</i>
<i>namaAtribut1: tipeData</i>
<i>namaAtribut2: tipeData</i>
<i>namaMethod1(): tipeData</i>
<i>namaMethod2(): tipeData</i>



Mahasiswa
+nim: int
+nama: String
+alamat: String
+kelas: String
+telepon: int
+tampilBiodata(): void

nama class

daftar atribut

daftar method

daftar method

Aturan penulisan class

- Berupa kata benda,
- Diawali dengan **HURUF BESAR**,
- Jika terdiri dari lebih dari 1 kata, maka antar kata satu dengan kata yang lain **disambungkan**, dan tiap huruf awal dari tiap kata menggunakan **HURUF BESAR**.

Implementasi Class

```
<modifier> class <nama_class> {  
  
    //deklarasi atribut dan method  
  
}
```

- Untuk membuat suatu Class, digunakan kata kunci class dan diikuti dengan nama Class yang akan dibuat, exp:

```
public class Mahasiswa{  
  
}
```

Latihan 1

- Jurusan Teknologi Informasi akan membangun system informasi pemberian nilai secara online kepada setiap mahasiswa yang menempuh perkuliahan. Setiap nilai mata kuliah yang akan dimasukkan dalam system di inputkan secara langsung oleh dosen yang bersangkutan. Dari system ini diharapkan mahasiswa akan lebih mudah untuk melakukan evaluasi terhadap setiap matakuliah yang sudah pernah diikuti.
- 1) Identifikasi, ada berapa class yang bisa dibuat dari sistem diatas ?
 - 2) Implementasikan class dalam bahasa pemrograman java.

Aturan penulisan atribut

- Aturan penulisan atribut adalah sebagai berikut:
- Berupa kata benda,
- Diawali dengan **HURUF KECIL**,
- Jika terdiri dari 2 atau lebih kata, kata pertama diawali **HURUF KECIL** sedangkan kata selanjutnya diawali **HURUF BESAR**. Dan antar kata disambung tidak (dipisah).

Deklarasi Atribut

- Untuk melakukan pendeklarasian atribut dapat dilakukan dengan sintaks sebagai berikut :

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

- Exp :

```
public int nim;  
public String nama;  
public String alamat;
```


Latihan 2

- Tambahkan secara lengkap attribute pada setiap class yang sudah anda identifikasi pada latihan sebelumnya (Latihan 1)?

Tipe Method

- Method dengan tipe data void, berarti tidak memiliki nilai balik, berarti tidak memerlukan kata kunci return di dalamnya.
- Method dengan tipe data bukan void, berarti memerlukan suatu nilai balik, yaitu harus memerlukan return di dalamnya

```
public void sayHello(){  
    System.out.println("Hello World!!");  
}
```

TIDAK PERLU
RETURN /
TIDAK ADA NILAI
KEMBALIAN

```
public int tambah (int a, int b){  
    int hasil = a+b;  
    return hasil;  
}
```

- TIPE DATA METHOD INT, BERARTI METHOD TSB HARUS MENGEMBALIKAN NILAI INT
- HARUS ADA RETURN

return hasil;

- HARUS ADA RETURN

- Deklarasi method dapat dilakukan dengan sintaks sebagai berikut :

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

Aturan penulisan method

- Aturan penulisan method adalah sebagai berikut:
- Berupa kata kerja,
- Diawali dengan **HURUF KECIL**,
- Jika terdiri dari 2 atau lebih kata, kata pertama diawali **HURUF KECIL** sedangkan kata selanjutnya diawali **HURUF BESAR**. Dan antar kata disambung tidak (dipisah).

Deklarasi method

```
public void tampil(){  
    System.out.println ("Halo PBO!!");  
}
```

```
public int tambah(int a, int b){  
    return a+b;  
}
```

Object

- Object adalah instansiasi dari sebuah class , exp :

```
NamaClass namaObject = new NamaClass();
```


- Mahasiswa mhs = new Mahasiswa();

Implementasi Object

```
public class Mahasiswa{  
    public int nim;  
    public int nama;  
    public int alamat;  
  
    public void biodataMahasiswa(){  
        System.out.println("Biodata Mahasiswa");  
        System.out.println("Nim          :"+nim);  
        System.out.println("Nama          :"+nama);  
        System.out.println("Alamat        :"+alamat);  
    }  
}
```


The background of the slide features decorative curved lines in the top-left and bottom-right corners. On the left side, there are several concentric, slightly curved lines, some solid and some dashed, in a light gray color. On the right side, there are similar curved lines, also in light gray, creating a sense of depth and movement.

```
public class TampilMahasiswa{
```


A large, solid orange rectangular block is positioned on the left side of the slide. It has a small triangular pointer at the bottom center, pointing downwards. This block appears to be a placeholder or a design element.

```
    public static void main(String[] args){  
        Mahasiswa mhs = new Mahasiswa();  
        mhs.nim = 1;  
        mhs.nama = "Very Sugiarto";  
        mhs.alamat = "Malang";  
        mhs.biodataMahasiswa();  
    }
```

```
}
```

Try – catch

- Untuk menangani ***error*** di Java, digunakan sebuah statement yang bernama *try - catch*.
- Statement tersebut digunakan untuk mengurung eksekusi yang menampilkan *error* dan dapat membuat program tetap berjalan tanpa dihentikan secara langsung.
- *Error* yang ditangani oleh *try – catch* biasa disebut dengan ***exception***.

- 
- Ada beberapa hal yang perlu diingat ketika akan menggunakan try - catch di Java:

- 1) Kita dapat membuat multiple try-catch,
- 2) Kita dapat menambahkan statement finally untuk menangani berbagai hal ketika error terjadi atau tidak,
- 3) Kita dapat membuat exception sendiri disamping menggunakan bawaan Java.

Latihan 3

- Buatlah implementasi dengan menggunakan bahasa pemrograman java dari soal berikut :
- Class **Persegi** dengan atribut **sisi** dengan tipe data integer.
- Terdapat tiga method :
 - ❑ Method data persegi untuk menampilkan data panjang sisi dari persegi tersebut.
 - ❑ Method luas persegi untuk menghitung luas dari persegi tersebut dengan rumus : $\text{sisi} \times \text{sisi}$.
 - ❑ Method keliling persegi untuk menghitung keliling dengan rumus : $4 \times \text{sisi}$.
- Tampilkan data persegi, nilai luas persegi dan keliling persegi pada class **TampilPersegi**.