

KUSUMA WEB ACADEMY

DASAR-DASAR JALUR JAVA

Modul 10: Inisialisasi Objek - Konstruktor & Keyword 'this'

Hak Cipta © 2026 Kusuma Web. All Rights Reserved.

Apa itu Konstruktor? Karakteristik & Fungsi Utama

Definisi & Peran

- ▶ Method khusus yang secara otomatis dipanggil/dijalankan tepat pada saat instansiasi objek (new).
- ▶ Digunakan untuk memberikan nilai awal (*setup/inisialisasi state*) pada atribut objek yang baru lahir.

Aturan Wajib Konstruktor

- ▶ Nama konstruktor **wajib sama persis** dengan nama Class-nya (termasuk huruf kapital).
- ▶ Konstruktor **tidak memiliki tipe return** apa pun (bahkan tidak menggunakan keyword void).

Konstruktor Default

Jika Anda tidak menulis konstruktor sama sekali, compiler Java otomatis menyisipkan satu konstruktor kosong tanpa parameter di balik layar (*default constructor*).

Jenis Konstruktor: Default vs Parameterized

Konstruktor dengan Parameter

- ▶ Memungkinkan programmer mengirimkan argumen data khusus di dalam tanda kurung objek saat diinstansiasi.

```
public class Laptop {
    String merk;

    // Parameterized Constructor
    public Laptop(String merkLaptop) {
        merk = merkLaptop;
    }
}
```

Cara Menggunakan Konstruktor

```
public class Main {
    public static void main(String[] args
    ) {
        // Objek diinisialisasi langsung
        dengan nilai awal
        Laptop lap1 = new Laptop("ASUS
        ROG");
        Laptop lap2 = new Laptop("MacBook
        M3");

        System.out.println(lap1.merk); //
        "ASUS ROG"
    }
}
```

Memahami Fungsi Keyword `this` pada Java

Mengapa Butuh Keyword `this`?

- ▶ Menjadi referensi khusus yang merujuk pada objek instansi saat ini yang sedang aktif berjalan.
- ▶ **Mengatasi Variable Shadowing:** Digunakan untuk membedakan secara tegas antara variabel instansi kelas (*instance variable*) dengan variabel parameter method/konstruktur yang memiliki nama identik.

Implementasi Menghindari Shadowing

```
public class Karyawan {  
    String nama; // Instance variable  
  
    // Nama parameter sama dengan nama instansi  
    public Karyawan(String nama) {  
        // this.nama merujuk ke instance variable  
        // nama (kanan) merujuk ke parameter lokal  
        this.nama = nama;  
    }  
}
```

Constructor Chaining: Memanggil Konstruktor Lain

Konsep Constructor Chaining

- ▶ Sebuah teknik di mana satu konstruktor memanggil konstruktor lain di dalam kelas yang sama menggunakan sintaks `this()`.
- ▶ Berguna untuk menghindari penulisan ulang kode inisialisasi yang berulang-ulang (*redundant*).

Aturan Sintaks Mutlak!

Panggilan `this()` harus selalu diposisikan pada ****baris pertama**** di dalam tubuh konstruktor yang memanggil.

Implementasi Chaining di Java

```
public class Produk {
    String nama;
    int harga;

    // Konstruktor 1 (Penuh)
    public Produk(String nama, int
        harga) {
        this.nama = nama;
        this.harga = harga;
    }

    // Konstruktor 2 (Harga Default
    1000)
    public Produk(String nama) {
        this(nama, 1000); // Memanggil
        Konstruktor 1
    }
}
```