

PRAKTIKUM PARADIGMA PEMROGRAMAN
MODUL 4



Disusun oleh :

Nama : Fidelia Ping

NIM : 245410012

Kelas : Informatika 1

PROGRAM STUDI INFORMATIKA
PROGRAM SARJANA
FAKULTAS TEKNOLOGI INFORMASI
UNIVERSITAS TEKNOLOGI DIGITAL INDONESIA
YOGYAKARTA
2025

MODUL 4

KONSTRUKTOR

A. TUJUAN PRAKTIKUM

Dapat membuat dan menggunakan konstruktor

B. PEMBAHASAN LISTING

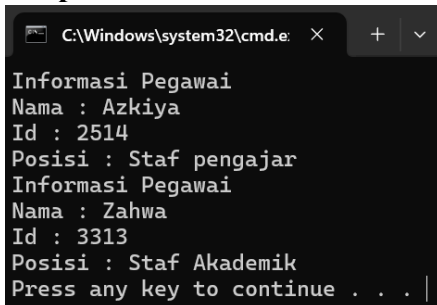
Praktik

```
public class Pegawai
{
    String NamaPegawai;
    int IdPegawai;
    String PosisiPegawai;
    Pegawai (String nama,int Id, String posisi) // konstruktor
    {
        NamaPegawai = nama;
        IdPegawai = Id;
        PosisiPegawai = posisi;
    }
    Pegawai () //konstruktor
    {
        NamaPegawai = "Azkiya";
        IdPegawai = 2514;
        PosisiPegawai = "Staf pengajar";
    }
    void Show()
    {
        System.out.println("Informasi Pegawai");
        System.out.println("Nama : "+NamaPegawai);
        System.out.println("Id : "+IdPegawai);
        System.out.println("Posisi : "+PosisiPegawai);
    }
    public static void main(String args[])
    {
        Pegawai pegawai1 = new Pegawai();
        Pegawai pegawai2 = new Pegawai("Zahwa",3313,"Staf Akademik");
        pegawai1.Show();
        pegawai2.Show();
    }
}
```

Pembahasan program : pada program ini terdapat sebuah class `Pegawai` yang memiliki 3 atribut yaitu `NamaPegawai (String)`, `IdPegawai (int)`, `PosisiPegawai (String)`. Untuk menginisialisasi nilai dari atribut-atribut tersebut, digunakan konstruktor. Konstruktor adalah metode khusus yang akan dijalankan secara otomatis ketika sebuah objek dibuat dan digunakan untuk memberikan nilai awal pada atribut. Pada baris 12-17 konstruktor ini tidak memiliki parameter, sehingga nilai atribut akan otomatis diisi dengan nilai bawaan (default) yang sudah ditentukan dalam konstruktor. Objek yang dibuat menggunakan konstruktor terdapat pada baris 25. Pada baris 5-10 konstruktor ini memiliki parameter, sehingga nilai atribut dapat ditentukan sendiri saat membuat objek. Objek yang dibuat menggunakan konstruktor terdapat pada baris 26. Program ini menunjukkan bahwa konstruktor dapat dibuat lebih dari satu dalam satu kelas selama memiliki perbedaan jumlah atau tipe parameter. Hal ini memberikan

fleksibilitas dalam pembuatan objek, apakah ingin menggunakan nilai default atau memberikan nilai sendiri.

Output



```
C:\Windows\system32\cmd.e
Informasi Pegawai
Nama : Azkiya
Id : 2514
Posisi : Staf pengajar
Informasi Pegawai
Nama : Zahwa
Id : 3313
Posisi : Staf Akademik
Press any key to continue . . . |
```

Pembahasan output : Karena pegawai1(Azkiya) menggunakan konstruktor tanpa parameter, maka nilai atribut menggunakan nilai default dari konstruktor tersebut. Karena pegawai2(Zahwa) menggunakan konstruktor dengan parameter, nilai atribut diambil dari data yang dikirim saat objek dibuat.

Latihan

```
import java.util.Scanner;
class Buku {
    private String judul;
    private String penulis;
    private int tahunTerbit;
    // Konstruktor
    public Buku(String judul, String penulis, int tahunTerbit) {
        this.judul = judul;
        this.penulis = penulis;
        this.tahunTerbit = tahunTerbit;
    }
    // Method untuk menampilkan informasi buku
    public void tampilkanInfo() {
        System.out.println("Judul: " + judul);
        System.out.println("Penulis: " + penulis);
        System.out.println("Tahun Terbit: " + tahunTerbit);
    }
    // Getter dan Setter
    public String getJudul() {
        return judul;
    }
    public void setJudul(String judul) {
        this.judul = judul;
    }
    public String getPenulis() {
        return penulis;
    }
    public void setPenulis(String penulis) {
        this.penulis = penulis;
    }
    public int getTahunTerbit() {
        return tahunTerbit;
    }
    public void setTahunTerbit(int tahunTerbit) {
        this.tahunTerbit = tahunTerbit;
    }
}
public class Data_Buku {
    public static void main(String[] args) {
```

```

Scanner scanner = new Scanner(System.in);
System.out.print("Masukkan judul buku: ");
String judul = scanner.nextLine();
System.out.print("Masukkan penulis buku: ");
String penulis = scanner.nextLine();
System.out.print("Masukkan tahun terbit buku: ");
int tahunTerbit = scanner.nextInt();
scanner.nextLine(); // Consume newline left-over

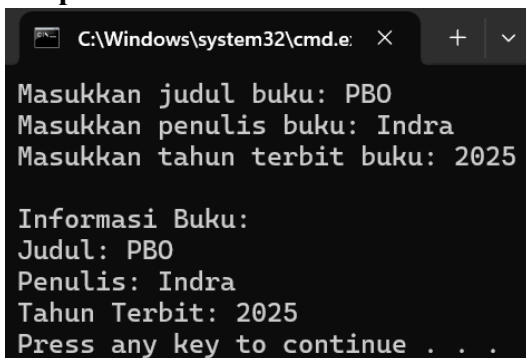
// Membuat objek buku
Buku buku = new Buku(judul, penulis, tahunTerbit);

// Menampilkan informasi buku
System.out.println("\nInformasi Buku:");
buku.tampilkanInfo();

```

Pembahasan program : pada program ini menggunakan class Buku untuk menyimpan data sebuah buku. Kelas ini memiliki tiga atribut yang dibuat private judul (String), penulis (String), tahunTerbit (int). pada baris 7-11 konstruktor ini akan dijalankan otomatis saat objek Buku dibuat dan berfungsi untuk memberikan nilai awal pada atribut objek berdasarkan input pengguna. Objek dibuat dengan pada baris 49, nilai atribut judul, penulis, dan tahunTerbit diberikan dari hasil input scanner di main(). Selain itu, class ini memiliki method tampilkanInfo() untuk mencetak detail buku, dan getter dan setter untuk mengubah dan mengambil nilai atribut jika dibutuhkan. Program ini menunjukkan penerapan konstruktor dengan parameter yang berguna untuk memberikan nilai awal pada objek saat dibuat, membaca data buku secara dinamis dari input pengguna, menampilkan informasi lengkap dari objek menggunakan method.

Output



```

C:\Windows\system32\cmd.e: x + v
Masukkan judul buku: PBO
Masukkan penulis buku: Indra
Masukkan tahun terbit buku: 2025

Informasi Buku:
Judul: PBO
Penulis: Indra
Tahun Terbit: 2025
Press any key to continue . . .

```

Pembahasan output : program ini meminta user untuk memasukkan infoemasi sebuah buku, yaitu : judul buku, penulis buku, dan tahun terbit buku. Setelah user memasukkan yang diminta oleh program, maka program akan menampilkan kembali hasil input dalam format informasi buku.

C. PEMBAHASAN TUGAS

Tugas

```

import java.util.Scanner;
class Komputer {
    private String Merk;
    private String Type;
    private int RAM;
    private int RAM_HD;
    private String Prosesor;

```

```

// Konstruktor
public Komputer(String Merk, String Type, int RAM, int RAM_HD, String
    Prosesor) {
    this.Merk = Merk;
    this.Type = Type;
    this.RAM = RAM;
    this.RAM_HD = RAM_HD;
    this.Prosesor = Prosesor;
}
// Method untuk menampilkan informasi komputer
public void tampilkanInfo() {
    System.out.println("Merk: " + Merk);
    System.out.println("Type: " + Type);
    System.out.println("RAM Komputer: " + RAM);
    System.out.println("RAM HD: " + RAM_HD);
    System.out.println("Prosesor Komputer: " + Prosesor);
}
// Getter dan Setter
public String getMerk() {
    return Merk;
}
public void setMerk(String Merk) {
    this.Merk = Merk;
}
public String getType() {
    return Type;
}
public void setType(String Type) {
    this.Type = Type;
}
public int getRAM() {
    return RAM;
}
public void setRAM(int RAM) {
    this.RAM = RAM;
}
public int getRAM_HD() {
    return RAM_HD;
}
public void setRAM_HD(int RAM_HD) {
    this.RAM_HD = RAM_HD;
}
public String getProsesor() {
    return Prosesor;
}
public void setProsesor(String Prosesor) {
    this.Prosesor = Prosesor;
}
}
public class KomputerApp {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.print("Masukkan Merk Komputer: ");
        String Merk = scanner.nextLine();
        System.out.print("Masukkan Type: ");
        String Type = scanner.nextLine();
        System.out.print("Masukkan RAM Komputer: ");
        int RAM = scanner.nextInt();
    }
}

```

```

        System.out.print("Masukkan RAM Hardisk Komputer: ");
        int RAM_HD = scanner.nextInt();
        scanner.nextLine(); // consume newline
        System.out.print("Masukkan Prosesor Komputer: ");
        String Prosesor = scanner.nextLine();
        // Membuat objek komputer
        Komputer komputer = new Komputer(Merk, Type, RAM, RAM_HD, Prosesor);
        // Menampilkan informasi komputer
        System.out.println("\nInformasi Komputer:");
        komputer.tampilkanInfo();
    }

```

Pembahasan program : program ini terdiri dari 2 class Komputer dan KomputerApp. Class ini digunakan untuk menyimpan data komputer dalam bentuk objek dengan atribut : Merk (String), Type (String), RAM (int), RAM_HD (int), dan Prosesor (String). Semua atribut menggunakan modifier private, sehingga tidak dapat diakses langsung dari luar kelas dan harus melalui method. Pada baris 9-15 Fungsi konstruktor dalam program ini digunakan untuk memberikan **nilai awal** pada atribut objek, nilai yang diberikan berasal dari input pengguna, dan akan otomatis dipanggil saat objek dibuat. Method **tampilkanInfo()** method ini bertugas untuk menampilkan seluruh data komputer yang sudah tersimpan di dalam objek. Getter dan setter disediakan agar atribut yang bersifat private tetap bisa diakses melalui getter, diubah melalui ini menunjukkan penerapan enkapsulasi dalam OOP. Kelas KomputerApp, kelas ini berfungsi menerima input dari user dan memprosesnya, program akan meminta user untuk memasukkan merk, type, RAM, kapasitas HD, dan prosesor. Kemudian data tersebut tersimpan dalam variabel local, dan data dikirim ke konstruktor untuk membuat objek pada baris ke 71. Kemudian program akan menampilkan data tersebut melalui data tersebut method pada baris 74. Program ini menunjukkan bagaimana konstruktor berperan penting dalam OOP untuk memberikan nilai awal pada sebuah objek menggunakan input dari pengguna, lalu informasi tersebut dapat ditampilkan kembali melalui sebuah method.

Output

```

Masukkan Merk Komputer: ASUS
Masukkan Type: Vivobook
Masukkan RAM Komputer: 8
Masukkan RAM Hardisk Komputer: 512
Masukkan Prosesor Komputer: IntelCOREi5

Informasi Komputer:
Merk: ASUS
Type: Vivobook
RAM Komputer: 8
RAM HD: 512
Prosesor Komputer: IntelCOREi5
Press any key to continue . . . |

```

Pembahasan output : pada program ini user diminta untuk memasukkan beberapa informasi tentang komputer, yaitu : merk komputer, tipe komputer, kpasitas RAM, kapasitas penyimpanan RAM HD, dan prosesor komputer. Setelah user memasukkan data yang diminta, maka program akan menampilkan Kembali informasi komputer tersebut.