

LAPORAN PRAKTIKUM
PERTEMUAN 8
“MODUL 9: METHOD TANPA NILAI KEMBALIAN”



Nama :

Muhammad Farel Alghazali (109092500033)

Asisten Praktikum :

Azaria Nanda Putri (2311102147)

Arvinanto Bahtiar (2211110014)

PROGRAM STUDI S1 REKAYASA PERANGKAT LUNAK
FAKULTAS INFORMATIKA
TELKOM UNIVERSITY PURWOKERTO

2025

I. DASAR TEORI

Pada saat membuat program, kita terkadang membuat kode yang sama yang kita gunakan berulang-ulang untuk keperluan yang sama. Misal kita ingin melakukan penjumlahan semua data nilai & penjumlahan semua data jumlah izin. Biasanya kita akan menuliskannya satu persatu, untuk nilai & izin, tapi hal ini bisa dibuat lebih ringkas jika kita menggunakan fungsi / method. Fungsi / method adalah sebuah potongan blok kode yang dapat dijalankan berulang kali pada blok kode lain. Pada bahasa pemrograman Java terdapat 2 jenis method yang bisa digunakan, yaitu method tanpa nilai kembalian dimana ia hanya menjalankan sebuah perintah dalam blok kodennya dan method dengan nilai kembalian dimana ia akan mengembalikan hasil proses dari kode yang dijalankan. Pada kali ini kita akan memahami lebih dalam tentang method tanpa nilai kembalian terlebih dahulu

II. GUIDE

1.) Tampilkan Identitas

a. Kode



```
1 public class tampilIdentitas{
2     void tampilkanIdentitas(){
3         System.out.println("Nama : Muhammad Farel Alghazali");
4         System.out.println("NIM : 109092500033");
5         System.out.println("Program Studi : Rekayasa Perangkat Lunak");
6     }
7
8     public static void main(String[] args) {
9         tampilIdentitas obj = new tampilIdentitas();
10        obj.tampilkanIdentitas();
11    }
12 }
```

b. Output

```
Nama : Muhammad Farel Alghazali
NIM : 109092500033
Program Studi : Rekayasa Perangkat Lunak
```

c. Penjelasan

i. Inisiasi method

Kita buat method bernama **tampilkanIdentitas** sebelum menuliskan fungsi **void main**, tujuannya agar method dapat dipanggil pada fungsi main nanti, karena program dibaca dari atas ke bawah.

ii. Tulis perintah dalam method

Kita akan mencetak beberapa baris kalimat pada layar, yaitu data nama, NIM, & prodi.

iii. Inisiasi method utama / main

Setelah selesai, kita buat method **main** sebagai method inti dalam program

iv. Membuat objek dari kelas

Untuk dapat menggunakan method yang sudah dibuat, kita perlu buat objek dari class utama kita yaitu **tampilIdentitas**.

v. Memanggil method dari objek

Terakhir kita panggil method yang sudah dibuat yang sudah kita inisiasikan ke dalam objek, dan kode akan dijalankan.

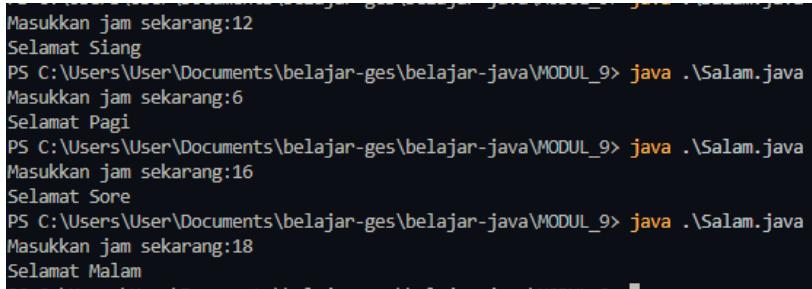
2.) Salam Waktu

a. Kode



```
1 import java.util.Scanner;
2
3 public class Salam{
4     void salamWaktu(int jam){
5         if(jam >= 0 && jam <= 10){
6             System.out.println("Selamat Pagi");
7         } else if(jam >= 11 && jam <= 14){
8             System.out.println("Selamat Siang");
9         } else if(jam >= 15 && jam < 18){
10            System.out.println("Selamat Sore");
11        }else if(jam >= 18){
12            System.out.println("Selamat Malam");
13        }
14    }
15
16    public static void main(String[] args){
17        Scanner input = new Scanner(System.in);
18        Salam obj = new Salam();
19
20        System.out.print("Masukkan jam sekarang:");
21        int jam = input.nextInt();
22
23        obj.salamWaktu(jam);
24    }
25 }
```

b. Output



```
Masukkan jam sekarang:12
Selamat Siang
PS C:\Users\User\Documents\belajar-ges\belajar-java\MODUL_9> java .\Salam.java
Masukkan jam sekarang:6
Selamat Pagi
PS C:\Users\User\Documents\belajar-ges\belajar-java\MODUL_9> java .\Salam.java
Masukkan jam sekarang:16
Selamat Sore
PS C:\Users\User\Documents\belajar-ges\belajar-java\MODUL_9> java .\Salam.java
Masukkan jam sekarang:18
Selamat Malam
```

c. Penjelasan

i. Import library

Panggil library scanner untuk bisa membaca inputan user

ii. Inisiasi method

Buat method **salamWaktu**, isikan parameter dengan variabel **jam** bertipe integer.

iii. Cek nilai jam

Kita cek nilai jam:

0–10: “Selamat Pagi”

11–14: “Selamat Siang”

15–18: “Selamat Sore”

18: “Selamat Malam”

iv. Inisiasi method main

Buat method **main** sebagai method utama dalam program.

v. **Buat objek dari class utama**

Untuk bisa menjalankan method yang dibuat, buat objek dari class utama

vi. **Terima input user**

Terima input user berupa variabel **jam** bertipe integer.

vii. **Panggil objek disertai nilai parameter**

Panggil method **salamWaktu** dari objek class lalu isi parameter dengan **jam**, maka akan muncul output sesuai **jam** yang diinputkan.

III. UNGUIDED

1. Show Student

- Kode



```
1 import java.util.Scanner;
2
3 public class contohVoid{
4     void Siswa(String nama, String nomorMatkul, int nilaiBhs, int nilaiBing, int nilaiMtk){
5         double rataRata = (nilaiBhs + nilaiBing + nilaiMtk) / 3;
6         System.out.println("Rata-rata nilai: " + rataRata);
7     }
8     public static void main(String[] args) {
9         Scanner input = new Scanner(System.in);
10        contohVoid obj = new contohVoid();
11
12        System.out.print("Masukkan nama: ");
13        String nama = input.nextLine();
14
15        System.out.print("Masukkan nomor mata kuliah: ");
16        String noMatkul = input.nextLine();
17
18        System.out.print("Masukkan nilai Bahasa: ");
19        int nilaiBhs = input.nextInt();
20
21        System.out.print("Masukkan nilai Bahasa Inggris: ");
22        int nilaiBing = input.nextInt();
23
24        System.out.print("Masukkan nilai Matematika: ");
25        int nilaiMtk = input.nextInt();
26
27        obj.Siswa(nama, noMatkul, nilaiBhs, nilaiBing, nilaiMtk);
28    }
29 }
```

- Output

```
Masukkan nama: Muhammad Farel Alghazali
Masukkan nomor mata kuliah: 007
Masukkan nilai Bahasa: 80
Masukkan nilai Bahasa Inggris: 96
Masukkan nilai Matematika: 70
Rata-rata nilai: 82.0
```

- Penjelasan

- Import library scanner untuk menerima inputan user
- Inisiasi method **Siswa** & isi parameternya dengan nama, nomor matkul, nilai bahasa, nilai bahasa Inggris & Matematika
- Hitung rata-rata semua nilai & cetak ke layar
- Inisiasi method main sebagai method utama
- Buat objek dari class utama agar method bisa digunakan
- Terima input user berupa nama, nomor matkul, nilai bahasa, nilai bahasa Inggris &

matematika

- Panggil method & isi parameter sesuai yang dibutuhkan

IV. KESIMPULAN

Di praktikum kali ini kita telah mengetahui cara untuk membuat sebuah blok kode / method yang terpisah dari fungsi utama yang tidak mengembalikan nilai. Cara ini terbukti efektif dan dapat meningkatkan efisiensi kode, karena kita dapat memisahkan bagian kode dan bisa digunakan berulang di tempat lain. Dengan cara ini juga kode yang dibuat bisa lebih mudah dibaca & dipahami orang lain & juga mesin.