

Nama : Rofiq Aristiyawan

Kelas/Absen : TI-1G/27

## **JOBSHEET 11**

### **Percobaan 1**

1. Nope. Bisa return value juga. "Tanpa parameter" ≠ "tanpa return".
2. Bisa. Tulis langsung main(). Tapi ini membuat kode jadi berantakan dan susah maintainance.
3. keuntungan pakai Menu():
  - Kode lebih rapi.
  - Bisa dipanggil ulang kapan saja.
  - Gampang dimodifikasi, tidak perlu mengedit di banyak tempat.
4. Program start → main() jalan → Menu() dipanggil → Java lompat ke fungsi Menu → tampilkan menu → balik lagi ke main().

### **Percobaan 2**

1. Biar fungsi bisa dynamic, tidak hardcore.
2. namaPelanggan untuk personalisasi, isMember untuk menentukan ada nya diskon atau treatment khusus.
3. Mirip, tapi beda konteks. Parameter itu variable yang diterima dari luar fungsi.
4. kalau true → dapat diskon/member benefits. Kalau false → menu normal.
5. Akan langsung error compile. Java strict soal parameter.

```
6.
package jobsheet11;

public class kafe27 {

    public static void Menu(String namaPelanggan, boolean isMember, String
kodePromo) {
        System.out.println("Selamat datang, " + namaPelanggan + "!");

        if (isMember) {
            System.out.println("Anda adalah member, dapatkan diskon 10% dari
program membership.");
        }
    }
}
```

```

        if (kodePromo.equals("DISKON50")) {
            System.out.println("Kode promo diterapkan! Anda mendapatkan DISKON
50% 🍷");
        } else if (kodePromo.equals("DISKON30")) {
            System.out.println("Kode promo diterapkan! Anda mendapatkan DISKON
30% 🍷");
        } else {
            System.out.println("Kode promo invalid");
        }

        System.out.println("==== MENU RESTO KAFE =====");
        System.out.println("1. Kopi Hitam - Rp 15,000");
        System.out.println("2. Cappuccino - Rp 20,000");
        System.out.println("3. Latte - Rp 22,000");
        System.out.println("4. Teh Tarik - Rp 12,000");
        System.out.println("5. Roti Bakar - Rp 10,000");
        System.out.println("6. Mie Goreng - Rp 18,000");
        System.out.println("=====");
        System.out.println("Silakan pilih menu yang Anda inginkan.");
    }

    public static void main(String[] args) {
        Menu("Andi", true, "DISKON30");
    }
}

```

7. Contoh pemanggilan (Budi, member, DISKON30):

```
Menu("Budi", true, "DISKON30");
```

8. Iya, kode menjadi reusable, clean, dan fleksibel.

### Percobaan 3

1. **Fungsi butuh return value** kalau hasil perhitungannya masih dipakai di luar fungsi.

**Contoh:** `hitungTotalHarga()` harus mengembalikan total harga supaya bisa ditampilkan di main.

**Fungsi tidak butuh return value** kalau hanya menampilkan sesuatu atau menjalankan proses tanpa menghasilkan data.

**Contoh:** `Menu()` hanya mencetak daftar menu, jadi cukup pakai void.

2. **Tipe kembalian:** `int` → karena fungsi mengembalikan total harga.

**Parameter:**

- `pilihanMenu` → nomor menu yang dipilih pelanggan.

- jumlahPesanan → jumlah porsi yang dipesan.

Keduanya dipakai untuk menghitung total harga pesanan.

### 3. modifikasi

```
package jobsheet11;

import java.util.Scanner;

public class kafe27 {

    public static void main(String[] args) {
        Scanner sc = new java.util.Scanner(System.in);

        Menu("Andi", true, "DISKON30");

        System.out.print("\nMasukkan nomor menu yang dipilih: ");
        int pilihanMenu = sc.nextInt();

        System.out.print("Masukkan jumlah item yang dipesan: ");
        int banyakItem = sc.nextInt();

        double totalBayar = hitungTotalHarga(pilihanMenu, banyakItem);

        System.out.println("\nTotal harga pesanan Anda: Rp " + totalBayar);

        sc.close();
    }

    private static double hitungTotalHarga(int pilihanMenu, int banyakItem) {
        throw new UnsupportedOperationException("Unimplemented method 'hitungTotalHarga'");
    }

    public static void Menu(String namaPelanggan, boolean isMember, String kodePromo) {
        System.out.println("Selamat datang, " + namaPelanggan + "!");

        if (isMember) {
            System.out.println("Anda adalah member, dapatkan diskon 10% dari program membership.");
        }

        if (kodePromo.equals("DISKON50")) {
```

```

        System.out.println("Kode promo diterapkan! Anda mendapatkan DISKON
50% 🍷");
    } else if (kodePromo.equals("DISKON30")) {
        System.out.println("Kode promo diterapkan! Anda mendapatkan DISKON
30% 🍷");
    } else {
        System.out.println("Kode promo invalid");
    }

    System.out.println("===== MENU RESTO KAFE =====");
    System.out.println("1. Kopi Hitam - Rp 15,000");
    System.out.println("2. Cappuccino - Rp 20,000");
    System.out.println("3. Latte - Rp 22,000");
    System.out.println("4. Teh Tarik - Rp 12,000");
    System.out.println("5. Roti Bakar - Rp 10,000");
    System.out.println("6. Mie Goreng - Rp 18,000");
    System.out.println("=====");
    System.out.println("Silakan pilih menu yang Anda inginkan.");
}

public static int hitungTotalHargaNoPresensi(int pilihanMenu, int
jumlahPesanan, String kodePromo) {
    int[] hargaItems = {15000, 20000, 22000, 12000, 10000, 18000};

    int totalHarga = hargaItems[pilihanMenu - 1] * jumlahPesanan;

    if (kodePromo.equals("DISKON50")) {
        System.out.println("Diskon 50% diterapkan!");
        totalHarga = totalHarga / 2;
    }
    else if (kodePromo.equals("DISKON30")) {
        System.out.println("Diskon 30% diterapkan!");
        totalHarga = (int)(totalHarga * 0.7);
    }
    else {
        System.out.println("Kode promo invalid (tidak ada diskon).");
    }

    return totalHarga;
}
}

```

#### 4. modifikasi

```
package jobsheet11;

import java.util.Scanner;

public class kafe27 {

    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);

        Menu("Andi", true, "DISKON30");

        System.out.print("\nBerapa jenis menu yang ingin dipesan? ");
        int jumlahMenu = sc.nextInt();

        int totalKeseluruhan = 0;

        for (int i = 1; i <= jumlahMenu; i++) {
            System.out.println("\nPesanan ke-" + i);

            System.out.print("Masukkan nomor menu: ");
            int pilihanMenu = sc.nextInt();

            System.out.print("Masukkan jumlah porsi: ");
            int banyakItem = sc.nextInt();

            int total = hitungTotalHargaNoPresensi(pilihanMenu, banyakItem,
"DISKON30");

            totalKeseluruhan += total;
        }

        System.out.println("\n=====");
        System.out.println("Total keseluruhan pesanan: Rp " + totalKeseluruhan);
        System.out.println("=====");

        sc.close();
    }

    public static void Menu(String namaPelanggan, boolean isMember, String
kodePromo) {
        System.out.println("Selamat datang, " + namaPelanggan + "!");

        if (isMember) {
```

```

        System.out.println("Anda adalah member, dapatkan diskon 10% dari
program membership.");
    }

    if (kodePromo.equals("DISKON50")) {
        System.out.println("Kode promo diterapkan! Anda mendapatkan DISKON
50% 🍷");
    } else if (kodePromo.equals("DISKON30")) {
        System.out.println("Kode promo diterapkan! Anda mendapatkan DISKON
30% 🍷");
    } else {
        System.out.println("Kode promo invalid");
    }

    System.out.println("===== MENU RESTO KAFE =====");
    System.out.println("1. Kopi Hitam - Rp 15,000");
    System.out.println("2. Cappuccino - Rp 20,000");
    System.out.println("3. Latte - Rp 22,000");
    System.out.println("4. Teh Tarik - Rp 12,000");
    System.out.println("5. Roti Bakar - Rp 10,000");
    System.out.println("6. Mie Goreng - Rp 18,000");
    System.out.println("=====");
    System.out.println("Silakan pilih menu yang Anda inginkan.");
}

public static int hitungTotalHargaNoPresensi(int pilihanMenu, int
jumlahPesanan, String kodePromo) {
    int[] hargaItems = {15000, 20000, 22000, 12000, 10000, 18000};

    int totalHarga = hargaItems[pilihanMenu - 1] * jumlahPesanan;

    if (kodePromo.equals("DISKON50")) {
        System.out.println("Diskon 50% diterapkan!");
        totalHarga = totalHarga / 2;
    }
    else if (kodePromo.equals("DISKON30")) {
        System.out.println("Diskon 30% diterapkan!");
        totalHarga = (int)(totalHarga * 0.7);
    }
    else {
        System.out.println("Kode promo invalid (tidak ada diskon).");
    }

    return totalHarga;
}

```

```
}
```

#### Percobaan 4

1. String... namaPengunjung digunakan karena varargs membuat fungsi bisa menerima **banyak nama sekaligus** tanpa jumlah yang tetap. Java otomatis mengubahnya menjadi array, jadi lebih fleksibel dan simpel.

2. Modifikasi

```
package jobsheet11;

public class PengunjungKafe13 {
    public static void main(String[] args) {
        daftarPengunjung("Ali", "Budi", "Citra");
        daftarPengunjung("Andi");
        daftarPengunjung("Doni", "Eti", "Fahmi", "Galih");
    }

    static void daftarPengunjung(String... namaPengunjung) {
        System.out.println("Daftar Nama Pengunjung:");
        for (String nama : namaPengunjung) {
            System.out.println("- " + nama);
        }
    }
}
```

3. **Tidak bisa.** Java hanya mengizinkan **satu varargs** dalam satu fungsi, dan varargs **harus diletakkan di posisi terakhir**.

4. Jika daftarPengunjung() dipanggil tanpa argumen, program **tetap berjalan** dan **tidak error**. Varargs menjadi **array kosong**, sehingga output hanya menampilkan judul tanpa nama.