

Jobsheet 11

Fungsi 1 (Pertemuan 14)



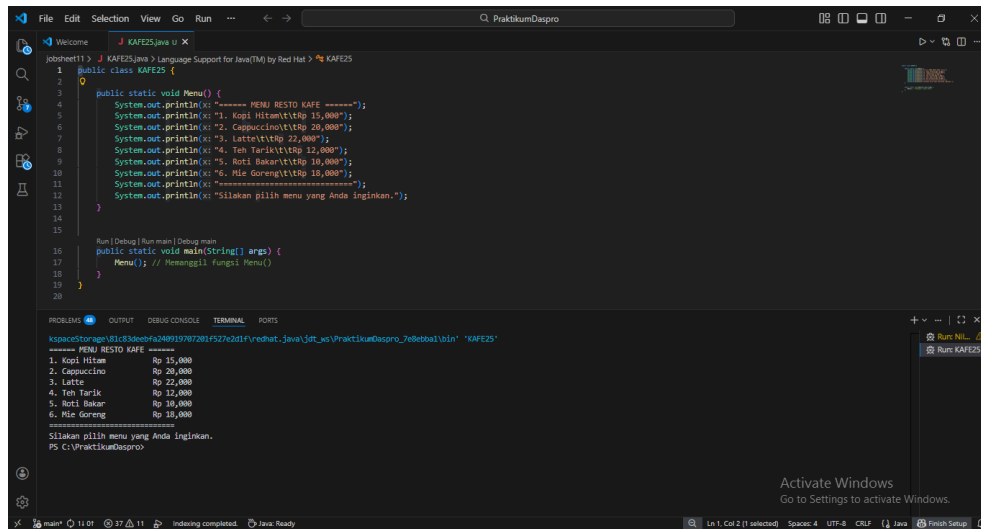
Nama : Salsabila Widyadhana

NIM : 254107020200

Kelas : TI-1H

Mata Kuliah : Praktikum Daspro

1. Percobaan 1: Membuat Fungsi Tanpa Parameter

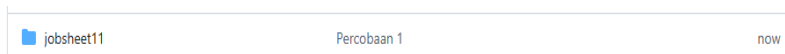


```
1 public class KAFE25 {
2
3     public static void Menu() {
4         System.out.println(x: "===== MENU RESTO KAFE =====");
5         System.out.println(x: "1. Kopi Hitam\tRp 15,000");
6         System.out.println(x: "2. Cappuccino\tRp 20,000");
7         System.out.println(x: "3. Latte\tRp 22,000");
8         System.out.println(x: "4. Teh Tarik\tRp 12,000");
9         System.out.println(x: "5. Roti Bakar\tRp 10,000");
10        System.out.println(x: "6. Mie Goreng\tRp 18,000");
11        System.out.println(x: "=====");
12        System.out.println(x: "Silakan pilih menu yang Anda inginkan.");
13    }
14
15    public static void main(String[] args) {
16        Menu(); // Memanggil Fungsi Menu()
17    }
18 }
19
20
```

OUTPUT

```
===== MENU RESTO KAFE =====
1. Kopi Hitam      Rp 15,000
2. Cappuccino     Rp 20,000
3. Latte           Rp 22,000
4. Teh Tarik      Rp 12,000
5. Roti Bakar     Rp 10,000
6. Mie Goreng     Rp 18,000
=====
Silakan pilih menu yang Anda inginkan.
PS C:\PraktikumDaspro>
```

Commit Github Percobaan 1 :



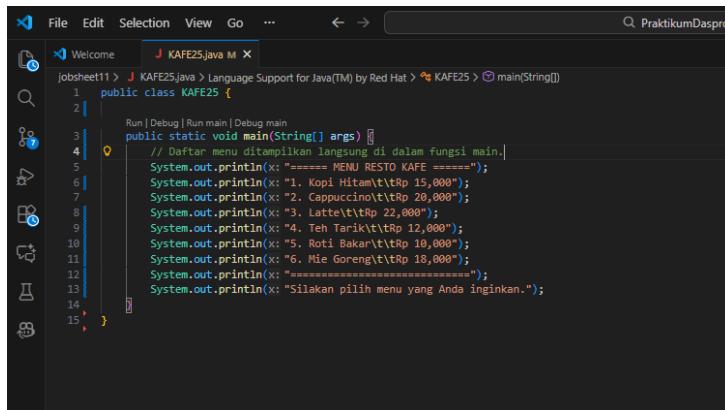
Pertanyaan :

1. Apakah fungsi tanpa parameter selalu harus bertipe void?
2. Apakah daftar menu pada program kafe dapat ditampilkan tanpa menggunakan fungsi Menu()? Modifikasi kode program tersebut untuk dapat menampilkan daftar menu tanpa menggunakan fungsi!
3. Jelaskan keuntungan menggunakan fungsi Menu() dibandingkan menulis semua perintah penampilan menu langsung di dalam fungsi main.
4. Uraikan secara singkat alur eksekusi program ketika fungsi Menu() dipanggil dari main (mulai dari program dijalankan sampai daftar menu tampil di layar).

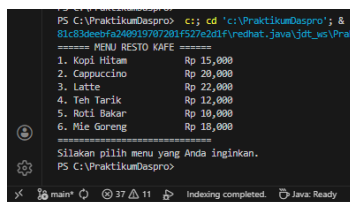
Jawab :

1. Tidak
2. Ya, Daftar menu dapat ditampilkan tanpa menggunakan fungsi Menu(). Semua perintah penampilan menu (System.out.println(...)) dapat dituliskan langsung di dalam fungsi main.

Modifikasi Percobaan 1 :



```
1 public class KAFE25 {
2
3     // Daftar menu ditampilkan langsung di dalam fungsi main.
4     public static void main(String[] args) {
5         System.out.println(x: "===== MENU RESTO KAFE =====");
6         System.out.println(x: "1. Kopi Hitam\t\tRp 15,000");
7         System.out.println(x: "2. Cappuccino\t\tRp 20,000");
8         System.out.println(x: "3. Latte\t\tRp 22,000");
9         System.out.println(x: "4. Teh Tarik\t\tRp 12,000");
10        System.out.println(x: "5. Roti Bakar\t\tRp 10,000");
11        System.out.println(x: "6. Mie Goreng\t\tRp 18,000");
12        System.out.println(x: "=====");
13        System.out.println(x: "Silakan pilih menu yang Anda inginkan.");
14    }
15 }
```



```
PS C:\PraktikumDaspro> cd 'c:\PraktikumDaspro'; & "
81c83d6e9f3d409107020f527e2d1f\redhat_java\get_us\Prak
===== MENU RESTO KAFE =====
1. Kopi Hitam      Rp 15,000
2. Cappuccino      Rp 20,000
3. Latte           Rp 22,000
4. Teh Tarik       Rp 12,000
5. Roti Bakar      Rp 10,000
6. Mie Goreng      Rp 18,000
=====
Silakan pilih menu yang Anda inginkan.
PS C:\PraktikumDaspro>
```

Commit Github Modifikasi Percobaan 1 :



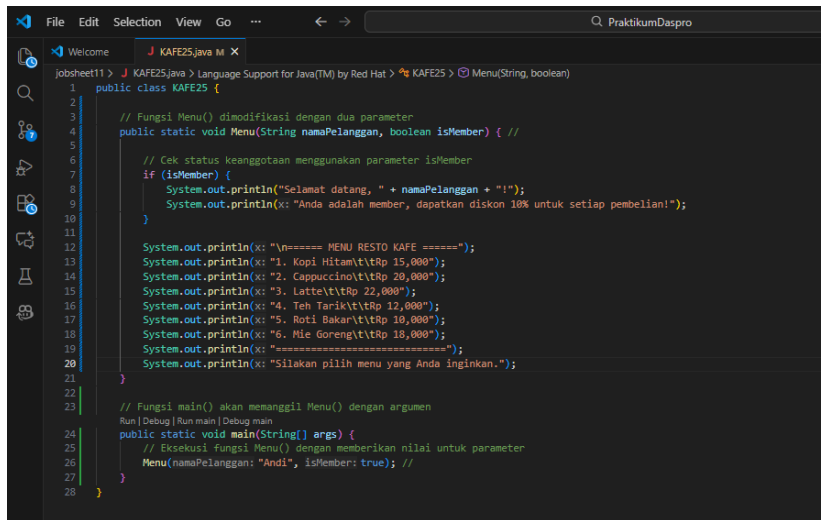
3. Keuntungan utama menggunakan fungsi Menu () adalah:

- Modularitas dan Keterbacaan: Fungsi Menu () mengelompokkan tugas spesifik (menampilkan menu) ke dalam blok kode yang terpisah, membuat fungsi main menjadi lebih ringkas dan mudah dipahami.
- Dapat Digunakan Kembali: Fungsi Menu () dapat dipanggil berkali-kali dari bagian program mana pun tanpa perlu menulis ulang baris-baris perintah menu yang sama.

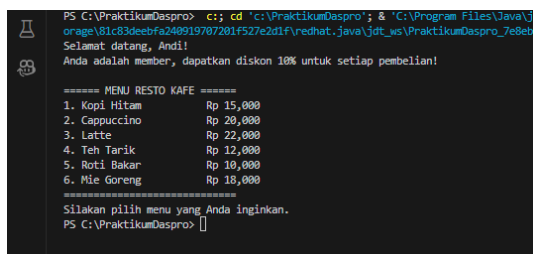
4. Alurnya adalah sebagai berikut :

- Mulai Program: Eksekusi program Java dimulai dari baris pertama fungsi main (public static void main(String[] args)).
- Panggilan Fungsi: Aliran kontrol mencapai baris Menu () ; di dalam main.
- Lompatan: Program menghentikan sementara eksekusi di main dan melompat ke baris pertama dari definisi fungsi Menu () .
- Eksekusi Fungsi: Semua perintah di dalam fungsi Menu () dieksekusi secara berurutan, menyebabkan daftar menu ditampilkan di layar.
- Kembali ke main(): Setelah semua baris di Menu () selesai dieksekusi, aliran kontrol kembali ke fungsi main, tepat setelah baris panggilan Menu () .
- Selesai: Program selesai dieksekusi karena tidak ada perintah lain di dalam main.

2. Percobaan 2: Membuat Fungsi Dengan Parameter



```
1 public class KAFE25 {
2
3     // Fungsi Menu() dimodifikasi dengan dua parameter
4     public static void Menu(String namaPelanggan, boolean isMember) { //
5
6         // Cek status keanggotaan menggunakan parameter isMember
7         if (isMember) {
8             System.out.println("Selamat datang, " + namaPelanggan + "!");
9             System.out.println("Anda adalah member, dapatkan diskon 10% untuk setiap pembelian!");
10        }
11
12        System.out.println("\n===== MENU RESTO KAFE =====");
13        System.out.println("1. Kopi Hitam\tRp 15,000");
14        System.out.println("2. Cappuccino\tRp 20,000");
15        System.out.println("3. Latte\tRp 22,000");
16        System.out.println("4. Teh Tarik\tRp 12,000");
17        System.out.println("5. Roti Bakar\tRp 10,000");
18        System.out.println("6. Mie Goreng\tRp 18,000");
19        System.out.println("=====");
20        System.out.println("Silakan pilih menu yang Anda inginkan.");
21    }
22
23    // Fungsi main() akan memanggil Menu() dengan argumen
24    public static void main(String[] args) {
25        // Eksekusi fungsi Menu() dengan memberikan nilai untuk parameter
26        Menu(namaPelanggan: "Andi", isMember: true); //
27    }
28 }
```



```
PS C:\PraktikumDaspro> cd 'c:\PraktikumDaspro'; & 'C:\Program Files\Java\jdk-8.0.602\bin\java.exe' -cp 'C:\PraktikumDaspro\KAFE25.jar' KAFE25
Selamat datang, Andi!
Anda adalah member, dapatkan diskon 10% untuk setiap pembelian!

===== MENU RESTO KAFE =====
1. Kopi Hitam      Rp 15,000
2. Cappuccino      Rp 20,000
3. Latte           Rp 22,000
4. Teh Tarik       Rp 12,000
5. Roti Bakar      Rp 10,000
6. Mie Goreng      Rp 18,000
=====
Silakan pilih menu yang Anda inginkan.
PS C:\PraktikumDaspro>
```

Commit Github Percobaan 2 :

jobsheet11	Percobaan 2	now
------------	-------------	-----

Pertanyaan :

1. Apakah kegunaan parameter di dalam fungsi?
2. Jelaskan mengapa pada percobaan ini fungsi Menu() menggunakan parameter namaPelanggan dan isMember?
3. Apakah parameter sama dengan variabel? Jelaskan.
4. Jelaskan bagaimana cara kerja parameter isMember pada fungsi Menu(). Apa perbedaan output ketika isMember bernilai true dan ketika false?
5. Apa yang akan terjadi jika memanggil fungsi Menu() tanpa menyertakan parameter namaPelanggan dan isMember?
6. Modifikasi kode di atas dengan menambahkan parameter baru kodePromo (String). Jika kodePromo adalah "DISKON50", tampilkan berikan diskon 50%. Jika kodePromo adalah "DISKON30", tampilkan berikan diskon 30%. Jika tidak ada kode promo yang berlaku, tampilkan kode invalid.

7. Berdasarkan fungsi Menu() di atas, jika nama pelanggan adalah "Budi", pelanggan tersebut member, dan menggunakan kode promo "DISKON30", tuliskan satu baris perintah pemanggilan fungsi menu yang benar.

8. Menurut Anda, apakah penggunaan parameter namaPelanggan dan isMember pada fungsi Menu() membuat program lebih mudah dibaca dan dikembangkan dibandingkan jika nilai-nilai tersebut ditulis langsung di dalam fungsi tanpa parameter? Jelaskan alasan Anda.

9. Commit dan push hasil modifikasi Anda ke Github dengan pesan “Modifikasi Percobaan 2”

Jawab :

1. Kegunaan parameter di dalam fungsi adalah sebagai penampung nilai input yang dikirimkan ke fungsi saat fungsi tersebut dipanggil.

2. Fungsi Menu() menggunakan parameter namaPelanggan (String) dan isMember (boolean) karena fungsi tersebut perlu informasi dinamis tentang pelanggan untuk melakukan tugas tambahan selain menampilkan menu.

3. Parameter adalah jenis variabel khusus yang dibuat hanya untuk menerima data input ke fungsi, sementara variabel umum digunakan untuk penyimpanan dan pemrosesan data di dalam fungsi.

4. Cara Kerja isMember:

1. Ketika fungsi Menu() dipanggil, nilai true atau false dilewatkan ke parameter isMember.
2. Program mengecek kondisi if (isMember).
3. Jika isMember bernilai true, blok kode di dalam if akan dieksekusi, yaitu menampilkan pesan diskon.
4. Jika isMember bernilai false, blok kode di dalam if dilewati, dan program langsung melanjutkan ke penampilan daftar menu.

Perbedaan Output:

- isMember = true (Contoh: Menu("Budi", true);): Output akan menampilkan sapaan member dan pesan diskon 10% sebelum daftar menu.
- isMember = false (Contoh: Menu("Budi", false);): Output hanya akan menampilkan sapaan biasa ("Selamat datang, Budi!") dan langsung diikuti oleh daftar menu, tanpa ada pesan diskon.

5. Ketika memanggil Menu() tanpa argumen, kompiler Java akan mendeteksi bahwa panggilan fungsi tersebut tidak cocok dengan tanda tangan fungsi yang didefinisikan, sehingga menghasilkan error yang menyatakan bahwa fungsi yang sesuai tidak dapat ditemukan (misalnya, *method Menu() in class KafeNoPresensi cannot be applied to given types; required: String, boolean; found: no arguments*).

6. Modifikasi Percobaan 2 beserta jawaban modifikasi kNo.7 :

```
File Edit Selection View Go ...
Welcome J KAFE25.java M X
johasheet11 > J KAFE25.java > Language Support for Java(TM) by Red Hat > % KAFE25 > @ main(String[])
1 public class KAFE25 {
2
3
4
5
6 public static void Menu(String namaPelanggan, boolean isMember, String kodePromo) {
7
8
9
10 System.out.println(s"===== MENU RESTO KAFE =====");
11 System.out.println("Selamat datang, " + namaPelanggan + "!");
12
13
14 // Cek Status Member (isMember)
15 if (isMember) {
16 System.out.println(s" >>> Anda adalah member, dapatkan diskon 18% untuk setiap pembelian!");
17 }
18
19 // LOGIKA DISKON KODE PROMO (Jawaban Pertanyaan No. 6)
20 if (kodePromo.equals(anObject: "DISKON38")) {
21 // Tampilkan diskon 38%
22 System.out.println(s" >>> Anda menggunakan kode promo DISKON38! Mendapatkan diskon 38%!");
23 } else if (kodePromo.equals(anObject: "DISKON30")) {
24 // Tampilkan diskon 30%
25 System.out.println(s" >>> Anda menggunakan kode promo DISKON30! Mendapatkan diskon 30%!");
26 } else if (!kodePromo.isEmpty()){
27 // Tampilkan kode invalid
28 System.out.println("Kode promo: " + kodePromo + " invalid atau tidak berlaku.");
29 }
30
31 // --- AKHIR LOGIKA DISKON PROMO ---
32
33 System.out.println(s" \n----- MENU RESTO KAFE -----");
34 System.out.println(s" 1. Kopi Hitam\t\tRp 15,000");
35 System.out.println(s" 2. Cappuccino\t\tRp 20,000");
36 System.out.println(s" 3. Latte\t\tRp 22,000");
37 System.out.println(s" 4. Teh Tarik\t\tRp 12,000");
38 System.out.println(s" 5. Roti Bakar\t\tRp 10,000");
39 System.out.println(s" 6. Mie Goreng\t\tRp 18,000");
40 System.out.println(s"=====");
41 System.out.println(s"Silakan pilih menu yang Anda inginkan.");
42
43 }
44
45 // --- FUNGSI UTAMA (MAIN) ---
46 Run | Debug | Run main | Debug main
47 public static void main(String[] args) {
48
49 // Jawaban Pertanyaan No. 7
50 // Pelanggan "Budi", member (true), kode promo "DISKON38"
51 System.out.println(s" --- Panggilan Fungsi (Jawaban No. 7) ---");
52 Menu(namaPelanggan: "Budi", isMember: true, kodePromo: "DISKON38");
53
54 System.out.println(s" \n");
55
56 // Contoh Panggilan Tambahan: Candra, member, kode promo invalid ("HEMAT")
57 System.out.println(s" --- Contoh Panggilan Tambahan ---");
58 Menu(namaPelanggan: "Candra", isMember: true, kodePromo: "HEMAT");
59
60 }
61 }
```

```
PS C:\PraktikumDaspro> cd 'C:\PraktikumDaspro'; & 'C:\Program Files\Java\
AppData\Roaming\Code\User\workspaceStorage\81c83debf2a240919707201f527e2d1fV
--- Panggilan Fungsi (Jawaban No. 7) ---
Selamat datang, Budi!
>>> Anda adalah member, dapatkan diskon 18% untuk setiap pembelian!
>>> Anda menggunakan kode promo DISKON38! Mendapatkan diskon 38%!

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

--- Contoh Panggilan Tambahan ---
Selamat datang, Candra!
>>> Anda adalah member, dapatkan diskon 18% untuk setiap pembelian!
Kode promo: HEMAT Invalid atau tidak berlaku.

===== MENU RESTO KAFE =====
1. Kopi Hitam      Rp 15,000
2. Cappuccino      Rp 20,000
3. Latte           Rp 22,000
4. Teh Tarik       Rp 12,000
5. Roti Bakar      Rp 10,000
6. Mie Goreng      Rp 18,000
=====
Silakan pilih menu yang Anda inginkan.
PS C:\PraktikumDaspro>
```

8. Ya,

Alasan:

- Keterbacaan main(): Fungsi main() menjadi sangat jelas dan ringkas. Dengan melihat Menu("Budi", true);, pembaca kode langsung tahu bahwa Budi sedang disambut dan dia adalah anggota, tanpa perlu melihat detail internal fungsi Menu().
- Fleksibilitas (Pengembangan): Jika ingin menyambut pelanggan baru (misalnya, Menu("Citra", false);), tidak perlu mengubah kode di dalam fungsi Menu(). Cukup memanggilnya dengan argumen yang berbeda. Jika nilai-nilai tersebut *hardcode* di dalam fungsi, harus mengubah dan mengompilasi ulang fungsi setiap kali pelanggan atau status keanggotaan berubah.

9.Commit Github Modifikasi Percobaan 2 :

jobsheet11	Modifikasi Percobaan 2	now
------------	------------------------	-----

3. Percobaan 3: Membuat Fungsi dengan Nilai Kembalian

```
File Edit Selection View Go ...
jobsheet11 > KAFE25.java M HitungTotalHarga25.java M
import java.util.Scanner;

public class HitungTotalHarga25 {
    // Scanner untuk menerima input dari pengguna
    static Scanner sc = new Scanner(System.in);

    // ... (Menampilkan Menu) ...
    public static void Menu() {
        System.out.println("===== MENU RESTO KAFE =====");
        System.out.println("1. Kopi Hitam\t\tRp 15,000");
        System.out.println("2. Cappuccino\t\tRp 20,000");
        System.out.println("3. Latte\t\t\tRp 22,000");
        System.out.println("4. Teh Tarik\t\tRp 12,000");
        System.out.println("5. Roti Bakar\t\tRp 10,000");
        System.out.println("6. Mie Goreng\t\tRp 18,000");
        System.out.println("=====");
    }

    // ... [PERCOBAAN 3 (Dengan Nilai Kembalian)] ...
    // Fungsi untuk menghitung total harga pesanan.
    // Fungsi ini mengembalikan nilai bertipe int (total harga).
    // Parameter pilihanMenu: Nomor menu yang dipilih (1-6)
    // Parameter banyakItem: Jumlah item yang dipesan
    // @return total harga pesanan
    public static int hitungTotalHarga(int pilihanMenu, int banyakItem) {
        // Array harga menu (sesuai urutan 1-6)
        int[] hargaMenu = {15000, 20000, 22000, 12000, 10000, 18000};

        // Cek validitas pilihan menu
        if (pilihanMenu <= 0 || pilihanMenu >= hargaMenu.length) {
            // Hitung harga total
            int hargaTotal = hargaMenu[pilihanMenu - 1]; // -1 karena array dimulai dari indeks 0
            int hargaTotal = hargaTotal * banyakItem;
            return hargaTotal; // Mengembalikan nilai total harga
        } else {
            // Jika pilihan tidak valid
            System.out.println("Pilihan menu tidak valid.");
            return 0;
        }
    }

    // ... FUNGSI UTAMA (MAIN) ...
    // Main (Default) Run main (Default main)
    public static void main(String[] args) {
        // 1. Tampilkan Menu
        Menu();

        // 2. Input dari pengguna
        System.out.print("Masukkan nomor menu yang ingin Anda pesan (1-6): ");
        int pilihanMenu = sc.nextInt();

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

        // 3. Panggil fungsi yang mengembalikan nilai dan simpan hasilnya
        int totalBayar = hitungTotalHarga(pilihanMenu, banyakItem);

        // 4. Tampilkan hasil
        if (totalBayar > 0) {
            System.out.println("-----");
            System.out.println("Total harga yang harus dibayar: Rp" + totalBayar);
            System.out.println("-----");
        }

        sc.close();
    }
}
```

```
PS C:\PraktikumDaspro> & 'C:\Program Files\Java\jdk-24\bin\jav
kspaceStorage\81c83deebfa240919707201f52e2d1f\redhat_java\jdk_
===== MENU RESTO KAFE =====
1. Kopi Hitam      Rp 15,000
2. Cappuccino      Rp 20,000
3. Latte            Rp 22,000
4. Teh Tarik       Rp 12,000
5. Roti Bakar      Rp 10,000
6. Mie Goreng      Rp 18,000
=====
Masukkan nomor menu yang ingin Anda pesan (1-6): 1
Masukkan jumlah item yang ingin dipesan: 2

Total harga yang harus dibayar: Rp30000
-----
PS C:\PraktikumDaspro>
PS C:\PraktikumDaspro>
```

Commit Github Percobaan 3 :

jobsheet11	Percobaan 3	7 minutes ago
------------	-------------	---------------

Pertanyaan :

1. Jelaskan secara singkat kapan suatu fungsi membutuhkan nilai kembalian (return value) dan kapan fungsi tidak perlu mengembalikan nilai. Berikan minimal satu contoh dari program kafe pada Percobaan 3 untuk masing-masing kasus.
2. Fungsi `hitungTotalHargaNoPresensi` saat ini mengembalikan total harga berdasarkan `pilihanMenu` dan `jumlahPesanan`. Sebutkan tipe data nilai kembalian dan dua buah parameter yang digunakan fungsi tersebut. Jelaskan arti masing-masing parameter dalam konteks program kafe.
3. Modifikasi kode di atas sehingga fungsi `hitungTotalHargaNoPresensi` dapat menerima `kodePromo`. Jika `kodePromo` adalah "DISKON50", maka mendapat diskon 50% dari `totalHarga` dan tampilkan diskon. Jika `kodePromo` adalah "DISKON30", maka mendapat diskon 30% dari `totalHarga` dan tampilkan diskon. Jika tidak ada kode promo yang berlaku, tampilkan kode invalid dan tidak ada pengurangan total harga `totalHarga`.
4. Modifikasi kode di atas sehingga bisa memilih beberapa jenis menu berbeda serta menampilkan total keseluruhan pesanan. Bagaimana memodifikasi program sehingga pengguna dapat: memesan lebih dari satu jenis menu (misalnya menu 1 dan 3 sekaligus), dan menampilkan total keseluruhan pesanan (gabungan dari semua jenis menu)?
5. Commit dan push hasil modifikasi Anda ke Github dengan pesan "Modifikasi Percobaan 3"

Jawab :

1. Membutuhkan Nilai Kembalian (Non-void): Ketika fungsi memiliki tugas untuk menghitung, memproses, atau mengambil suatu nilai, dan nilai hasil tersebut harus digunakan oleh bagian program lain yang memanggilnya.

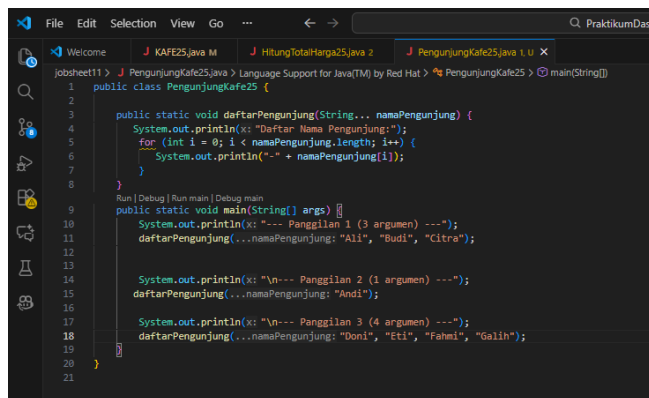
- Contoh dari Program Kafe: Fungsi `hitungTotalHarga(pilihanMenu, banyakItem)` harus mengembalikan nilai kembalian (tipe `int` atau `double`) karena hasil perhitungannya (total harga) diperlukan oleh fungsi `main` untuk ditampilkan kepada pengguna.

2. - Parameter `int pilihanMenu` Nilai input yang menunjukkan jenis menu mana yang dipesan oleh pelanggan (misalnya, 1 untuk Kopi Hitam, 3 untuk Latte). Nilai ini menentukan harga satuan.

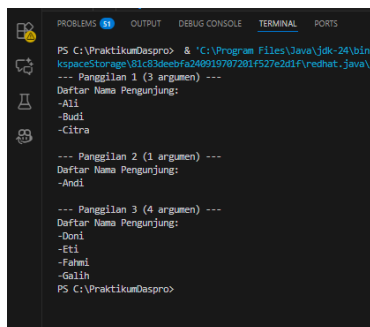
- Parameter `int banyakItem` Nilai input yang menunjukkan jumlah atau kuantitas dari item menu yang dipilih. Nilai ini digunakan sebagai pengali dalam perhitungan total harga.

3. Modifikasi Percobaan 3 beserta jawaban modifikasi percobaan 3 no.4 :

4. Percobaan 4: Fungsi Varargs



```
1 public class PengunjungKafe25 {
2
3     public static void daftarPengunjung(String... namaPengunjung) {
4         System.out.println("Daftar Nama Pengunjung:");
5         for (int i = 0; i < namaPengunjung.length; i++) {
6             System.out.println("-" + namaPengunjung[i]);
7         }
8     }
9
10    public static void main(String[] args) {
11        System.out.println("--- Panggilan 1 (3 argumen) ---");
12        daftarPengunjung(...namaPengunjung: "Ali", "Budi", "Citra");
13
14        System.out.println("\n--- Panggilan 2 (1 argumen) ---");
15        daftarPengunjung(...namaPengunjung: "Andi");
16
17        System.out.println("\n--- Panggilan 3 (4 argumen) ---");
18        daftarPengunjung(...namaPengunjung: "Doni", "Eti", "Fahmi", "Galih");
19    }
20 }
21
```



```
PS C:\PraktikumDaspro> & 'C:\Program Files\Java\jdk-24\bin\
kspaceStorage\51c83deebfa240919707201f527e2d1f\redhat_java\
--- Panggilan 1 (3 argumen) ---
Daftar Nama Pengunjung:
-Ali
-Budi
-Citra

--- Panggilan 2 (1 argumen) ---
Daftar Nama Pengunjung:
-Andi

--- Panggilan 3 (4 argumen) ---
Daftar Nama Pengunjung:
-Doni
-Eti
-Fahmi
-Galih
PS C:\PraktikumDaspro>
```

Commit Github Percobaan 4 :

jobsheet11	Percobaan 4	3 minutes ago
------------	-------------	---------------

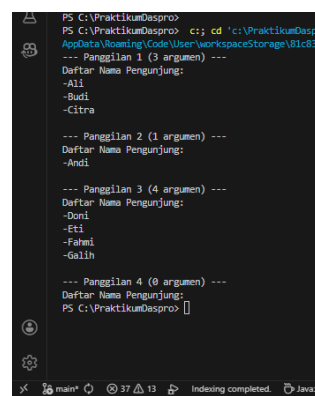
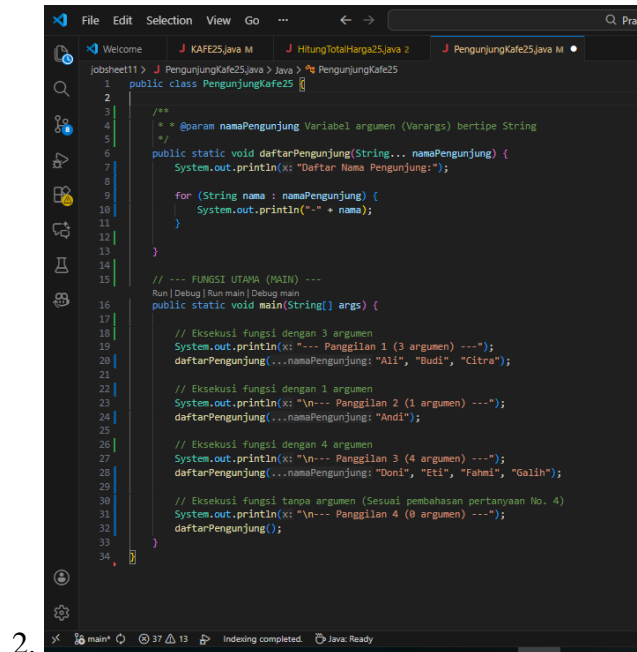
Pertanyaan :

1. Jelaskan mengapa penulisan parameter di praktikum 4 ditulis dengan String... namaPengunjung!
2. Modifikasi method daftarPengunjung menggunakan for-each loop.
3. Bisakah menggunakan dua tipe data varargs dalam satu fungsi? Jelaskan jawaban Anda berdasarkan aturan varargs di Java, dan berikan contohnya!
4. Jelaskan apa yang terjadi jika fungsi daftarPengunjung dipanggil tanpa argumen. Apakah program akan error saat kompilasi, error saat dijalankan, atau tetap berjalan? Jika tetap berjalan, bagaimana output yang dihasilkan?
5. Commit dan push hasil modifikasi Anda ke Github dengan pesan “Modifikasi Percobaan 4”

Jawab :

1. Java memperlakukan Varargs sebagai array satu dimensi dengan tipe data yang sesuai. Oleh karena itu, di dalam fungsi, namaPengunjung dapat diolah layaknya sebuah array,

misalnya dengan menggunakan `namaPengunjung.length` untuk mengetahui jumlah argumen atau mengakses elemennya menggunakan indeks (`namaPengunjung[i]`).



3. Tidak. Aturan Varargs di Java:

1. Sebuah fungsi (method) hanya boleh memiliki satu parameter varargs.
2. Jika parameter varargs digunakan, ia harus menjadi parameter terakhir dalam daftar parameter fungsi tersebut.

Alasan: Jika terdapat lebih dari satu varargs, atau jika varargs tidak diletakkan di akhir, kompiler tidak dapat menentukan di mana akhir dari argumen varargs pertama dan awal dari argumen varargs berikutnya. Ini akan menciptakan ambiguitas saat fungsi dipanggil.

1. Contoh yang salah menyebabkan error :

// ERROR: Tidak boleh ada dua varargs

```
public static void cetakData(String... nama, int... usia) { ... }
```

2. Contoh yang benar varargs harus diakhir :

// BENAR: Varargs diletakkan setelah parameter biasa

```
public static void cetakLaporan(String judul, String... dataIsian) {  
  
    // ...  
  
}
```

4. Program akan tetap berjalan dan tidak akan mengalami *error* (baik saat kompilasi maupun saat dijalankan). Mekanisme Varargs: Parameter `String... namaPengunjung` secara otomatis akan diubah menjadi array kosong (`String[]`) dengan panjang nol (`namaPengunjung.length = 0`).

Output yang Dihasilkan:

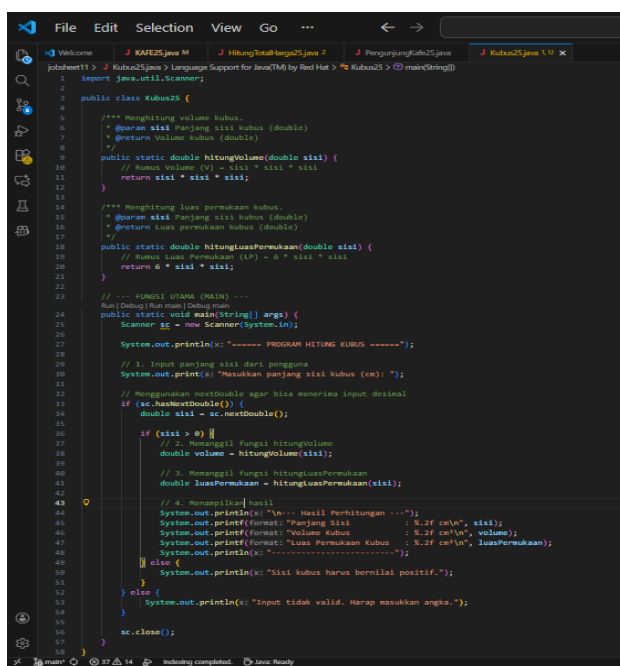
- Program akan mencetak baris pertama: "Daftar Nama Pengunjung:".
- *Loop* (for atau for-each) untuk menampilkan nama tidak akan dijalankan sama sekali karena panjang array adalah nol.

5.

jobsheet11	Modifikasi Percobaan 4	5 minutes ago
------------	------------------------	---------------

5. Tugas

1. Tugas 1



```
File Edit Selection View Go ***
Welcome J KAFE25.java M J HitungTotalKargo25.java J PengunjungKafe25.java J Kubus25.java
jobsheet11 > J Kubus25.java > Language Support for Java(TM) by Red Hat > Kubus25 > (D main(String[]))
1 import java.util.Scanner;
2
3 public class Kubus25 {
4
5     /** Menghitung volume kubus.
6     * @param sisi Panjang sisi kubus (double)
7     * @return volume kubus (double)
8     */
9     public static double hitungVolume(double sisi) {
10         // Rumus volume (V) = sisi * sisi * sisi
11         return sisi * sisi * sisi;
12     }
13
14     /** Menghitung luas permukaan kubus.
15     * @param sisi panjang sisi kubus (double)
16     * @return luas permukaan kubus (double)
17     */
18     public static double hitungLuasPermukaan(double sisi) {
19         // Rumus luas permukaan (LP) = 6 * sisi * sisi
20         return 6 * sisi * sisi;
21     }
22
23     // ... FUNGSI UTAMA (MAIN) ...
24     Run (Debug) Run main (Debug main)
25     public static void main(String[] args) {
26         Scanner sc = new Scanner(System.in);
27
28         System.out.println("===== PROGRAM HITUNG KUBUS =====");
29
30         // 1. Input panjang sisi dari pengguna
31         System.out.print("Masukkan panjang sisi kubus (cm): ");
32
33         // Menggunakan nextDouble() agar bisa menerima input desimal
34         if (sc.hasNextDouble()) {
35             double sisi = sc.nextDouble();
36
37             if (sisi > 0) {
38                 // 2. Memanggil fungsi hitungVolume
39                 double volume = hitungVolume(sisi);
40
41                 // 3. Memanggil fungsi hitungLuasPermukaan
42                 double luasPermukaan = hitungLuasPermukaan(sisi);
43
44                 // 4. Menampilkan hasil
45                 System.out.println("\n--- Hasil Perhitungan ---");
46                 System.out.printf("Panjang Sisi : %.2f cm\n", sisi);
47                 System.out.printf("Volume Kubus : %.2f cm³\n", volume);
48                 System.out.printf("Luas Permukaan Kubus : %.2f cm²\n", luasPermukaan);
49                 System.out.println("=====");
50             } else {
51                 System.out.println("Sisi kubus harus bernilai positif.");
52             }
53         } else {
54             System.out.println("Input tidak valid. Harap masukkan angka.");
55         }
56     }
57 }
58
59 sc.close();
60 }
```

```
PS C:\PraktikumDaspro> & 'C:\Program Files\Java\jdk-24\bin\
f527e2d1f\redhat.java\jdt_ws\PraktikumDaspro_7e8ebba1\bin'
===== PROGRAM HITUNG KUBUS =====
Masukkan panjang sisi kubus (cm): 11

--- Hasil Perhitungan ---
Panjang Sisi      : 11,00 cm
Volume Kubus     : 1331,00 cm³
Luas Permukaan Kubus : 726,00 cm²

PS C:\PraktikumDaspro> |
```

Commit Github Tugas 1 :

jobsheet11	Tugas 1	now
------------	---------	-----

2.Tugas 2

```
File Edit Selection View Go ...
public class NilaiMahasiswa25 {
    static Scanner sc = new Scanner(System.in);

    // Fungsi untuk menerima array nilai mahasiswa dan mengembalikannya
    // sebagai nilaiArray Array 1 dimensi ber tipe double untuk nilai nilai.
    public static void isiArray(double[] nilaiArray) {
        // Mengisi parameter berupa array nilai mahasiswa dan mengembalikannya
        // sebagai nilaiArray Array 1 dimensi nilai mahasiswa
        // Loop sebanyak jumlah elemen array (N)
        for (int i = 0; i < nilaiArray.length; i++) {
            System.out.print("Masukkan nilai mahasiswa ke-" + (i + 1) + ": ");
            // Menunggu user memasukkan nilai decimal
            nilaiArray[i] = sc.nextDouble();
        }
        System.out.println("\n");
    }

    // Fungsi untuk menampilkan array nilai mahasiswa dan menampilkan
    // sebagai nilaiArray Array 1 dimensi nilai mahasiswa
    public static void tampilArray(double[] nilaiArray) {
        System.out.println("\n");
        // Loop sebanyak jumlah elemen array (N)
        for (int i = 0; i < nilaiArray.length; i++) {
            System.out.print("Nilai mahasiswa ke-" + (i + 1) + ": ");
            System.out.print("-----\n");
        }
    }

    // Fungsi parameter berupa array nilai mahasiswa, menghitung dan mengembalikan total nilainya.
    // Menghasilkan nilai total double untuk return value
    // sebagai nilaiArray Array 1 dimensi nilai mahasiswa
    // sebagai total nilai mahasiswa ke-" + (i + 1) + ": ");
    public static double hitung(double[] nilaiArray) {
        double totalNilai = 0;
        // Loop untuk menjumlahkan semua elemen array
        for (double nilai : nilaiArray) {
            totalNilai += nilai;
        }
        return totalNilai;
    }

    // Fungsi utama program
    public static void main(String[] args) {
        // Mengisi array N dimensi dengan nilai mahasiswa
        double[] nilaiMahasiswa = new double[N];
        // Mengisi array N dimensi dengan nilai mahasiswa
        isiArray(nilaiMahasiswa);
        // Menampilkan array nilai mahasiswa
        tampilArray(nilaiMahasiswa);
        // Menghitung total nilai dan mengembalikan nilai total
        double totalNilaiKumulatif = hitung(nilaiMahasiswa);
        // Menampilkan total nilai yang dikembalikan oleh fungsi hitung
        System.out.println("\n");
        System.out.print("Total Nilai Keseluruhan Mahasiswa : ");
        System.out.print("-----\n");
        System.out.print(totalNilaiKumulatif);
        System.out.println("\n");
    }
}
```



```
File Edit Selection View Go ...
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
===== PROGRAM OLAH REKAP PENJUALAN CAFE =====
Masukkan jumlah menu yang akan diolah (N): 5
Masukkan jumlah hari penjualan: 1

--- Memulai Input Data Penjualan (modifikasi 3b dengan input pengguna) ---
Masukkan nama menu ke-1: 2
Masukkan nama menu ke-2: 3
Masukkan nama menu ke-3: 4
Masukkan nama menu ke-4: 5
Masukkan nama menu ke-5: 6

--- Masukkan Jumlah Penjualan per Hari ---
Menu: 2
Penjualan Hari ke-1: 2
Menu: 3
Penjualan Hari ke-1: 5
Menu: 4
Penjualan Hari ke-1: 11
Menu: 5
Penjualan Hari ke-1: 25
Menu: 6
Penjualan Hari ke-1: 7
Input data penjualan selesai.

=====
REKAP PENJUALAN CAFE
=====
MENU | HARI 1 | TOTAL
-----|-----|-----
2 | 2 | 2
3 | 5 | 5
4 | 11 | 11
5 | 25 | 25
6 | 7 | 7
=====

--- MENU DENGAN TOTAL PENJUALAN TERTINGGI ---
Menu Terlaris : 5
Total Penjualan : 25 item.
=====

--- DATA RATA PENJUALAN HARIAN (UNTUK SETIAP MENU) ---
2 : 2,00 item per hari
3 : 5,00 item per hari
4 : 11,00 item per hari
5 : 25,00 item per hari
6 : 7,00 item per hari
=====

PS C:\PraktikumDaspro
```

Commit Github Tugas 3 :

jobsheet11	Tugas 3	now
------------	---------	-----