

Nama: Ahmad Sofyan Badawi

NIM: 244107020073

Kelas: TI-1B

Percobaan 1

Langkah-langkah:

1. Buat repository baru dengan nama daspro-jobsheet11. Selanjutnya buat class baru, dan simpan file tersebut dengan nama **Kafe04.java**.
2. Buat fungsi Menu di dalam class tersebut.

```
1 public static void Menu() {  
2     System.out.println("==== MENU RESTO KAFE =====");  
3     System.out.println("1. Kopi Hitam - Rp 15,000");  
4     System.out.println("2. Cappuccino - Rp 20,000");  
5     System.out.println("3. Latte - Rp 22,000");  
6     System.out.println("4. Teh Tarik - Rp 12,000");  
7     System.out.println("5. Roti Bakar - Rp 10,000");  
8     System.out.println("6. Mie Goreng - Rp 18,000");  
9     System.out.println("=====");  
10    System.out.println("Silakan pilih menu yang Anda inginkan.");  
11 }
```

3. Buat fungsi main di dalam class tersebut, dan eksekusi fungsi Menu dari dalam fungsi main.

```
1 public static void main(String[] args) {  
2     Menu();  
3 }
```

4. Push dan commit ke git!

Pertanyaan:

1. Apakah fungsi tanpa parameter selalu digunakan untuk fungsi “void” saja?
2. Apakah bisa perintah menampilkan menu yang ada dituliskan tanpa fungsi Menu? modifikasi kode program tersebut untuk dapat menampilkan daftar menu tanpa menggunakan fungsi!
3. Apakah keuntungan menggunakan fungsi di dalam program?

Jawaban:

1. Tidak, fungsi tanpa parameter tidak selalu harus memiliki fungsi tipe **void**. Dalam Java, kita dapat memiliki metode (fungsi) tanpa parameter yang mengembalikan nilai. Tipe pengembalian metode tersebut bisa berupa tipe data apa pun, seperti **int**, **String**, **boolean**, dan lain-lain.
2. Bisa, hanya memberikan perintah sout saja seperti biasa dengan syarat sebelumnya sudah membuat struktur dasar program Java yang terdiri dari fungsi main().

```

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

```

3. Berfungsi untuk memudahkan pembagian tugas dan menghindari perulangan pembuatan kode

Percobaan 2

Langkah-langkah:

1. Ubah fungsi Menu dengan dua buah parameter bertipe String dan boolean di dalam class **Kafe04.java**.

```

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

    if (isMember) {
        System.out.println("Anda adalah member, dapatkan diskon 10% untuk setiap pembelian!");
    }

    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.");
}

```

2. Eksekusi fungsi UcapanTambahan dari dalam fungsi main.

```

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

```

Pertanyaan:

1. Apakah kegunaan parameter di dalam fungsi?
2. Apakah parameter sama dengan variabel? jelaskan!
3. Bagaimana cara kerja parameter isMember pada method Menu
4. Apa yang akan terjadi jika memanggil Menu tanpa menyertakan parameter namaPelanggan dan isMember?

5. 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

Jawaban:

1. Berfungsi untuk menerima data dari luar, sehingga fungsi dapat dengan mudah beroperasi pada data tersebut.
2. Parameter dengan variabel tentu berbeda. Parameter berfungsi untuk nilai yang akan diteruskan ke fungsi saat fungsi tersebut dipanggil dan parameter hanya ada di konteks fungsi. Sedangkan, variabel berfungsi untuk menyimpan nilai yang dapat diubah selama eksekusi program.
3. Parameter isMember digunakan untuk menentukan apakah pelanggan berhak mendapatkan diskon member. Jika isMember bernilai true, maka program akan menampilkan pesan tentang diskon untuk member tersebut.
4. Karena metode Menu didefinisikan untuk tiga parameter, maka jika tidak menyertakan parameter akan terjadi error.
- 5.

```
21      System.out.println(x:"Silahkan pilih menu yang Anda inginkan.");
22
23      if (kodePromo.equalsIgnoreCase(anotherString:"DISKON50")) {
24          System.out.println(x:"Mendapatkan diskon 50%");
25      } else if (kodePromo.equalsIgnoreCase(anotherString:"DISKON30")) {
26          System.out.println(x:"Mendapatkan diskon 30%");
27      } else {
28          System.out.println(x:"Kode diskon invalid");
29      }
30  }
31
32  Run | Debug
33  public static void main(String[] args) {
34      Menu(namaPelanggan:"Andi", isMember:true, kodePromo:"DISKON50");
35  }
36  }
```

PROBLEMS 2 OUTPUT DEBUG CONSOLE PORTS SEARCH ERROR

TERMINAL Run:

```
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
=====
Silahkan pilih menu yang Anda inginkan.
Mendapatkan diskon 50%
```

Percobaan 3

Langkah-langkah:

1. Pada Class yang sama buat fungsi **hitungTotalHarga04** di dalam class tersebut yang mengembalikan nilai total harga dan parameter masukan pilihan Menu dan banyaknya pesanan

```
public static int hitungTotalHarga(int pilihanMenu, int banyakItem) {
    int[] hargaItems = {15000, 20000, 22000, 12000, 10000, 18000};

    int hargaTotal = hargaItems[pilihanMenu - 1] * banyakItem;
    return hargaTotal;
}
```

2. Modifikasi fungsi main di dalam class tersebut, dan eksekusi fungsi **PenerimaUcapan** dari dalam fungsi main.

```

System.out.print("\nMasukkan nomor menu yang ingin Anda pesan: ");
int pilihanMenu = sc.nextInt();
System.out.print("Masukkan jumlah item yang ingin dipesan: ");
int banyakItem = sc.nextInt();

int totalHarga = hitungTotalHarga(pilihanMenu, banyakItem);

System.out.println("Total harga untuk pesanan Anda: Rp" + totalHarga);

```

Pertanyaan:

1. Jelaskan mengenai, kapan suatu fungsi membutuhkan nilai kembalian (return)!
2. Modifikasi kode diatas sehingga fungsi hitungTotalHarga dapat menerima kodePromo. Jika kodePromo adalah "DISKON50", maka mendapat diskon 50% dari totalHarga. Jika kodePromo adalah "DISKON30", maka mendapat diskon 30% dari totalHarga tampilkan berikan diskon 30%. Jika tidak ada kode promo yang berlaku, tampilkan kode invalid dan tidak ada pengurangan total harga totalHarga.
3. Modifikasi kode diatas sehingga bisa memilih beberapa jenis menu berbeda serta menampilkan total keseluruhan pesanan.

Jawaban:

1. Suatu fungsi membutuhkan nilai kembalian (return) ketika hasil dari proses yang dilakukan oleh fungsi tersebut perlu digunakan di tempat lain dalam program.
2. Program ada pada github

```

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
=====
Silahkan pilih menu yang Anda inginkan.

Masukkan jumlah item yang ingin dipesan: 1
Ada menu tambahan (Y/N): n
Masukkan kode Promo untuk total pesanan Anda (atau tekan Enter jika tidak ada): DISKON50
Mendapatkan diskon 50%
=====
Total harga untuk semua pesanan Anda: Rp 20000.0
Total setelah diskon: Rp 10000.0
=====

```

3. Program ada pada github

```
=====
Total harga untuk semua pesanan Anda: Rp 20000.0
Total setelah diskon: Rp 10000.0
=====
```

Percobaan 4

Langkah-langkah:

1. Buat class baru, simpan file tersebut dengan nama **PengunjungCafe04.java**.
2. Buatlah fungsi `daftarPengunjung` (bertipe void) di dalam class tersebut dengan menggunakan parameter `varArgs`

```
System.out.print("\nMasukkan nomor menu yang ingin Anda pesan: ");
int pilihanMenu = sc.nextInt();
System.out.print("Masukkan jumlah item yang ingin dipesan: ");
int banyakItem = sc.nextInt();

int totalHarga = hitungTotalHarga(pilihanMenu, banyakItem);

System.out.println("Total harga untuk pesanan Anda: Rp" + totalHarga);
```

3. Buat fungsi `main` di dalam class tersebut, dan eksekusi fungsi `namaPengunjung` dari dalam fungsi `main`.

```
daftarPengunjung("Ali", "Budi", "Citra");
```

Pertanyaan:

1. Jelaskan mengapa penulisan parameter di praktikum 4 di tulis dengan `String... namaPengunjung`!
2. Modifikasi method `daftarPengunjung` menggunakan `for-each` loop!
3. Bisakah menggunakan dua tipe data `varaargs` dalam satu fungsi? Berikan contohnya!

Jawaban:

1. `String...` memberikan fleksibilitas dalam jumlah argumen yang dapat diterima oleh fungsi. Dengan cara ini, pengguna fungsi tidak perlu khawatir tentang berapa banyak nama pengunjung yang akan dimasukkan. Mereka dapat memberikan satu, beberapa, atau bahkan tidak ada nama sama sekali, dan fungsi tetap akan berfungsi dengan baik.
- 2.

```

public static void daftarPengunjung(Object... namaPengunjung) {
    System.out.println(x:"Daftar Nama Pengunjung: ");
    //for (int i = 0; i < namaPengunjung.length; i++) {
    //    System.out.println("- " + namaPengunjung[i]);
    //}
    for (Object nama : namaPengunjung) {
        System.out.println("- " + nama);
    }
}

```

3. Bisa, tetapi harus mengganti tipe data String menjadi Object

```

public static void daftarPengunjung(Object... namaPengunjung) {
    System.out.println(x:"Daftar Nama Pengunjung: ");
    //for (int i = 0; i < namaPengunjung.length; i++) {
    //    System.out.println("- " + namaPengunjung[i]);
    //}
    for (Object nama : namaPengunjung) {
        System.out.println("- " + nama);
    }
}

```

Percobaan 5

Langkah-langkah:

1. Buat class baru, simpan file tersebut dengan nama Percobaan604.java.
2. Buatlah program untuk menghitung luas persegi panjang dan volume balok tanpa menggunakan fungsi

```

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

    int p,l,t,L,vol;

    System.out.println("Masukkan panjang");
    p=input.nextInt();
    System.out.println("Masukkan lebar");
    l=input.nextInt();
    System.out.println("Masukkan tinggi");
    t=input.nextInt();

    L=p*l;
    System.out.println("Luas Persegi panjang adalah "+L);

    vol=p*l*t;
    System.out.println("Volume balok adalah "+vol);
}

```

3. Program menghitung luas persegi dan volume balok di atas jika dibuatkan fungsi maka terdapat 3 fungsi yaitu hitungLuas, hitungVolume dan fungsi main, seperti di bawah ini:

Fungsi hitungLuas

```
static int hitungLuas (int pjpg, int lb){  
    int Luas=pjpg*lb;  
    return Luas;  
}
```

Fungsi hitungVolume

```
static int hitungVolume (int tinggi, int a, int b){  
    int volume= hitungLuas(a,b)*tinggi;  
    return volume;  
}
```

Fungsi main

```
public static void main(String[] args) {  
    Scanner input =new Scanner (System.in);  
    int p,l,t,L, vol;  
    System.out.println("Masukkan panjang");  
    p=input.nextInt();  
    System.out.println("Masukkan lebar");  
    l=input.nextInt();  
    System.out.println("Masukkan tinggi");  
    t=input.nextInt();  
  
    L=hitungLuas(p,l);  
    System.out.println("Luas Persegi Panjang adalah "+L);  
    vol=hitungVolume(t,p,l);  
    System.out.println("Volume Balok adalah "+vol);  
}
```

4. Lakukan validasi hasil dengan memberikan contoh input pada program tersebut!
Tampilkan hasilnya!

Pertanyaan:

1. Sebutkan tahapan dan urutan eksekusi pada percobaan 6 di atas!
2. Apakah output dari program di bawah ini kemudian jelaskan alur jalannya program tersebut!


```

1 public class programKu {
2     public static void TampilHinggaKei(int i) {
3         for (int j = 1; j <= i; j++) {
4             System.out.print(j);
5         }
6     }
7
8     public static int Jumlah (int bil1, int bil2) {
9         return (bil1 + bil2);
10    }
11
12    public static void TampilJumlah (int bil1, int bil2) {
13        TampilHinggaKei(Jumlah(bil1, bil2));
14    }
15
16    public static void main (String[] args) {
17        int temp = Jumlah(1, 1);
18        TampilJumlah(temp, 5);
19    }
20 }

```

3. Pada saat apakah fungsi yang kita buat harus menggunakan parameter atau tidak? Pada saat apakah fungsi yang kita buat harus memiliki nilai kembalian atau tidak? Jelaskan!

Jawaban:

1.
 - hitungLuas(p, l) berfungsi untuk menghitung luas persegi panjang.
 - hitungVolume(t, l, p) berfungsi untuk menghitung volume balok.
 1. Deklarasikan class dan method pada program
 2. Masuk ke main dan membuat Scanner untuk nantinya user bisa memberikan nilai input
 3. Mengambil input yang sudah dimasukkan (p l, t).
 4. Menghitung luas persegi panjang serta balok kemudian menampilkan hasilnya.
2. Program dimulai dari metode **main**, di mana pertama-tama dihitung jumlah dari **1** dan **1**, yang hasilnya adalah **2**. Kemudian, program memanggil metode **tampilJumlah** dengan argumen **2** dan **5**. Di dalam metode ini, jumlah dari **2** dan **5** dihitung, menghasilkan **7**. Setelah itu, metode **TampilHinggaKei** dipanggil dengan argumen **7**, yang mencetak angka dari **1** hingga **7** secara berurutan tanpa spasi.
3. - Parameter
Fungsi menggunakan **parameter** ketika kita perlu memberi **input** ke dalam fungsi untuk diproses. Misalnya, kalau kita ingin menjumlahkan dua angka yang berbeda setiap kali, kita perlu memberikan angka-angka itu sebagai parameter.

Fungsi **tidak perlu** parameter jika tidak memerlukan input apa pun. Misalnya, jika fungsi hanya mencetak pesan yang sama setiap kali dipanggil.

- Nilai kembalian

Fungsi memerlukan **nilai kembalian** jika kita ingin **menghasilkan hasil** dari perhitungan atau proses yang dilakukan dalam fungsi itu. Misalnya, menghitung luas, kita ingin mendapatkan hasilnya untuk digunakan lagi.

Fungsi **tidak perlu** nilai kembalian jika tujuannya hanya untuk **melakukan aksi** atau **menampilkan sesuatu**, bukan untuk mengembalikan hasil.

Tugas

1. Buatlah sebuah class KubusNoAbsen yang di dalamnya terdapat fungsi untuk menghitung volume kubus dan luar permukaan kubus!

Program:

```
1  import java.util.Scanner;
2  public class Kubus04 {
3      Run | Debug
4      public static void main(String[] args) {
5          Scanner sc = new Scanner(System.in);
6          System.out.print("Masukkan sisi kubus: ");
7          double sisi = sc.nextDouble();
8
9          System.out.println("Volume Kubus: " + hitungVolume(sisi));
10         System.out.println("Luas Permukaan Kubus: " + hitungLuasPermukaan(sisi));
11     }
12
13     public static double hitungVolume(double sisi) {
14         double volume = sisi * sisi * sisi;
15         return volume;
16     }
17
18     public static double hitungLuasPermukaan(double sisi) {
19         double luasPermukaan = 6 * sisi * sisi;
20         return luasPermukaan;
21     }
22 }
```

Output:

```
C:\Users\AHMAD SOFYAN\Documents> java Kubus04
Masukkan sisi kubus: 3
Volume Kubus: 27.0
Luas Permukaan Kubus: 54.0
```

2. Ibu Mariana mengajar café. Berikut adalah rekap penjualan 5 menu dari hari pertama hingga ketujuh:

	Hari ke 1	Hari ke 2	Hari ke 3	Hari ke 4	Hari ke 5	Hari ke 6	Hari ke 7
Kopi	20	20	25	20	10	60	10
Teh	30	80	40	10	15	20	25
Es Degan	5	9	20	25	10	5	45
Roti Bakar	50	8	17	18	10	30	6
Gorengan	15	10	16	15	10	10	55

Tambahkan fungsi untuk mengambil informasi dari data di atas dengan rincian sebagai berikut :

- a. Fungsi untuk menginputkan data penjualan
- b. Fungsi untuk menampilkan seluruh data penjualan dari hari pertama hingga hari terakhir
- c. Fungsi untuk menampilkan Menu yang memiliki penjualan tertinggi
- d. Fungsi untuk menampilkan rata-rata penjualan untuk setiap menu

Program ada pada GitHub