

Nama: Bisma Adhiaksa

Kelas: TI-1B

NIM: 244107020216

### Percobaan 1

1. Buat file dengan nama **Kafe(NoAbsen).java**
2. Buat fungsi menu seperti berikut

```
public static void Menu() {  
    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("Silahkan pilih menu yang Anda inginkan.");  
}
```

3. Buat fungsi main dalam class

```
17  
18     public static void main(String[] var0) {  
19         Menu();  
20     }  
21 }  
22
```

### 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 berarti fungsi ini tidak menerima inputan saat dipanggil. Dan untuk "void" ini berarti fungsi tidak mengembalikan nilai apapun

2. Bisa,

```
1 public class Kafe07 {
2     Run main | Debug main | Run | Debug
3     public static void main (String[] args){
4         System.out.println(x:"==== MENU RESTO KAFE ===");
5         System.out.println(x:"1. Kopi Hitam - Rp 15,000");
6         System.out.println(x:"2. Cappuccino - Rp 20,000");
7         System.out.println(x:"3. Latte - Rp 22,000");
8         System.out.println(x:"4. Teh Tarik - Rp 12,000");
9         System.out.println(x:"5. Roti Bakar - Rp 10,000");
10        System.out.println(x:"6. Mie Goreng - Rp 18,000");
11        System.out.println(x:"=====");
12        System.out.println(x:"Silahkan pilih menu yang Anda inginkan.");
13    }
```

3. Penggunaan fungsi dalam program memudahkan pemanggilan saat programnya panjang, sehingga tidak perlu megulang membuat program.

## Percobaan 2

1. Ubah fungsi Menu, dengan menambahkan parameter nama pelanggan dan status ke-memberan pelanggan

```
1 public class Kafe07 {
2     public static void Menu(String namaPelanggan, boolean isMember){
3         System.out.println("Selamat datang, " + namaPelanggan + "!");
4
5         if(isMember){
6             System.out.println(x:"Anda adalah member, dapatkan diskon 10% untuk setiap pembelian!");
7         }
8
9         System.out.println(x:"==== MENU RESTO KAFE ===");
10        System.out.println(x:"1. Kopi Hitam - Rp 15,000");
11        System.out.println(x:"2. Cappuccino - Rp 20,000");
12        System.out.println(x:"3. Latte - Rp 22,000");
13        System.out.println(x:"4. Teh Tarik - Rp 12,000");
14        System.out.println(x:"5. Roti Bakar - Rp 10,000");
15        System.out.println(x:"6. Mie Goreng - Rp 18,000");
16        System.out.println(x:"=====");
17        System.out.println(x:"Silahkan pilih menu yang Anda inginkan.");
18    }
```

2. Tambahkan parameter pada pemanggilan fungsi menu

```
Run main | Debug main | Run | Debug
public static void main (String[] args){
    Menu(namaPelanggan:"Andi", isMember: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. Kegunaan parameter adalah untuk mengisi nilai pada variabel dalam fungsi, penggunaan variabel bisa membuat fungsi lebih fleksibel karena tidak ada variabel tetap yang digunakan.
2. Bisa dibilang sama, hanya saja parameter lebih ke variabel khusus. Perbedaannya karena parameter hanya berlaku dalam fungsi parameter dideklarasikan, sedangkan variabel tergantung pada di mana variabel dideklarasikan. Parameter dideklarasikan di tanda kurung dalam tanda kurung fungsi, variabel dideklarasikan.
3. Parameter isMember akan menerima data bertipe Boolean saat pemanggilan fungsi Menu
4. Akan terjadi eror

5.

```
if (kodePromo.equals(anObject:"DISKON50")) {  
    System.out.println(x:"Diskon yang diterima sebesar 50%");  
}else if (kodePromo.equals(anObject:"DISKON30")){  
    System.out.println(x:"Diskon yang diterima sebesar 30%");  
}else{  
    System.out.println(x:"Kode Invalid");  
}
```

### Percobaan 3

1. Pada class yang sama, buat fungsi **hitungTotalHarga(NoAbsen)** yang mengembalikan nilai total harga dan parameter masukkan pilihan menu

dan

banyak

pesanan

```
public static int hitungTotalHarga07(int pilihanMenu, int banyakItem){  
    int hargaItems[] = {15000, 20000, 22000, 12000, 10000, 18000};  
  
    int hargaTotal = hargaItems[pilihanMenu - 1] * banyakItem;  
    return hargaTotal;  
}
```

2. Modifikasi fungsi main dengan menambahkan fungsi menerima inputan

```
public static void main (String[] args){  
    Menu(namaPelanggan:"Andi",kodePromo:"DISKON50" ,isMember:true);  
  
    Scanner sc = new Scanner(System.in);  
    System.out.print(s:"\nMasukkan nomor menu yang ingin Anda pesan: ");  
    int pilihanMenu = sc.nextInt();  
    System.out.print(s:"Masukkan jumlah item yang ingin Anda pesan: ");  
    int banyakItem = sc.nextInt();  
  
    int totalHarga = hitungTotalHarga07(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. 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. Saat fungsi menggunakan tipe non-void

```

public static double hitungTotalHarga07(int pilihanMenu, int banyakItem) {
    int hargaItems[] = {15000, 20000, 22000, 12000, 10000, 18000};
    return hargaItems[pilihanMenu - 1] * banyakItem;
}

public static double hitungDiskon07(String kodePromo, double totalHarga07) {
    double diskon, hargaTotal = totalHarga07;

    if (kodePromo.equals(anObject:"DISKON50")) {
        diskon = 0.5;
        System.out.println(x:"\nDiskon yang diterima sebesar 50%");
    } else if (kodePromo.equals(anObject:"DISKON30")) {
        diskon = 0.3;
        System.out.println(x:"\nDiskon yang diterima sebesar 30%");
    } else {
        System.out.println(x:"\nKode Promo Invalid");
        diskon = 0;
    }

    return hargaTotal - (hargaTotal * diskon);
}

```

2.

```

do {
    if (jumlahPesanan >= pilihanMenu.length) {
        int tempPilihanMenu[] = new int[pilihanMenu.length + 1];
        int tempBanyakItem[] = new int[banyakItem.length + 1];
        System.arraycopy(pilihanMenu, srcPos:0, tempPilihanMenu, destPos:0, pilihanMenu.length);
        System.arraycopy(banyakItem, srcPos:0, tempBanyakItem, destPos:0, banyakItem.length);
        pilihanMenu = tempPilihanMenu;
        banyakItem = tempBanyakItem;
    }

    System.out.print(s:"\nMasukkan nomor menu yang ingin Anda pesan: ");
    pilihanMenu[jumlahPesanan] = sc.nextInt();
    System.out.print(s:"Masukkan jumlah item yang ingin Anda pesan: ");
    banyakItem[jumlahPesanan] = sc.nextInt();
    sc.nextLine();

    bayarKotor += hitungTotalHarga07(pilihanMenu[jumlahPesanan], banyakItem[jumlahPesanan]);
    jumlahPesanan++;

    System.out.print(s:"Apakah Anda ingin menambah pesanan (y/n)? ");
    String tambah = sc.nextLine();

    if (!tambah.equalsIgnoreCase(anotherString:"y")) {
        pilihan = false;
    }
} while (pilihan);

System.out.print(s:"Tolong masukkan kode promo (huruf kapital): ");
String kodePromo = sc.nextLine();

tampilanPesanan(pilihanMenu, banyakItem, jumlahPesanan);
double totalHarga = hitungDiskon07(kodePromo, bayarKotor);

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

```

3.

#### Percobaan 4

1. Buat file baru dengan nama **PengunjungCafe(NoAbsen).java**

2. Buat fungsi daftar pengunjung

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

Run main | Debug main | Run | Debug

3. Buat fungsi main

```
public static void main(String[] args) {  
    daftarPengunjung(...namaPengunjung:"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. Penulisan String... namaPengunjung berarti parameter itu akan berubah menjadi String array sesuai dengan berapa String inputan yang diberikan.

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

- 2.
3. Tidak bisa

### Percobaan 5: Pembuatan Kode Program, Dengan Fungsi versus Tanpa Fungsi

1. Buat file baru dengan nama **Percobaan6(NoAbsen).java**, lalu buat fungsi hitung luas dan hitung volume

```
public static int hitungLuas(int panjang, int lebar){  
    int luas = panjang * lebar;  
    return luas;  
}  
  
public static int hitungVolume(int tinggi, int p, int l){  
    int volume = hitungLuas(p, l) * tinggi;  
    return volume;  
}
```

2. Lalu buat struktur main

```
public static void main(String[] args) {  
    Scanner input = new Scanner(System.in);  
  
    int p,l,t,L,vol;  
  
    System.out.print(s:"Masukkan panjang: ");  
    p = input.nextInt();  
    System.out.print(s:"Masukkan lebar: ");  
    l = input.nextInt();  
    System.out.print(s:"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);  
}
```

```
Masukkan panjang: 5  
Masukkan lebar: 2  
Masukkan tinggi: 9  
Luas Persegi Panjang adalah 10  
Volume balok adalah 90
```

3. Berikut hasil run

## 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. Program akan langsung mengeksekusi program main, awalnya menerima input, menghitung luas dengan fungsi menghitung luas dan menampilkannya, lalu menghitung volume dengan fungsi menghitung volume dan menampilkannya
2. outputnya adalah tampilan perbaris angka 1 sampai 7, alurnya dimulai dari pembuatan variabel temp yang berisi nilai dari fungsi Jumlah yaitu 2, lalu angka 2 dimasukkan sebagai parameter pada fungsi TampilJumlah. Pada TampilJumlah dikerjakan terlebih dahulu fungsi Jumlah(2, 5) yang hasilnya 7, lalu akan mengerjakan fungsi TampilHinggaKei(7) yang akan menghasilkan output barisan angka 1 sampai 7
3. fungsi menggunakan parameter saat dibutuhkan nilai dari luar fungsi, sedangkan fungsi harus memiliki nilai kembalian saat fungsi merupakan tipe non-void



### 3. Tugas

#### Waktu Pengerjaan: 100 menit

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

```
Kubus07.java
1  import java.util.Scanner;
2  public class Kubus07 {
3      public static double hitungVolume(double sisi){
4          double volume = sisi * sisi * sisi;
5          return volume;
6      }
7
8      public static double hitungLuasPermukaan(double sisi){
9          double luasPerm = sisi * sisi * 6;
10         return luasPerm;
11     }
12
13     public static void main(String[] args) {
14         Scanner sc = new Scanner(System.in);
15         System.out.print("Masukkan sisi kubus: ");
16         double sisi = sc.nextInt();
17
18         System.out.println("Volume kubus: " + hitungVolume(sisi));
19         System.out.println("Luas Permukaan kubus: " + hitungLuasPermukaan(sisi));
20     }
21 }
22
```

1.

```

1  import java.util.Scanner;
2  public class CafeMariana {
3      public static void inputPenjualan(int jumlahMenu[], int hari, String namaMenu[], int penjualan[][], double
4          Scanner sc = new Scanner(System.in);
5          for(int i = 0; i < namaMenu.length; i++){
6              for (int j = 0; j < hari; j++) {
7                  System.out.print("Masukkan penjualan " + namaMenu[i] + " hari ke-" + (j+1) + ": ");
8                  penjualan[i][j] = sc.nextInt();
9                  jumlahMenu[i] += penjualan[i][j];
10             }
11             if(jualTinggi > jumlahMenu[i]){
12                 jualTinggi = jumlahMenu[i];
13                 menuJualTinggi = namaMenu[i];
14             }
15             rataPenjualan[i] =(jumlahMenu[i])/7;
16         }
17     }
18
19     public static void tampilanPenjualan(int hari, String namaMenu[], int penjualan[][], double rataPenjualan
20         for(int i = 0; i <= hari; i++){
21             System.out.println("Penjualan hari ke-" + (i+1) + ": ");
22             for (int j = 0; j < namaMenu.length; j++) {
23                 System.out.println("Penjualan menu " + namaMenu[j] + ": " + penjualan[j][i]);
24             }
25         }
26         for(int i = 0; i < namaMenu.length; i++){
27             System.out.println("Rata-rata penjualan menu " + namaMenu[i] + ": " + rataPenjualan[i]);
28         }
29         System.out.println("Menu dengan penjualan tertinggi: " + menuJualTinggi);
30     }
31
32     Run main | Debug main | Run | Debug
33     public static void main(String[] args) {
34         int penjualan[][], hari = 7, jumlahMenu[];
35         double rataPenjualan[], jualTinggi = 0;
36         String menuJualTinggi = null;
37
38         String namaMenu[] = {"Kopi", "Teh", "Es Degan", "Roti Bakar", "Gorengan"};
39         penjualan = new int[namaMenu.length][hari];
40         jumlahMenu = new int [namaMenu.length];
41         rataPenjualan = new double[namaMenu.length];
42
43         inputPenjualan(jumlahMenu, hari, namaMenu, penjualan, jualTinggi, menuJualTinggi, rataPenjualan);
44         tampilanPenjualan(hari, namaMenu, penjualan, rataPenjualan, menuJualTinggi);
45     }

```

2.