

TUGAS MATA KULIAH
PRAKTIKUM DASAR PEMROGRAMAN

JOBSHEET 11

FUNGSI 1



OLEH:

MARTA PRAMA DANISWARA

244107020205

PROGRAM STUDI D-IV TEKNIK INFORMATIKA

JURUSAN TEKNOLOGI INFORMASI

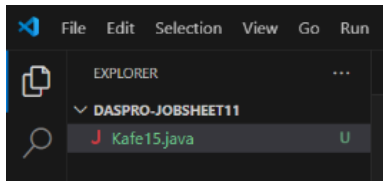
POLITEKNIK NEGERI MALANG

MALANG

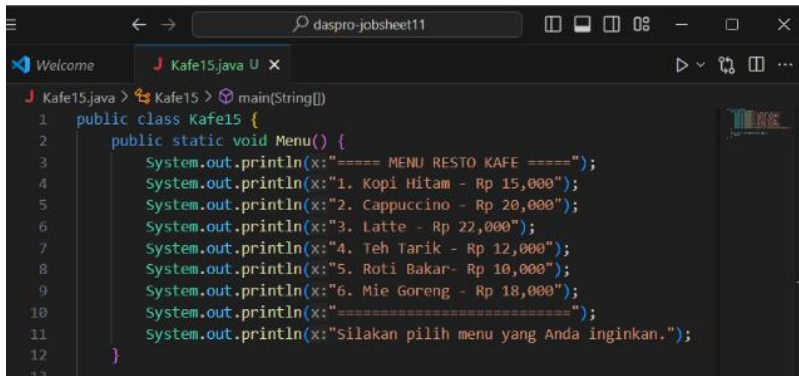
2024

2.1 PERCOBAAN 1: Membuat Fungsi Tanpa Parameter

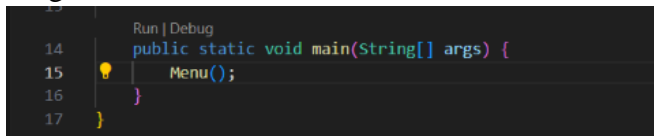
1. Buat repository baru dengan nama **daspro-jobsheet11** di GitHub. Kemudian, buat file baru dengan nama **Kafe15.java**.



2. Buat fungsi **Menu** di dalam *class* tersebut.



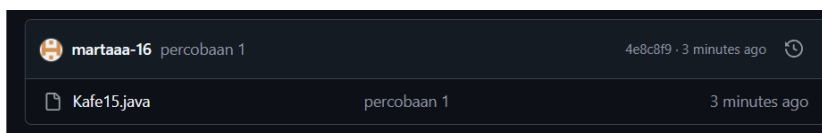
3. Buat fungsi **main** di dalam *class* tersebut, kemudian eksekusi fungsi **Menu** dari dalam fungsi **main**.



4. Run program.



5. Push dan commit ke GitHub.



Jawaban

1. Tidak, fungsi tanpa parameter tidak selalu harus menjadi fungsi void.
2. Iya, bisa.

Modifikasi kode program:

```
J Kafe15.java > Kafe15
1 public class Kafe15 {
2     public static void main(String[] args) {
3         System.out.println(x:"==== MENU RESTO KAFE ===");
4         System.out.println(x:"1. Kopi Hitam - Rp 15,000");
5         System.out.println(x:"2. Cappuccino - Rp 20,000");
6         System.out.println(x:"3. Latte - Rp 22,000");
7         System.out.println(x:"4. Teh Tarik - Rp 12,000");
8         System.out.println(x:"5. Roti Bakar- Rp 10,000");
9         System.out.println(x:"6. Mie Goreng - Rp 18,000");
10        System.out.println(x:"=====");
11        System.out.println(x:"Silakan pilih menu yang Anda inginkan.");
12    }
13 }
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
==== 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.
```

3. Keuntungan menggunakan fungsi di dalam program, yaitu:
 - Mempermudah pengelolaan, pembacaan, pengujian, dan pemeliharaan kode.
 - Membuat kode lebih modular, dapat digunakan ulang, dan efisien.

2.2 PERCOBAAN 2: Membuat Fungsi Dengan Parameter

1. Ubah fungsi **Menu** dengan dua parameter bertipe *String* dan *boolean* di dalam class **Kafe15.java**.

```
J Kafe15.java > Kafe15
1 public class Kafe15 {
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%");
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:"Silakan pilih menu yang Anda inginkan.");
18    }
19 }
```

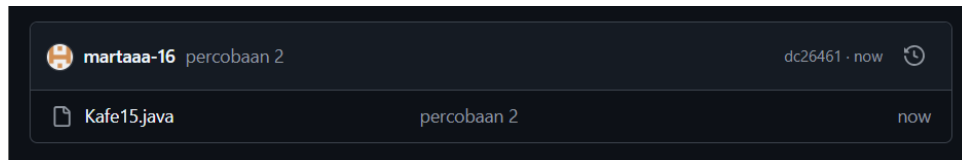
2. Eksekusi fungsi **UcapanTambahan** dari dalam fungsi *main*.

```
19
20     Run | Debug
21     public static void main(String[] args) {
22         Menu(namaPelanggan:"Andi", isMember:true);
23     }
```

3. Run program.

```
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.
```

4. Push dan commit ke GitHub.



Jawaban

1. Kegunaan parameter di dalam fungsi, yaitu:
 - Memberikan input ke fungsi.
 - Membuat fungsi lebih fleksibel dan serbaguna.
 - Mengurangi pengulangan kode, sehingga kode lebih efisien.
 - Mendukung logika berbasis input.
 - Mempermudah generalisasi fungsi untuk berbagai kebutuhan.
2. Tidak sama. **Parameter** digunakan untuk menerima nilai dari luar fungsi, sedangkan **variabel** adalah tempat penyimpanan data yang dapat digunakan untuk berbagai tujuan di dalam program.
3. Cara kerja parameter *isMember* pada method **Menu**:
 - a. Parameter *isMember* digunakan sebagai pengendali logika di dalam metode untuk menampilkan atau menyembunyikan pesan terkait keanggotaan.
 - b. Nilai *boolean* (true atau false) menentukan hasil eksekusi blok **if**:
 - true: Menampilkan pesan diskon.
 - false: Tidak menampilkan pesan diskon.
4. Yang terjadi adalah kompilasi program akan **error**.

5. Modifikasi kode program.

```
J Kafe15.java > Kafe15
1 public class Kafe15 {
2     public static void Menu(String namaPelanggan, boolean isMember, String kodePromo) {
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         if (kodePromo.equalsIgnoreCase(anotherString:"DISKON50")) {
10            System.out.println(x:"Selamat, Anda mendapatkan diskon sebesar 50%!");
11        } else if (kodePromo.equalsIgnoreCase(anotherString:"DISKON30")) {
12            System.out.println(x:"Selamat, Anda mendapatkan diskon sebesar 30%!");
13        } else {
14            System.out.println(x:"Kode invalid!");
15        }
16
17        System.out.println(x:"==== MENU RESTO KAFE =====");
18        System.out.println(x:"1. Kopi Hitam - Rp 15,000");
19        System.out.println(x:"2. Cappuccino - Rp 20,000");
20        System.out.println(x:"3. Latte - Rp 22,000");
21        System.out.println(x:"4. Teh Tarik - Rp 12,000");
22        System.out.println(x:"5. Roti Bakar- Rp 10,000");
23        System.out.println(x:"6. Mie Goreng - Rp 18,000");
24        System.out.println(x:"=====");
25        System.out.println(x:"Silakan pilih menu yang Anda inginkan.");
26    }
27
28    Run | Debug
29    public static void main(String[] args) {
30        Menu(namaPelanggan:"Andi", isMember:true, kodePromo:"Diskon50");
31    }
```

2.3 PERCOBAAN 3: Membuat Fungsi dengan Nilai Kembalian

1. Pada *class* yang sama, buat fungsi **hitungTotalHarga15** di dalam *class* tersebut yang mengembalikan nilai total harga dan parameter masukan pilihan Menu dan banyaknya pesanan.

```
28
29     public static int hitungTotalHarga15(int pilihanMenu, int banyakItem){
30         int[] hargaItems = {15000, 20000, 22000, 12000, 10000, 18000};
31         int hargaTotal = hargaItems[pilihanMenu - 1] * banyakItem;
32         return hargaTotal;
33     }
```

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

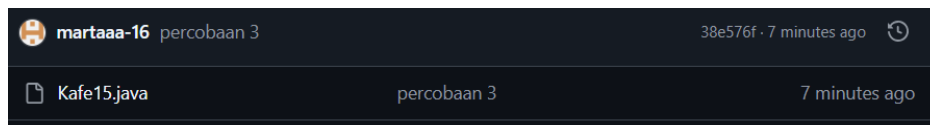
```
Run | Debug
34     public static void main(String[] args) {
35         Scanner sc = new Scanner(System.in);
36         Menu(namaPelanggan:"Andi", isMember:true, kodePromo:"DISKON50");
37         System.out.print(s:"\nMasukkan nomor menu yang ingin Anda pesan: ");
38         int pilihanMenu = sc.nextInt();
39         System.out.print(s:"Masukkan jumlah item yang ingin dipesan: ");
40         int banyakItem = sc.nextInt();
41
42         int totalHarga = hitungTotalHarga15(pilihanMenu, banyakItem);
43
44         System.out.println("Total harga untuk pesanan Anda: Rp " + totalHarga);
45     }
46 }
```

3. Run program.

```
selamat Datang, Andi!
Anda adalah member, dapatkan diskon 10% untuk setiap pembelian!
Selamat, Anda mendapatkan diskon sebesar 50%!
===== 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.

Masukkan nomor menu yang ingin Anda pesan: 4
Masukkan jumlah item yang ingin dipesan: 2
Total harga untuk pesanan Anda: Rp 24000
```

4. Push dan commit ke GitHub.



Jawaban

1. Fungsi membutuhkan nilai kembalian (return) jika:
 - Perlu memberikan hasil perhitungan atau pemrosesan.
 - Perlu mengembalikan status atau kondisi tertentu.
 - Fungsi akan digunakan untuk memberikan data yang diproses lebih lanjut.
 - Hasilnya akan digunakan oleh fungsi lain dalam rangkaian logika program.
2. Modifikasi kode program:

```
2 public class Kafe15 {
35     public static int hitungHargaSetelahDiskon(int totalHarga, String kodePromo) {
36         double diskon = 0;
37
38         if (kodePromo.equalsIgnoreCase(anotherString:"DISKON50")) {
39             diskon = 0.50;
40         } else if (kodePromo.equalsIgnoreCase(anotherString:"DISKON30")) {
41             diskon = 0.30;
42         }
43
44         return (int) (totalHarga * (1 - diskon));
45     }
46
47     public static void main(String[] args) {
48         Scanner sc = new Scanner(System.in);
49         Menu(namaPelanggan:"Andi", isMember:true, kodePromo:"DISKON50");
50         System.out.print(s:"\nMasukkan nomor menu yang ingin Anda pesan: ");
51         int pilihanMenu = sc.nextInt();
52         System.out.print(s:"Masukkan jumlah item yang ingin dipesan: ");
53         int banyakItem = sc.nextInt();
54
55         int totalHarga = hitungTotalHarga15(pilihanMenu, banyakItem);
56         int hargaSetelahDiskon = hitungHargaSetelahDiskon(totalHarga, kodePromo:"DISKON50");
57
58         System.out.println("\nTotal harga untuk pesanan Anda: Rp " + totalHarga);
59         System.out.println("Harga setelah diskon: Rp " + hargaSetelahDiskon);
60     }
}
```

Output:

```
Selamat Datang, Andi!  
Anda adalah member, dapatkan diskon 10% untuk setiap pembelian!  
Selamat, Anda mendapatkan diskon sebesar 50%!  
===== 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.  
  
Masukkan nomor menu yang ingin Anda pesan: 2  
Masukkan jumlah item yang ingin dipesan: 3  
  
Total harga untuk pesanan Anda: Rp 60000  
Harga setelah diskon: Rp 30000
```

3. Modifikasi kode program:

```
J Kafe15.java > Kafe15 > main(String[])  
2 public class Kafe15 {  
  
    Run | Debug  
    public static void main(String[] args) {  
46         Scanner sc = new Scanner(System.in);  
47         Menu(namaPelanggan:"Andi", isMember:true, kodePromo:"DISKON50");  
48  
49  
50         int totalHargaSemuaPesanan = 0;  
51         String pemesanan = "y";  
52         while (pemesanan.equals(anObject:"y")) {  
53             System.out.print(s:"\nMasukkan nomor menu yang ingin Anda pesan: ");  
54             int pilihanMenu = sc.nextInt();  
55             System.out.print(s:"Masukkan jumlah item yang ingin dipesan: ");  
56             int banyakItem = sc.nextInt();  
57  
58             int totalHarga = hitungTotalHarga15(pilihanMenu, banyakItem);  
59             totalHargaSemuaPesanan += totalHarga;  
60  
61             System.out.println("Total harga untuk pesanan ini: Rp " + totalHarga);  
62  
63             System.out.print(s:"Apakah Anda ingin memesan lagi? (y/n): ");  
64             pemesanan = sc.next();  
65         }  
66  
67         System.out.println("\nTotal keseluruhan pesanan Anda: Rp " + totalHargaSemuaPesanan);  
68  
69         int hargaSetelahDiskon = hitungHargaSetelahDiskon(totalHargaSemuaPesanan, kodePromo:"DISKON50");  
70  
71         System.out.println("Harga setelah diskon: Rp " + hargaSetelahDiskon);  
72     }  
73 }
```

Output:

```
Selamat Datang, Andi!
Anda adalah member, dapatkan diskon 10% untuk setiap pembelian!
Selamat, Anda mendapatkan diskon sebesar 50%!
===== 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.

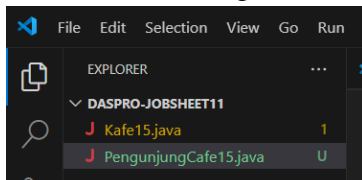
Masukkan nomor menu yang ingin Anda pesan: 1
Masukkan jumlah item yang ingin dipesan: 2
Total harga untuk pesanan ini: Rp 30000
Apakah Anda ingin memesan lagi? (y/n): y

Masukkan nomor menu yang ingin Anda pesan: 6
Masukkan jumlah item yang ingin dipesan: 3
Total harga untuk pesanan ini: Rp 54000
Apakah Anda ingin memesan lagi? (y/n): n

Total keseluruhan pesanan Anda: Rp 84000
Harga setelah diskon: Rp 42000
PS D:\PRAK-DASPRO\11\daspro-jobsheet11> █
```

2.4 PERCOBAAN 4: Fungsi Varargs

1. Buat file baru dengan nama **PengunjungCafe15.java**.



2. Buatlah fungsi **daftarPengunjung** (bertipe *void*) di dalam *class* tersebut dengan menggunakan parameter *varArgs*.

```
PengunjungCafe15.java > ...
1 public class PengunjungCafe15 {
2     public static void daftarPengunjung(String... namaPengunjung) {
3         System.out.println(x:"Daftar Nama Pengunjung: ");
4         for (int i = 0; i < namaPengunjung.length; i++) {
5             System.out.println("- " + namaPengunjung[i]);
6         }
7     }
8 }
```

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

```
Run | Debug
8 public static void main(String[] args) {
9     daftarPengunjung(...namaPengunjung:"Ali", "Budi", "Citra");
10 }
11 }
```


4. Run program.

```
Daftar Nama Pengunjung:  
- Ali  
- Budi  
- Citra
```

5. Push dan commit ke GitHub.

 marta-16 percobaan 4	d27ac4a · 1 minute ago	
 Kafe15.java	percobaan 3	9 minutes ago
 PengunjungCafe15.java	percobaan 4	1 minute ago

Jawaban

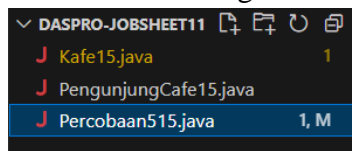
1. Karena penulisan parameter menggunakan **String... namaPengunjung** digunakan untuk membuat metode yang lebih fleksibel dan mudah diadaptasi sesuai dengan kebutuhan tanpa harus mendeklarasikan parameter dalam jumlah yang banyak.
2. Modifikasi program menggunakan **for each-loop**:

```
PengunjungCafe15.java > PengunjungCafe15  
1 public class PengunjungCafe15 {  
2     public static void daftarPengunjung(String... namaPengunjung) {  
3         System.out.println(x:"Daftar Nama Pengunjung: ");  
4         for (String nama : namaPengunjung) {  
5             System.out.println("- " + nama);  
6         }  
7     }  
8     Run | Debug  
9     public static void main(String[] args) {  
10         daftarPengunjung(...namaPengunjung:"Ali", "Budi", "Citra");  
11     }  
12 }
```

3. Tidak bisa. Hanya bisa ada satu parameter varArgs dalam satu metode.

2.5 PERCOBAAN 5: Pembuatan Kode Program, Dengan Fungsi versus Tanpa Fungsi

1. Buat file baru dengan nama **Percobaan515.java**.



2. Buatlah program untuk menghitung luas persegi panjang dan volume balok tanpa menggunakan fungsi.

```

J Percobaan515.java > Percobaan515
1 import java.util.Scanner;
2
3 public class Percobaan515 {
4     Run | Debug
5     public static void main(String[] args) {
6         Scanner input = new Scanner(System.in);
7
8         int p, l, t, L, vol;
9
10        System.out.print(s:"Masukkan panjang: ");
11        p = input.nextInt();
12        System.out.print(s:"Masukkan lebar: ");
13        l = input.nextInt();
14        System.out.print(s:"Masukkan tinggi: ");
15        t = input.nextInt();
16
17        L = p * l;
18        System.out.print("\nLuas persegi panjang adalah " + L);
19
20        vol = p * l * t;
21        System.out.print("\nVolume balok adalah " + vol);
22    }
23 }

```

3. Program menghitung luas persegi dan volume balok menggunakan fungsi:

Fungsi hitungLuas

```

J Percobaan515.java > Percobaan515 > main(String[])
1 import java.util.Scanner;
2
3 public class Percobaan515 {
4     //fungsi hitung Luas
5     public static int hitungLuas(int p, int l) {
6         int L = p * l;
7         return L;
8     }
9 }

```

Fungsi hitungVolume

```

10 public static int hitungVolume(int t, int a, int b) {
11     int vol = hitungLuas(a, b) * t;
12     return vol;
13 }
14

```

Fungsi main

```

15 Run | Debug
16 public static void main(String[] args) {
17     Scanner input = new Scanner(System.in);
18     int p, l, t, L, vol;
19     System.out.print(s:"Masukkan panjang: ");
20     p = input.nextInt();
21     System.out.print(s:"Masukkan lebar: ");
22     l = input.nextInt();
23     System.out.print(s:"Masukkan tinggi: ");
24     t = input.nextInt();
25
26     L = hitungLuas(p, l);
27     System.out.print("\nLuas persegi panjang adalah " + L);
28
29     vol = hitungVolume(t, p, l);
30     System.out.print("\nVolume balok adalah " + vol);
31 }

```

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

a. Output program tanpa fungsi

```

Masukkan panjang: 15
Masukkan lebar: 7
Masukkan tinggi: 8

Luas persegi panjang adalah 105
Volume balok adalah 840

```

- b. Output program menggunakan fungsi

```
Masukkan panjang: 15
Masukkan lebar: 7
Masukkan tinggi: 8

Luas persegi panjang adalah 105
Volume balok adalah 840
```

5. Push dan commit ke GitHub.

martaaa-16 percobaan 5		bb87c4d · now
Kafe15.java	percobaan 3	38 minutes ago
PengunjungCafe15.java	percobaan 5	11 minutes ago
Percobaan515.java	percobaan 5	now

Jawaban

- Tahapan dan urutan eksekusi program:
 - Program tanpa fungsi:
 - Deklarasi variable **p, l, t, L, vol**
 - Membaca input pengguna dengan memasukkan nilai panjang, lebar, dan tinggi.
 - Menghitung luas.
 - Menampilkan hasil perhitungan luas.
 - Menghitung volume.
 - Menampilkan hasil perhitungan volume.
 - Program menggunakan fungsi:
 - Program dimulai di method **main**.
 - Pengguna diminta memasukkan nilai panjang (p), lebar (l), dan tinggi (t).
 - Fungsi **hitungLuas(p, l)** dipanggil untuk menghitung luas persegi panjang.
 - Di dalam fungsi **hitungLuas**, luas dihitung ($L = p * l$), dan nilai L dikembalikan ke **main**.
 - Fungsi **hitungVolume(t, p, l)** dipanggil untuk menghitung volume balok.
 - Di dalam fungsi **hitungVolume**, fungsi **hitungLuas** dipanggil kembali untuk menghitung luas dasar.
 - Volume dihitung ($vol = L * t$), dan nilai *vol* dikembalikan ke **main**.
 - Program menampilkan hasil perhitungan luas dan volume.
 - Program selesai.
- Outputnya adalah

```
1234567
```

Alur programnya, yaitu:

- Deklarasi dan Definisi Fungsi
 - Fungsi **TampilHinggaKei(int i)**: mencetak angka dari 1 hingga i dalam satu baris.

- Fungsi **Jumlah(int bil1, int bil2)**: menerima dua parameter bil1 dan bil2, kemudian mengembalikan hasil penjumlahan kedua angka tersebut.
 - Fungsi **TampilJumlah(int bil1, int bil2)**: fungsi pertama akan menghitung jumlah kedua parameter dengan memanggil fungsi **Jumlah**, kemudian hasil jumlah tersebut akan diteruskan sebagai parameter untuk fungsi **TampilHinggaKei**.
- b. Alur Eksekusi Program di **main()**
- Pada saat eksekusi dimulai, program berada pada method **main()**.
 - Langkah 1: Memanggil Fungsi **Jumlah(1, 1)**:
 - Untuk menjumlahkan keduanya dan mengembalikan hasilnya ($1 + 1 = 2$).
 - Nilai yang dikembalikan (2) disimpan dalam variabel *temp*.
 - Langkah 2: Memanggil Fungsi **TampilJumlah(temp, 5)**:
 - Setelah mendapatkan nilai *temp* = 2, fungsi **TampilJumlah** dipanggil dengan parameter 2 dan 5.
 - Di dalam fungsi **TampilJumlah**, fungsi **Jumlah(2, 5)** dipanggil terlebih dahulu.
 - Fungsi **Jumlah(2, 5)** mengembalikan hasil penjumlahan $2 + 5 = 7$.
 - Langkah 3: Memanggil Fungsi **TampilHinggaKei(7)**:
 - Hasil 7 dari fungsi **Jumlah(2, 5)** kemudian diteruskan sebagai parameter ke fungsi **TampilHinggaKei(7)**.
 - Fungsi **TampilHinggaKei** akan mencetak angka dari 1 hingga 7 dalam satu baris, menghasilkan output 1234567.
3. Fungsi menggunakan **parameter** ketika kita perlu memberikan **input** atau **data eksternal** yang dibutuhkan untuk menghitung atau memproses sesuatu di dalam fungsi tersebut. Parameter memungkinkan fungsi untuk bekerja dengan berbagai nilai tanpa mengubah definisinya.
- Fungsi harus memiliki **nilai kembalian** (return) ketika fungsi tersebut **menghasilkan nilai** yang perlu digunakan atau diproses lebih lanjut oleh bagian lain dari program.

TUGAS

1. Buat sebuah class **Kubus15.java**

- Kode program:

```
J Kubus15.java > Kubus15
1  import java.util.Scanner;
2
3  public class Kubus15 {
4      //fungsi hitung Luas
5      public static int hitungLuas(int s) {
6          int L = s * s;
7          return L;
8      }
9
10     //fungsi hitung Volume
11     public static int hitungVolume(int s) {
12         int vol = hitungLuas(s) * s;
13         return vol;
14     }
15
16     //fungsi hitung Luas Permukaan
17     public static int hitungLuasPermukaan(int s) {
18         int luasPermukaan = 6 * hitungLuas(s);
19         return luasPermukaan;
20     }
21
22     //fungsi main
23     public static void main(String[] args) {
24         Scanner input = new Scanner(System.in);
25         int s, L, vol, luasPermukaan;
26         System.out.print(s: "Masukkan panjang sisi: ");
27         s = input.nextInt();
28
29         vol = hitungVolume(s);
30         System.out.print("\nVolume kubus adalah " + vol);
31
32         luasPermukaan = hitungLuasPermukaan(s);
33         System.out.print("\nLuas permukaan kubus adalah " + luasPermukaan);
34     }
35 }
```

- Output:

```
Masukkan panjang sisi: 5
Volume kubus adalah 125
Luas permukaan kubus adalah 150
```

2. Tugas 2

- Kode program:

```
J Tugas215.java > Tugas215 > inputDataPenjualan(int[][], Scanner)
1  import java.util.Scanner;
2
3  public class Tugas215 {
4      Run | Debug
5      public static void main(String[] args) {
6          int[][] dataPenjualan = {
7              {20, 20, 25, 20, 10, 60, 10}, // Kopi
8              {30, 80, 40, 10, 15, 20, 25}, // Teh
9              {5, 9, 20, 25, 10, 5, 45},    // Es Degan
10             {50, 8, 17, 18, 30, 6, 5},    // Roti Bakar
11             {15, 10, 16, 15, 10, 10, 55} // Gorengan
12         };
13
14         Scanner sc = new Scanner(System.in);
15
16         while (true) {
17             System.out.println(x: "\n===== Menu Cafe =====");
18             System.out.println(x: "1. Input Data Penjualan");
19             System.out.println(x: "2. Seluruh Data Penjualan");
20             System.out.println(x: "3. Menu dengan Penjualan Tertinggi");
21             System.out.println(x: "4. Rata-rata Penjualan Setiap Menu");
22             System.out.println(x: "5. Selesai");
23             System.out.print(s: "Pilih 1 opsi: ");
24             int pilih = sc.nextInt();
25
26             switch (pilih) {
27                 case 1:
28                     inputDataPenjualan(dataPenjualan, sc);
29                     break;
30                 case 2:
31                     seluruhDataPenjualan(dataPenjualan);
32                     break;
33                 case 3:
34                     menuPenjualanTertinggi(dataPenjualan);
35                     break;
36                 case 4:
37                     rata2PenjualanTiapMenu(dataPenjualan);
38                     break;
39                 case 5:
40                     // Selesai
41                     break;
42             }
43         }
44     }
45 }
```

```

Tugas215.java > Tugas215 > inputDataPenjualan(int[][], Scanner)
3   public class Tugas215 {
4       public static void main(String[] args) {
37           break;
38       case 5:
39           System.out.println(x:"Selesai...");
40           return;
41       default:
42           System.out.println(x:"Pilihan tidak valid. Coba ulangi lagi!");
43       }
44   }
45   }
46
47   // Fungsi untuk input data penjualan
48   public static void inputDataPenjualan(int[][] dataPenjualan, Scanner sc) {
49       System.out.println(x:"Input Data Penjualan:");
50       for (int i = 0; i < dataPenjualan.length; i++) {
51           for (int j = 0; j < dataPenjualan[i].length; j++) {
52               System.out.print("Masukkan penjualan untuk menu " + (i + 1) + ", hari ke-" + (j + 1) + ": ");
53               dataPenjualan[i][j] = sc.nextInt();
54           }
55       }
56   }
57
58   // Fungsi untuk menampilkan seluruh data penjualan
59   public static void seluruhDataPenjualan(int[][] dataPenjualan) {
60       System.out.println(x:"Data Penjualan:");
61       for (int i = 0; i < dataPenjualan.length; i++) {
62           System.out.print("Menu " + (i + 1) + ": ");
63           for (int j = 0; j < dataPenjualan[i].length; j++) {
64               System.out.print(dataPenjualan[i][j] + " ");
65           }
66           System.out.println();
67       }
68   }
69
70   // Fungsi untuk menampilkan Menu yang memiliki penjualan tertinggi
71   public static void menuPenjualanTertinggi(int[][] dataPenjualan) {
72       int maxPenjualan = 0;

```

```

Tugas215.java > Tugas215 > inputDataPenjualan(int[][], Scanner)
3   public class Tugas215 {
71       public static void menuPenjualanTertinggi(int[][] dataPenjualan) {
74
75           for (int i = 0; i < dataPenjualan.length; i++) {
76               int totalPenjualan = 0;
77               for (int j = 0; j < dataPenjualan[i].length; j++) {
78                   totalPenjualan += dataPenjualan[i][j];
79               }
80               if (totalPenjualan > maxPenjualan) {
81                   maxPenjualan = totalPenjualan;
82                   indeksMenu = i;
83               }
84           }
85
86           System.out.println("Menu dengan penjualan tertinggi adalah " + (indeksMenu + 1) + " dengan total penjualan " + maxPenjualan);
87       }
88
89       // Fungsi untuk menampilkan rata-rata penjualan untuk setiap menu
90       public static void rata2PenjualanTiapMenu(int[][] dataPenjualan) {
91           System.out.println(x:"Rata-rata penjualan untuk tiap menu:");
92           for (int i = 0; i < dataPenjualan.length; i++) {
93               int totalPenjualan = 0;
94               for (int j = 0; j < dataPenjualan[i].length; j++) {
95                   totalPenjualan += dataPenjualan[i][j];
96               }
97               double rata2 = (double) totalPenjualan / dataPenjualan[i].length;
98               System.out.println("Menu " + (i + 1) + ": " + rata2);
99           }
100       }
101   }

```

- Output:

```
===== Menu Cafe =====
1. Input Data Penjualan
2. Seluruh Data Penjualan
3. Menu dengan Penjualan Tertinggi
4. Rata-rata Penjualan Setiap Menu
5. Selesai
Pilih 1 opsi: 1
Input Data Penjualan:
Masukkan penjualan untuk menu 1, hari ke-1: 20
Masukkan penjualan untuk menu 1, hari ke-2: 20
Masukkan penjualan untuk menu 1, hari ke-3: 25
Masukkan penjualan untuk menu 1, hari ke-4: 20
Masukkan penjualan untuk menu 1, hari ke-5: 10
Masukkan penjualan untuk menu 1, hari ke-6: 60
Masukkan penjualan untuk menu 1, hari ke-7: 10
Masukkan penjualan untuk menu 2, hari ke-1: 30
Masukkan penjualan untuk menu 2, hari ke-2: 80
Masukkan penjualan untuk menu 2, hari ke-3: 40
Masukkan penjualan untuk menu 2, hari ke-4: 10
Masukkan penjualan untuk menu 2, hari ke-5: 15
Masukkan penjualan untuk menu 2, hari ke-6: 20
Masukkan penjualan untuk menu 2, hari ke-7: 25
Masukkan penjualan untuk menu 3, hari ke-1: 5
Masukkan penjualan untuk menu 3, hari ke-2: 9
Masukkan penjualan untuk menu 3, hari ke-3: 20
Masukkan penjualan untuk menu 3, hari ke-4: 25
Masukkan penjualan untuk menu 3, hari ke-5: 10
Masukkan penjualan untuk menu 3, hari ke-6: 5
Masukkan penjualan untuk menu 3, hari ke-7: 45
Masukkan penjualan untuk menu 4, hari ke-1: 50
Masukkan penjualan untuk menu 4, hari ke-2: 8
Masukkan penjualan untuk menu 4, hari ke-3: 17
Masukkan penjualan untuk menu 4, hari ke-4: 18
Masukkan penjualan untuk menu 4, hari ke-5: 10
Masukkan penjualan untuk menu 4, hari ke-6: 30
Masukkan penjualan untuk menu 4, hari ke-7: 6
Masukkan penjualan untuk menu 5, hari ke-1: 15
Masukkan penjualan untuk menu 5, hari ke-2: 10
Masukkan penjualan untuk menu 5, hari ke-3: 16
Masukkan penjualan untuk menu 5, hari ke-4: 15
Masukkan penjualan untuk menu 5, hari ke-5: 10
```

```
Masukkan penjualan untuk menu 5, hari ke-5: 10
Masukkan penjualan untuk menu 5, hari ke-6: 10
Masukkan penjualan untuk menu 5, hari ke-7: 55
```

```
===== Menu Cafe =====
1. Input Data Penjualan
2. Seluruh Data Penjualan
3. Menu dengan Penjualan Tertinggi
4. Rata-rata Penjualan Setiap Menu
5. Selesai
Pilih 1 opsi: 2
Data Penjualan:
Menu 1: 20 20 25 20 10 60 10
Menu 2: 30 80 40 10 15 20 25
Menu 3: 5 9 20 25 10 5 45
Menu 4: 50 8 17 18 10 30 6
Menu 5: 15 10 16 15 10 10 55
```

```
===== Menu Cafe =====
1. Input Data Penjualan
2. Seluruh Data Penjualan
3. Menu dengan Penjualan Tertinggi
4. Rata-rata Penjualan Setiap Menu
5. Selesai
Pilih 1 opsi: 3
Menu dengan penjualan tertinggi adalah 2 dengan total penjualan 220
```

```
===== Menu Cafe =====
1. Input Data Penjualan
2. Seluruh Data Penjualan
3. Menu dengan Penjualan Tertinggi
4. Rata-rata Penjualan Setiap Menu
5. Selesai
Pilih 1 opsi: 4
Rata-rata penjualan untuk tiap menu:
Menu 1: 23.571428571428573
Menu 2: 31.428571428571427
Menu 3: 17.0
Menu 4: 19.142857142857142
Menu 5: 18.714285714285715
```

```
===== Menu Cafe =====
1. Input Data Penjualan
1. Input Data Penjualan
2. Seluruh Data Penjualan
3. Menu dengan Penjualan Tertinggi
4. Rata-rata Penjualan Setiap Menu
5. Selesai
Pilih 1 opsi: 5
Selesai...
```

```
===== Menu Cafe =====
1. Input Data Penjualan
2. Seluruh Data Penjualan
3. Menu dengan Penjualan Tertinggi
4. Rata-rata Penjualan Setiap Menu
5. Selesai
Pilih 1 opsi: 7
Pilihan tidak valid. Coba ulangi lagi!
```