

# **LAPORAN HASIL PRAKTIKUM**

Jobsheet 11



Disusun oleh:

Muhammad Hafiz

Kelas 1H/ Teknik Informatika

254107020056

**PROGRAM STUDI D-IV TEKNIK INFORMATIKA**

**JURUSAN TEKNOLOGI INFORMASI**

**POLITEKNIK NEGERI MALANG**

**2025/2026**

## 1. Percobaan 1

Kode:


```
1  package jobsheet11;
2
3  public class Kafe17 {
4      public static void Menu() {
5          System.out.println(x: "==== MENU RESTO KAFE ====");
6          System.out.println(x: "1. Kopi Hitam - Rp 15,000");
7          System.out.println(x: "2. Cappucino - Rp 20,000");
8          System.out.println(x: "3. Latte - Rp 22,000");
9          System.out.println(x: "4. Teh Tarik - Rp 12,000");
10         System.out.println(x: "5. Roti Bakar - Rp 10,000");
11         System.out.println(x: "6. Mie Goreng - Rp 18,000");
12         System.out.println(x: "=====");
13         System.out.println(x: "Silahkan pilih menu yang Anda inginkan.");
14     }
15     public static void main(String[] args) {
16         Menu();
17     }
18 }
19
20
```

Output:

```
==== MENU RESTO KAFE ====
1. Kopi Hitam - Rp 15,000
2. Cappucino - 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.
PS C:\Users\Muhammad Hafiz\PraktikumDaspro17>
```

1. Apakah fungsi tanpa parameter selalu harus bertipe void?
  - Tidak harus, tetapi jika fungsi tanpa parameter memang tidak mengembalikan nilai, maka harus menggunakan tipe 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

- 



```
1 package jobsheet11;
2
3 public class Kafe17 {
4     public static void main(String[] Args) {
5         System.out.println(x: "==== MENU RESTO KAFE =====");
6         System.out.println(x: "1. Kopi Hitam - Rp 15,000");
7         System.out.println(x: "2. Cappuccino - Rp 20,000");
8         System.out.println(x: "3. Latte - Rp 22,000");
9         System.out.println(x: "4. Teh Tarik - Rp 12,000");
10        System.out.println(x: "5. Roti Bakar - Rp 10,000");
11        System.out.println(x: "6. Mie Goreng - Rp 18,000");
12        System.out.println(x: "=====");
13        System.out.println(x: "Silahkan pilih menu yang Anda inginkan.");
14    }
15 }
16
17
18
```

3. Jelaskan keuntungan menggunakan fungsi Menu() dibandingkan menulis semua perintah penampilan menu langsung di dalam fungsi main.
  - Kode terlihat lebih rapi
  - Efisien → Tidak perlu *copy paste* kode. Cukup panggil Menu();
  - Mudah diedit
4. Uraikan secara singkat alur eksekusi program ketika fungsi Menu() dipanggil dari main (Mulai dari program dijalankan sampai daftar menu tampil di layar)
  - Program dimulai dengan menjalankan fungsi utama main. Selanjutnya, perintah Menu(); dieksekusi. Alur program melompat dari fungsi main menuju definisi fungsi Menu. Program kemudian menjalankan program secara berurutan. Pada saat inilah daftar menu muncul di terminal. Setelah mencapai penutup kurawal fungsi Menu, alur program Kembali ke dalam fungsi main. Karena tidak ada lagi kode setelah pemanggilan menu, program akhirnya berakhir.

## 2. Percobaan 2

Kode:

```
1 package jobsheet11;
2
3 public class Kafe17 {
4     public static void Menu(String namaPelanggan, boolean isMember) {
5         System.out.println("Selamat Datang, " + namaPelanggan + "!");
6
7         if (isMember){
8             System.out.println(x: "Anda adalah member, dapatkan diskon 10% untuk setiap pembelian!");
9         }
10        System.out.println(x: "==== MENU RESTO KAFE =====");
11        System.out.println(x: "1. Kopi Hitam - Rp 15,000");
12        System.out.println(x: "2. Cappuccino - Rp 20,000");
13        System.out.println(x: "3. Latte - Rp 22,000");
14        System.out.println(x: "4. Teh Tarik - Rp 12,000");
15        System.out.println(x: "5. Roti Bakar - Rp 10,000");
16        System.out.println(x: "6. Mie Goreng - Rp 18,000");
17        System.out.println(x: "=====");
18        System.out.println(x: "Silahkan pilih menu yang Anda inginkan.");
19    }
20    Run main | Debug main | Run | Debug
21    public static void main(String[] args) {
22        Menu(namaPelanggan: "Andi", isMember: true);
23    }
24 }
25 }
```

Output:

```
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.
PS C:\Users\Muhammad Hafiz\PraktikumDaspro17>
```

1. Apakah kegunaan parameter di dalam fungsi?

- Kegunaan utama parameter dalam fungsi adalah sebagai input agar fungsi tersebut lebih fleksibel, dapat digunakan kembali, dan dapat disesuaikan dengan nilai yang berbeda.

2. Jelaskan mengapa pada percobaan ini fungsi **Menu()** menggunakan parameter **namaPelanggan** dan **isMember**?
  - Penggunaan parameter tersebut bertujuan untuk membuat fungsi tersebut menjadi dinamis terhadap data pengguna yang berbeda.
3. Apakah parameter sama dengan variabel? Jelaskan.
  - Ya, parameter adalah sebuah variabel yang didefinisikan dalam *signature* sebuah metode atau konstruktor, berfungsi sebagai input untuk menerima nilai dari luar.
4. Jelaskan bagaimana cara kerja parameter **isMember** pada fungsi **Menu()**. Apa perbedaan output ketika **isMember** bernilai true dan ketika false?
  - Jika **isMember** bernilai benar, maka program akan menjalankan perintah cetak dibawahnya. Jika salah, maka program akan melewati perintah tersebut.
  - Jika **isMember** bernilai true → program akan menganggap pelanggan adalah member.
  - Jika **isMember** bernilai false → program akan menganggap pelanggan bukan member. Maka, baris kode di dalam kurung kurawal akan dilewati dan akan mencetak header menu.
5. Apa yang akan terjadi jika memanggil fungsi **Menu()** tanpa menyertakan parameter **namaPelanggan** dan **isMember**?
  - Kode akan mengalami error.
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.

```

public class Kafel7 {
    public static void Menu(String namaPelanggan, boolean isMember, String kodePromo) {
        System.out.println("Selamat Datang, " + namaPelanggan + "!");

        if (isMember){
            System.out.println(x: "Anda adalah member, dapatkan diskon 10% untuk setiap pembelian!");
        }
        if (kodePromo.equalsIgnoreCase(anotherString: "DISKON50")){
            System.out.println(x: "Berikan diskon 50%");
        }
        else if(kodePromo.equalsIgnoreCase(anotherString: "DISKON30")){
            System.out.println(x: "Berikan diskon 30%");
        }
        else {
            System.out.println(x: "Kode promo tidak valid");
        }
        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.");
    }
    Run main | Debug main | Run | Debug
    public static void main(String[] args) {
        Menu(namaPelanggan: "Andi", isMember: true, kodePromo: "DISKON50");
    }
}

```

7. Berdasarkan fungsi Menu() di atas, jika nama pelanggan adalah “Budi”, pelanggan tersebut member, dan menggunakan kode promo “DISKON30” tuliskan satu baris pemanggilan fungsi menu yang benar.
  - Menu(“Budi”, true, “DISKON30”);
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?
  - Menurut saya, penggunaan parameter sangat berpengaruh karena penggunaan parameter ini membuat program menjadi jauh lebih fleksibel, dapat digunakan kembali, dan mudah untuk dibaca.



### 3. Percobaan 3

Kode:

```
1 package jobsheet11;
2 import java.util.Scanner;
3 public class KafeI7 {
4     public static void Menu(String namaPelanggan, boolean isMember, String kodePromo) {
5         System.out.println("Selamat Datang, " + namaPelanggan + "!");
6
7         if (isMember) {
8             System.out.println(x: "Anda adalah member, dapatkan diskon 10% untuk setiap pembelian!");
9         }
10        if (kodePromo.equalsIgnoreCase(anotherString: "DISKON50")) {
11            System.out.println(x: "Berikan diskon 50%");
12        }
13        else if (kodePromo.equalsIgnoreCase(anotherString: "DISKON30")) {
14            System.out.println(x: "Berikan diskon 30%");
15        }
16        else {
17            System.out.println(x: "Kode promo tidak valid");
18        }
19        System.out.println(x: "---- MENU RESTO KAFE ----");
20        System.out.println(x: "1. Kopi Hitam - Rp 15000");
21        System.out.println(x: "2. Cappuccino - Rp 20,000");
22        System.out.println(x: "3. Latte - Rp 22,000");
23        System.out.println(x: "4. Teh Tarik - Rp 12,000");
24        System.out.println(x: "5. Roti Bakar - Rp 10,000");
25        System.out.println(x: "6. Mie Goreng - Rp 18,000");
26        System.out.println(x: "-----");
27        System.out.println(x: "Silahkan pilih menu yang Anda inginkan.");
28    }
29    public static int hitungTotalHargaI7(int pilihanMenu, int banyakItem) {
30        int[] hargaItems = {15000, 20000, 22000, 12000, 10000, 18000};
31
32        int hargaTotal = hargaItems[pilihanMenu - 1] * banyakItem;
33        return hargaTotal;
34    }
35    public static void main(String[] args) {
36        Scanner sc = new Scanner(System.in);
37        Menu(namaPelanggan: "Andi", isMember: true, kodePromo: "DISKON50");
38        Menu(namaPelanggan: "Budi", isMember: true, kodePromo: "DISKON30");
39
40        System.out.print(s: "\nMasukkan nomor menu yang ingin Anda pesani: ");
41        int pilihanMenu = sc.nextInt();
42        System.out.print(s: "Masukkan jumlah item yang ingin dipesan: ");
43        int banyakItem = sc.nextInt();
44
45        int totalHarga = hitungTotalHargaI7(pilihanMenu, banyakItem);
46
47        System.out.println("Total harga untuk pesanan Anda: Rp " + totalHarga);
48    }
49 }
50
51
52
53
```

Output:

```
Selamat Datang, Andi!
Anda adalah member, dapatkan diskon 10% untuk setiap pembelian!
Berikan diskon 50%
===== MENU RESTO KAFE =====
1. Kopi Hitam - Rp 15000
2. Cappucino - 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.
Selamat Datang, Budi!
Anda adalah member, dapatkan diskon 10% untuk setiap pembelian!
Berikan diskon 30%
===== MENU RESTO KAFE =====
1. Kopi Hitam - Rp 15000
2. Cappucino - 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 nomor menu yang ingin Anda pesan: 1
Masukkan jumlah item yang ingin dipesan: 2
Total harga untuk pesanan Anda: Rp30000
PS C:\Users\Muhammad Hafiz\PraktikumDaspro17>
```

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.
  - Fungsi membutuhkan *return value* ketika hasil dari proses di dalam fungsi tersebut akan digunakan kembali oleh bagian program lain. Fungsi tidak membutuhkan *return value* ketika tujuannya hanya untuk melakukan sebuah aksi atau perintah langsung (seperti menampilkan teks ke layar) tanpa perlu melapor balik ke pemanggilnya.
  - Contoh dengan *return value*: Fungsi **hitungTotalHarga**. Fungsi ini menghitung harga total lalu mengembalikannya ke fungsi main. Di main, nilai tersebut ditangkap untuk ditampilkan.
  - Contoh tanpa *return value*: Fungsi **Menu**. Fungsi ini tugasnya hanya menampilkan daftar menu dan sapaan ke layar. Setelah teks muncul, tugasnya selesai. Fungsi **main** tidak perlu menyimpan “Teks menu” tersebut ke dalam variabel.



2. Fungsi **hitungTotalHarga** saat ini mengembalikan total harga berdasarkan **pilihanMenu** dan **jumlahPesanan**. Sebutkan tipe data nilai kembalian dan dua buah parameter yang digunakan untuk fungsi tersebut. Jelaskan arti masing-masing parameter dalam konteks program kafe.
- Tipe data nilai kembaliannya adalah int karena menghitung angka tanpa desimal.
  - Parameter yang digunakan: int **pilihanMenu** → Parameter ini mewakili nomor urut menu yang dipilih oleh pelanggan.
  - Parameter yang digunakan: int **banyakItem** → Parameter ini mewakili jumlah porsi atau kuantitas item yang dipesan.
3. Modifikasi kode di atas sehingga fungsi **hitungTotalHarga** 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**.

```
public static int hitungTotalHarga1(int pilihanMenu, int banyakItem, String kodePromo){
    int[] hargaItems = {15000, 20000, 22000, 12000, 10000, 18000};

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

    if (kodePromo.equalsIgnoreCase("DISKON50")){
        System.out.println(x: "Selamat! Anda mendapatkan diskon 50%");
        hargaTotal = (int) (hargaTotal * 0.5);
    }
    else if (kodePromo.equalsIgnoreCase("DISKON30")){
        System.out.println(x: "Selamat! Anda mendapatkan diskon 30%");
        hargaTotal = (int) (hargaTotal * 0.3);
    }
    else {
        System.out.println(x: "Kode tidak valid!");
    }

    return hargaTotal;
}
```

```

public static void main(String[] args) {
    Scanner sc = new Scanner(System.in);
    Menu(namaPelanggan: "Andi", isMember: true, kodePromo: "DISKON50");
    Menu(namaPelanggan: "Budi", isMember: true, kodePromo: "DISKON30");

    System.out.print(s: "\nMasukkan nomor menu yang ingin Anda pesan: ");
    int pilihanMenu = sc.nextInt();
    System.out.print(s: "Masukkan jumlah item yang ingin dipesan: ");
    int banyakItem = sc.nextInt();
    sc.nextLine();
    System.out.print(s: "Masukkan kode promo: ");
    String kodePromo = sc.nextLine();
    int totalHarga = hitungTotalHarga17(pilihanMenu, banyakItem, kodePromo);

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

```

```

Selamat Datang, Andi!
Anda adalah member, dapatkan diskon 10% untuk setiap pembelian!
==== MENU RESTO KAFE ====
1. Kopi Hitam - Rp 15000
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 nomor menu yang ingin Anda pesan: 1
Masukkan jumlah item yang ingin dipesan: 2
Masukkan kode promo: DISKON50
Selamat! Anda mendapatkan diskon 50%
Total harga untuk pesanan Anda: Rp15000
PS C:\Users\Muhammad Hafiz\PraktikumDaspro17>

```

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 menu** (misalnya menu 1 dan 3 sekaligus), dan menampilkan **total keseluruhan** pesanan (gabungan dari semua jenis menu)?

```

public static int hitungTotalHarga17(int pilihanMenu, int banyakItem){
    int[] hargaItems = {15000, 20000, 22000, 12000, 10000, 18000};

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

    return hargaTotal;
}

```

```

public static void main(String[] args) {
    Scanner sc = new Scanner(System.in);
    int totalKeseluruhan = 0;
    String lanjut;
    System.out.print(s: "Masukkan nama Pelanggan: ");
    String nama = sc.nextLine();

    Menu(nama, isMember: false);

    do{
        System.out.print(s: "\nMasukkan nomor menu yang ingin Anda pesan: ");
        int pilihanMenu = sc.nextInt();

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

        int hargaItemSementara = hitungTotalHarga17(pilihanMenu, banyakItem);
        totalKeseluruhan += hargaItemSementara;
        System.out.println("Harga item ini: Rp " + hargaItemSementara);
        System.out.println(x: "Apakah Anda ingin memesan menu lain? (y/t)");
        lanjut = sc.next();
        sc.nextLine();
    }while (lanjut.equalsIgnoreCase(anotherString: "y"));
    System.out.print(s: "Masukkan kode promo: ");
    String kodePromo = sc.nextLine();
    if (kodePromo.equalsIgnoreCase(anotherString: "DISKON50")){
        System.out.println(x: "Selamat! Anda mendapatkan diskon 50%");
        totalKeseluruhan = (int) (totalKeseluruhan * 0.5);
    }
    else if (kodePromo.equalsIgnoreCase(anotherString: "DISKON30")){
        System.out.println(x: "Selamat! Anda mendapatkan diskon 30%");
        totalKeseluruhan = (int) (totalKeseluruhan * 0.3);
    }
    else {
        System.out.println(x: "Kode tidak valid!");
    }

    System.out.println("Total keseluruhan pesanan Anda: Rp " + totalKeseluruhan);
}

```

#### 4. Percobaan 4

Kode:

```
1 package jobsheet11;
2
3 public class PengunjungCafe17 {
4     public static void daftarPengunjung(String...namaPengunjung){
5         System.out.println(x: "Daftar Nama Pengunjung:");
6         for (int i = 0; i < namaPengunjung.length; i++){
7             System.out.println("- " + namaPengunjung[i]);
8         }
9     }
10
11     Run main | Debug main | Run | Debug
12     public static void main(String[] args) {
13         daftarPengunjung(...namaPengunjung: "Ali", "Budi", "Citra");
14         daftarPengunjung(...namaPengunjung: "Ali");
15         daftarPengunjung(...namaPengunjung: "Doni", "Eti", "Fahmi", "Galih");
16     }
17 }
18
```

Output:

```
Daftar Nama Pengunjung:
- Ali
- Budi
- Citra
Daftar Nama Pengunjung:
- Ali
Daftar Nama Pengunjung:
- Doni
- Eti
- Fahmi
- Galih
PS C:\Users\Muhammad Hafiz\PraktikumDaspro17>
```

1. Jelaskan mengapa penulisan parameter di praktikum 4 ditulis dengan **String...namaPengunjung**!
  - Penulisan parameter tersebut disebut dengan variabel argument atau Varargs.
2. Modifikasi method **daftarPengunjung** menggunakan **for-each** loop.

```
public class PengunjungCafe17 {
    public static void daftarPengunjung(String...namaPengunjung){
        System.out.println(x: "Daftar Nama Pengunjung:");
        for (String nama : namaPengunjung){
            System.out.println("- " +nama);
        }
    }
}
```

- 
3. Bisakah menggunakan dua tipe data varargs dalam satu fungsi? Jelaskan jawaban Anda berdasarkan aturan varargs di Java, dan berikan contohnya.

- Tidak bisa, aturan varargs adalah hanya boleh menggunakan satu varargs dan harus di posisi terakhir.

```
// ERROR: Varargs harus di posisi terakhir
void fungsiSalahPosisi(String... nama, int umur) {
}
}
```

- 
4. Jelaskan apa yang terjadi jika fungsi daftarPengunjung dipanggil tanpa argument. Apakah program akan error saat kompilasi, error saat dijalankan, atau tetap berjalan? Jika tetap berjalan, bagaimana output yang dihasilkan?
- Program akan tetap berjalan. Output yang akan dihasilkan adalah: “Daftar Nama Pengunjung:”

Tugas:

1. Kode:

```
1 package jobsheet11;
2 import java.util.Scanner;
3
4 public class Kubus17 {
5
6
7     public double hitungVolume(double sisi) {
8         return sisi * sisi * sisi;
9     }
10
11
12     public double hitungLuasPermukaan(double s) {
13         return 6 * (s * s);
14     }
15
16     Run main | Debug main | Run | Debug
17     public static void main(String[] args) {
18
19         Scanner input = new Scanner(System.in);
20         Kubus17 kubus = new Kubus17();
21
22         System.out.print(s: "Masukkan panjang sisi kubus: ");
23         double sisi = input.nextDouble();
24
25         double v = kubus.hitungVolume(sisi);
26         double lp = kubus.hitungLuasPermukaan(sisi);
27
28         System.out.println("Volume : " + v);
29         System.out.println("Luas Permukaan : " + lp);
30     }
31 }
32
33
```

Output:

```
Masukkan panjang sisi kubus: 5
Volume : 125.0
Luas Permukaan : 150.0
PS C:\Users\Muhammad Hafiz\PraktikumDaspro17>
```



## 2. Kode:

```
4 public class NilaiMahasiswa17 {
5
6
7     public static void isianArray(int[] nilai) {
8         Scanner sc = new Scanner(System.in);
9         for (int i = 0; i < nilai.length; i++) {
10             System.out.print("Masukkan nilai mahasiswa ke-" + (i + 1) + ": ");
11             nilai[i] = sc.nextInt();
12         }
13     }
14
15
16     public static void tampilArray(int[] arr) {
17         System.out.println(x: "\nDaftar Nilai Mahasiswa:");
18         for (int i = 0; i < arr.length; i++) {
19             System.out.println("Mahasiswa ke-" + (i + 1) + ": " + arr[i]);
20         }
21     }
22
23
24     public static int hitTot(int[] arr) {
25         int total = 0;
26         for (int nilai : arr) {
27             total += nilai;
28         }
29         return total;
30     }
31
32
33     Run main | Debug main | Run | Debug
34     public static void main(String[] args) {
35         Scanner sc = new Scanner(System.in);
36
37         System.out.print(s: "Masukkan jumlah mahasiswa: ");
38         int mahasiswa = sc.nextInt();
39
40         int[] nilaiMhs = new int[mahasiswa];
41
42         isianArray(nilaiMhs);
43         tampilArray(nilaiMhs);
44
45         int totalNilai = hitTot(nilaiMhs);
46         System.out.println("\nTotal nilai seluruh mahasiswa = " + totalNilai);
47     }
48 }
49
50
```

## Output:

```
Masukkan jumlah mahasiswa: 5
Masukkan nilai mahasiswa ke-1: 65
Masukkan nilai mahasiswa ke-2: 88
Masukkan nilai mahasiswa ke-3: 87
Masukkan nilai mahasiswa ke-4: 78
Masukkan nilai mahasiswa ke-5: 44

Daftar Nilai Mahasiswa:
Mahasiswa ke-1: 65
Mahasiswa ke-2: 88
Mahasiswa ke-3: 87
Mahasiswa ke-4: 78
Mahasiswa ke-5: 44

Total nilai seluruh mahasiswa = 362
PS C:\Users\Muhammad Hafiz\PraktikumDaspro17>
```

### 3. Kode:

```
8 static void inputPenjualan(int[][] penjualanMenu, String[] namaMenu) {
9     for (int i = 0; i < penjualanMenu.length; i++) {
10         System.out.println("Input penjualan untuk menu: " + namaMenu[i]);
11         for (int hari = 0; hari < penjualanMenu[0].length; hari++) {
12             System.out.print(" Hari ke: " + (hari + 1) + ": ");
13             Scanner sc = new Scanner(System.in);
14             penjualanMenu[i][hari] = sc.nextInt();
15         }
16     }
17 }
18
19
20 static void tampilkanLabel(int[][] penjualanMenu, String[] namaMenu) {
21     System.out.println(x: "\n--- TABEL PENJUALAN ---");
22     for (int i = 0; i < penjualanMenu.length; i++) {
23         System.out.print(namaMenu[i] + ": ");
24         for (int hari = 0; hari < penjualanMenu[0].length; hari++) {
25             System.out.print(penjualanMenu[i][hari] + " ");
26         }
27         System.out.println();
28     }
29 }
30
31
32 static int hitungTotalPerMenu(int[] dataPenjualan) {
33     int total = 0;
34     for (int jumlah : dataPenjualan) {
35         total += jumlah;
36     }
37     return total;
38 }
39
40
41 static void tampilkanMenuTertinggi(int[][] penjualanMenu, String[] namaMenu) {
42     int totalTertinggi = -1;
43     String menuTertinggi = "";
44
45     for (int i = 0; i < penjualanMenu.length; i++) {
46         int totalMenuIni = hitungTotalPerMenu(penjualanMenu[i]);
47         if (totalMenuIni > totalTertinggi) {
48             totalTertinggi = totalMenuIni;
49             menuTertinggi = namaMenu[i];
50         }
51     }
52
53     System.out.println(x: "\nMenu dengan penjualan tertinggi:");
54     System.out.println(menuTertinggi + " (Total: " + totalTertinggi + ")");
55 }
```

```
58 static void tampilkanRataRataPerMenu(int[][] penjualanMenu, String[] namaMenu) {
59     System.out.println(x: "\n--- RATA-RATA PENJUALAN ---");
60     for (int i = 0; i < penjualanMenu.length; i++) {
61         double rataRata = (double) hitungTotalPerMenu(penjualanMenu[i]) / penjualanMenu[0].length;
62         System.out.println(namaMenu[i] + ": " + rataRata);
63     }
64 }
65
66
67 Run main | Debug main | Run | Debug
68 public static void main(String[] args) {
69     Scanner scanner = new Scanner(System.in);
70
71     System.out.print(s: "Masukkan jumlah menu: ");
72     int jumlahMenu = scanner.nextInt();
73     scanner.nextLine();
74
75     System.out.print(s: "Masukkan jumlah hari penjualan: ");
76     int jumlahHari = scanner.nextInt();
77     scanner.nextLine();
78
79
80     String[] namaMenu = new String[jumlahMenu];
81     System.out.println(x: "\nMasukkan nama-nama menu:");
82     for (int i = 0; i < jumlahMenu; i++) {
83         System.out.print("Menu ke: " + (i + 1) + ": ");
84         namaMenu[i] = scanner.nextLine();
85     }
86
87
88     int[][] penjualanMenu = new int[jumlahMenu][jumlahHari];
89
90     RekagPenjualanCafe17 menghitung = new RekagPenjualanCafe17();
91
92     |
93     inputPenjualan(penjualanMenu, namaMenu);
94     tampilkanLabel(penjualanMenu, namaMenu);
95     tampilkanMenuTertinggi(penjualanMenu, namaMenu);
96     tampilkanRataRataPerMenu(penjualanMenu, namaMenu);
97
98 }
99
100 }
```

Output:

```
Masukkan jumlah menu: 2
Masukkan jumlah hari penjualan: 3

Masukkan nama-nama menu:
Menu ke-1: Nasi Goreng
Menu ke-2: Mie Goreng
Input penjualan untuk menu: Nasi Goreng
  Hari ke-1: 12
  Hari ke-2: 11
  Hari ke-3: 10
Input penjualan untuk menu: Mie Goreng
  Hari ke-1: 9
  Hari ke-2: 8
  Hari ke-3: 7

=== TABEL PENJUALAN ===
Nasi Goreng: 12 11 10
Mie Goreng: 9 8 7

Menu dengan penjualan tertinggi:
Nasi Goreng (Total: 33)

=== RATA-RATA PENJUALAN ===
Nasi Goreng: 11.0
Mie Goreng: 8.0
PS C:\Users\Muhammad Hafiz\PraktikumDaspro17>
```