

JOBSCHEET 11

FUNGSI 1



Disusun oleh:

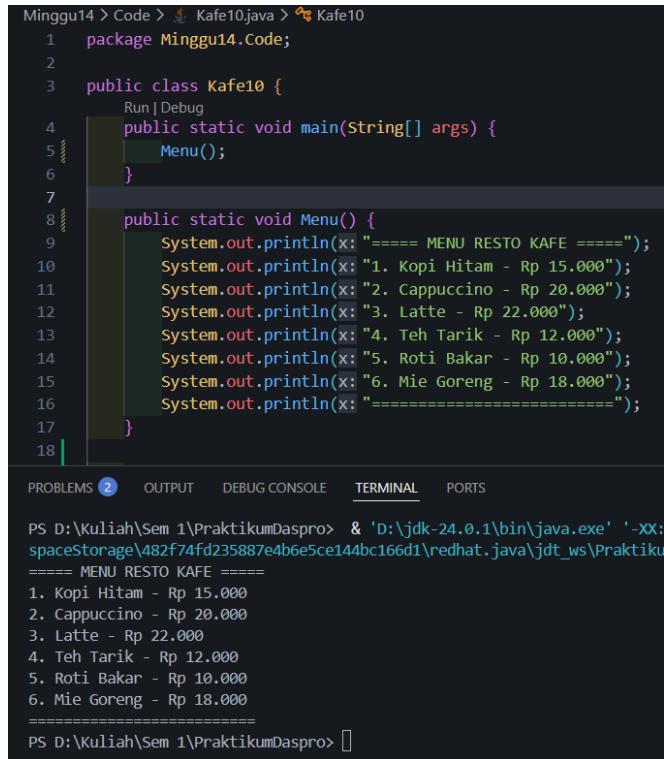
Nama : Galih Candra Kirana
No. Absen : 10
Kelas : TI 1G
NIM : 254107020080

1. Tujuan

- Mahasiswa mampu memahami penggunaan fungsi static pada Java dengan parameter dan mengembalikan nilai.
- Mahasiswa mampu membuat program menggunakan fungsi static dan mengeksekusi fungsi tersebut.

2. Praktikum

2.1. Percobaan 1: Membuat Fungsi Tanpa Parameter



```
Minggu14 > Code > Kafe10.java > Kafe10
1 package Minggu14.Code;
2
3 public class Kafe10 {
4     Run | Debug
5     public static void main(String[] args) {
6         Menu();
7     }
8
9     public static void Menu() {
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    }
18 }

PROBLEMS ② OUTPUT DEBUG CONSOLE TERMINAL PORTS

PS D:\Kuliah\Sem 1\PraktikumDaspro> & 'D:\jdk-24.0.1\bin\java.exe' '-XX:spaceStorage\482f74fd235887e4b6e5ce144bc166d1\redhat.java\jdt_ws\Praktiku
===== 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
=====
PS D:\Kuliah\Sem 1\PraktikumDaspro>
```

Pertanyaan:

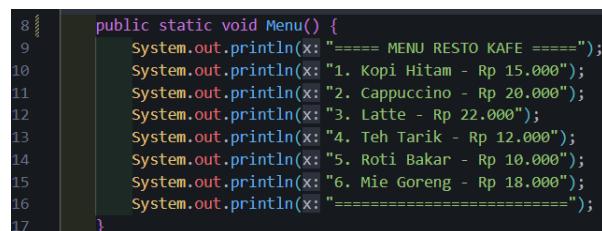
1. Apakah fungsi tanpa parameter selalu harus bertipe void?

Jawab: Tidak harus, bisa memakai tipe data lain tetapi harus direturn.

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!

Jawab: Bisa, caranya tinggal taruh daftar menu ke fungsi main.

Modifikasi:



```
8
9     public static void Menu() {
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    }
```

- 3. Jelaskan keuntungan menggunakan fungsi Menu() dibandingkan menulis semua perintah penampilan menu langsung di dalam fungsi main.**

Jawab: Keuntungannya jika menggunakan fungsi Menu() adalah bisa dipanggil berulang-ulang, lebih efektif, dan lebih modular.

- 4. Uraikan secara singkat alur eksekusi program ketika fungsi Menu() dipanggil dari main (mulai dari program dijalankan sampai daftar menu tampil di layar).**

Jawab: Uraian singkatnya ketika program dijalankan, JVM mengeksekusi fungsi main. Di dalam main terdapat pemanggilan fungsi Menu, akhirnya JVM mencari fungsi Menu dan langsung mencetak daftar menu.

2.2. Percobaan 2: Membuat Fungsi Dengan Parameter

The screenshot shows a Java code editor with the file `Kafe10.java`. The code defines a class `Kafe10` with a `main` method that calls a `Menu` method with parameters `"Andi"` and `true`. The `Menu` method prints a welcome message and a menu of items. A break point is set at the start of the `Menu` method. The terminal below shows the execution of the program, starting with the welcome message "Selamat datang, Andi!", followed by the menu items, and ending with a separator line. The terminal also shows the path `PS D:\Kuliah\Sem 1\PraktikumDaspro>`.

```
3  public class Kafe10 {  
4      Run | Debug  
5      public static void main(String[] args) {  
6          Menu(namaPelanggan: "Andi", isMember: true);  
7      }  
8  
9      public static void Menu(String namaPelanggan, boolean isMember) {  
10         System.out.println("Selamat datang, " + namaPelanggan + "!");  
11         if (isMember) {  
12             System.out.println("Anda adalah member, dapatkan diskon 10% untuk seluruh pembelian");  
13         }  
14  
15         System.out.println("===== MENU RESTO KAFE =====");  
16         System.out.println("1. Kopi Hitam - Rp 15.000");  
17         System.out.println("2. Cappuccino - Rp 20.000");  
18         System.out.println("3. Latte - Rp 22.000");  
19         System.out.println("4. Teh Tarik - Rp 12.000");  
20         System.out.println("5. Roti Bakar - Rp 10.000");  
21         System.out.println("6. Mie Goreng - Rp 18.000");  
22         System.out.println("=====");  
23     }  
24 }
```

PROBLEMS 2 OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
Selamat datang, Andi!  
Anda adalah member, dapatkan diskon 10% untuk seluruh 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  
=====
```

PS D:\Kuliah\Sem 1\PraktikumDaspro>

Pertanyaan

- 1. Apakah kegunaan parameter di dalam fungsi?**

Jawab: Parameter berguna untuk wadah sementara menampung nilai.

- 2. Jelaskan mengapa pada percobaan ini fungsi Menu() menggunakan parameter `namaPelanggan` dan `isMember`?**

Jawab: Karena kedua parameter tersebut digunakan untuk mengetahui nama pelanggan dan menanyakan apakah pelanggan tersebut member.

- 3. Apakah parameter sama dengan variabel? Jelaskan.**

Jawab: hampir mirip, tetapi berbeda dari segi asal nilai, lokasi deklarasi, dan tujuan.

- 4. Jelaskan bagaimana cara kerja parameter `isMember` pada fungsi `Menu()`. Apa perbedaan output ketika `isMember` bernilai `true` dan ketika `false`?**
- Jawab: Cara kerja parameter `isMember` adalah menanyakan apakah dia member kafe tersebut atau bukan. Jika `true` maka pelanggan tersebut akan mendapat diskon 10% untuk seluruh pembelian, jika `false` maka pelanggan tersebut tidak mendapat diskon.
- 5. Apa yang akan terjadi jika memanggil fungsi `Menu()` tanpa menyertakan parameter `namaPelanggan` dan `isMember`?**
- Jawab: Yang terjadi adalah kode program tidak bisa dijalankan karena 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.**

Modifikasi:

```

4  public static void main(String[] args) {
5    Menu(namaPelanggan: "Andi", isMember: true, kodePromo: "DISKON50");
6  }
7
8  public static void Menu(String namaPelanggan, boolean isMember, String kodePromo) {
9    System.out.println("Selamat datang, " + namaPelanggan + "!");
10
11   if (isMember) {
12     System.out.println("Anda adalah member, dapatkan diskon 10% untuk seluruh pembelian");
13   }
14   if (kodePromo.equalsIgnoreCase("DISKON50")) {
15     System.out.println("Anda mendapat diskon 50% untuk seluruh pembelian");
16   } else if (kodePromo.equalsIgnoreCase("DISKON30")) {
17     System.out.println("Anda mendapat diskon 30% untuk seluruh pembelian");
18   } else {
19     System.out.println("Kode invalid");
20   }

```

- 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.**

Implementasi:

```

4  public static void main(String[] args) {
5    |   Menu(namaPelanggan: "Budi", isMember: true, kodePromo: "DISKON30");
6  }

```

- 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.**

Jawab: Saya setuju dengan statement tersebut karena fungsi yang menggunakan parameter lebih mudah dibaca dan dikembangkan serta dapat dipanggil dengan data lain, sehingga fungsi tersebut lebih fleksibel. Berbeda dengan fungsi tanpa parameter yang kaku, sehingga tidak bisa menggunakan data pelanggan lain.

- 9. Commit dan push hasil modifikasi Anda ke Github dengan pesan "Modifikasi Percobaan 2"**

Modifikasi Percobaan 2
galihcandra committed 2 minutes ago

2.3. Percobaan 3: Membuat Fungsi dengan Nilai Kembalian

```
public static void main(String[] args) {
    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 anda pesan: ");
    int banyakItem = sc.nextInt();

    int totalHarga = hitungTotalHarga(pilihanMenu, banyakItem);

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

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

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

PROBLEMS ② OUTPUT DEBUG CONSOLE TERMINAL PORTS

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: 2
Masukkan jumlah item yang ingin anda pesan: 2
Total harga untuk pesanan anda: Rp40000

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.

Jawab: Fungsi tidak membutuhkan kembalian nilai saat tipe datanya berupa void, jika tipe data lain maka harus mengembalikan nilai. Contoh fungsi yang tidak membutuhkan nilai kembalian adalah fungsi menu. Sedangkan fungsi yang butuh nilai kembalian adalah fungsi hitungTotalHarga.

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.

Jawab: Tipe data nilai kembalian fungsi tersebut adalah int.

Parameter pertama yang ada di fungsi tersebut yaitu pilihanMenu yang berfungsi untuk menanyakan pelanggan mau membeli menu apa di kafe. Parameter kedua yaitu jumlahPesanan yang berfungsi untuk menanyakan pelanggan mau membeli berapa banyak menu yang sudah dipilih tadi.

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.

Modifikasi:

```
3  public class KafeLo {
4      static Scanner sc = new Scanner(System.in);
5      static String kodePromo;
6      Run|Debug
7      public static void main(String[] args) {
8          Menu(namaPelanggan:"Budi", isMember: true, kodePromo: "DISKON30");
9          System.out.print("Masukkan nomor menu yang ingin anda pesan: ");
10         int pilihanMenu = sc.nextInt();
11         System.out.print("Masukkan jumlah item yang ingin anda pesan: ");
12         int banyakItem = sc.nextInt();
13         sc.nextLine();
14         System.out.print("Masukkan kode promo: ");
15         kodePromo = sc.nextLine();
16
17         int totalHarga = hitungTotalHarga(pilihanMenu, banyakItem);
18
19         System.out.println("Total harga untuk pesanan anda: Rp " + totalHarga);
```

```

58     public static int hitungTotalHarga(int pilihanMenu, int banyakItem) {
59         int[] hargaItems = { 15000, 20000, 22000, 12000, 10000, 18000 };
60         int hargaTotal = hargaItems[pilihanMenu - 1] * banyakItem;
61
62         if (kodePromo.equalsIgnoreCase("DISKON50")) {
63             hargaTotal *= 0.5;
64             System.out.println("Anda mendapat diskon 50% untuk seluruh pembelian");
65         } else if (kodePromo.equalsIgnoreCase("DISKON30")) {
66             hargaTotal *= 0.3;
67             System.out.println("Anda mendapat diskon 30% untuk seluruh pembelian");
68         } else {
69             System.out.println("Kode invalid! tidak ada diskon");
70         }
71     return hargaTotal;

```

- 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)?**

Modifikasi:

```

3  public class Kafe10 {
4      static Scanner sc = new Scanner(System.in);
5      static String lanjut;
6      static int pilihanMenu, banyakItem, totalHarga = 0;
7
8      Run | Debug
9      public static void main(String[] args) {
10         Menu();
11         System.out.print("Masukkan nomor menu yang ingin anda pesan: ");
12         pilihanMenu = sc.nextInt();
13         System.out.print("Masukkan jumlah item yang ingin anda pesan: ");
14         banyakItem = sc.nextInt();
15         sc.nextLine();
16         System.out.print("Apakah ingin memesan lagi? (y/n): ");
17         lanjut = sc.nextLine();
18         totalHarga += hitungTotalHarga(pilihanMenu, banyakItem);
19     } while (!lanjut.equalsIgnoreCase("n"));
20
21     System.out.print("Masukkan kode promo: ");
22     String kodePromo = sc.nextLine();
23
24     totalHarga = hitungDiskon(totalHarga, kodePromo);
25
26     System.out.println("Total harga untuk pesanan anda: Rp" + totalHarga);
27 }

```

```

53     public static int hitungTotalHarga(int pilihanMenu, int banyakItem) {
54         int[] hargaItems = { 15000, 20000, 22000, 12000, 10000, 18000 };
55         return hargaItems[pilihanMenu - 1] * banyakItem;
56     }
57
58     public static int hitungDiskon(int total, String kodePromo) {
59         if (kodePromo.equalsIgnoreCase("DISKON50")) {
60             total *= 0.5;
61             System.out.println("Anda mendapat diskon 50% untuk seluruh pembelian");
62         } else if (kodePromo.equalsIgnoreCase("DISKON30")) {
63             total *= 0.7;
64             System.out.println("Anda mendapat diskon 30% untuk seluruh pembelian");
65         } else {
66             System.out.println("Kode invalid! tidak ada diskon");
67         }
68     }
69 }

```

- 5. Commit dan push hasil modifikasi Anda ke Github dengan pesan “Modifikasi Percobaan 3”**

Modifikasi Percobaan 3
galihcandra committed

2.4. Percobaan 4: Fungsi Varargs

```

Minggu14 > Code > PengunjungCafe10.java > PengunjungCafe10 > daftarPengunjung(String...)
● 1 package Minggu14.Code;
2
3 public class PengunjungCafe10 {
4     Run | Debug
5     public static void main(String[] args) {
6         daftarPengunjung(...namaPengunjung: "Ali", "Budi", "Citra");
7         daftarPengunjung(...namaPengunjung: "Andi");
8         daftarPengunjung(...namaPengunjung: "Doni", "Eti", "Fahmi", "Galih");
9     }
10
11     public static void daftarPengunjung(String... namaPengunjung) {
12         System.out.println("Daftar nama pengunjung: ");
13         for (int i = 0; i < namaPengunjung.length; i++) {
14             System.out.println("- " + namaPengunjung[i]);
15         }
16     }

```

PROBLEMS 2 OUTPUT DEBUG CONSOLE TERMINAL PORTS

```

Daftar nama pengunjung:
- Ali
- Budi
- Citra
Daftar nama pengunjung:
- Andi
Daftar nama pengunjung:
- Doni
- Eti
- Fahmi
- Galih

```

Pertanyaan

1. Jelaskan mengapa penulisan parameter di praktikum 4 ditulis dengan String... namaPengunjung!

Jawab: Karena parameter tersebut adalah varargs yang digunakan untuk mengatasi jumlah nilai parameter yang belum diketahui.

2. Modifikasi method daftarPengunjung menggunakan for-each loop.

Modifikasi:

```
10  public static void daftarPengunjung(String... namaPengunjung) {  
11      System.out.println("Daftar nama pengunjung: ");  
12      for (String nama : namaPengunjung) {  
13          System.out.println("- " + nama);  
14      }  
15  }
```

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

Jawab: Tidak bisa, menurut materi pertemuan 14 terdapat rule varargs yaitu setiap fungsi hanya diperkenankan memiliki 1 variable arguments dan variabel argument diletakkan di paling akhir jika memiliki lebih dari 1 parameter.

Contohnya: public void contoh(int... angka, String... nama) {} // ERROR karena ada 2 varargs.

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?

Jawab: Yang terjadi program tetap berjalan dan tidak error, tetapi saat program dijalankan tidak semua output dicetak karena ketika fungsi daftarPengunjung yang dipanggil tanpa argumen, varargs akan menjadi array kosong sehingga tidak ada nama yang dicetak dalam daftar pengunjung.

5. Commit dan push hasil modifikasi Anda ke Github dengan pesan “Modifikasi Percobaan 4”



3. Tugas

3.1. Tugas 1: Kubus

```
Minggu14 > Code > Kubus10java > Kubus10
1 package Minggu14.Code;
2
3 public class Kubus10 {
4     Run | Debug
5     public static void main(String[] args) {
6         System.out.println("===== PROGRAM HITUNG VOLUME & LUAS PERMUKAAN KUBUS =====");
7         int sisiInput = 5;
8         System.out.println("- Panjang sisi kubus: 5");
9         System.out.println("- Volume: " + hitungVolume(sisiInput) + "cm³");
10        System.out.println("- Luas Permukaan: " + hitungLuasPermukaan(sisiInput) + "cm²");
11        System.out.println("===== PROGRAM SELESAI =====");
12    }
13
14    public static int hitungVolume(int sisi) {
15        return sisi * sisi * sisi;
16    }
17
18    public static int hitungLuasPermukaan(int sisi) {
19        return 6 * sisi * sisi;
20    }
21 }
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
===== PROGRAM HITUNG VOLUME & LUAS PERMUKAAN KUBUS =====
Panjang sisi kubus: 5
- Volume: 125cm³
- Luas Permukaan: 150cm²
===== PROGRAM SELESAI =====
```

PS D:\Kuliah\Sem 1\PraktikumDaspro>

3.2. Tugas 2: NilaiMahasiswa

```
Minggu14 > Code > NilaiMahasiswa.java > NilaiMahasiswa > tampilArray(int[])
1 public class NilaiMahasiswa {
2     static Scanner sc = new Scanner(System.in);
3     static int[] mhs;
4     static int nMhs = 0;
5     Run | Debug
6     public static void main(String[] args) {
7         System.out.println("===== PROGRAM NILAI MAHASISWA =====");
8         System.out.print("Masukkan jumlah mahasiswa: ");
9         nMhs = sc.nextInt();
10
11        System.out.println("----- INPUT -----");
12        mhs = isianArray();
13        System.out.println("----- OUTPUT -----");
14        tampilArray(mhs);
15        System.out.println("Total nilai seluruh mahasiswa: " + hitTot(mhs));
16        System.out.println("===== PROGRAM SELESAI =====");
17    }
18
19    public static int[] isianArray() {
20        int[] tempMhs = new int[nMhs];
21        for (int i = 0; i < tempMhs.length; i++) {
22            System.out.print("- Input nilai mahasiswa ke " + (i + 1) + ": ");
23            tempMhs[i] = sc.nextInt();
24        }
25        return tempMhs;
26    }
27
28    public static void tampilArray(int[] nilaiMhs) {
29        for (int i = 0; i < nilaiMhs.length; i++) {
30            System.out.println("- Nilai mahasiswa ke " + (i + 1) + ": " + nilaiMhs[i]);
31        }
32    }
33 }
34
35    public static int hitTot(int[] nilaiMhs) {
36        int totalNilai = 0;
37        for (int i = 0; i < mhs.length; i++) {
38            totalNilai += nilaiMhs[i];
39        }
40        return totalNilai;
41 }
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
===== PROGRAM NILAI MAHASISWA =====
Masukkan jumlah mahasiswa: 3
----- INPUT -----
- Input nilai mahasiswa ke 1: 90
- Input nilai mahasiswa ke 2: 88
- Input nilai mahasiswa ke 3: 85
----- OUTPUT -----
- Nilai mahasiswa ke 1: 90
- Nilai mahasiswa ke 1: 90
- Nilai mahasiswa ke 2: 88
- Nilai mahasiswa ke 3: 85
Total nilai seluruh mahasiswa: 263
===== PROGRAM SELESAI =====
```

3.3. Tugas 3: Rekap Penjualan Cafe

```
Minggu14 > Code > RekapPenjualanCafe10.java > RekapPenjualanCafe10 > tampilData(int[], String[])
1 package Minggu14.Code;
2
3 import java.util.Scanner;
4
5 public class RekapPenjualanCafe10 {
6     static Scanner sc = new Scanner(System.in);
7     static int[][] rekapJual = new int[5][7];
8     static String namaMenu[] = { "Kopi", "Teh", "Es Kelapa Muda", "Roti Bakar", "Gorengan" };
9
10    Run | Debug
11    public static void main(String[] args) {
12        System.out.println(x: === REKAP PENJUALAN CAFE ===);
13        System.out.println(x: ===== INPUT =====);
14        rekapJual = inputData(namaMenu);
15
16        System.out.println(x: ===== OUTPUT =====);
17        tampilData(rekapJual, namaMenu);
18        tampilMenuTerjualTinggi(rekapJual, namaMenu);
19        tampilRata2(rekapJual, namaMenu);
20        System.out.println(x: ===== PROGRAM SELESAI =====);
21    }
22
23    public static int[][] inputData(String[] namaMenu) {
24        int[][] tempRekap = new int[5][7];
25        for (int i = 0; i < tempRekap.length; i++) {
26            System.out.println(namaMenu[i] + ": ");
27            for (int j = 0; j < tempRekap[i].length; j++) {
28                System.out.print("- hari ke " + (j + 1) + ": ");
29                tempRekap[i][j] = sc.nextInt();
30            }
31            System.out.println();
32        }
33        return tempRekap;
34    }
35
36    public static void tampilData(int[][] rekap, String[] namaMenu) {
37        System.out.println(x: ----- DATA -----);
38        int jumlahHari = rekap[0].length;
39        System.out.printf(format: "%-15s", ...args: "");
40
41        for (int h = 0; h < jumlahHari; h++) {
42            System.out.printf(format: "%-12s", "Hari ke " + (h + 1));
43        }
44        System.out.println();
45
46        for (int i = 0; i < rekap.length; i++) {
47            System.out.printf(format: "%-15s", namaMenu[i]);
48            for (int j = 0; j < rekap[i].length; j++) {
49                System.out.printf(format: "%-12d", rekap[i][j]);
50            }
51            System.out.println();
52        }
53    }
54
55    public static void tampilMenuTerjualTinggi(int[][] rekap, String[] namaMenu) {
56        int terjualTertinggi = 0;
57        int indeksMenu = 0;
58
59        for (int i = 0; i < rekap.length; i++) {
60            for (int j = 0; j < rekap[i].length; j++) {
61                if (rekap[i][j] > terjualTertinggi) {
62                    terjualTertinggi = rekap[i][j];
63                    indeksMenu = i;
64                }
65            }
66        }
67        System.out.println(x: ---- PENJUALAN TERTINGGI ----);
68        System.out.printf(format: "- Nama menu : %s\n", namaMenu[indeksMenu]);
69        System.out.printf(format: "- Total jual : %d\n", terjualTertinggi);
70
71    public static void tampilRata2(int[][] rekapData, String[] namaMenu) {
72        double[] totalData = new double[5];
73
74        System.out.println(x: ----- RATA-RATA -----);
75        for (int i = 0; i < rekapData.length; i++) {
76            for (int j = 0; j < rekapData[i].length; j++) {
77                totalData[i] += rekapData[i][j];
78            }
79            double rata2 = totalData[i] /= 7;
80            System.out.printf(format: "%-20s : %.2f\n", namaMenu[i], rata2);
81        }
82    }
83 }
```