

Nama : Agrisyandi Maraja Hutabarat

NIM : 254107020071

Matkul : Praktikum Dasar Pemrograman

Jobsheet 11

Percobaan 1

```
package jobsheet11;

public class Kafe01 {

    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");
    }

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

```
==== 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
```

Pertanyaan

1. Apakah fungsi tanpa parameter selalu harus bertipe void?
Tidak, fungsi tanpa parameter tidak harus bertipe 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!
Bisa, daftar menu dapat ditampilkan tanpa menggunakan fungsi dengan menambahkannya dalam bentuk "sout" di "main".

```
Run main | Debug main | Run | Debug
public static void main(String[] args) {
    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");
}
```

```
==== 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
```

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

Dengan menggunakan fungsi Menu(), program yang dibuat tampak lebih bersih, dan lebih fleksibel jika kita ingin mengganti detail dari menu tersebut tanpa mempengaruhi program di "main".

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

Program akan mulai mengeksekusi dari fungsi utama yaitu main, dalam fungsi main tersebut, program menemukan pemanggilan terhadap fungsi Menu() yang sudah dibuat sebelumnya, program akan memproses pemanggilan tersebut secara berurutan hingga selesai, dan akhirnya, program berakhir dengan hadirnya tampilan menu sebagai outputnya.

Percobaan 2

```
package jobsheet11;

public class Kafe01 {

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

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

```
Selamat Datang Andi!
Anda adalah member, dapatkan diskon 10% untuk setiap pembelian!
Anda telah mendapatkan tambahan diskon sebesar 30%
==== 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
```

Pertanyaan

1. Apakah kegunaan parameter di dalam fungsi?
Agar fungsi dapat menerima data yang berasal dari luar fungsi tersebut.
2. Jelaskan mengapa pada percobaan ini fungsi **Menu()** menggunakan parameter namaPelanggan dan isMember?
Karena tujuannya bukan cuma untuk menampilkan daftar menu statis, tapi juga menambahkan input-output baru yang fleksibel bagi pengguna.
3. Apakah parameter sama dengan variabel? Jelaskan.
Tidak, parameter adalah jenis khusus dari variabel yang tugas utamanya adalah menerima nilai (argumen) dari pemanggil fungsi.

4. Jelaskan bagaimana cara kerja parameter `isMember` pada fungsi **Menu()**. Apa perbedaan output ketika `isMember` bernilai `true` dan ketika `false`?

Parameter tersebut berfungsi sebagai switch boolean yang menentukan apakah suatu bagian kode tertentu (pesan diskon) harus dieksekusi atau tidak. Ketika `isMember` bernilai `true`, maka anda akan dinyatakan sebagai member, sementara jika `false`, maka tidak akan menampilkan apapun terkait member karena belum diberikan kondisinya.

5. Apa yang akan terjadi jika memanggil fungsi `Menu()` tanpa menyertakan parameter `namaPelanggan` dan `isMember`?

Program akan menjadi error atau kesalahan kompilasi.

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.
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.

```
public static void main(String[] args) {  
    Menu(namaPelanggan: "Budi", isMember: true, kodePromo: "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? Jelaskan alasan Anda.

= Ya, menurut saya, penggunaan parameter `namaPelanggan` dan `isMember` pada fungsi `Menu()` jauh lebih baik dan membuat program lebih mudah dibaca dan dikembangkan dibandingkan jika nilai-nilai tersebut ditulis langsung di dalam fungsi tanpa parameter. Karena dengan adanya parameter membuat kode program menjadi lebih fleksibel dan dinamis

9. **Commit dan push hasil modifikasi Anda ke Github dengan pesan “Modifikasi Percobaan 2!”**

jobsheet11

Modifikasi Percobaan 2

now

Percobaan 3

```
package jobsheet11;

import java.util.Scanner;

public class Kafe01 {

    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: "Anda telah mendapatkan tambahan diskon sebesar 50%");
        } else if (kodePromo.equalsIgnoreCase(anotherString: "DISKON30")) {
            System.out.println(x: "Anda telah mendapatkan tambahan diskon sebesar 30%");
        } else {
            System.out.println(x: "Kode promo yang anda masukkan tidak valid, silahkan coba lagi!");
        }
        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");
    }

    public static int hitungTotalHarga01(int pilihanMenu, int banyakItem) {
        int[] hargaItems = {15000, 20000, 22000, 12000, 10000, 18000};

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

    Run main | Debug main | Run | Debug
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        Menu(namaPelanggan: "Andi", 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();

        int totalHarga = hitungTotalHarga01(pilihanMenu, banyakItem);

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

```
Selamat Datang Andi!
Anda adalah member, dapatkan diskon 10% untuk setiap pembelian!
Anda telah mendapatkan tambahan diskon sebesar 30%
===== MENU RESTO KAFE =====
1. Kopi Hitam - Rp 15,000
2. Cappuccino - Rp 20,000
3. Latte - Rp 22,000
4. Teh Tarik - Rp 12,000
5. Roti Bakar - Rp 10,000
6. Mie Goreng - Rp 18,000
=====
Silahkan pilih menu yang Anda inginkan

Masukkan nomor menu yang ingin Anda pesan: 1
Masukkan jumlah item yang ingin dipesan: 2
Total harga untuk pesanan Anda: Rp30000
```


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.

Ketika fungsi disuruh untuk melakukan perhitungan dan menggunakan tipe data int nilai kembalian ini perlu digunakan untuk menampilkan hasil perhitungannya
Contohnya

```
public static int hitungTotalHarga01(int pilihanMenu, int banyakItem) {  
    int[] hargaItems = {15000, 20000, 22000, 12000, 10000, 18000};  
  
    int hargaTotal = hargaItems[pilihanMenu - 1] * banyakItem;  
    return hargaTotal;  
}
```

Dan ketika fungsi disuruh hanya untuk melakukan print out kata kata dan menggunakan tipe data String, Boolean, dsb. Nilai kembalian ini tidak perlu digunakan untuk menampilkan hasil

Contohnnya

```
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: "Anda telah mendapatkan tambahan diskon sebesar 50%");  
    } else if (kodePromo.equalsIgnoreCase(anotherString: "DISKON30")) {  
        System.out.println(x: "Anda telah mendapatkan tambahan diskon sebesar 30%");  
    } else {  
        System.out.println(x: "Kode promo yang anda masukkan tidak valid, silahkan coba lagi!");  
    }  
}
```

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.

Tipe data yang digunakan pada fungsi **hitungTotalHarga01** ini adalah int(integer), lalu parameternya yang pertama adalah int pilihanMenu untuk menentukan menu makanan/minuman apa yang ingin dipesan berdasarkan no urut menunya, yang kedua adalah int banyakItem yakni untuk menentukan berapa banyak item atau jumlah barang yang ingin dipesan .

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.

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)?

```
package jobsheet11;

import java.util.Scanner;

public class Kafe01 {

    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: "DISKONS0")) {
            System.out.println(x: "Anda telah mendapatkan tambahan diskon sebesar 50%");
        } else if (kodePromo.equalsIgnoreCase(anotherString: "DISKON30")) {
            System.out.println(x: "Anda telah mendapatkan tambahan diskon sebesar 30%");
        } else {
            System.out.println(x: "Kode promo yang anda masukkan tidak valid, silahkan coba lagi!");
        }
        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");
    }

    public static int hitungTotalHarga01(int pilihanMenu, int banyakItem, String kodePromo) {
        int[] hargaItems = {15000, 20000, 22000, 12000, 10000, 18000};

        if (pilihanMenu < 1 || pilihanMenu > hargaItems.length) {
            return 0;
        }

        int hargaTotal = hargaItems[pilihanMenu - 1] * banyakItem;
        double diskon = 0;
        double hargaAkhir = hargaTotal;

        if (kodePromo.equalsIgnoreCase(anotherString: "DISKONS0")) {
            diskon = hargaTotal * 0.50;
            hargaAkhir = hargaTotal - diskon;
            System.out.println(x: "Kode Promo: DISKONS0 (50%) berlaku");
            System.out.println("Besar Diskon: Rp" + diskon);
        } else if (kodePromo.equalsIgnoreCase(anotherString: "DISKON30")) {
            diskon = hargaTotal * 0.30;
            hargaAkhir = hargaTotal - diskon;
            System.out.println(x: "Kode Promo: DISKON30 (30%) berlaku");
            System.out.println("Besar Diskon: Rp" + diskon);
        } else {
            System.out.println("Kode Promo: " + kodePromo + " **INVALID**");
            System.out.println(x: "Tidak ada diskon");
        }

        return hargaTotal;
    }
}
```

```

Run main | Debug main | Run | Debug
public static void main(String[] args) {
    Scanner sc = new Scanner(System.in);
    Menu(namaPelanggan: "Budi", isMember: true, kodePromo: "DISKON50");
    String kodePromo = "DISKON50";
    int totalKeseluruhan = 0;
    String tambahMenu;

    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();
        sc.nextLine();

        int totalHarga = hitungTotalHarga01(pilihanMenu, banyakItem, kodePromo);

        if (totalHarga > 0) {
            totalKeseluruhan += totalHarga;
            System.out.println("Harga item ini: Rp" + totalHarga);
        } else {
            System.out.println(x: "Nomor menu tidak valid. Pesanan diabaikan.");
        }

        System.out.print(s: "Mau Pesan menu lain? (yes/no): ");
        tambahMenu = sc.nextLine();

    } while (tambahMenu.equalsIgnoreCase(anotherString: "yes"));

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

```

```

Selamat Datang Budi!
Anda adalah member, dapatkan diskon 10% untuk setiap pembelian!
Anda telah mendapatkan tambahan 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
=====
Silahkan pilih menu yang Anda inginkan

Masukkan nomor menu yang ingin Anda pesan: 1
Masukkan jumlah item yang ingin dipesan: 3
Kode Promo: DISKON50 (50%) berlaku
Besar Diskon: Rp22500.0
Harga item ini: Rp45000
Mau Pesan menu lain? (yes/no): no
Total harga untuk pesanan Anda: Rp45000

```

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

jobsheet11	Modifikasi Percobaan 3	now
------------	------------------------	-----

Percobaan 4

```
package jobsheet11;

public class PengunjungCafe01 {

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

    Run main | Debug main | Run | Debug
    public static void main(String[] args) {
        daftarPengunjung(...namaPengunjung: "Ali", "Budi", "Citra");
        daftarPengunjung(...namaPengunjung: "Andi");
        daftarPengunjung(...namaPengunjung: "Doni", "Eti", "Fahmi", "Galih");
    }
}
```

Dafter Nama Pengunjung:
- Ali
- Budi
- Citra
Dafter Nama Pengunjung:
- Andi
Dafter Nama Pengunjung:
- Doni
- Eti
- Fahmi
- Galih

Pertanyaan

1. Jelaskan mengapa penulisan parameter di praktikum 4 ditulis dengan String... namaPengunjung!
Fitur ini disebut sebagai Varargs (Variable Arguments) dalam Java, digunakan karena fleksibilitasnya yang memungkinkan fungsi daftarPengunjung dipanggil dengan jumlah nama yang berbeda beda, sederhana karena membuat fungsi lebih ringkas dan lebih bersih.
2. Modifikasi method daftarPengunjung menggunakan for-each loop.

```
package jobsheet11;

public class PengunjungCafe01 {

    static void daftarPengunjung(String... namaPengunjung) {
        System.out.println(x: "Dafter Nama Pengunjung: ");
        for (String nama : namaPengunjung) {
            System.out.println("- " + nama);
        }
    }

    Run main | Debug main | Run | Debug
    public static void main(String[] args) {
        daftarPengunjung(...namaPengunjung: "Ali", "Budi", "Citra");
        daftarPengunjung(...namaPengunjung: "Andi");
        daftarPengunjung(...namaPengunjung: "Doni", "Eti", "Fahmi", "Galih");
    }
}
```

```
Daftar Nama Pengunjung:  
- Ali  
- Budi  
- Citra  
Daftar Nama Pengunjung:  
- Andi  
Daftar Nama Pengunjung:  
- Doni  
- Eti  
- Fahmi  
- Galih
```

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

= Tidak, kita tidak bisa menggunakan dua tipe data Varargs dalam satu fungsi di Java. Aturan ini berlaku untuk tipe data yang sama maupun tipe data yang berbeda. Dan untuk aturan di Java varargs hanya boleh satu per fungsi dan harus ditempatkan dipaling akhir

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?

Program akan tetap berjalan tanpa terjadinya error, baik saat kompilasi maupun saat dijalankan. Untuk output tidak ada yang akan ditampilkan di hasil outputnya

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

jobsheet11	Modifikasi Percobaan 4	now
------------	------------------------	-----

Tugas

1. Buatlah sebuah class **KubusNoPresensi** yang di dalamnya terdapat fungsi untuk menghitung volume kubus dan luar permukaan kubus! **Commit dan push program Anda ke Github dengan pesan “Tugas 1”**

```
package jobsheet11;

public class Kubus01 {
    public static int hitungVolume(int sisi) {
        return sisi*sisi*sisi;
    }

    public static int hitungLuasPermukaan(int sisi) {
        return 6*sisi*sisi;
    }

    Run main | Debug main | Run | Debug
    public static void main(String[] args) {
        int panjangSisi = 10;

        System.out.println("Panjang sisi: "+ panjangSisi);

        int volume = hitungVolume(panjangSisi);
        System.out.println("Volume kubus: "+ volume);

        int luasPermukaan = hitungLuasPermukaan(panjangSisi);
        System.out.println("Luas Permukaan kubus: "+ luasPermukaan);
    }
}
```

jobsheet11 Tugas 1 1 minute ago

2. Di suatu kelas, terdapat N mahasiswa yang mengikuti satu mata kuliah. Nilai tiap mahasiswa untuk mata kuliah tersebut akan diinputkan oleh pengguna dan disimpan dalam sebuah array 1 dimensi berukuran N. Buatlah sebuah class NilaiMahasiswaNoPresensi yang di dalamnya terdapat beberapa fungsi berikut:
 - a. Fungsi isianArray untuk:
 - Mengisi elemen-elemen array dengan nilai yang diinputkan oleh user (jumlah elemen = N, dibaca dari input)
 - b. Fungsi tampilArray untuk:
 - Menerima parameter berupa array nilai mahasiswa
 - Menampilkan seluruh nilai yang telah diinputkan
 - c. Fungsi hitTot untuk:
 - Menerima parameter berupa array nilai mahasiswa
 - Menghitung dan mengembalikan total nilai seluruh mahasiswa (return tipe int atau double)
 - d. Fungsi main untuk:
 - Membaca nilai N (jumlah mahasiswa) dari input pengguna
 - Membuat array dengan ukuran N
 - Memanggil fungsi isianArray, tampilArray, dan hitTot
 - Menampilkan total nilai yang dikembalikan oleh fungsi hitTot**Commit dan push program Anda ke Github dengan pesan “Tugas 2”**

```

package jobsheet11;

import java.util.Scanner;

public class NilaiMahasiswa01 {
    public static Scanner sc = new Scanner(System.in);

    public static void isianArray(int[] nilaiArray) {
        System.out.println(x: "\nNilai Mahasiswa");
        System.out.println(x: "=====");
        for (int i = 0; i < nilaiArray.length; i++) {
            System.out.print("Masukkan Nilai Mahasiswa ke-" + (i + 1) + ": ");
            nilaiArray[i] = sc.nextInt();
        }
    }

    public static void tampilArray(int[] nilaiArray) {
        System.out.println(x: "\nDaftar Nilai Mahasiswa");
        System.out.print(s: "Nilai: [");

        for (int i = 0; i < nilaiArray.length; i++) {
            System.out.print(nilaiArray[i]);
            if (i < nilaiArray.length - 1) {
                System.out.print(s: ", ");
            }
        }
        System.out.println(x: "]");
    }

    public static double hitTot(int[] nilaiArray) {
        double total = 0;

        for (int nilai : nilaiArray) {
            total += nilai;
        }

        return total;
    }

    Run main | Debug main | Run | Debug
    public static void main(String[] args) {
        System.out.print(s: "Masukkan jumlah Mahasiswa(N): ");
        int N = sc.nextInt();

        int[] nilaiMahasiswa = new int[N];

        isianArray(nilaiMahasiswa);
        tampilArray(nilaiMahasiswa);
        double totalNilai = hitTot(nilaiMahasiswa);

        System.out.printf(format: "Total nilai seluruh Mahasiswa: %.2f\n", totalNilai);
    }
}

```


3. Ibu Mariana mengelola sebuah kafe. 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 Kelapa Muda	5	9	20	25	10	5	45
Roti Bakar	50	8	17	18	10	30	6
Gorengan	15	10	16	15	10	10	55

a. Buatlah sebuah class `RekapPenjualanCafeNoPresensi` yang digunakan untuk menyimpan dan mengolah data penjualan 5 menu kafe dari hari pertama hingga hari ketujuh. Di dalam class tersebut, buat dan gunakan beberapa fungsi (method) berikut:

- Fungsi untuk menginputkan data penjualan setiap menu untuk setiap hari (petunjuk: menggunakan array 2 dimensi).
- Fungsi untuk menampilkan seluruh data penjualan dari hari pertama hingga hari terakhir dalam bentuk tabel
- Fungsi untuk menampilkan menu yang memiliki total penjualan tertinggi selama tujuh hari (tampilkan nama menu dan total penjualannya).
- Fungsi untuk menampilkan rata-rata penjualan untuk setiap menu selama tujuh hari (tampilkan nama menu dan nilai rata-ratanya).

Gunakan fungsi main untuk:

- Memanggil fungsi input data penjualan,
- Menampilkan seluruh data penjualan,
- Menampilkan menu dengan penjualan tertinggi, dan
- Menampilkan rata-rata penjualan untuk setiap menu.

b. Modifikasi program pada 3 (a) sehingga jumlah hari penjualan dan jumlah menu dibaca dari input pengguna, lalu sesuaikan ukuran array dan proses perhitungan agar tetap berjalan dengan benar.

Commit dan push program Anda ke Github dengan pesan “Tugas 3”

```

package jobsheet11;

import java.util.Scanner;

public class RekapPenjualanCafe01 {
    public static void dataPenjualanCafe(Scanner sc, int[][] dataJual, int jmlMenu, int jmlHari) {
        sc.nextLine();

        for (int i = 0; i < jmlMenu; i++) {
            System.out.println("Input Penjualan Menu ke-" + (i + 1));
            for (int j = 0; j < jmlHari; j++) {
                System.out.print("Hari ke-" + (j + 1) + ": ");
                dataJual[i][j] = sc.nextInt();
            }
        }
    }

    public static void menampilkanData(int[][] dataJual, int jmlMenu) {
        int JML_HARI = dataJual[0].length;
        System.out.println("\n--- Rekap Penjualan ---");

        System.out.printf(format: "%-15s", ...args: "Menu");
        for (int h = 1; h <= JML_HARI; h++) {
            System.out.printf(format: "HARI %-3d", h);
        }
        System.out.println();
        for (int i = 0; i < jmlMenu; i++) {
            System.out.printf(format: "%-15s", "Menu " + (i + 1));
            for (int j = 0; j < JML_HARI; j++) {
                System.out.printf(format: "%-9d", dataJual[i][j]);
            }
            System.out.println();
        }
    }

    public static void dataPenjualanTertinggi(int[][] dataJual, int jmlMenu) {
        int menuTertinggi = 0;
        int totPenjualan = -1;

        for (int i = 0; i < dataJual.length; i++) {
            int totalSaatIni = 0;
            for (int j = 0; j < dataJual[i].length; j++) {
                totalSaatIni += dataJual[i][j];
            }

            if (totalSaatIni > totPenjualan) {
                totPenjualan = totalSaatIni;
                menuTertinggi = i;
            }
        }

        System.out.println("Menu dengan penjualan tertinggi adalah " + (menuTertinggi + 1));
        System.out.println("Total Penjualan adalah " + totPenjualan + " item");
    }
}

```

```

public static void RataRata(int[][] dataJual, int jmlMenu) {
    int jumlahHari = dataJual[0].length;
    System.out.println(x: "\n--- Rata-rata Penjualan ---");

    for (int i = 0; i < dataJual.length; i++) {
        double total = 0;
        for (int j = 0; j < jumlahHari; j++) {
            total += dataJual[i][j];
        }
        double rataRata = total / jumlahHari;

        String namaMenu = "Menu " + (i + 1);
        System.out.printf(namaMenu + " : %.2f unit\n", rataRata);
    }
}

```

Run main | Debug main | Run | Debug

```

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

    System.out.print(s: "Masukkan jumlah menu: ");
    int jmlMenu = sc.nextInt();
    System.out.print(s: "Masukkan jumlah hari penjualan: ");
    int jmlHari = sc.nextInt();

    int[][] dataPenjualan = new int[jmlMenu][jmlHari];

    dataPenjualanCafe(sc, dataPenjualan, jmlMenu, jmlHari);
    menampilkanData(dataPenjualan, jmlMenu);
    dataPenjualanTertinggi(dataPenjualan, jmlMenu);
    RataRata(dataPenjualan, jmlMenu);
}
}

```