

Nama : Galang Satriyo Anorrogo Winnada

NIM : 254107020231

Kelas : 1 – H

Mata Kuliah : Praktikum Daspro

JOBSHEET 11

1. Tujuan

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

2. Praktikum

2.1 Percobaan 1: Membuat Fungsi Tanpa Parameter

Waktu Percobaan: 40 menit

1. Buat folder baru pada repositori lokal Anda, beri nama jobsheet11. Selanjutnya buat class baru, dan simpan file tersebut dengan nama KafeNoPresensi.java.

2. Buat fungsi Menu di dalam class tersebut.

```
1 public class Kafe11 {
2     public static void menu() {
3         System.out.println("==== MENU RESTO KAFE ===");
4         System.out.println("1. Kopi Hitam - Rp 15,000 ");
5         System.out.println("2. Cappuccino - Rp 20,000 ");
6         System.out.println("3. Latte - Rp 22,000 ");
7         System.out.println("4. Teh Tarik - Rp 12,000 ");
8         System.out.println("5. Roti Bakar - Rp 10,000 ");
9         System.out.println("6. Mie Goreng - Rp 18,000");
10        System.out.println("=====");
11        System.out.println("Silahkan pilih menu yang anda inginkan. ");
12    }
13 }
14
```

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

```
==== 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.
```

4. Commit program Anda ke Github dengan pesan “Percobaan 1”

Pertanyaan

1. Apakah fungsi tanpa parameter selalu 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!
3. Jelaskan keuntungan menggunakan fungsi Menu() dibandingkan menulis semua perintah penampilan menu langsung di dalam fungsi main.
4. Uraikan secara singkat alur eksekusi program ketika fungsi Menu() dipanggil dari main (mulai dari program dijalankan sampai daftar menu tampil di layar).

Jawaban

1. Tidak Fungsi tanpa parameter tidak harus bertipe void.
Fungsi tanpa parameter tetap bisa mengembalikan nilai, misalnya int, float, atau String.
2. Ya, bisa.
Daftar menu dapat ditampilkan langsung di dalam main() tanpa membuat fungsi Menu().
3. a. Kode lebih rapi dan terstruktur
Bagian menampilkan menu dipisah ke dalam fungsi khusus, sehingga main() lebih bersih.
b. Menghindari pengulangan kode
Jika menu perlu ditampilkan lebih dari sekali, cukup memanggil:
c. Mudah diperbaiki atau dimodifikasi
Jika harga/tulisan menu berubah, cukup edit di satu tempat saja.
d. Memudahkan pembacaan dan pemahaman program
Pembaca kode langsung tahu fungsi Menu() untuk apa, tanpa harus melihat seluruh isi main.
e. Meningkatkan *reusability*
Menu bisa digunakan oleh program lain atau bagian lain tanpa mengubah struktur utama.
4. Program dijalankan, komputer mulai dari fungsi main().
Di dalam main(), komputer membaca perintah:
Program melompat ke definisi fungsi Menu().
Semua perintah di dalam Menu() dieksekusi, misalnya:
menampilkan judul menu
menampilkan daftar makanan/minuman
Setelah semua perintah di dalam fungsi Menu() selesai, program kembali lagi ke baris setelah pemanggilan fungsi di main().

Program melanjutkan eksekusi hingga main selesai.

2.2 Percobaan 2: Membuat Fungsi Dengan Parameter

Waktu Percobaan: 40 menit

1. Buka kembali class KafeNoPresensi.java yang sudah dibuat pada Percobaan 1.
2. Ubah fungsi Menu dengan dua buah parameter bertipe String dan Boolean

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

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

        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(namaPelanggan: "Andi", isMember: true);
    }
}
```

3. Eksekusi fungsi Menu() dari dalam fungsi main.

```
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.
```

4. Commit program Anda ke Github dengan pesan "Percobaan 2"

Pertanyaan

1. Apakah kegunaan parameter di dalam fungsi?
2. Jelaskan mengapa pada percobaan ini fungsi Menu() menggunakan parameter namaPelanggan dan isMember?
3. Apakah parameter sama dengan variabel? Jelaskan.
4. Jelaskan bagaimana cara kerja parameter isMember pada fungsi Menu(). Apa perbedaan output ketika isMember bernilai true dan ketika false?
5. Apa yang akan terjadi jika memanggil fungsi Menu() tanpa menyertakan parameter namaPelanggan dan isMember?
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.

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.

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

JAWABAN

1. Parameter digunakan untuk **mengirimkan nilai** dari luar fungsi agar fungsi bisa bekerja berdasarkan data yang diberikan.
Tanpa parameter, fungsi tidak bisa menerima input dari pemanggilnya.
2. Karena:
namaPelanggan digunakan untuk menampilkan nama pelanggan pada kalimat sambutan.
isMember digunakan untuk menentukan apakah pelanggan mendapat diskon member atau tidak.
Dengan parameter, fungsi bisa digunakan untuk banyak orang dan banyak kondisi (member / bukan member), tanpa mengubah isi kode fungsi.
3. Parameter mirip variabel, tetapi tidak sama.
Perbedaannya:
Parameter menerima nilai dari luar fungsi saat fungsi dipanggil.
Variabel biasa menyimpan nilai di dalam fungsi itu sendiri.
Parameter = jembatan nilai dari luar → masuk ke fungsi.
4. isMember bertipe boolean → hanya bernilai true atau false.
Jika `isMember == true`
Akan menampilkan pesan diskon member.
Output ada tulisan tentang diskon.
Jika `isMember == false`
Bagian dalam `if` tidak dijalankan.
Tidak ada pesan diskon.
5. Hasilnya: ERROR (compile-time error)
Karena fungsi membutuhkan dua parameter, jadi harus diberi nilai untuk `namaPelanggan` dan `isMember`.

```

public class Kafe1 {
    public static void menu(String namaPelanggan, boolean isMember, String kodePromo) {
        System.out.println("Selamat Datang, " + namaPelanggan + "!");

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

        if (kodePromo.equals("DISKON50")) {
            System.out.println("Kode promo valid: Anda mendapatkan diskon 50%!");
        } else if (kodePromo.equals("DISKON30")) {
            System.out.println("Kode promo valid: Anda mendapatkan diskon 30%!");
        } else {
            System.out.println("Kode promo invalid!");
        }

        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. ");
    }
}

Run | Debug
public static void main(String[] args) {
    menu(namaPelanggan: "Andir", isMember: true, kodePromo: "DISKON50");
}

```

6.

```

Selamat Datang, Andir!
Anda adalah member, dapatkan diskon 10% untuk setiap pembelian!
Kode promo valid: Anda mendapatkan diskon 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.

```

7.

8. Ya, sangat memudahkan.

Karena:

Fungsi menjadi reusable (dapat dipakai berkali-kali dengan nama dan status berbeda).

Kode lebih rapi karena nilai tidak ditulis langsung di dalam fungsi.

Saat ingin mengubah data pelanggan, cukup ubah saat pemanggilan fungsi, bukan dalam isi fungsi.

Fungsi menjadi lebih fleksibel, tidak tergantung satu nilai saja.

Tanpa parameter, fungsi jadi kaku dan tidak bisa digunakan untuk pelanggan lain.

2.3 Percobaan 3: Membuat Fungsi dengan Nilai Kembalian

1. Buka kembali class KafeNoPresensi.java yang sudah digunakan pada Percobaan 1 dan 2.

2. Pada class yang sama buat fungsi hitungTotalHargaNoPresensi di dalam class tersebut yang mengembalikan nilai total harga dan parameter masukan pilihan Menu dan banyaknya pesanan. Tambahkan daftar harga untuk setiap menu kafe dalam variabel berupa Array.

```

public class Kafe1 {
    public static void menu(String namaPelanggan, boolean isMember, String kodePromo) {
        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. ");
    }
}

Run | Debug
public static void main(String[] args) {
    menu(namaPelanggan: "Andir", isMember: true, kodePromo: "DISKON50");
    Scanner sc = new Scanner(System.in);
    System.out.println("Masukan nomor menu yang ingin anda pesan: ");
    int pilihanMenu = sc.nextInt();
    System.out.println("Masukan jumlah item yang ingin dipesan: ");
    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;
}

```

3. Modifikasi fungsi main di dalam class tersebut, dan eksekusi fungsi Menu() dari dalam fungsi main untuk menampilkan daftar menu kafe.
4. Selanjutnya modifikasi fungsi main sehingga program dapat meminta input dari pengguna untuk pilihanMenu, banyakItem (jumlah porsi yang dipesan). Fungsi main juga memanggil fungsi hitungTotalHargaNoPresensi (pilihanMenu, banyakItem) dan menyimpan nilai kembalian (return value) ke dalam variabel totalBayar, lalu program menampilkan totalBayar ke layar.
5. Jalankan program dan amati apakah nilai total yang ditampilkan sudah sesuai dengan harga dan jumlah pesanan.

```
Selamat Datang, Andir!
Anda adalah member, dapatkan diskon 10% untuk setiap pembelian!
kode promo valid: Anda mendapatkan diskon 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.

Masukan nomor menu yang ingin anda pesan:
3
Masukan jumlahitem yang ingin dipesan:
2
Total harga untuk pesanan anda: Rp44000
PS C:\Users\shelfy\OneDrive\Documents\MATA_KULIAH\KODINGAN\JOBSHEET\PraktikumDaspro>
```

6. Commit program Anda ke Github dengan pesan “Percobaan 3”

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

Jawaban

1. Fungsi membutuhkan nilai kembalian (return value) ketika:

Fungsi menghasilkan suatu nilai yang ingin dipakai kembali oleh pemanggil. Fungsi tidak memerlukan nilai kembalian ketika:

Fungsi hanya melakukan proses / menampilkan sesuatu, tanpa harus memberikan nilai balik.

2. int karena mengembalikan total harga dalam bentuk angka.

Parameter:

1. pilihanMenu (int)
Nomor menu yang dipilih pelanggan (1–6).
2. banyakItem (int)
Jumlah item yang dipesan.

Maknanya dalam konteks kafe:

- pilihanMenu menentukan makanan/minuman mana yang dihitung.
- banyakItem menentukan berapa banyak item yang dibeli.

3.

```
public class Kafe {  
    public static void menu(String namaPelanggan, boolean isMember, String kodePromo) {  
  
        System.out.println("===== MENU RESTO KAFE =====");  
        System.out.println("1. Kopi Hitam      - Rp 15,000");  
        System.out.println("2. Espresso     - Rp 20,000");  
        System.out.println("3. Latte        - Rp 22,000");  
        System.out.println("4. Tea Lada     - Rp 12,000");  
        System.out.println("5. Roti Bakar   - Rp 10,000");  
        System.out.println("6. Hias Goreng  - Rp 18,000");  
        System.out.println("=====");  
        System.out.println("Silahkan pilih menu yang anda inginkan.\n");  
    }  
  
    public static int hitungTotalHarga(int pilihanMenu, int banyakItem, String kodePromo) {  
  
        int[] hargaItem = {10000, 20000, 22000, 12000, 10000, 18000};  
        int hargaTotal = hargaItem[pilihanMenu - 1] * banyakItem;  
  
        if (kodePromo.equalsIgnoreCase("DISKON50")) {  
            System.out.println("Diskon 50% diterapkan.");  
            hargaTotal = hargaTotal / 2;  
        } else if (kodePromo.equalsIgnoreCase("DISKON30")) {  
            System.out.println("Diskon 30% diterapkan.");  
            hargaTotal = hargaTotal - (hargaTotal * 30 / 100);  
        } else {  
            System.out.println("Kode promo invalid, tidak ada diskon.");  
        }  
    }  
}
```

4.

```
public class Kafe {  
    public static int hitungTotalHarga(int pilihanMenu, int banyakItem, String kodePromo)  
    {  
        // ...  
    }  
}  
  
Run|Debug  
public static void main(String[] args) {  
  
    Scanner sc = new Scanner(System.in);  
  
    System.out.print("Masukkan nama pelanggan: ");  
    String nama = sc.nextLine();  
  
    System.out.print("Apakah Anda member? (true/false): ");  
    boolean isMember = sc.nextBoolean();  
  
    sc.nextLine();  
  
    System.out.print("Masukkan kode promo (DISKON50/DISKON30 atau lainnya): ");  
    String kodePromo = sc.nextLine();  
  
    menu(nama, isMember, kodePromo);  
  
    System.out.print("Berapa jenis menu yang ingin dipesan? ");  
    int jumlahJenis = sc.nextInt();  
  
    int totalKeseluruhan = 0;  
  
    for (int i = 1; i <= jumlahJenis; i++) {  
        System.out.println("===== Pesanan ke-" + i + " =====");  
    }  
}
```

2.4 Percobaan 4: Fungsi Varargs

Waktu Percobaan: 40 menit

1. Buat class baru, simpan file tersebut dengan nama `PengunjungCafeNoPresensi.java`.
2. Buatlah fungsi `daftarPengunjung` (bertipe `void`) di dalam class tersebut dengan menggunakan parameter `varArgs`

```
1 public class PengunjungKafe11 {
2     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 }
9
```

3. Buat fungsi main di dalam class tersebut, dan eksekusi fungsi `namaPengunjung` dari dalam fungsi main.
4. Jalankan program dan amati hasilnya

```
Daftar nama pengunjung:
-Ali
-Budi
-Citra
PS C:\Users\shelfy\OneDrive\Documents\MATA_KULIAH\KODINGAN\JOBSHEET\PraktikumDaspro>
```

5. Tambahkan dua baris berikut pada fungsi main di bawah kode pada nomor 3 di atas:
6. Jalankan program. Amati perbedaan output ketika jumlah argumen yang dikirim berbeda-beda.

```
Daftar nama pengunjung:
-Ali
-Budi
-Citra
Daftar nama pengunjung:
-ANDI
Daftar nama pengunjung:
-Doni
-Eti
-Fahmi
-Galih
PS C:\Users\shelfy\OneDrive\Documents\MATA_KULIAH\KODINGAN\JOBSHEET\PraktikumDaspro>
```

7. Commit program Anda ke Github dengan pesan “Percobaan 4”

Pertanyaan

1. Jelaskan mengapa penulisan parameter di praktikum 4 ditulis dengan `String...namaPengunjung`!
2. Modifikasi method `daftarPengunjung` menggunakan `for-each` loop.

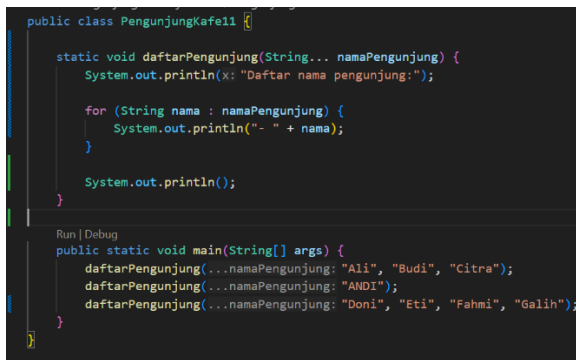
3. Bisakah menggunakan dua tipe data varargs dalam satu fungsi? Jelaskan jawaban Anda berdasarkan aturan varargs di Java, dan berikan contohnya!
4. Jelaskan apa yang terjadi jika fungsi `daftarPengunjung` dipanggil tanpa argumen. Apakah program akan error saat kompilasi, error saat dijalankan, atau tetap berjalan? Jika tetap berjalan, bagaimana output yang dihasilkan?
5. Commit dan push hasil modifikasi Anda ke Github dengan pesan "Modifikasi Percobaan 4"

Jawaban

1. Karena `String...` adalah varargs (variable arguments).

Fungsi dengan varargs boleh menerima:

- 0 argumen
- 1 argumen
- banyak argumen (tanpa batas)



```
public class PengunjungKafe11 {  
    static void daftarPengunjung(String... namaPengunjung) {  
        System.out.println(x: "Daftar nama pengunjung:");  
        for (String nama : namaPengunjung) {  
            System.out.println("- " + nama);  
        }  
        System.out.println();  
    }  
}  
  
Run | Debug  
public static void main(String[] args) {  
    daftarPengunjung(...namaPengunjung: "Ali", "Budi", "Citra");  
    daftarPengunjung(...namaPengunjung: "ANDI");  
    daftarPengunjung(...namaPengunjung: "Doni", "Eti", "Fahmi", "Galih");  
}
```

2.

3. Tidak bisa. Java hanya mengizinkan *satu varargs* per fungsi.

Aturan Java:

- Varargs hanya boleh satu dalam deklarasi metode.
- Varargs **harus berada di posisi terakhir** dari parameter.

4. Tidak error. Program tetap berjalan.

Karena varargs = array kosong ketika tidak diberikan argumen.

Tugas

1. - Implementasikan flowchart yang telah dibuat pada latihan Dasar Pemrograman (Teori) ke dalam kode program Java.

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

Jawab

1.

```
public class Tugas111 {  
    public static double hitungVolume(double sisi) {  
        return sisi * sisi * sisi;  
    }  
  
    public static double hitungLuasPermukaan(double sisi) {  
        return 6 * sisi * sisi;  
    }  
  
    public static void main(String[] args) {  
        double sisi = 5;  
  
        System.out.println("Sisi kubus: " + sisi);  
        System.out.println("Volume kubus: " + hitungVolume(sisi));  
        System.out.println("Luas permukaan kubus: " + hitungLuasPermukaan(sisi));  
    }  
}
```

```
Sisi kubus: 5.0  
Volume kubus: 125.0  
Luas permukaan kubus: 150.0
```

TUGAS 2

1. 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 “Tugas2”

JAWAB

```

Sheet1 > J:\nlaMahasiswa1\java> %nlaMahasiswa1> mainString()
4 public class nlaMahasiswa1 {
6     public static void isianArray(int[] arr, Scanner sc) {
10     }
11 }
12
13 public static void tampilArray(int[] arr) {
14     System.out.println("Daftar Nilai Mahasiswa:");
15     for (int i = 0; i < arr.length; i++) {
16         System.out.println("Mahasiswa " + (i + 1) + " : " + arr[i]);
17     }
18 }
19
20 public static int hitTot(int[] arr) {
21     int total = 0;
22     for (int nilai : arr) {
23         total += nilai;
24     }
25     return total;
26 }
27
28 Run (Debug)
29 public static void main(String[] args) {
30     Scanner sc = new Scanner(System.in);
31     System.out.print("Masukkan jumlah mahasiswa (N): ");
32     int N = sc.nextInt();
33     int[] nilai = new int[N];
34     isianArray(nilai, sc);
35     tampilArray(nilai);
36     int totalNilai = hitTot(nilai);
37     System.out.println("\nTotal nilai seluruh mahasiswa: " + totalNilai);
38 }

```

1.

```

Masukkan jumlah mahasiswa (N): 5
Masukkan nilai mahasiswa ke-1: 89
Masukkan nilai mahasiswa ke-2: 98
Masukkan nilai mahasiswa ke-3: 87
Masukkan nilai mahasiswa ke-4: 78
Masukkan nilai mahasiswa ke-5: 89

Daftar Nilai Mahasiswa:
Mahasiswa 1: 89
Mahasiswa 2: 98
Mahasiswa 3: 87
Mahasiswa 4: 78
Mahasiswa 5: 89

Total nilai seluruh mahasiswa: 441
PS C:\Users\shelly\OneDrive\Documents\WATA KULTAH\KODINGAN\JOBSHEET\PraktikumDaspro>

```

TUGAS 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

- 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 “Tugas3”

JAWAB

1.

```
Masukkan jumlah menu: 2
Masukkan jumlah hari: 2

=== Input Nama Menu ===
Nama menu ke-1: NASI GORENG
Nama menu ke-2: SEGO PECEL

=== Input Data Penjualan ===
Menu: NASI GORENG
Hari ke-1 : 20
Hari ke-2 : 32
Menu: SEGO PECEL
Hari ke-1 : 34
Hari ke-2 : 54

=== Tabel Penjualan ===
Menu/Hari      H1      H2
NASI GORENG     20     32
SEGO PECEL      34     54

Menu dengan total tertinggi: SEGO PECEL (88)

=== Rata-rata Penjualan ===
NASI GORENG : 26.0
SEGO PECEL : 44.0
PS C:\Users\shelfy\OneDrive\Documents\NATA KULTAH\KODINGAN\JOBSHEET\PraktikumDaspro>
```