

Praktek Daspro Ke-11

Nama : Aqila Herfian Fa'aizah Safwa
NIM : 254107020041
Mata Kuliah : Prakt_Daspro
Pertemuan Ke- : 11

2.1 Pertanyaan

```
public class kafe03 {  
    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("Silakan pilih menu yang Anda inginkan.");  
    }  
}
```

Run | Debug | Run main | Debug main
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  
=====  
Silakan pilih menu yang Anda inginkan.
```

1. Apakah fungsi tanpa parameter selalu harus bertipe void?
 - Tidak, fungsi tanpa parameter tidak harus memakai void. Void digunakan ketika nilai akan dikembalikan. Yang wajib adalah, jika tidak menggunakan void maka harus memakai return dengan nilai yang sesuai dengan tipe
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!

- Karena dalam menginisialisasi hanya hingga baris ke-4 dan kolom ke-1. Hal tersebut terisi null karena dalam tipe data String, nilai default berupa null/kosong.
3. Jelaskan keuntungan menggunakan fungsi Menu() dibandingkan menulis semua perintah penampilan menu langsung di dalam fungsi main
- Menu dapat dipanggil di banyak bagian program. Jika menu berubah, bisa diubah di satu tempat saja.
4. Uraikan secara singkat alur eksekusi program ketika fungsi Menu() dipanggil dari main (mulai dari program dijalankan sampai daftar menu tampil di layar).
- Java membaca baris menu.
 - Program pindah ke fungsi menu.
 - Seluruh perintah print di menu dijalankan satu per satu.
 - Judul menu tercetak, dilanjutkan dengan daftar makanan dan minuman tampil berurutan.
 - Setelah selesai, program keluar dari fungsi menu
 - Program Kembali ke main lalu selesai karena tidak ada perintah lain.

2.2 Pertanyaan

```

jobsheet11 / kafe03.java / Language Support for Java (IV) by Red Hat / kafe03 / main(String[])
1 public class kafe03 {
2     public static void Menu(String namaPelanggan, boolean isMmember) {
3         System.out.println("Selamat datang " + namaPelanggan + "!");
4
5         if (isMmember) {
6             System.out.println("Anda adalah member! dapatkan 10% untuk setiap pembelian!");
7         }
8
9         System.out.println("===== MENU RESTO KAFE =====");
10        System.out.println("1. Kopi Hitam Rp 15,000");
11        System.out.println("2. Cappuccino Rp 20,000");
12        System.out.println("3. Latte Rp 22,000");
13        System.out.println("4. Teh Tarik Rp 12,000");
14        System.out.println("5. Roti Bakar Rp 10,000");
15        System.out.println("6. Mie Goreng Rp 18,000");
16        System.out.println("===== Silakan pilih menu yang Anda inginkan.");
17    }
18
19
20    Run | Debug | Run main | Debug main
21    public static void main(String[] args) {
22        Menu("Andi", true);
23    }
24

```

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

1. Apakah kegunaan parameter di dalam fungsi?
 - Mengirim data ke dalam fungsi, membuat fungsi lebih fleksibel, memungkinkan 1 fungsi digunakan berbagai kondisi.
2. Jelaskan mengapa pada percobaan ini fungsi Menu() menggunakan parameter namaPelanggan dan isMember?
 - namaPelanggan digunakan untuk menampilkan nama pelanggan di layar. isMember digunakan untuk mengecek status member. Jika benar maka muncul pesan diskon 10%, jika tidak, pesan diskon tidak muncul.
3. Apakah parameter sama dengan variabel? Jelaskan.
 - Tidak, parameter adalah variable khusus milik fungsi. Nilainya dikirim dari luar saat fungsi dipanggil, sedangkan variable biasa dibuat dan digunakan di dalam blok program tanpa dikirim dari luar.
4. Jelaskan bagaimana cara kerja parameter isMember pada fungsi Menu(). Apa perbedaan output ketika isMember bernilai true dan ketika false?
 - isMember bertipe boolean, nilainya dikirim dari main saat menu dipanggil. Nilai ini langsung dipakai di kondisi if (isMember). Jika kondisi terpenuhi/true, program menampilkan pesan diskon.
5. Apa yang akan terjadi jika memanggil fungsi Menu() tanpa menyertakan parameter namaPelanggan dan isMember?
 - Program akan error saat di-compile karena jumlah parameter tidak sesuai.
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 kafe03 {
    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 ("DISKON50".equals(kodePromo)) {
            System.out.println("kode promo DISKON50 digunakan! anda mendapatkan diskon 50%!");
        } else if ("DISKON30".equals(kodePromo)) {
            System.out.println("kode promo DISKON30 digunakan! anda mendapatkan diskon 30%!");
        } else {
            System.out.println("kode promo invalid!");
        }
        System.out.println("===== MENU RESTO KAFE =====");
    }
}

```

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.

```

Run | Debug | Run main | Debug main
public static void main(String[] args) {
    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? Jelaskan alasan Anda.
- Ya, dengan adanya parameter program menjadi lebih mudah dibaca dan dikembangkan. Hal ini dikarenakan data yang tidak terkunci di dalam fungsi, sehingga nilai bisa berubah dari luar. Jika ingin mengembangkan juga cukup dengan menambahkan parameter saja.

2.3 Pertanyaan

```

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

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

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

    Menu ();
    System.out.print("\nMasukkan nomer menu yang ingin anda pesan: ");
    int pilihanMenu = sc.nextInt();
    System.out.print("Masukkan jumlah item yang ingin dipesan: ");
    int banyakMenu = sc.nextInt();

    int hargaTotal = hitungTotalHarga(pilihanMenu, banyakMenu);

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

```

===== MENU RESTO KAFE =====

1. Kopi Hitam Rp 15,000
2. Cappuccino Rp 20,000
3. Latte Rp 22,000
4. Teh Tarik Rp 12,000
5. Roti Bakar Rp 10,000
6. Mie Goreng Rp 18,000

=====
Silakan pilih menu yang Anda inginkan.

Masukkan nomer menu yang ingin anda pesan: 3

Masukkan jumlah item yang ingin dipesan: 2

Total harga untuk pesanan anda: Rp. 44000

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 return diperlukan jika hasilnya akan dipakai lagi di proses lain. Contoh: hitungTotalHarga mengembalikan nilai total harga.
 - Fungsi return tidak diperlukan jika hanya menampilkan output atau menjalankan aksi. Contoh: menu hanya menampilkan daftar menu ke layar.
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.

- pilihanMenu bertipe int, nomor menu yang dipilih pelanggan.
 - banyakMenu bertipe int, jumlah porsi yang dipesan dari menu tersebut.
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.

```
public static int hitungTotalHarga(int pilihanMenu, int banyakMenu, String kodePromo) {  
    int [] hargaItems = {15000, 20000, 22000, 12000, 10000, 18000};  
  
    int hargaTotal = hargaItems[pilihanMenu - 1] * banyakMenu;  
  
    if ("DISKON50".equals(kodePromo)) {  
        System.out.println("kode promo DISKON50 digunakan! anda mendapatkan diskon 50%!");  
        hargaTotal = hargaTotal - (hargaTotal * 50 / 100);  
    } else if ("DISKON30".equals(kodePromo)) {  
        System.out.println("kode promo DISKON30 digunakan! anda mendapatkan diskon 30%!");  
        hargaTotal = hargaTotal - (hargaTotal * 30 / 100);  
    } else {  
        System.out.println("kode promo invalid!");  
    }  
  
    return hargaTotal;  
}  
  
Run | Debug | Run main | Debug main  
public static void main(String[] args) {  
    Scanner sc = new Scanner(System.in);  
  
    Menu ();  
  
    System.out.print("\nMasukkan nomer menu yang ingin anda pesan: ");  
    int pilihanMenu = sc.nextInt();  
    System.out.print("Masukkan jumlah item yang ingin dipesan: ");  
    int banyakMenu = sc.nextInt();  
    sc.nextLine();  
    System.out.print("Masukkan kode promo: ");  
    String kodePromo = sc.nextLine();  
●
```

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)?

```

Run | Debug | Run main | Debug main
public static void main(String[] args) {
    Scanner sc = new Scanner(System.in);
    int hargaTotal = 0;
    Menu();

    do {
        System.out.print("\nMasukkan nomer menu yang ingin anda pesan: ");
        int pilihanMenu = sc.nextInt();
        System.out.print("Masukkan jumlah item yang ingin dipesan: ");
        int banyakMenu = sc.nextInt();
        sc.nextLine();
        System.out.print("Masukkan kode promo: ");
        String kodePromo = sc.nextLine();

        int subTotal = hitungTotalHarga(pilihanMenu, banyakMenu, kodePromo);
        hargaTotal += subTotal;

        System.out.print("Tambahkan pesanan lagi? (ya/tidak): ");
        String lanjut = sc.nextLine();

        if (lanjut.equalsIgnoreCase("tidak")) {
            break;
        }
    } while (true);
    System.out.print("Total harga untuk pesanan anda: Rp. " + hargaTotal);
}

```

2.4 Pertanyaan

```

public class pengunjungKafe03 {
    static void daftarPengunjung(String...namaPengunjung) {
        System.out.println("Daftar nama pengunjung: ");
        for (int i = 0; i < namaPengunjung.length; i++) {
            System.out.println("- " + namaPengunjung[i]);
        }
    }

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

```

```
Daftar nama pengunjung:
```

- Ali
- Budi
- Citra

```
Daftar nama pengunjung:
```

- Andi

```
Daftar nama pengunjung:
```

- Doni
- Eti
- Fahmi
- Galih

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

- Karena jumlah data yang dikirim tidak tetap.

2. Modifikasi method daftarPengunjung menggunakan for-each loop.

```
public class pengunjungKafe03 {  
    static void daftarPengunjung(String...namaPengunjung) {  
        System.out.println("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 memakai 2 varargs dalam satu fungsi. Contoh: public static void menu(String... nama, int... nilai) adalah contoh yang salah.

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 dan tidak error. Varargs otomatis dianggap sebagai array kosong. For-each tidak menampilkan apapun. Yang muncul hanyalah judul.

Tugas 1

```
1 import java.util.Scanner;
2 public class kubus03 {
3     public static int hitungVolume(int sisi){
4         int volume = sisi * sisi;
5         return volume;
6     }
7
8     public static int hitungLuasPermukaan(int sisi){
9         int luas = 6 * sisi * sisi;
10        return luas;
11    }
12
13    Run | Debug | Run main | Debug main
14    public static void main(String[] args) {
15        Scanner sc = new Scanner(System.in);
16
17        System.out.print("Masukkan panjang sisi kubus: ");
18        int sisi = sc.nextInt();
19
20        int volume = hitungVolume(sisi);
21        int luas = hitungLuasPermukaan(sisi);
22
23        System.out.println("\nVolume kubus: " + volume);
24        System.out.println("Luas kubus: " + luas);
25    }
26}
```

PROBLEMS 2 OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
$ E:\PDASPRO\github\PraktikumDaspro\jobsheet11> javac kubus03.java
$ E:\PDASPRO\github\PraktikumDaspro\jobsheet11> java kubus03
Masukkan panjang sisi kubus: 2
```

```
Volume kubus: 4
Luas kubus: 24
```

Tugas 2

```
3  public class nilaiMahasiswa03 {
4      public static void isiNilai(int[] nilai) {
7          for (int i = 0; i < nilai.length; i++) {
8              System.out.print("Masukkan nilai mahasiswa ke-" + (i + 1) + ": ");
9              nilai[i] = sc.nextInt();
10         }
11     }
12
13     public static void tampilNilai(int[] nilai) {
14         System.out.println("Daftar nilai mahasiswa: ");
15         for (int i = 0; i < nilai.length; i++) {
16             System.out.println("Mahasiswa ke-" + (i + 1) + ": " + nilai[i]);
17         }
18     }
19
20     public static int hitungTotal(int[] nilai) {
21         int total = 0;
22
23         for (int i = 0; i < nilai.length; i++) {
24             total += nilai[i];
25         }
26         return total;
27     }
28
29     Run | Debug | Run main | Debug main
30     public static void main(String[] args) {
31         Scanner sc = new Scanner(System.in);
32
33         System.out.print("Masukkan jumlah mahasiswa: ");
34         int N = sc.nextInt();
35
36         int[] nilai = new int[N];
37
38         isiNilai(nilai);
            tampilNilai(nilai);
```

Tugas 3

```
1  public class rekapPenjualanKafe03 {
2      public static void tampilData(int[][] penjualan) {
3          System.out.println("Tabel penjualan: ");
4
5          for (int i = 0; i < penjualan.length; i++) {
6              System.out.print("Menu " + (i + 1) + ": ");
7              for (int j = 0; j < penjualan[i].length; j++) {
8                  System.out.print(penjualan[i][j] + " ");
9              }
10             System.out.println();
11         }
12     }
13     public static void menuTerlaris(int[][] penjualan) {
14         int menuMax = 0;
15         int totalMax = 0;
16
17         for (int i = 0; i < penjualan.length; i++) {
18             int total = 0;
19             for (int j = 0; j < penjualan[i].length; j++) {
20                 total += penjualan[i][j];
21             }
22             if (total > totalMax) {
23                 totalMax = total;
24                 menuMax = i;
25             }
26         }
27         System.out.println("\nMenu Terlaris: Menu " + (menuMax + 1));
28         System.out.println("Total terjual: " + totalMax);
29     }
30     public static void rataMenu(int[][] penjualan) {
31         System.out.println("\nRata-rata penjualan tiap menu: ");
32
33         for (int i = 0; i < penjualan.length; i++) {
34             int total = 0;
35             for (int j = 0; j < penjualan[i].length; j++) {
36                 total += penjualan[i][j];
```