

Nama : Yudhistira Putra Hartanto

Kelas/Absen : 1G/29

NIM : 254107020083

Prodi : D-IV Teknik Informatika

Praktikum

1. Percobaan 1



```
1 package jobsheet11;
2
3 public class Kafe29 {
4     public static void main(String[] args) {
5         Menu();
6     }
7
8     public static void Menu(){
9         System.out.println("===== MENU RESTO KAFE =====");
10        System.out.println("1. Kopi Hitam - Rp 15,000");
11        System.out.println("2. Cappucino - 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("===== Silahkan pilih menu yang Anda inginkan.");
17    }
18 }
19
20
```

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

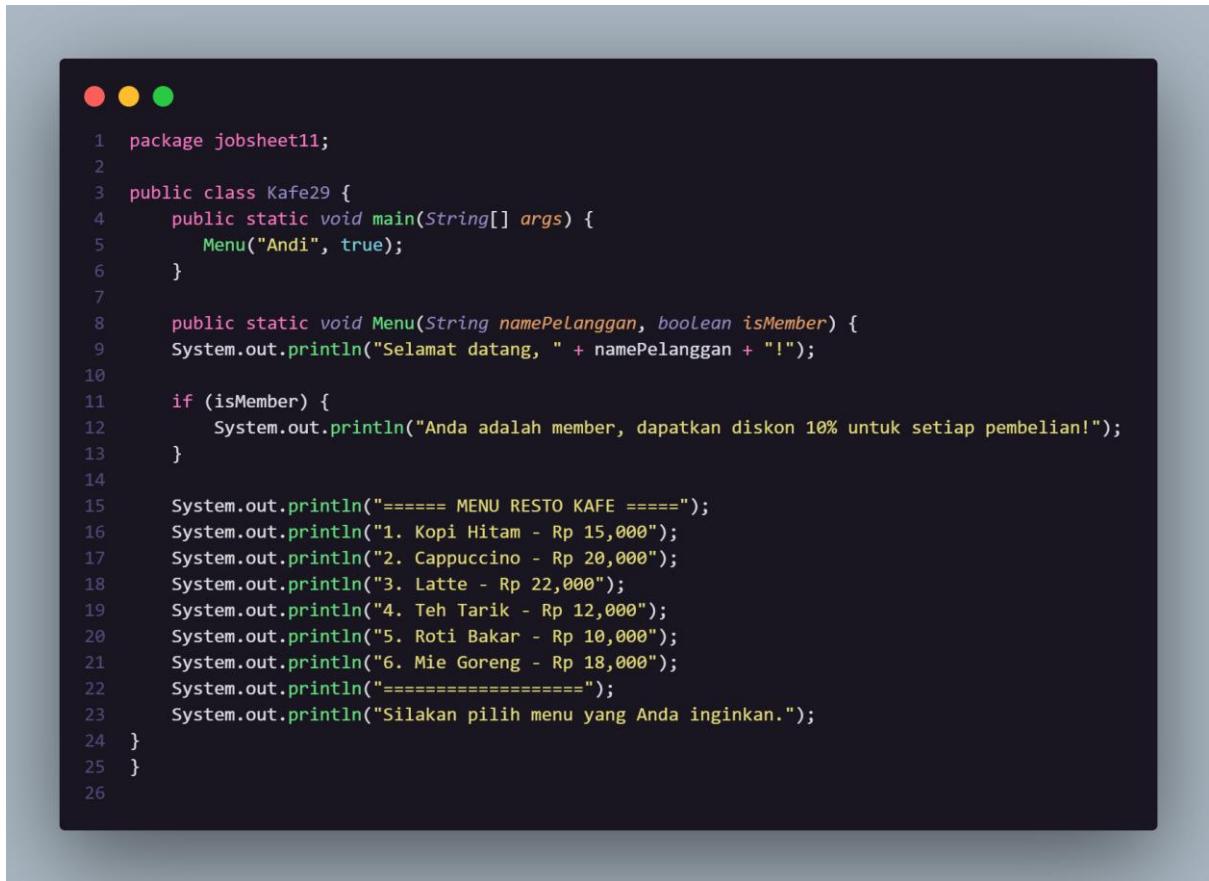
1. Tidak. Fungsi tanpa parameter tidak harus selalu bertipe void. Fungsi dapat mengembalikan nilai dengan tipe data tertentu seperti int, String, double, dll.
2. Ya bisa



```
1 package jobsheet11;
2
3 public class Kafe29 {
4     public static void main(String[] args) {
5         System.out.println("===== MENU RESTO KAFE =====");
6         System.out.println("1. Kopi Hitam - Rp 15,000");
7         System.out.println("2. Cappuccino - Rp 20,000");
8         System.out.println("3. Latte - Rp 22,000");
9         System.out.println("4. Teh Tarik - Rp 12,000");
10        System.out.println("5. Roti Bakar - Rp 10,000");
11        System.out.println("6. Mie Goreng - Rp 18,000");
12        System.out.println("=====");
13        System.out.println("Silahkan pilih menu yang Anda inginkan.");
14    }
15 }
16
```

3. Fungsi Menu() dapat dipanggil berulang kali dari berbagai bagian program tanpa menulis ulang kode yang sama. Jika ada perubahan pada menu, cukup mengubah di satu tempat saja (dalam fungsi Menu()). Kode menjadi lebih terstruktur dan mudah dibaca karena logika penampilan menu dipisahkan. Fungsi main menjadi lebih sederhana dan fokus pada alur program utama
4. Berikut adalah langkah-langkahnya :
 - a. JVM mencari metode main sebagai titik masuk program
 - b. Program mulai mengeksekusi perintah dalam fungsi main
 - c. Baris Menu(); , program:
 - i. Menyimpan posisi eksekusi saat ini di main()
 - ii. Melompat ke definisi fungsi Menu()
 - d. Program mengeksekusi semua perintah System.out.println() dalam fungsi Menu() secara berurutan
 - e. Setelah semua perintah dalam Menu() selesai dieksekusi, program kembali ke main() pada posisi setelah pemanggilan Menu()
 - f. Karena tidak ada perintah lain setelah Menu(), program berakhir

2. Percobaan 2



```
1 package jobsheet11;
2
3 public class Kafe29 {
4     public static void main(String[] args) {
5         Menu("Andi", true);
6     }
7
8     public static void Menu(String namePelanggan, boolean isMember) {
9         System.out.println("Selamat datang, " + namePelanggan + "!");
10
11     if (isMember) {
12         System.out.println("Anda adalah member, dapatkan diskon 10% untuk setiap 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     System.out.println("Silakan pilih menu yang Anda inginkan.");
24 }
25 }
26
```

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 berfungsi untuk menerima input/data dari luar fungsi, sehingga fungsi dapat bekerja dengan data yang dinamis dan dapat digunakan kembali dengan nilai yang berbeda-beda.
2.
 - namaPelanggan: Mem-personalisasi salam selamat datang dengan nama pelanggan yang berbeda-beda
 - isMember: Menentukan apakah perlu menampilkan pesan diskon khusus untuk member atau tidak
3. Tidak sama, karena parameter dideklarasikan dalam tanda kurung fungsi. Hal ini berfungsi sebagai tempat menerima nilai dari luar fungsi. Nilainya ditentukan saat fungsi dipanggil
4. Parameter isMember bertipe boolean yang menentukan eksekusi blok if. Jika isMember = true maka akan menampilkan pesan "Anda adalah member, dapatkan diskon 10% untuk setiap pembelian!". Jika isMember = false maka tidak akan menampilkan pesan diskon, langsung menampilkan menu
5. Program akan error kompilasi karena fungsi Menu() sekarang memerlukan dua parameter (String dan boolean) yang harus diikutkan saat pemanggilan.
- 6.



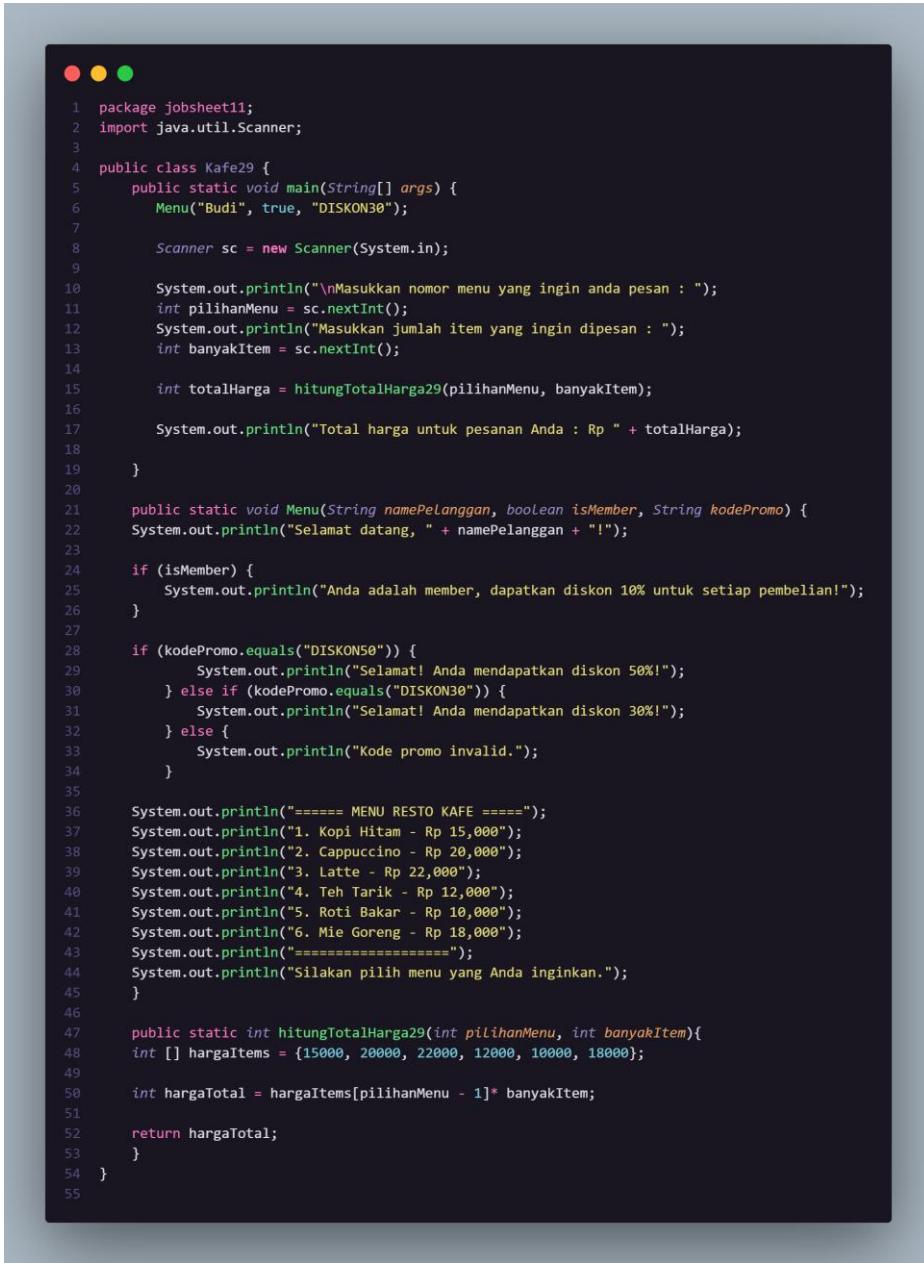
```
1 package jobsheet11;
2
3 public class Kafe29 {
4     public static void main(String[] args) {
5         Menu("Budi", true, "DISKON30");
6     }
7
8     public static void Menu(String namePelanggan, boolean isMember, String kodePromo) {
9         System.out.println("Selamat datang, " + namePelanggan + "!");
10
11        if (isMember) {
12            System.out.println("Anda adalah member, dapatkan diskon 10% untuk setiap pembelian!");
13        }
14
15        if (kodePromo.equals("DISKON50")) {
16            System.out.println("Selamat! Anda mendapatkan diskon 50%!");
17        } else if (kodePromo.equals("DISKON30")) {
18            System.out.println("Selamat! Anda mendapatkan diskon 30%!");
19        } else {
20            System.out.println("Kode promo invalid.");
21        }
22
23        System.out.println("===== MENU RESTO KAFE =====");
24        System.out.println("1. Kopi Hitam - Rp 15,000");
25        System.out.println("2. Cappuccino - Rp 20,000");
26        System.out.println("3. Latte - Rp 22,000");
27        System.out.println("4. Teh Tarik - Rp 12,000");
28        System.out.println("5. Roti Bakar - Rp 10,000");
29        System.out.println("6. Mie Goreng - Rp 18,000");
30        System.out.println("=====");
31        System.out.println("Silakan pilih menu yang Anda inginkan.");
32    }
33 }
34
```

7.

```
● ● ●
1 package jobsheet11;
2
3 public class Kafe29 {
4     public static void main(String[] args) {
5         Menu("Budi", true, "DISKON30");
6     }
7
8     public static void Menu(String namePelanggan, boolean isMember, String kodePromo) {
9         System.out.println("Selamat datang, " + namePelanggan + "!");
10
11     if (isMember) {
12         System.out.println("Anda adalah member, dapatkan diskon 10% untuk setiap pembelian!");
13     }
14
15     if (kodePromo.equals("DISKON50")) {
16         System.out.println("Selamat! Anda mendapatkan diskon 50%!");
17     } else if (kodePromo.equals("DISKON30")) {
18         System.out.println("Selamat! Anda mendapatkan diskon 30%!");
19     } else {
20         System.out.println("Kode promo invalid.");
21     }
22
23     System.out.println("===== MENU RESTO KAFE =====");
24     System.out.println("1. Kopi Hitam - Rp 15,000");
25     System.out.println("2. Cappuccino - Rp 20,000");
26     System.out.println("3. Latte - Rp 22,000");
27     System.out.println("4. Teh Tarik - Rp 12,000");
28     System.out.println("5. Roti Bakar - Rp 10,000");
29     System.out.println("6. Mie Goreng - Rp 18,000");
30     System.out.println("=====");
31     System.out.println("Silakan pilih menu yang Anda inginkan.");
32 }
33 }
34
```

8. Ya, program lebih mudah dibaca dan dikembangkan dibandingkan jika nilai-nilai tersebut ditulis langsung di dalam fungsi tanpa parameter. Karena fungsi dapat digunakan untuk banyak pelanggan dengan nama dan status member berbeda, tidak perlu mengubah kode fungsi ketika ada pelanggan baru, perubahan logika hanya dilakukan di satu tempat, fungsi menjadi lebih generik dan mudah dipahami

3. Percobaan 3



```
1 package jobsheet11;
2 import java.util.Scanner;
3
4 public class Kafe29 {
5     public static void main(String[] args) {
6         Menu("Budi", true, "DISKON30");
7
8         Scanner sc = new Scanner(System.in);
9
10        System.out.println("\nMasukkan nomor menu yang ingin anda pesan : ");
11        int pilihanMenu = sc.nextInt();
12        System.out.println("Masukkan jumlah item yang ingin dipesan : ");
13        int banyakItem = sc.nextInt();
14
15        int totalHarga = hitungTotalHarga29(pilihanMenu, banyakItem);
16
17        System.out.println("Total harga untuk pesanan Anda : Rp " + totalHarga);
18
19    }
20
21    public static void Menu(String namePelanggan, boolean isMember, String kodePromo) {
22        System.out.println("Selamat datang, " + namePelanggan + "!");
23
24        if (isMember) {
25            System.out.println("Anda adalah member, dapatkan diskon 10% untuk setiap pembelian!");
26        }
27
28        if (kodePromo.equals("DISKON50")) {
29            System.out.println("Selamat! Anda mendapatkan diskon 50%!");
30        } else if (kodePromo.equals("DISKON30")) {
31            System.out.println("Selamat! Anda mendapatkan diskon 30%!");
32        } else {
33            System.out.println("Kode promo invalid.");
34        }
35
36        System.out.println("===== MENU RESTO KAFE =====");
37        System.out.println("1. Kopi Hitam - Rp 15,000");
38        System.out.println("2. Cappuccino - Rp 20,000");
39        System.out.println("3. Latte - Rp 22,000");
40        System.out.println("4. Teh Tarik - Rp 12,000");
41        System.out.println("5. Roti Bakar - Rp 10,000");
42        System.out.println("6. Mie Goreng - Rp 18,000");
43        System.out.println("=====");
44        System.out.println("Silakan pilih menu yang Anda inginkan.");
45    }
46
47    public static int hitungTotalHarga29(int pilihanMenu, int banyakItem){
48        int [] hargaItems = {15000, 20000, 22000, 12000, 10000, 18000};
49
50        int hargaTotal = hargaItems[pilihanMenu - 1]* banyakItem;
51
52        return hargaTotal;
53    }
54 }
```

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 tanpa nilai kembalian (void), Digunakan ketika fungsi hanya melakukan aksi/tugas tanpa menghasilkan nilai. Contoh: Menu() - hanya menampilkan menu ke layar
 - Fungsi dengan nilai kembalian, Digunakan ketika fungsi perlu menghasilkan nilai yang akan digunakan di bagian lain program. Contoh: hitungTotalHarga() - mengembalikan nilai total harga untuk digunakan dalam perhitungan selanjutnya
2. Tipe kembalian: int yaitu mengembalikan total harga dalam bentuk integer
- Parameter 1: int pilihanMenu - nomor menu yang dipilih pelanggan (1-6)
 - Parameter 2: int banyakItem - jumlah item yang dipesan

3.

```
1 package jobsheet11;
2 import java.util.Scanner;
3
4 public class Kafe29 {
5     public static void main(String[] args) {
6         Menu("Budi", true, "DISKON30");
7
8         Scanner sc = new Scanner(System.in);
9
10        System.out.println("\nMasukkan nomor menu yang ingin anda pesan : ");
11        int pilihanMenu = sc.nextInt();
12        System.out.println("Masukkan jumlah item yang ingin dipesan : ");
13        int banyakItem = sc.nextInt();
14
15        int totalHarga = hitungTotalHarga29(pilihanMenu, banyakItem, "Diskon30");
16
17        System.out.println("Total harga untuk pesanan Anda : Rp " + totalHarga);
18    }
19
20    public static void Menu(String namePelanggan, boolean isMember, String kodePromo) {
21        System.out.println("Selamat datang, " + namePelanggan + "!");
22
23        if (isMember) {
24            System.out.println("Anda adalah member, dapatkan diskon 10% untuk setiap pembelian!");
25        }
26
27        if (kodePromo.equals("DISKON50")) {
28            System.out.println("Selamat! Anda mendapatkan diskon 50%!");
29        } else if (kodePromo.equals("DISKON30")) {
30            System.out.println("Selamat! Anda mendapatkan diskon 30%!");
31        } else {
32            System.out.println("Kode promo invalid.");
33        }
34
35        System.out.println("===== MENU RESTO KAFE =====");
36        System.out.println("1. Kopi Hitam - Rp 15,000");
37        System.out.println("2. Cappuccino - Rp 20,000");
38        System.out.println("3. Latte - Rp 22,000");
39        System.out.println("4. Teh Tarik - Rp 12,000");
40        System.out.println("5. Roti Bakar - Rp 10,000");
41        System.out.println("6. Mie Goreng - Rp 18,000");
42        System.out.println("=====");
43        System.out.println("Silakan pilih menu yang Anda inginkan.");
44    }
45
46    public static int hitungTotalHarga29(int pilihanMenu, int banyakItem, String kodePromo){
47        int [] hargaItems = {15000, 20000, 22000, 12000, 10000, 18000};
48
49        int hagaTotal = hargaItems[pilihanMenu - 1]* banyakItem;
50        int diskon = 0;
51
52        if(kodePromo.equalsIgnoreCase("DISKON50")){
53            diskon = hagaTotal*50/100;
54            System.out.println("Diskon 50% : Rp " + diskon);
55        }
56        else if (kodePromo.equalsIgnoreCase("DISKON30")) {
57            diskon = hagaTotal * 30 / 100;
58            System.out.println("Diskon 30% : Rp " + diskon);
59        } else {
60            System.out.println("Kode promo invalid.");
61        }
62
63        int totalBayar = hagaTotal - diskon;
64        System.out.println("Subtotal untuk menu ini: Rp " + totalBayar);
65        return totalBayar;
66    }
67}
68}
```

4.

```

1 package jobsheet11;
2 import java.util.Scanner;
3
4 public class Kafe29 {
5     public static void main(String[] args) {
6         Menu("Budi", true, "DISKON30");
7
8         Scanner sc = new Scanner(System.in);
9         int totalKeseluruhan = 0;
10        char pesanlagi;
11
12        do {
13            System.out.println("\nMasukkan nomor menu yang ingin anda pesan : ");
14            int pilihanMenu = sc.nextInt();
15            System.out.println("Masukkan jumlah item yang ingin dipesan : ");
16            int banyakItem = sc.nextInt();
17
18            int totalHarga = hitungTotalHarga29(pilihanMenu, banyakItem, "DISKON30");
19            totalKeseluruhan += totalHarga;
20
21            System.out.println("Mau tambah pesanan ? (y/n)");
22            pesanlagi = sc.next().charAt(0);
23
24        }while (pesanlagi == 'y' || pesanlagi == 'Y');
25
26
27        System.out.println("Total yang harus dibayar: Rp " + totalKeseluruhan);
28        System.out.println("Terima kasih atas pesanan Anda!");
29    }
30
31    public static void Menu(String namePelanggan, boolean isMember, String kodePromo) {
32        System.out.println("Selamat datang, " + namePelanggan + "!");
33
34        if (isMember) {
35            System.out.println("Anda adalah member, dapatkan diskon 10% untuk setiap pembelian!");
36        }
37
38        if (kodePromo.equals("DISKON50")) {
39            System.out.println("Selamat! Anda mendapatkan diskon 50%!");
40        } else if (kodePromo.equals("DISKON30")) {
41            System.out.println("Selamat! Anda mendapatkan diskon 30%!");
42        } else {
43            System.out.println("Kode promo invalid.");
44        }
45
46        System.out.println("===== MENU RESTO KAFE =====");
47        System.out.println("1. Kopi Hitam - Rp 15,000");
48        System.out.println("2. Cappuccino - Rp 20,000");
49        System.out.println("3. Latte - Rp 22,000");
50        System.out.println("4. Teh Tarik - Rp 12,000");
51        System.out.println("5. Roti Bakar - Rp 10,000");
52        System.out.println("6. Mie Goreng - Rp 18,000");
53        System.out.println("=====");
54        System.out.println("Silakan pilih menu yang Anda inginkan.");
55    }
56
57    public static int hitungTotalHarga29(int pilihanMenu, int banyakItem, String kodePromo){
58        int [] hargaItems = {15000, 20000, 22000, 12000, 10000, 18000};
59
60        int hagaTotal = hargaItems[pilihanMenu - 1]* banyakItem;
61        int diskon = 0;
62
63        if(kodePromo.equalsIgnoreCase("DISKON50")){
64            diskon = hagaTotal*50/100;
65            System.out.println("Diskon 50% : Rp " + diskon);
66        }
67        else if (kodePromo.equalsIgnoreCase("DISKON30")) {
68            diskon = hagaTotal * 30 / 100;
69            System.out.println("Diskon 30%: Rp " + diskon);
70        } else {
71            System.out.println("Kode promo invalid.");
72        }
73
74        int totalBayar = hagaTotal - diskon;
75        System.out.println("Subtotal untuk menu ini: Rp " + totalBayar);
76        return totalBayar;
77    }
78}

```

4. Percobaan 4

```
1 package jobsheet11;
2
3 public class PengunjungCafe29 {
4     static void daftarPengunjung(String... namePengunjung) {
5         System.out.println("Daftar Nama Pengunjung:");
6         for(int i = 0; i < namePengunjung.length; i++) {
7             System.out.println("- " + namePengunjung[i]);
8         }
9     }
10
11    public static void main(String[] args) {
12        daftarPengunjung("Ali", "Budi", "Citra");
13
14        daftarPengunjung("Andi");
15        daftarPengunjung("Doni", "Eti", "Fahmi", "Galih");
16    }
17 }
18 }
```

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. Penulisan String... namaPengunjung menggunakan varargs (variable arguments) yang memungkinkan fungsi menerima jumlah parameter yang variabel.
- 2.

```
● ● ●  
1 package jobsheet11;  
2  
3 public class PengunjungCafe29 {  
4     static void daftarPengunjung(String... namePengunjung) {  
5         System.out.println("Daftar Nama Pengunjung:");  
6         for(String nama : namePengunjung) {  
7             System.out.println("- " + nama);  
8         }  
9     }  
10    public static void main(String[] args) {  
11        daftarPengunjung("Ali", "Budi", "Citra");  
12        daftarPengunjung("Andi");  
13        daftarPengunjung("Doni", "Eti", "Fahmi", "Galih");  
14    }  
15}  
16  
17}
```

3. Tidak bisa karena setiap fungsi hanya diperkenankan memiliki 1 variable arguments dan Variabel argument diletakkan di paling akhir jika memiliki lebih dari 1 parameter.

Contoh yang salah :

```
void method(String... gfg, int... q)  
{  
    // Compile time error as there  
    // are two varargs  
}
```

(1)

```
void method(int... gfg, String q)  
{  
    // Compile time error as vararg  
    // appear before normal argument  
}
```

(2)

Contoh yang benar :

```
● ● ●  
1 static void contohBenar(String judul, int... numbers) {  
2     System.out.println(judul);  
3     for(int num : numbers) {  
4         System.out.println(num);  
5     }  
6 }  
7  
8 // Pemanggilan:  
9 contohBenar("Data Angka:", 1, 2, 3, 4, 5);
```

4. Program akan tetap berjalan normal tanpa error. Varargs dapat menerima nol argumen, dan dalam hal ini array yang diterima akan kosong.