

JOB SHEET 11

Fungsi 1

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

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

```
1 public static void main(String[] args) {  
2     Menu();  
3 }
```

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).

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*.

```

1  public static void Menu(String namaPelanggan, boolean isMember) {
2      System.out.println("Selamat datang, " + namaPelanggan + "!");
3
4      if (isMember) {
5          System.out.println("Anda adalah member, dapatkan diskon 10% untuk setiap pembelian!");
6      }
7
8      System.out.println("===== MENU RESTO KAFE =====");
9      System.out.println("1. Kopi Hitam - Rp 15,000");
10     System.out.println("2. Cappuccino - Rp 20,000");
11     System.out.println("3. Latte - Rp 22,000");
12     System.out.println("4. Teh Tarik - Rp 12,000");
13     System.out.println("5. Roti Bakar - Rp 10,000");
14     System.out.println("6. Mie Goreng - Rp 18,000");
15     System.out.println("=====");
16     System.out.println("Silakan pilih menu yang Anda inginkan.");
17 }
    
```

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

```

1  public static void main(String[] args) {
2      Menu("Andi", true);
3
4  }
    
```

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"**

2.3 Percobaan 3: Membuat Fungsi dengan Nilai Kembalian

Waktu Percobaan: 40 menit

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.

```
1 public static int hitungTotalHarga(int pilihanMenu, int banyakItem) {
2     int[] hargaItems = {15000, 20000, 22000, 12000, 10000, 18000};
3
4     int hargaTotal = hargaItems[pilihanMenu - 1] * banyakItem;
5     return hargaTotal;
6 }
```

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.

```

1
2 System.out.print("\nMasukkan nomor menu yang ingin Anda pesan: ");
3 int pilihanMenu = sc.nextInt();
4 System.out.print("Masukkan jumlah item yang ingin dipesan: ");
5 int banyakItem = sc.nextInt();
6
7 int totalHarga = hitungTotalHarga(pilihanMenu, banyakItem);
8
9 System.out.println("Total harga untuk pesanan Anda: Rp" + totalHarga);

```

5. Jalankan program dan amati apakah nilai total yang ditampilkan sudah sesuai dengan harga dan jumlah pesanan.
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"

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

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

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

```
daftarPengunjung("Ali", "Budi", "Citra");
```

4. Jalankan program dan amati hasilnya
5. Tambahkan dua baris berikut pada fungsi main di bawah kode pada nomor 3 di atas:

```
daftarPengunjung("Andi");
daftarPengunjung("Doni", "Eti", "Fahmi", "Galih");
```

6. Jalankan program. Amati perbedaan output ketika jumlah argumen yang dikirim berbeda-beda.
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"



3. Tugas

Waktu Pengerjaan: 100 menit

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 "Tugas1"**
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 "Tugas2"

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 “Tugas3”