

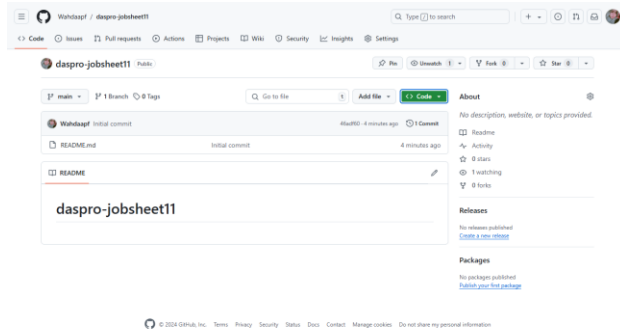
Nama: Wahda Adella Putri Febriana

NIM : 244107020156

Kelas: 1B

Percobaan 1

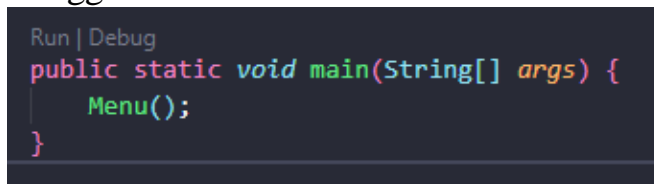
1. Buat Repository “daspro-jobsheet11”



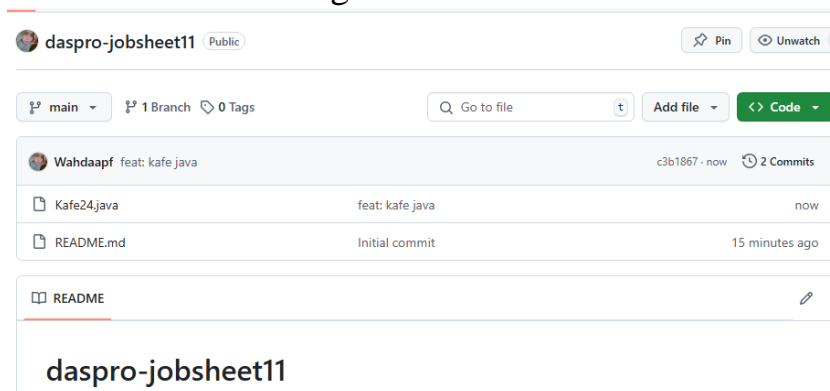
2. Clone dan Open di VSC, buat file Kafe24.java, dan buat fungsi menu



3. Panggil menu di main



4. Push dan commit di github



Pertanyaan 1

1. Tidak, fungsi tanpa parameter tidak selalu digunakan hanya untuk fungsi “void” saja. Sebuah fungsi tanpa parameter dapat mengembalikan nilai apa pun, tergantung pada deklarasi tipenya
2. Tidak, karena kodingan print berada pada fungsi menu, jika menu tidak di panggil maka, print tidak akan nampil.

Modifikasi agar tidak perlu memanggil fungsi

```
Run | Debug
public static void main(String[] args) {
    System.out.println(x:"====MENU RESTO KAFE====");
    System.out.println(x:"1. Kopi Hitam\t - Rp.15.000");
    System.out.println(x:"2. Cappuccino\t - Rp.20.000");
    System.out.println(x:"3. Latte\t\t - Rp.22.000");
    System.out.println(x:"4. Teh Tarik\t - Rp.12.000");
    System.out.println(x:"5. Roti Bakar\t - Rp.10.000");
    System.out.println(x:"6. Mie Goreng\t - Rp.18.000");
    System.out.println(x:"=====");
    System.out.println(x:"Silahkan pilih menu yang anda inginkan.");
}
```

3. a. Membuat kode lebih rapi, modular, dan mudah dipahami.
b. Mengurangi duplikasi kode dan meningkatkan efisiensi.
c. Mempermudah debugging, pemeliharaan, dan pengujian.
d. Memungkinkan penggunaan kembali kode dan fleksibilitas yang lebih besar.

Percobaan 2

1. Update code fungsi menu

```
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!");
    }
}
```

2. Update code pemanggilan fungsi menu

```
Run | Debug
public static void main(String[] args) {
    Menu(namaPelanggan:"Andi", isMember:true);
}
```

Pertanyaan 2

1. Membuat fungsi menjadi flexible, dinamis, dan dapat digunakan Kembali dalam berbagai konteks

2. Tidak, parameter digunakan tempat menampung sementara untuk diteruskan ke fungsi, variable yaitu tempat penampung data umum untuk menyimpan nilai Dimana saja
3. Parameter isMember digunakan untuk mengecek apakah pelanggan merupakan member atau bukan, tipe data nya Boolean, saat pemanggilan fungsi menu() isMember akan ditaruh sebagai parameter kedua dengan opsi true atau false, lalu melakukan pengecekan di fungsi menu(), jika true maka akan memunculkan print “anda adalah member dan mendapatkan diskon sebesar 10%”
4. Error, dikarenakan fungsi nya membutuhkan parameter
5. Modifikasi kode program

```
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.equals(anObject:"DISKON30")) {
        System.out.println(x:"Selamat Anda Mendapatkan diskon 30%");
    } else if (kodePromo.equals(anObject:"DISKON50")) {
        System.out.println(x:"Selamat Anda Mendapatkan diskon 50%");
    } else {
        System.out.println(x:"Kode tidak valid");
    }
    System.out.println(x:"====MENU RESTO KAFE====");
    System.out.println(x:"1. Kopi Hitam\t - Rp.15.000");
    System.out.println(x:"2. Cappuccino\t - Rp.20.000");
    System.out.println(x:"3. Latte\t\t - Rp.22.000");
    System.out.println(x:"4. Teh Tarik\t - Rp.12.000");
    System.out.println(x:"5. Roti Bakar\t - Rp.10.000");
    System.out.println(x:"6. Mie Goreng\t - Rp.18.000");
    System.out.println(x:"=====");
    System.out.println(x:"Silahkan pilih menu yang anda inginkan.");
}

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

Output

```
Selamat Datang Andi!
Anda adalah member, dapatkan diskon 10% untuk setiap pembelian!
Selamat 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.
=====
Selamat Datang Andi!
Anda adalah member, dapatkan diskon 10% untuk setiap pembelian!
Selamat Anda Mendapatkan diskon 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.
=====
Selamat Datang Andi!
Anda adalah member, dapatkan diskon 10% untuk setiap pembelian!
Kode tidak valid
====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.
```

Percobaan 3

1. Buat file HitungTotalHarga24.java dan fungsi hitungTotalHarga

```
public class HitungTotalHarga24 {  
    public static int hitungTotalHarga(int pilihanMenu, int banyakItem) {  
        int[] hargaItems = {15000, 20000, 22000, 12000, 10000, 18000};  
  
        int hargaTotal = hargaItems[pilihanMenu - 1] * banyakItem;  
        return hargaTotal;  
    }  
}
```

2. Buat fungsi penerimaPesanan, dan panggil di main

```
public static void penerimaPesanan() {  
    Scanner input = new Scanner(System.in);  
  
    System.out.print(s:"\nMasukkan nomor menu yang ingin anda pesan: ");  
    int pilihanMenu = input.nextInt();  
    System.out.print(s:"Masukkan jumlah item yang ingin dipesan: ");  
    int banyakItem = input.nextInt();  
  
    int totalHarga = hitungTotalHarga(pilihanMenu, banyakItem);  
  
    System.out.println("Total harga untuk pesanan anda: Rp." + totalHarga);  
  
    input.close();  
}  
  
Run | Debug  
public static void main(String[] args) {  
    penerimaPesanan();  
}
```

Pertanyaan

1. Suatu fungsi membutuhkan nilai kembalian (**return**) ketika fungsi tersebut perlu **mengembalikan hasil perhitungan, data, atau nilai tertentu** ke pemanggilnya.
2. Output

```
Masukkan nomor menu yang ingin anda pesan: 2  
Masukkan jumlah item yang ingin dipesan: 2  
Masukkan Kode Promo: kfo  
Kode tidak valid  
Total harga untuk pesanan anda: Rp.40000  
PS D:\Project\Belajar\Java\daspro-jobsheet11> d;;  
a\jdk-22\bin\java.exe' '-XX:+ShowCodeDetailsInException  
orkspaceStorage\03c7c0d39d54738cd2e9af7052fe52ee\res  
  
Masukkan nomor menu yang ingin anda pesan: 2  
Masukkan jumlah item yang ingin dipesan: 2  
Masukkan Kode Promo: DISKON50  
Total harga untuk pesanan anda: Rp.20000  
PS D:\Project\Belajar\Java\daspro-jobsheet11> |
```

Kode

```
public static int hitungTotalHarga(int pilihanMenu, int banyakItem, String kodePromo) {  
    int[] hargaItems = {15000, 20000, 22000, 12000, 10000, 18000};  
  
    int hargaTotal = hargaItems[pilihanMenu - 1] * banyakItem;  
  
    if(kodePromo.equals(anObject:"DISKON50")) {  
        hargaTotal = hargaTotal * 30/100;  
    } else if (kodePromo.equals(anObject:"DISKON50")) {  
        hargaTotal = hargaTotal * 50/100;  
    } else {  
        System.out.println(s:"Kode tidak valid");  
    }  
    return hargaTotal;  
}  
  
public static void penerimaPesanan() {  
    Scanner input = new Scanner(System.in);  
  
    System.out.print(s:"\nMasukkan nomor menu yang ingin anda pesan: ");  
    int pilihanMenu = input.nextInt();  
    System.out.print(s:"Masukkan jumlah item yang ingin dipesan: ");  
    int banyakItem = input.nextInt();  
    System.out.print(s:"Masukkan Kode Promo: ");  
    String kodePromo = input.next();  
  
    int totalHarga = hitungTotalHarga(pilihanMenu, banyakItem, kodePromo);  
  
    System.out.println("Total harga untuk pesanan anda: Rp." + totalHarga);  
  
    input.close();  
}  
  
Run | Debug  
public static void main(String[] args) {  
    penerimaPesanan();  
}
```

3. output

```
=====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
=====

Masukkan nomor menu yang ingin anda pesan (atau 0 untuk selesai): 2
Masukkan jumlah item untuk menu tersebut: 2
Masukkan Kode Promo (atau '-' jika tidak ada): -
Total harga untuk menu ini: Rp.40000

Masukkan nomor menu yang ingin anda pesan (atau 0 untuk selesai): 1
Masukkan jumlah item untuk menu tersebut: 2
Masukkan Kode Promo (atau '-' jika tidak ada): -
Total harga untuk menu ini: Rp.30000

Masukkan nomor menu yang ingin anda pesan (atau 0 untuk selesai): 0

=====
Total keseluruhan pesanan anda: Rp.70000
Terima kasih telah memesan!
```

Kodingan

```
public static void penerimaPesanan() {
    Scanner input = new Scanner(System.in);
    int totalKeseluruhan = 0;

    System.out.println(x:"=====MENU RESTO KAFE=====");
    System.out.println(x:"1. Kopi Hitam\t - Rp.15.000");
    System.out.println(x:"2. Cappuccino\t - Rp.20.000");
    System.out.println(x:"3. Latte\t - Rp.22.000");
    System.out.println(x:"4. Teh Tarik\t - Rp.12.000");
    System.out.println(x:"5. Roti Bakar\t - Rp.10.000");
    System.out.println(x:"6. Mie Goreng\t - Rp.18.000");
    System.out.println(x:"=====");

    while (true) {
        System.out.print("\nMasukkan nomor menu yang ingin anda pesan (atau 0 untuk selesai): ");
        int pilihanMenu = input.nextInt();
        if (pilihanMenu == 0) {
            break; // Keluar dari loop jika selesai memilih
        }

        System.out.print("Masukkan jumlah item untuk menu tersebut: ");
        int banyakItem = input.nextInt();

        System.out.print("Masukkan Kode Promo (atau '-' jika tidak ada): ");
        String kodePromo = input.next();

        int totalHarga = hitungTotalHarga(pilihanMenu, banyakItem, kodePromo);
        totalKeseluruhan += totalHarga;

        System.out.println("Total harga untuk menu ini: Rp." + totalHarga);

    }

    System.out.println(x:"\n=====");
    System.out.println("Total keseluruhan pesanan anda: Rp." + totalKeseluruhan);
    System.out.println(x:"Terima kasih telah memesan!");

    input.close();
}
```

Percobaan 4

1. Buat file pengunjungKafeNo24.java, dan tambahkan kodingan berikut

```
public class PengunjungKafe24 {

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

    public static void main(String[] args) {
        daftarPengunjung (...namaPengunjung:"Ali", "Budi", "Citra");
    }
}
```

Pertanyaan

1. bertujuan untuk memungkinkan metode menerima **jumlah argumen yang tidak ditentukan sebelumnya**.
2. Ubah menjadi for each

```
public static void daftarPengunjung (String... namaPengunjung) {  
    System.out.println(x:"Daftar Nama Pengunjung: ");  
    for (String nama : namaPengunjung) { // Menggunakan for-each loop  
        System.out.println("-" + nama);  
    }  
}
```

3. Tidak, tidak dapat menggunakan dua parameter varargs (...) dalam satu fungsi. Karena parameter varargs akan menangkap semua argumen yang sesuai, sehingga akan membingungkan compiler mengenai argumen mana yang harus masuk ke parameter varargs tertentu.

Percobaan 5

1. Buat file percobaan6.java, dan kode berikut

```
public class Percobaan6 {  
    Run | Debug  
    public static void main(String[] args) {  
        Scanner input = new Scanner(System.in);  
  
        int p,l,t,L,vol;  
  
        System.out.print(s:"Masukkan Panjang: ");  
        p = input.nextInt();  
        System.out.print(s:"Masukkan Lebar: ");  
        l = input.nextInt();  
        System.out.print(s:"Masukkan Tinggi: ");  
        t = input.nextInt();  
  
        L= p * l;  
        System.out.println("Luas Persegi Panjang Adalah " + L);  
  
        vol = p * l * t;  
        System.out.println("Volume Persegi Panjang Adalah " + vol);  
  
        input.close();  
    }  
}
```

2. Buat fungsi untuk menghitung luas dan volume

```
static int hitungLuas(int p, int l) {  
    int luas = p * l;  
    return luas;  
}  
  
static int hitungVolume(int p, int l, int t) {  
    int volume = p * l * t;  
    return volume;  
}
```

3. Perubahan kode main

```
L= hitungLuas(p, l);  
System.out.println("Luas Persegi Panjang Adalah " + L);  
  
vol = hitungVolume(p, l, t);  
System.out.println("Volume Persegi Panjang Adalah " + vol);
```

4. Hasil run

```
Masukkan Panjang: 10  
Masukkan Lebar: 2  
Masukkan Tinggi: 5  
Luas Persegi Panjang Adalah 20  
Volume Persegi Panjang Adalah 100  
PS D:\Project\Belajar\Java\daspro-jobsheet11>  
odeDetailsInExceptionMessages' '-cp' 'C:\Users\  
03c7c0d39d54738cd2e9af7052fe52ee\redhat.java\jd  
Masukkan Panjang: 12  
Masukkan Lebar: 3  
Masukkan Tinggi: 5  
Luas Persegi Panjang Adalah 36  
Volume Persegi Panjang Adalah 180  
PS D:\Project\Belajar\Java\daspro-jobsheet11>  
odeDetailsInExceptionMessages' '-cp' 'C:\Users\  
03c7c0d39d54738cd2e9af7052fe52ee\redhat.java\jd  
Masukkan Panjang: 34  
Masukkan Lebar: 67  
Masukkan Tinggi: 3  
Luas Persegi Panjang Adalah 2278  
Volume Persegi Panjang Adalah 6834
```

Pertanyaan

1. Menginputkan Panjang, lebar dan tinggi lalu memanggil fungsi hitungLuas dan hitungVolume dengan parameter yang diinputkan user lalu di print hasilnya
2. Outputnya 1 2 3 4 5 6 7
Urutan system nya yaitu
 - a. `int temp = Jumlah(1, 1) -> hasil nya 2`
 - b. `TampilJumlah(temp, 5)`
 - c. Di dalam tampilJumlah memanggil fungsi tampilHinggaKeI(Jumlah(2, 5))
 - d. Berarti tampilHinggaKeI(7)
 - e. Di dalam sana melakukan perulangan sebanyak parameter I, lalu di print i nya
3. A. Fungsi harus menggunakan parameter jika Fungsi membutuhkan data dari luar untuk bekerja dan Fungsi perlu beradaptasi untuk berbagai scenario

- B. Fungsi tidak perlu parameter jika Data yang digunakan sudah tersedia di dalam fungsi dan Fungsi tidak memerlukan variasi input.
- C. Fungsi harus memiliki nilai kembalian (return) jika Fungsi menghasilkan data atau hasil yang perlu digunakan kembali. Dan Fungsi harus memberikan status, jawaban, atau validasi.
- D. Fungsi tidak perlu nilai kembalian jika Fungsi hanya melakukan tindakan tanpa menghasilkan hasil yang perlu digunakan lebih lanjut.