

Muhammad Ichsan Rahmat Ramadhan
231524015
1A – D4 Teknik Informatika

Soal 1

Encapsulation: Variabel instance dalam class sebaiknya diubah menjadi private, dan diakses melalui metode getter dan setter.

Single Responsibility Principle: Setiap class sebaiknya memiliki tanggung jawab yang jelas. Misalnya, class Restaurant sebaiknya hanya bertanggung jawab untuk mengelola daftar menu, sedangkan class Makanan bertanggung jawab untuk menyimpan informasi tentang makanan.

Constructor: Menggunakan constructor untuk inisialisasi objek Makanan.

Static: Menggunakan static hanya jika diperlukan, seperti untuk variabel id pada class Restaurant.

Akan sulit jika kita ingin menambahkan stok dan lain-lain jika algoritmanya seperti itu, mungkin kita dapat menambahkan pengurangan stok di class Restaurant.

Tetapi kita juga perlu menambahkan class lain untuk membuat pemesanan, karena task tersebut jauh berbeda dengan task yang ada di modul kita sementara

Ini adalah Program yang sudah di modifikasi:
Restaurant.java



```
1 public class Restaurant {
2     private Makanan[] daftarMenu;
3     private static byte id=0;
4
5     public Restaurant () {
6         this.daftarMenu = new Makanan[10];
7     }
8
9     public void tambahMenuMakanan (Makanan makanan) {
10         this.daftarMenu[id] = makanan;
11         id++;
12     }
13
14     public void tampilMenuMakanan () {
15         for (int i=0; i<id; i++) {
16             if(!isOutOfStock (i)) {
17                 System.out.println(daftarMenu[i].getNama() + "[" + daftarMenu[i].getStok() + "]" + "\trp. " + daftarMenu[i].getHarga());
18             }
19         }
20     }
21
22     public boolean isOutOfStock (int id) {
23         return daftarMenu[id].getStok() == 0;
24     }
25 }
26
```

RestaurantMain.java

```
public class RestaurantMain {
    public static void main(String[] args) {
        Restaurant menu = new Restaurant();

        Makanan balaBala = new Makanan("Bala-Bala", 1_000, 20);
        menu.tambahMenuMakanan(balaBala);

        Makanan gehu = new Makanan("Gehu", 1_000, 20);
        menu.tambahMenuMakanan(gehu);

        Makanan tahu = new Makanan("Tahu", 1_000, 0);
        menu.tambahMenuMakanan(tahu);

        Makanan molen = new Makanan("Molen", 1_000, 20);
        menu.tambahMenuMakanan(molen);

        menu.tampilMenuMakanan();
    }
}
```

Makanan.java

```
*Restaurant.java  RestaurantMain.java  Makanan.java X
1  public class Makanan {
2      private String nama;
3      private double harga;
4      private int stok;
5
6  public Makanan(String nama, double harga, int stok) {
7      this.nama = nama;
8      this.harga = harga;
9      this.stok = stok;
10 }
11
12 public String getNama() {
13     return nama;
14 }
15
16 public double getHarga() {
17     return harga;
18 }
19
20 public int getStok() {
21     return stok;
22 }
23 }
24
```

Soal 2

Algoritma yang saya buat untuk melakukan pemesanan dan tambahan pengurangan stok.

Restaurant.java

```
import java.util.Scanner;

public class Restaurant {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);

        // Inisialisasi produk
        Produk[] menu = {
            new Produk("Nasi Goreng", 15000, 0, 10),
            new Produk("Mie Goreng", 12000, 0, 10),
            new Produk("Ayam Goreng", 20000, 0, 10)
        };

        // Menampilkan menu
        boolean pesanLagi = true;
        while (pesanLagi) {
            System.out.println("Menu Makanan:");
            for (int i = 0; i < menu.length; i++) {
                System.out.println((i + 1) + ". " + menu[i].getNamaMakanan() + " - Rp" + menu[i].getHarga() + "(Stok : " + menu[i].getStok() + ")");
            }

            System.out.print("Pilih menu (nomor): ");
            int pilihan = scanner.nextInt() - 1;
            scanner.nextLine(); // Membersihkan newline

            System.out.print("Jumlah yang dipesan: ");
            int jumlah = scanner.nextInt();
            scanner.nextLine(); // Membersihkan newline

            if (jumlah > menu[pilihan].getStok()) {
                System.out.println("Stok " + menu[pilihan].getNamaMakanan() + " tidak mencukupi.");
            } else {
                menu[pilihan].setJumlah(jumlah);
                menu[pilihan].kurangiStok(jumlah);

                System.out.print("Pesan lagi? (y/n): ");
                String jawab = scanner.nextLine();
                pesanLagi = jawab.equalsIgnoreCase("y");
            }
        }

        // Menampilkan pesanan
        System.out.println("Pesanan:");
        int totalBayar = 0;
        for (Produk item : menu) {
            if (item.getJumlah() > 0) {
                int harga = item.getTotalHarga();
                System.out.println(item.getNamaMakanan() + " - " + item.getJumlah() + " x Rp" + harga);
                totalBayar += harga;
            }
        }
        System.out.println("Total Bayar: Rp" + totalBayar);

        scanner.close();
    }
}
```

Produk.java

```
public class Produk {  
    private String namaMakanan;  
    private int harga;  
    private int jumlah;  
    private int stok;  
  
    public Produk(String namaMakanan, int harga, int jumlah, int stok) {  
        this.namaMakanan = namaMakanan;  
        this.harga = harga;  
        this.jumlah = jumlah;  
        this.stok = stok;  
    }  
  
    public String getNamaMakanan() {  
        return namaMakanan;  
    }  
  
    public int getHarga() {  
        return harga;  
    }  
  
    public int getJumlah() {  
        return jumlah;  
    }  
  
    public void setJumlah(int jumlah) {  
        this.jumlah = jumlah;  
    }  
  
    public int getStok() {  
        return stok;  
    }  
  
    public void kurangiStok(int jumlah) {  
        stok -= jumlah;  
    }  
  
    public int getTotalHarga() {  
        return harga * jumlah;  
    }  
}
```

Penjualan.java

```
1 public class Penjualan {
2     private String namaMakanan;
3     private int totalHarga;
4     private int jumlah;
5     private int sisaStok;
6
7     public Penjualan(String namaMakanan, int totalHarga, int sisaStok) {
8         this.namaMakanan = namaMakanan;
9         this.totalHarga = totalHarga;
10        this.sisaStok = sisaStok;
11    }
12
13    public String getNamaMakanan() {
14        return namaMakanan;
15    }
16
17    public int getTotalHarga() {
18        return totalHarga;
19    }
20
21    public int getJumlah() {
22        return jumlah;
23    }
24
25    public void setJumlah(int jumlah) {
26        this.jumlah = jumlah;
27    }
28
29    public int getSisaStok() {
30        return sisaStok;
31    }
32 }
33
```

Gambaran aplikasi

```
Menu Makanan:
1. Nasi Goreng - Rp15000(Stok : - 10)
2. Mie Goreng - Rp12000(Stok : - 10)
3. Ayam Goreng - Rp20000(Stok : - 10)
Pilih menu (nomor): 1
Jumlah yang dipesan: 2
Pesan lagi? (y/n): y
Menu Makanan:
1. Nasi Goreng - Rp15000(Stok : - 8)
2. Mie Goreng - Rp12000(Stok : - 10)
3. Ayam Goreng - Rp20000(Stok : - 10)
Pilih menu (nomor): 2
Jumlah yang dipesan: 5
Pesan lagi? (y/n): n
Pesanan:
Nasi Goreng - 2 x Rp30000
Mie Goreng - 5 x Rp60000
Total Bayar: Rp90000
```

Link github:

<https://github.com/rahmatichsan21>