

Tugas Minggu - 5

*(Sebagai pemenuhan salah satu tugas mata kuliah PBO praktek
Program Studi D3 Teknik Informatika)*



Disusun oleh :
Muhammad Ali Kusnadin (201511016)

**D3 – 2A
PROGRAM STUDI D3 TEKNIK INFORMATIKA
POLITEKNIK NEGERI BANDUNG
TAHUN 2021**

1. Kasus – 2

- a. Membuat atribut menjadi private

Sebelum:

```
public class Restaurant {  
    public String[] nama_makanan;  
    public double[] harga_makanan;  
    public int[] stok;  
    public static byte id=0;
```

Setelah :

```
public class Restaurant {  
    private String[] nama_makanan;  
    private double[] harga_makanan;  
    private int[] stok;  
    private static byte id=0;  
  
    public Restaurant() {
```

Agar atribut di dalam kelas Restaurant tidak mudah di akses oleh orang. Hanya dengan modul modul yang sudah ditetapkan oleh developer.

- b. Membuat method di RestaurantMain

Sebelum :

```
public class RestaurantMain {  
    public static void main(String[] args) {  
        Restaurant menu = new Restaurant();  
        menu.tambahMenuMakanan("Bala-Bala", 1_000, 20);  
        Restaurant.nextId();  
        menu.tambahMenuMakanan("Gehu", 1_000, 20);  
        Restaurant.nextId();  
        menu.tambahMenuMakanan("Tahu", 1_000, 0);  
        Restaurant.nextId();  
        menu.tambahMenuMakanan("Molen", 1_000, 20);  
  
        menu.tampilMenuMakanan();  
    }  
}
```

Setelah :

```
public class RestaurantMain {  
    static void inisialisasiMakanan(Restaurant menu) {  
        menu.tambahMenuMakanan("Bala-Bala", 1_000, 20);  
        Restaurant.nextId();  
        menu.tambahMenuMakanan("Gehu", 1_000, 20);  
        Restaurant.nextId();  
        menu.tambahMenuMakanan("Tahu", 1_000, 0);  
        Restaurant.nextId();  
        menu.tambahMenuMakanan("Molen", 1_000, 20);  
    }  
}
```

Agar pada method main tidak terlalu complex, lalu pada id saya pindahkan ke kelas Makanan dan dibuatkan getternya lalu untuk nextId saya tempatkan di 'tambahMenuMakanan'

- c. Membuat kelas makanan

```

2
3 public class Makanan {
4     private String[] nama_makanan;
5     private double[] harga_makanan;
6     private int[] stok;
7     private static byte idMakanan=0;
8
9     public Makanan() {
10         this.nama_makanan = new String[10];
11         this.harga_makanan = new double[10];
12         this.stok = new int[10];
13     }
14     public String getNama_makanan(byte id) {
15         return this.nama_makanan[id];
16     }
17     public void setNama_makanan(String nama_makanan) {
18         this.nama_makanan[idMakanan] = nama_makanan;
19     }
20     public double getHarga_makanan(byte id) {
21         return harga_makanan[id];
22     }
23     public void setHarga_makanan(double harga_makanan) {
24         this.harga_makanan[idMakanan] = harga_makanan;
25     }
26     public int getStok(byte id) {
27         return stok[id];
28     }
29     public void setStok(int stok,int id) {
30         this.stok[id] += stok;
31     }
32
33     public byte cariIdMakanan(String nama) {
34         for(byte i = 0;i<this.nama_makanan.length;i++) {
35             if(nama.compareTo(this.nama_makanan[i])==0) {
36                 return i;
37             }
38         }
39         return -1;
40     }
41
42     public static void nextIdMakanan() {
43         idMakanan++;
44     }
45     public static byte getIdMakanan() {
46         return idMakanan;
47     }
48     // public static void setIdMakanan(byte idMakanan) {

```

Jadi pada kelas Restaurant hal yang berkaitan dengan makanan di pindahkan di kelas makanan

Pada kelas Restaurant :

```

2
3 public class Restaurant {
4     // private String[] nama_makanan;
5     // private double[] harga_makanan;
6     // private int[] stok;
7     private Makanan menu_makanan;
8     // private static byte idMakanan=0;
9
10    public Restaurant() {
11        // this.nama_makanan = new String[10];
12        // this.harga_makanan = new double[10];
13        // this.stok = new int[10];
14        menu_makanan = new Makanan();
15    }
16
17    public void tambahMenuMakanan(String nama, double harga, int stok) {
18        // this.nama_makanan[idMakanan] = nama;
19        // this.harga_makanan[idMakanan] = harga;
20        // this.stok[idMakanan] = stok;
21        menu_makanan.setNama_makanan(nama);
22        menu_makanan.setHarga_makanan(harga);
23        menu_makanan.setStok(stok, Makanan.getIdMakanan());
24        Makanan.nextIdMakanan();
25    }
26
27    public void tampilMenuMakanan() {
28        for(byte i = 0; i <= Makanan.getIdMakanan(); i++) { //mengganti tipe data i menjadi byte dari int
29            if(!isOutOfStock(i)) {
30                //System.out.println(nama_makanan[i] + "+" + stok[i] + "+" + "\tRp. " + harga_makanan[i]);
31                System.out.println(menu_makanan.getNama_makanan(i) + "+" + menu_makanan.getStok(i) + "+" + "\tRp. " + menu_ma
32            }
33        }
34    }
35
36    public boolean isOutOfStock(byte id) {
37        if(menu_makanan.getStok(id) == 0) {
38            return true;
39        } else {
40            return false;
41        }
42    }
43
44    // public static void main(String[] args) {
45    //     idMakanan++;
46    // }
47
48    public void pesanMenuMakanan(String nama, byte jumlah) {
49        byte idTemp = menu_makanan.cariIdMakanan(nama);
50        if(menu_makanan.getStok(idTemp) >= jumlah) {
51            menu_makanan.setStok(-jumlah, idTemp);
52        } else {
53            if(menu_makanan.getStok(idTemp) > 0) {
54                System.out.println("maaf stok " + menu_makanan.getNama_makanan(idTemp) + " tidak ada sesuai dengan jumlah");
55            } else {
56                System.out.println("maaf stok " + menu_makanan.getNama_makanan(idTemp) + " sedang kosong");
57            }
58        }
59    }
60
61    }
62
63    }

```

d. Menambahkan method pemesanan pada kelas Restaurant

```

43
44 public void pesanMenuMakanan(String nama, byte jumlah) {
45     byte idTemp = menu_makanan.cariIdMakanan(nama);
46     if(menu_makanan.getStok(idTemp) >= jumlah) {
47         menu_makanan.setStok(-jumlah, idTemp);
48     } else {
49         if(menu_makanan.getStok(idTemp) > 0) {
50             System.out.println("maaf stok " + menu_makanan.getNama_makanan(idTemp) + " tidak ada sesuai dengan jumlah")
51         } else {
52             System.out.println("maaf stok " + menu_makanan.getNama_makanan(idTemp) + " sedang kosong");
53         }
54     }
55 }
56
57 }

```

Jadi pada RestaurantMain:

```

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

        menu.tampilMenuMakanan();
        System.out.println();
        menu.pesanMenuMakanan("Bala-Bala", (byte) 10);
        menu.pesanMenuMakanan("Tahu", (byte) 10);

        menu.tampilMenuMakanan();
    }

```

Dan hasil output nya :

```

<terminated> RestaurantMain [Java Application]
Bala-Bala[20]    Rp. 1000.0
Gehu[20]         Rp. 1000.0
Molen[20]        Rp. 1000.0

maaf stok Tahu sedang kosong
Bala-Bala[10]    Rp. 1000.0
Gehu[20]         Rp. 1000.0
Molen[20]        Rp. 1000.0

```

2. Kasus – 3

Untuk menampilkan Output:

```

Rinnegan
Mangekyou Sharingan
Sharingan
Susanoo

```

Saya menambahkan pemanggilan `setDojutsu()` di kelas itachi pada method `printDojutsu()`. Lalu dikarenakan terdapat override di super kelas dari sasuke yaitu itachi, jadi saya panggil `print_Dojutsu` yang ada di kelas itachi ke kelas sasuke.

```

public class Itachi extends Rikudo{
    private String kekkeiGenkai = "Susanoo";
    private String Dojutsu = super.Dojutsu;

    void printkekkeiGenkai() {
        System.out.println(this.kekkeiGenkai);
    }
    void printDojutsu() {
        System.out.println(this.Dojutsu);
        setDojutsu();
        System.out.println(this.Dojutsu);
    }
    private void setDojutsu() {
        this.Dojutsu = "Mangekyou Sharingan";
    }
}

```

```

void printDojutsu() {
    super.printDojutsu();
    System.out.println(this.Dojutsu);
}
}

```

- Cara kerja dari program
 1. Pertama tama variable s yang bertipe sasuke di inisialisasi, lalu variable memanggil printDojutsu();
 2. Ada dua printDojutsu yang pertama ada di kelas itachi dan sasuke. Sasuke merupakan turunan dari itachi. Tetapi karena s di inisialisasi dengan sasuke maka yang akan di tampilkan adalah printDojutsu yang berada di sasuke.
 3. Lalu variable kedua i yang bertipe Itachi di inisialisasikan Sasuke jadi walaupun s merupakan superclass tapi jika di inisialisasikan oleh kelas warisannya maka print Dojutsu nya akan mengarah ke kelas sasuke
 4. Lalu i memanggil printKekeiGenkai untuk menampilkan KekeiGenkai dari Itachi