

# **W4 - FUNDAMENTAL PROGRAMMING STRUCTURES IN JAVA**

## **LAPORAN PRAKTIKUM**

Disusun untuk memenuhi tugas Mata Kuliah Pemrograman Berorientasi Objek

Disusun oleh

Nazwa Fitriyani Zahra      211511051



**PROGRAM STUDI D3 TEKNIK INFORMATIKA**

**JURUSAN TEKNIK KOMPUTER DAN INFORMATIKA**

**POLITEKNIK NEGERI BANDUNG**

**2022**

## A. Soal 1

```
1 package soall;
2 public class Barang {
3     String kode_barang;
4     String nama_barang;
5     private int stok;
6
7     public int getStok() {
8         return stok;
9     }
10    public void setStok(int stok) {
11        this.stok = stok;
12    }
13    public Barang(String kode, String nama, int stk) {
14        kode_barang = kode;
15        nama_barang = nama;
16        stok = stk;
17    }
18    public void addStok (int stok) {
19        this.stok += stok;
20    }
21 }
```

```
1 package soall;
2 public class Inventory {
3     Barang[] barangs;
4     void initBarang() {
5         barangs = new Barang[2];
6         barangs[0] = new Barang("001", "Baju", 10);
7         barangs[1] = new Barang("002", "Celana", 20);
8     }
9     void showBarang() {
10        System.out.println(barangs[0].nama_barang + "(" + barangs[0].getStok() + ")");
11        System.out.println(barangs[1].nama_barang + "(" + barangs[1].getStok() + ")");
12    }
13    void pengadaan() {
14        initBarang();
15        // barangs[0].stok += 20;
16        // barangs[0].stok -= 30;
17        // barangs[0].stok *= 30;
18        barangs[0].addStok(20);
19        showBarang();
20    }
21    public static void main(String[] args) {
22        Inventory beli = new Inventory();
23        beli.pengadaan();
24    }
25 }
```

Hasil Output :

```
<terminated> Inventory [Java Application] C:\Program Files\Java\jdk-18.0.2.1\bin\javaw.exe (Sep 13, 2022, 9:43:51 AM – 9:43:52 AM) [p
Baju (30)
Celana (20)
```

Masalah pada soal : Carilah solusi, agar variable “stok” dibungkus/ dilindungi sehingga tidak bisa dilakukan operasi aritmatika selain hanya tambah saja.

Jawaban Masalah pada soal : diberikan getter, setter, dan add,

Permasalahan :

Terkadang masih bingung dalam penggunaan enkapsulasi ketika di beri kasus

Solusi :

Bertanya pada teman mengenai encapsulasinya

Sumber :

Nama Teman yang membantu : Terbantu oleh yang presentasi (Rizki H)

## B. Soal 2

```
1 package soal2;  
2 public class Item {  
3     private String name;  
4     private Item() {  
5         name = "Ipin";  
6     }  
7     public Item(String name) {  
8         this();  
9         System.out.println(this.name);  
10    }  
11 }
```

```
1 package soal2;  
2 public class Item {  
3     private String name;  
4     private Item() {  
5         name = "Ipin";  
6     }  
7     public Item(String name) {  
8         //Dilakukan penambahan this(), agar dapat memanggil  
9         //name dalam satu file  
10        this();  
11        System.out.println(this.name);  
12    }  
13 }
```

Hasil Output :

```
<terminated> UpinIpin [Java Application] C:\Program Files\  
Ipin
```

Masalah pada soal : Bagaimana cara agar output menjadi 'Ipin'

Jawaban Masalah pada soal : menambahkan this() pada baris sebelum dilakukan print dengan parameter this.name

Permasalahan : -

Solusi : -

Sumber :

Nama Teman yang membantu : Terbantu oleh yang presentasi (M. Hafizh)

## C. Soal 3

```

1 package soal3;
2 class KelasSatu{
3     {
4         System.out.println(11);
5     }
6     static
7     {
8         System.out.println(2);
9     }
10    public KelasSatu(int i)
11    {
12        System.out.println(3);
13    }
14    public KelasSatu()
15    {
16        System.out.println(4);
17    }
18 }

```

```

1 package soal3;
2 class KelasDua{
3     {
4         System.out.println(5);
5     }
6     public static void main(String[] args)
7     {
8
9         System.out.println(6);
10        KelasSatu satu = new KelasSatu();
11        KelasSatu dua = new KelasSatu(10);
12    }
13    //Baris pertama dulu
14    //Static
15    //Gaada static gaada publik
16    //Public tanpa parameter
17    //Gaada static gaada publik
18    //Public pake parameter
19 }

```

Hasil Output :

```

<terminated> KelasDua [Java Application] C:\Program Files\Java
6
2
11
4
11
3

```

Masalah pada soal : bagaimana urutan konstruksi objek tersebut, dan mengapa urutannya seperti itu

Jawaban Masalah pada soal :

Angka 6 muncul pertama karena karena di method main.

b. Angka 2 kemudian muncul karena terdapat di dalam blok static pada class.

c. Angka 11 selanjutnya muncul sebagai hasil eksekusi dari instance initialization

blocks

d. Angka 4 lalu muncul saat dilakukan pemanggilan terhadap constructor yang ada pada class, yaitu constructor yang tidak memiliki parameter.

e. Angka 11 lalu muncul sebagai hasil eksekusi dari instance initialization blocks karna blok ini adalah blok yang dieksekusi setiap class diinisialisasi dan sebelum constructor dipanggil.

f. Angka 3 muncul saat dilakukan pemanggilan terhadap constructor yang ada pada class, yaitu constructor yang berparameter integer.

Permasalahan : Masih kurang mengerti tentang constructor dan urutan pemanggilannya

Solusi : mencari google

Nama Teman yang membantu : google