

# **PRAKTIKUM PERTEMUAN 4**

*Disusun untuk memenuhi tugas mata kuliah Pemrograman Berorientasi Objek (Praktek)*



Disusun oleh:

Bandyaga Adiansyah Sugandi

NIM 231511037

2B – D3

Jurusan Teknik Komputer dan Informatika

Politeknik Negeri Bandung

2024

## A. SOAL 1

### Screenshot hasil

```
"C:\Program Files\Java\jdk-22\bin\java.exe" "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA Community Edition
Memproses penjualan...
5 Laptop terjual.
Stok setelah penjualan: 5
Menambah stok...
Stok setelah penambahan: 15
Memperbarui harga produk...
Harga baru: 14500000
Memperbarui harga produk...
Harga tidak valid!
Harga baru: 14500000

Process finished with exit code 0
```

### Pembahasan, Screenshot Jawaban dan Solusi

⇒ Soal 1 membahas program enkapsulasi class Sales dan class Product ke dalam sebuah package direktori com/polban/jtk/sales/(file .class). Package tersebut kemudian diimport pada Main program Soal1.java dengan cara:

**import com.polban.jtk.sales.Product;**

**import com.polban.jtk.sales.Sales;**

Masing-masing merepresentasikan isi dari class-class tersebut pada direktori com/polban/jtk/sales.

⇒ Pengujian enkapsulasi dilakukan dengan beberapa hal yaitu:

- Membuat object product dengan quantity 10
- Membuat object sales
- Melakukan penjualan sebanyak 5 quantity
- Melakukan restock product
- Memperbaharui harga product
- Memperbaharui harga product dengan nilai minus (negatif)
- Mengubah format output agar tidak berupa 1.4E7 pada harga product (program di atas menggunakan cara 2), cara (1) dicontohkan pada Soal 3
  1. Menggunakan printf kemudian panggil method getPrice()  
**System.out.printf("Harga baru: %.2f\n", product.getPrice());**
  2. Menggunakan println kemudian method getPrice() cast menjadi long  
**System.out.println("Harga baru: " + (long) product.getPrice());**

```
public class Soal1 {  ± basganajaah
    public static void main(String[] args) {  ± basganajaah
        Product product = new Product( productName: "Laptop", price: 15000000, stock: 10);
        Sales sales = new Sales(product);

        // jual product sebanyak quantity 5
        sales.sellProduct( quantity: 5);

        // restock product sebanyak quantity 10
        sales.restockProduct( quantity: 10);

        // atur harga menjadi 14500000
        sales.updateProductPrice( newPrice: 14500000);

        // atur harga menjadi nilai negatif sembarang
        sales.updateProductPrice( newPrice: -14500000);
    }
}
```

## B. SOAL 2

### Screenshot hasil

```
"C:\Program Files\Java\jdk-22\bin\java.exe" "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA Community Edition
Baju (60)
Celana (20)

Process finished with exit code 0
```

### Pembahasan, Screenshot Jawaban dan Solusi

- ⇒ Soal 2 membahas program enkapsulasi class Barang dan class Inventori yang mana class Inventori juga merupakan Main Class (Main Program). Class-class tersebut terdapat pada package direktori com/polban/jtk/inventory/(file .class). Package tersebut kemudian digunakan pada Main program di class Inventori.java dan Barang.java dan dengan cara:

**package com.polban.jtk.inventory;**

Package tersebut terletak pada satu folder yang sama yaitu folder inventory. Tidak perlu import karena terletak di lokasi package yang sama.

- ⇒ Ada permasalahan dimana stock barang dapat dimanipulasi menggunakan operasi aritmatika selain penjumlahan. Pada logikanya ketika menambah stock, stock sekarang akan ditambah dengan stock baru menggunakan aritmatika penjumlahan. Maka dari itu agar program di Soal 2 ini tidak dapat dimanipulasi oleh operasi selain penjumlahan, maka buat method tambahStok di Barang.java:

```
public int getStok() { 2 usages  ⚡ basganajaah
    return stok;
}

public String getNamaBarang() { 2 usages  ⚡ basganajaah
    return nama_barang;
}

public void tambahStok(int jumlah) { 1 usage  ⚡ basganajaah
    if (jumlah > 0) {
        this.stok += jumlah;
    } else {
        System.out.println("Jumlah stok yang ditambah harus lebih dari 0");
    }
}
```

Panggil method nya dengan cara dibawah ini pada Inventori.java:

```
void pengadaan() { 1 usage  ⚡ basganajaah
    initBarang();
    barangs[0].tambahStok( jumlah: 50);
    showBarang();
}
```

Dengan ini maka jika ingin menambah stok, harus dilakukan dengan cara memanggil method tambahStok yang meminta paramter jumlah stok yang ingin ditambah dan akan diproses apabila nilai jumlah tersebut bernilai positif atau lebih dari 0.

### C. SOAL 3

#### Screenshot hasil

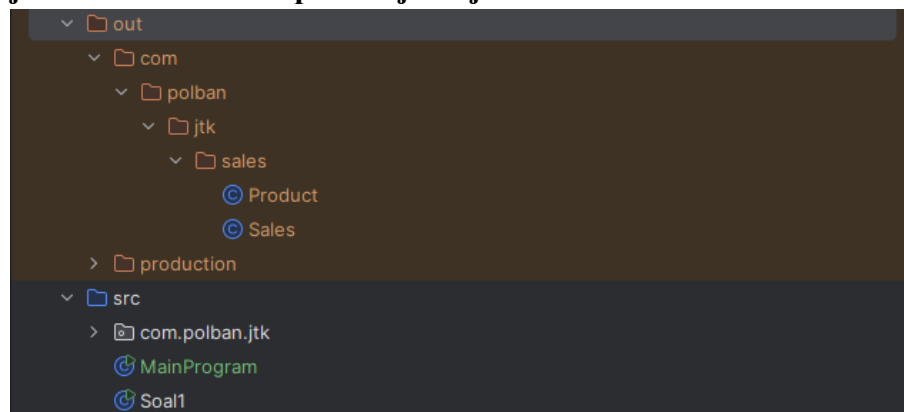
```
"C:\Program Files\Java\jdk-22\bin\java.exe" "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA Community Edition
Memproses penjualan...
5 Laptop terjual.
Stok setelah penjualan: 5
Menambah stok...
Stok setelah penambahan: 15
Memperbarui harga produk...
Harga baru: 14500000,00
Memperbarui harga produk...
Harga tidak valid!
Harga baru: 14500000,00

Process finished with exit code 0
```

#### Pembahasan, Screenshot Jawaban dan Solusi

⇒ Soal 3 membahas program yang sama dengan Soal 1 di mana program enkapsulasi class dilakukan sedikit perubahan dengan melakukan compile dan membuat library (.jar) pada file .java yang terdapat pada package com/polban/jtk/sales/\*.java dengan cara:

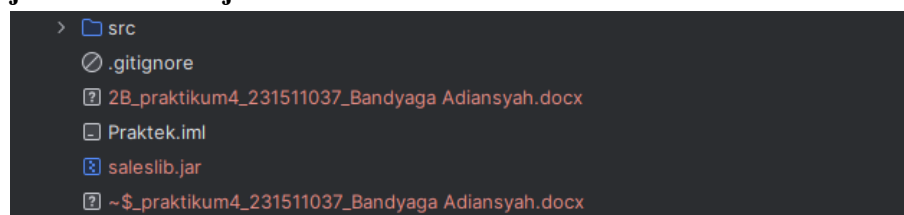
**javac -d out src/com/polban/jtk/\*.java**



Ini akan membuat sebuah class yang terletak di folder out/com/polban/jtk/sales/ yang merupakan hasil compile tadi.

⇒ Kemudian membuat file saleslib.jar dengan cara:

**jar cvf saleslib.jar -C out**



Ini akan menghasilkan file saleslib.jar seperti gambar di atas.

⇒ Buat program baru yang bernama Main yang berisi code program main class di Soal 1 namun juga dilakukan import jar file dengan cara:

**import com.polban.jtk.sales.\*;**

Program akan berjalan sama seperti pada Soal 1.

Referensi Syntax dan Methods:

W3Schools: <https://www.w3schools.com/java>

Tutorialspoint: [https://www.tutorialspoint.com/java/number\\_parseint.htm](https://www.tutorialspoint.com/java/number_parseint.htm)

Geeksforgeeks: <https://www.geeksforgeeks.org/biginteger-class-in-java/>

ChatGPT: <https://chatgpt.com/>

Lampiran:

Github: <https://github.com/basganajaah/Pemrograman-Berorientasi-Objek---Praktek>