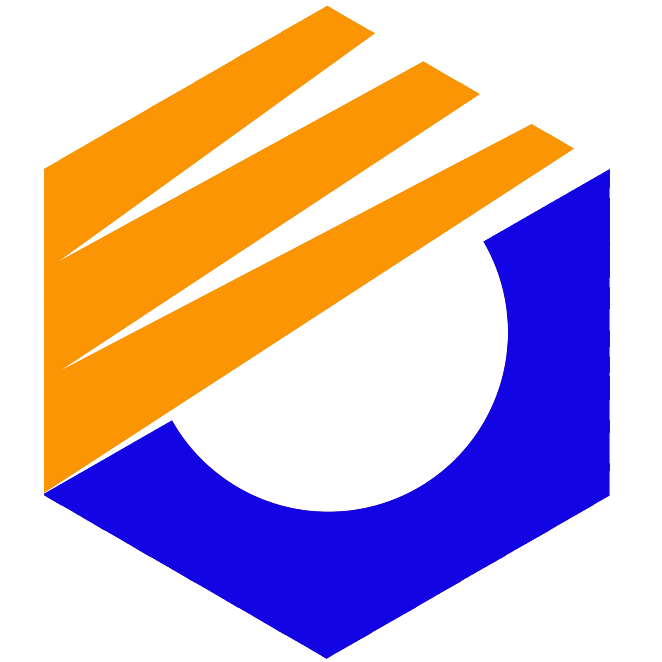
**Tugas Praktikum 4**

**Pemrograman Berorientasi Objek**



Disusun oleh:

Restu Akbar

231511088

**Jurusan Teknik Komputer dan Informatika**

**Politeknik Negeri Bandung**

**Link GITHUB: https://github.com/akbarst3/PBO-Archive.git**

**SOAL1**

Kode:

Main:

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

public class Soal1 {

    public static void main(String[] args) {

        Product produk1 = new Product("Laptop", 1000000000, 10);

        Sales sales1 =  new Sales(produk1);

        sales1.sellProduct(3);

        sales1.restockProduct(5);

        sales1.updateProductPrice(15000000);

    }

}

Kelas-kelas:

Product.java

package com.polban.jtk.sales;

*// Kelas Product*

public class Product {

*// Atribut private untuk menyembunyikan informasi*

    private String productName;

    private double price;

    private int stock;

*// Constructor*

    public Product(String productName, double price, int stock) {

        this.productName = productName;

        this.price = price;

        this.stock = stock;

    }

*// Getter dan Setter untuk mengontrol akses ke atribut*

    public String getProductName() {

        return productName;

    }

    public double getPrice() {

        return price;

    }

    public void setPrice(double price) {

        if (price > 0) {

            this.price = price;

        } else {

            System.out.println("Harga tidak valid!");

        }

    }

    public int getStock() {

        return stock;

    }

    public void addStock(int amount) {

        if (amount > 0) {

            this.stock += amount;

        } else {

            System.out.println("Jumlah stok tidak valid!");

        }

    }

    public void sellProduct(int quantity) {

        if (quantity > 0 && quantity <= stock) {

            stock -= quantity;

            System.out.println(quantity + " " + productName + " terjual.");

        } else {

            System.out.println("Jumlah stok tidak cukup untuk penjualan.");

        }

    }

}

Sales.java

package com.polban.jtk.sales;

*// Kelas Sales untuk mengelola transaksi penjualan*

public class Sales {

    private Product product;

*// Constructor*

    public Sales(Product product) {

        this.product = product;

    }

*// Metode untuk melakukan penjualan*

    public void sellProduct(int quantity) {

        System.out.println("Memproses penjualan...");

        product.sellProduct(quantity);

        System.out.println("Stok setelah penjualan: " +

                product.getStock());

    }

*// Metode untuk menambah stok produk*

    public void restockProduct(int quantity) {

        System.out.println("Menambah stok...");

        product.addStock(quantity);

        System.out.println("Stok setelah penambahan: " +

                product.getStock());

    }

*// Metode untuk memperbarui harga produk*

    public void updateProductPrice(double newPrice) {

        System.out.println("Memperbarui harga produk...");

        product.setPrice(newPrice);

        System.out.println("Harga baru: " + (int) product.getPrice());

    }

}

Output:

A screen shot of a computer screen

Description automatically generated

Penjelasan:

Di kode ini saya mengubah apa yang ditampilkan di method updateProductPrice() milik kelas sales agar menampilkan data harga baru yang sudah dicasting ke integer, sehingga output yang dihasilkan merupakan bilangan bulat penuh bukan menggunakan simbol seperti E karena itu merupakan bawaan dari tipe data double.

**SOAL2**

Kode:

Main:

import com.polban.jtk.inventory.Inventori;

public class Soal2 {

    public static void main(String[] args) {

        Inventori main = new Inventori();

        main.main(args);

    }

}

Kelas-kelas:

Barang.java

package com.polban.jtk.inventory;

public class Barang {

    String kode\_barang;

    String nama\_barang;

    private int stok;

    public Barang(String kode, String nama, int stk) {

        kode\_barang = kode;

        nama\_barang = nama;

        stok = stk;

    }

    public void tambahStok (int jumlah) {

        stok += jumlah;

    }

    public int getStok() {

        return this.stok;

    }

}

Inventori.java

package com.polban.jtk.inventory;

public class Inventori {

    Barang[] barangs;

    void initBarang() {

        barangs = new Barang[2];

        barangs[0] = new Barang("001", "Baju", 10);

        barangs[1] = new Barang("002", "Celana", 20);

    }

    void showBarang() {

        for (int i = 0; i < barangs.length; i++) {

            System.out.println(barangs[i].nama\_barang + "(" + barangs[i].getStok() + ")");

        }

    }

    void pengadaan() {

        initBarang();

        for (int i = 0; i < barangs.length; i++) {

            barangs[i].tambahStok(5);

        }

        showBarang();

    }

    public static void main(String[] args) {

        Inventori beli = new Inventori();

        beli.pengadaan();

    }

}

Output:



Penjelasan:

Di kode ini saya mengubah atribut stok pada kelas barang menjadi private agar tidak bisa diubah di luar dari kelasnya. Kemudian menambahkan method tambah stok agar perubahan stok hanya dapat dijumlahkan melalui method tambahStok(). Lalu juga meambahkan getter agar kelas lain dapat mengakses atribut dari stok. Terakhir, ada penyesuaian method pengadaan() milik kelas inventori agar menggunaka method-method baru dari barang untuk megakses jumlah stok dari object yang sudah dibuat.

**Soal3**

Kasus ini merupakan penerapan file jar dalam java, yaitu penggunaan file library atau package yang dapat mengefisiensikan pembuatan project dalam java. Saya di sini sudah berhasil membuat file jar dan membuatnya dapat digunakan dalam main baru dalam project java

A screenshot of a computer program

Description automatically generated

Output:

A black background with white text

Description automatically generated