

**Nama : Nelsi**

**Kelas : C2'23**

**Nim : 2309106120**

## **Posttest 4**

**Programnya :**

```
import java.util.*;
```

```
abstract class Person {
```

```
    protected String nama;
```

```
    protected String kontak;
```

```
    public Person(String nama, String kontak) {
```

```
        this.nama = nama;
```

```
        this.kontak = kontak;
```

```
    }
```

```
    public abstract void display();
```

```
}
```

```
class Supplier extends Person {
```

```
    private String alamat;
```

```
    public Supplier(String nama, String kontak, String alamat) {
```

```
        super(nama, kontak);
```

```
        this.alamat = alamat;
```

```
    }
```

```
    public String getNama() {
```

```
        return nama;
```

```
    }
```

*@Override*

```
public void display() {  
    System.out.println("Supplier: " + nama + ", Kontak: " + kontak + ", Alamat: " + alamat);  
}  
}
```

```
class BahanBaku {  
    private String nama;  
    private int stok;  
    private double harga;  
  
    public BahanBaku(String nama, int stok, double harga) {  
        this.nama = nama;  
        this.stok = stok;  
        this.harga = harga;  
    }  
  
    public String getNama() {  
        return nama;  
    }  
  
    public int getStok() {  
        return stok;  
    }  
  
    public double getHarga() {  
        return harga;  
    }  
  
    public void setStok(int stok) {  
        this.stok = stok;  
    }  
}
```

```
}
```

```
public void setHarga(double harga) {
```

```
    this.harga = harga;
```

```
}
```

```
public void tampilkan() {
```

```
    System.out.println("Nama: " + nama + ", Stok: " + stok + ", Harga: " + harga);
```

```
}
```

```
}
```

```
class Transaksi {
```

```
    private String namaBarang;
```

```
    private int jumlah;
```

```
    private double total;
```

```
public Transaksi(String namaBarang, int jumlah, double total) {
```

```
    this.namaBarang = namaBarang;
```

```
    this.jumlah = jumlah;
```

```
    this.total = total;
```

```
}
```

```
public void tampilkan() {
```

```
    System.out.println("Transaksi - Barang: " + namaBarang + ", Jumlah: " + jumlah + ", Total: " + total);
```

```
}
```

```
}
```

```
public class SistemWarung {
```

```
    static Scanner input = new Scanner(System.in);
```

```
    static ArrayList<BahanBaku> daftarBahan = new ArrayList<>();
```

```
    static ArrayList<Supplier> daftarSupplier = new ArrayList<>();
```

```
    static ArrayList<Transaksi> daftarTransaksi = new ArrayList<>();
```

```

public static void main(String[] args) {
    int pilihan;
    do {
        System.out.println("\n--- SISTEM WARUNG SEMBAKO ---");
        System.out.println("1. Kelola Bahan Baku");
        System.out.println("2. Kelola Supplier");
        System.out.println("3. Catat Transaksi");
        System.out.println("4. Cari Bahan Baku");
        System.out.println("5. Urutkan Bahan Baku berdasarkan Nama");
        System.out.println("0. Keluar");
        System.out.print("Pilih menu: ");
        pilihan = input.nextInt();
        input.nextLine();

        switch (pilihan) {
            case 1 -> kelolaBahanBaku();
            case 2 -> kelolaSupplier();
            case 3 -> catatTransaksi();
            case 4 -> cariBahan();
            case 5 -> urutkanBahan();
            case 0 -> System.out.println("Terima kasih!");
            default -> System.out.println("Pilihan tidak valid!");
        }
    } while (pilihan != 0);
}

```

```

public static void kelolaBahanBaku() {
    int pilih;
    do {
        System.out.println("\n--- Kelola Bahan Baku ---");
        System.out.println("1. Tambah Bahan");
    }
}

```

```

System.out.println("2. Tampilkan Semua");
System.out.println("3. Edit Stok/Harga");
System.out.println("4. Hapus Bahan");
System.out.println("0. Kembali");

System.out.print("Pilih: ");
pilih = input.nextInt();
input.nextLine();

switch (pilih) {
    case 1 -> {
        System.out.print("Nama bahan: ");
        String nama = input.nextLine();
        System.out.print("Stok: ");
        int stok = input.nextInt();
        System.out.print("Harga: ");
        double harga = input.nextDouble();
        daftarBahan.add(new BahanBaku(nama, stok, harga));
    }
    case 2 -> daftarBahan.forEach(BahanBaku::tampilkan);
    case 3 -> {
        System.out.print("Nama bahan yang ingin diedit: ");
        String namaEdit = input.nextLine();
        for (BahanBaku b : daftarBahan) {
            if (b.getNama().equalsIgnoreCase(namaEdit)) {
                System.out.print("Stok baru: ");
                b.setStok(input.nextInt());
                System.out.print("Harga baru: ");
                b.setHarga(input.nextDouble());
            }
        }
    }
    case 4 -> {

```

```

        System.out.print("Nama bahan yang ingin dihapus: ");
        String namaHapus = input.nextLine();
        daftarBahan.removeIf(b -> b.getNama().equalsIgnoreCase(namaHapus));
    }
}
} while (pilih != 0);
}

```

```

public static void kelolaSupplier() {
    System.out.print("Nama supplier: ");
    String nama = input.nextLine();
    System.out.print("Kontak: ");
    String kontak = input.nextLine();
    System.out.print("Alamat: ");
    String alamat = input.nextLine();
    daftarSupplier.add(new Supplier(nama, kontak, alamat));

    System.out.println("\nDaftar Supplier:");
    for (Supplier s : daftarSupplier)
        s.display();
}

```

```

public static void catatTransaksi() {
    System.out.print("Nama barang: ");
    String namaBarang = input.nextLine();
    System.out.print("Jumlah beli: ");
    int jumlah = input.nextInt();

    for (BahanBaku b : daftarBahan) {
        if (b.getNama().equalsIgnoreCase(namaBarang)) {
            double total = jumlah * b.getHarga();
            daftarTransaksi.add(new Transaksi(namaBarang, jumlah, total));
        }
    }
}

```

```
        System.out.println("Total harga: " + total);  
        return;  
    }  
}  
System.out.println("Barang tidak ditemukan.");  
}
```

```
public static void cariBahan() {  
    System.out.print("Masukkan nama bahan: ");  
    String cari = input.nextLine();  
    for (BahanBaku b : daftarBahan) {  
        if (b.getNama().equalsIgnoreCase(cari)) {  
            b.tampilkan();  
            return;  
        }  
    }  
    System.out.println("Bahan tidak ditemukan.");  
}
```

```
public static void urutkanBahan() {  
    daftarBahan.sort(Comparator.comparing(BahanBaku::getNama));  
    System.out.println("\nData setelah diurutkan:");  
    daftarBahan.forEach(BahanBaku::tampilkan);  
}  
}
```

## Outputnya :

### 1. Kelola Bahan Baku

```
--- SISTEM WARUNG SEMBAKO ---
1. Kelola Bahan Baku
2. Kelola Supplier
3. Catat Transaksi
4. Cari Bahan Baku
5. Urutkan Bahan Baku berdasarkan Nama
0. Keluar
Pilih menu:
```

```
--- SISTEM WARUNG SEMBAKO ---
1. Kelola Bahan Baku
2. Kelola Supplier
3. Catat Transaksi
4. Cari Bahan Baku
5. Urutkan Bahan Baku berdasarkan Nama
0. Keluar
Pilih menu: 1
```

```
--- Kelola Bahan Baku ---
1. Tambah Bahan
2. Tampilkan Semua
3. Edit Stok/Harga
4. Hapus Bahan
0. Kembali
Pilih: 1
Nama bahan: beras
Stok: 20
Harga: 120
Bahan berhasil ditambahkan!
```

```
--- Kelola Bahan Baku ---
1. Tambah Bahan
2. Tampilkan Semua
3. Edit Stok/Harga
4. Hapus Bahan
0. Kembali
Pilih: 2
Nama: beras, Stok: 20, Harga: 120.0
```

```
--- Kelola Bahan Baku ---
1. Tambah Bahan
2. Tampilkan Semua
3. Edit Stok/Harga
4. Hapus Bahan
0. Kembali
Pilih: 3
Nama bahan yang ingin diedit: beras
Stok baru: 10
Harga baru: 30
Data berhasil diupdate.
```

```
--- Kelola Bahan Baku ---
1. Tambah Bahan
2. Tampilkan Semua
3. Edit Stok/Harga
4. Hapus Bahan
0. Kembali
Pilih: 2
Nama: beras, Stok: 10, Harga: 30.0
```

```
--- Kelola Bahan Baku ---
1. Tambah Bahan
2. Tampilkan Semua
3. Edit Stok/Harga
4. Hapus Bahan
0. Kembali
Pilih: 4
Nama bahan yang ingin dihapus: beras
Bahan berhasil dihapus.
```



## 2. Kelola supplier

```
--- SISTEM WARUNG SEMBAKO ---
1. Kelola Bahan Baku
2. Kelola Supplier
3. Catat Transaksi
4. Cari Bahan Baku
5. Urutkan Bahan Baku berdasarkan Nama
0. Keluar
Pilih menu: 2
Nama supplier: Nelsi
Kontak: 082159540367
Alamat: m.yamin

Daftar Supplier:
Supplier: Nelsi, Kontak: 082159540367, Alamat: m.yamin
```

## 3. Catat Transaksi

```
--- SISTEM WARUNG SEMBAKO ---
1. Kelola Bahan Baku
2. Kelola Supplier
3. Catat Transaksi
4. Cari Bahan Baku
5. Urutkan Bahan Baku berdasarkan Nama
0. Keluar
Pilih menu: 3
Nama barang: beras
Jumlah beli: 1
Total harga: 8.0
```

## 4. Cari bahan baku

```
--- SISTEM WARUNG SEMBAKO ---
1. Kelola Bahan Baku
2. Kelola Supplier
3. Catat Transaksi
4. Cari Bahan Baku
5. Urutkan Bahan Baku berdasarkan Nama
0. Keluar
Pilih menu: 4
Masukkan nama bahan: beras
Nama: beras, Stok: 30, Harga: 8.0
```

## 5. Urutkan bahan baku berdasarkan nama

```
--- SISTEM WARUNG SEMBAKO ---
1. Kelola Bahan Baku
2. Kelola Supplier
3. Catat Transaksi
4. Cari Bahan Baku
5. Urutkan Bahan Baku berdasarkan Nama
0. Keluar
Pilih menu: 5

Data setelah diurutkan:
Nama: beras, Stok: 30, Harga: 8.0
```

## 6. Keluar

```
--- SISTEM WARUNG SEMBAKO ---
1. Kelola Bahan Baku
2. Kelola Supplier
3. Catat Transaksi
4. Cari Bahan Baku
5. Urutkan Bahan Baku berdasarkan Nama
0. Keluar
Pilih menu: 0
Terima kasih telah menggunakan sistem.
PS C:\Users\lenovo>
```