

LAPORAN Tugas Teori & Praktikum

Abstract Class & Interface

Diajukan untuk memenuhi salah satu tugas praktikum Mata kuliah Pemrograman Berorientasi Object



Disusun Oleh:
Muhammad Azzam Izzuddin (231511054)
2B-D3
Jurusan Teknik Komputer dan Informatika

Program Studi D-3 Teknik Informatika
Politeknik Negeri Bandung

Daftar isi

Soal:	3
Jawaban:.....	3
Code Program:	3
Diagram kelas:	5
Main Class:	6
screenshot aplikasi yang sudah dijalankan.....	8
penjelasan dari source code tersebut.	8

Soal:

1. Cari/buat program yang mengimplementasikan konsep abstract class, interface dan implementation dalam satu kasus.
2. Buat diagram kelasnya
2. Buat main classnya juga dengan menjalankan aplikasi yang mengimplementasikan konsep data.
3. Buat dokumennya dengan isi:
 - cover,
 - screenshot source code,
 - screenshot aplikasi yang sudah dijalankan
 - penjelasan dari source code tersebut.

Jawaban:

Inheritance:

Makanan dan Minuman mewarisi dari kelas abstrak MenuItem.

Makanan dan Minuman mengimplementasikan antarmuka DapatDidiskon.

Interface Implementation:

Makanan dan Minuman mengimplementasikan antarmuka DapatDidiskon, memungkinkan keduanya menerima diskon.

Code Program:

MenuItem.java

```
import java.util.ArrayList;

// Kelas Abstrak
public abstract class MenuItem {
    protected String nama;
    protected double harga;

    public MenuItem(String nama, double harga) {
        this.nama = nama;
        this.harga = harga;
    }

    public abstract void tampilkanInfo();

    public double getHarga() {
        return harga;
    }

    public void setHarga(double harga) {
        this.harga = harga;
    }
}
```

DapatDiskon.java

```
// Antarmuka
public interface DapatDidiskon {
    void terapkanDiskon(double persentase);
}
```

Makanan.java

```
// Kelas Implementasi untuk Makanan
public class Makanan extends MenuItem implements DapatDidiskon {
    private String jenis; // Jenis makanan (misalnya: pembuka, utama, penutup)

    public Makanan(String nama, double harga, String jenis) {
        super(nama, harga);
        this.jenis = jenis;
    }

    @Override
    public void tampilkanInfo() {
        System.out.printf("Item Makanan: %s, Harga: $%.2f, Jenis: %s\n", nama, harga,
jenis);
    }

    @Override
    public void terapkanDiskon(double persentase) {
        double jumlahDiskon = harga * (persentase / 100);
        harga -= jumlahDiskon;
    }
}
```

Minuman.java

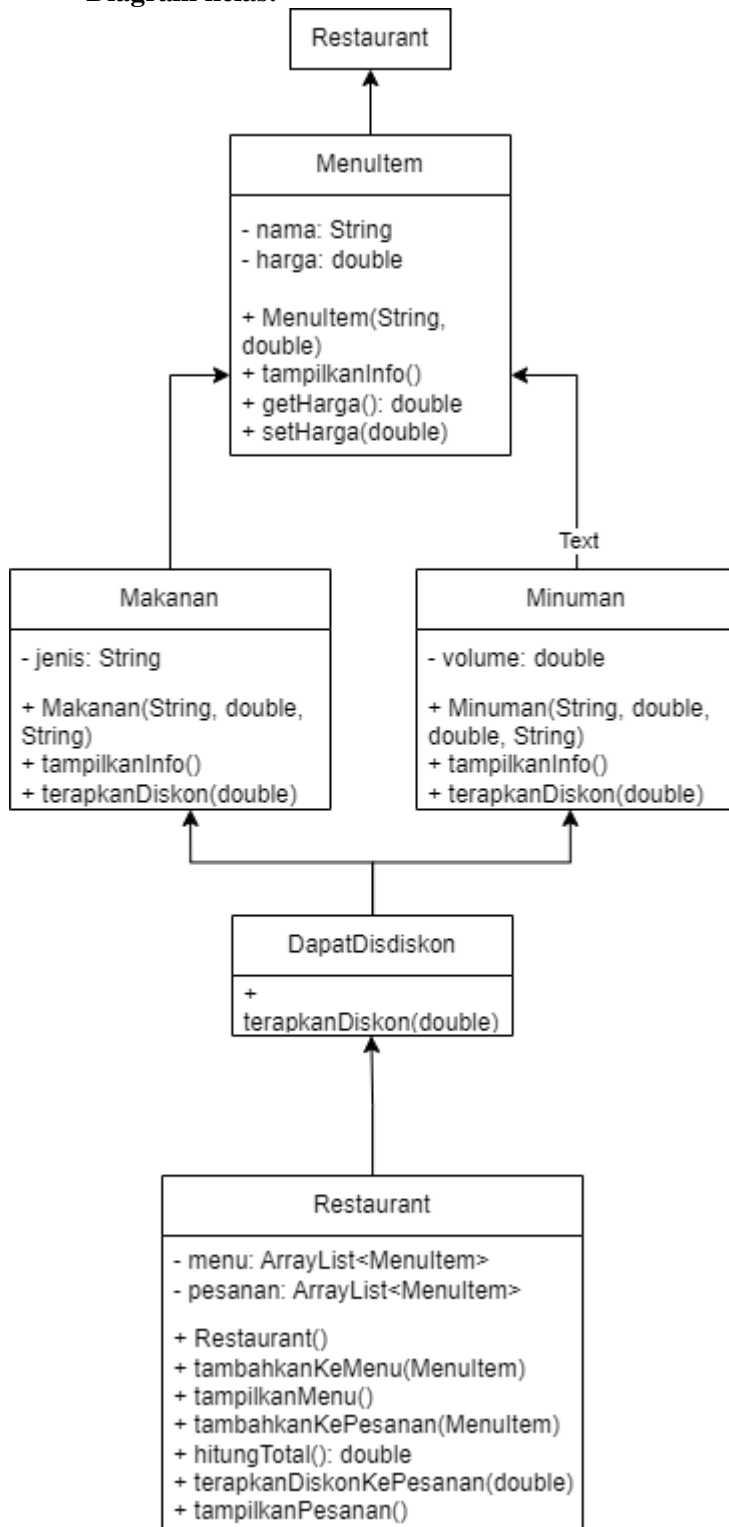
```
// Kelas Implementasi untuk Minuman
public class Minuman extends MenuItem implements DapatDidiskon {
    private double volume; // dalam liter
    private String jenis; // Jenis minuman (misalnya: soda, jus, alkohol)

    public Minuman(String nama, double harga, double volume, String jenis) {
        super(nama, harga);
        this.volume = volume;
        this.jenis = jenis;
    }

    @Override
    public void tampilkanInfo() {
        System.out.printf("Item Minuman: %s, Harga: $%.2f, Volume: %.1f L, Jenis: %s\n",
nama, harga, volume, jenis);
    }

    @Override
    public void terapkanDiskon(double persentase) {
        double jumlahDiskon = harga * (persentase / 100);
        harga -= jumlahDiskon;
    }
}
```

Diagram kelas:



Main Class:

Restaurant.java

```
import java.util.ArrayList;

// Kelas Utama untuk Sistem Manajemen Restoran
public class Restaurant {
    private ArrayList<MenuItem> menu;
    private ArrayList<MenuItem> pesanan;

    public Restaurant() {
        menu = new ArrayList<>();
        pesanan = new ArrayList<>();
    }

    public void tambahkanKeMenu(MenuItem item) {
        menu.add(item);
    }

    public void tampilkanMenu() {
        System.out.println("Menu Restoran:");
        for (MenuItem item : menu) {
            item.tampilkanInfo();
        }
        System.out.println();
    }

    public void tambahkanKePesanan(MenuItem item) {
        pesanan.add(item);
        System.out.printf("Ditambahkan %s ke pesanan.\n", item.nama);
    }

    public double hitungTotal() {
        double total = 0;
        for (MenuItem item : pesanan) {
            total += item.getHarga();
        }
        return total;
    }

    public void terapkanDiskonKePesanan(double persentase) {
        for (MenuItem item : pesanan) {
            if (item instanceof DapatDidiskon) {
                DapatDidiskon itemDiskon = (DapatDidiskon) item;
                itemDiskon.terapkanDiskon(persentase);
            }
        }
    }

    public void tampilkanPesanan() {
        System.out.println("Pesanan Saat Ini:");
        for (MenuItem item : pesanan) {
            item.tampilkanInfo();
        }
    }
}
```

```

    }
    double totalHarga = hitungTotal();
    System.out.printf("Total Harga: $%.2f\n", totalHarga);
}

public static void main(String[] args) {
    // Membuat restoran
    Restaurant restaurant = new Restaurant();

    // Menambahkan item ke menu
    Makanan pizza = new Makanan("Pizza", 12.99, "Utama");
    Makanan salad = new Makanan("Salad", 8.49, "Pembuka");
    Makanan esKrim = new Makanan("Es Krim", 4.99, "Penutup");

    Minuman soda = new Minuman("Soda", 1.99, 0.5, "Soda");
    Minuman jusJeruk = new Minuman("Jus Jeruk", 2.49, 0.25, "Jus");
    Minuman bir = new Minuman("Bir", 3.99, 0.33, "Alkohol");

    // Menambahkan item ke menu restoran
    restaurant.tambahkanKeMenu(pizza);
    restaurant.tambahkanKeMenu(salad);
    restaurant.tambahkanKeMenu(esKrim);
    restaurant.tambahkanKeMenu(soda);
    restaurant.tambahkanKeMenu(jusJeruk);
    restaurant.tambahkanKeMenu(bir);

    // Menampilkan menu restoran
    restaurant.tampilkanMenu();

    // Menambahkan item ke pesanan
    restaurant.tambahkanKePesanan(pizza);
    restaurant.tambahkanKePesanan(soda);

    // Menampilkan pesanan saat ini
    restaurant.tampilkanPesanan();

    // Menerapkan diskon ke pesanan
    System.out.println("\nMenerapkan diskon 10% ke pesanan.");
    restaurant.terapkanDiskonKePesanan(10);

    // Menampilkan pesanan setelah diskon
    restaurant.tampilkanPesanan();
}
}

```

screenshot aplikasi yang sudah dijalankan

```
PS D:\A\pbo\mingguke7\restaurant> javac Restaurant.java MenuItem.java Makanan.java Minuman.java DapatDidiskon.java
PS D:\A\pbo\mingguke7\restaurant> java Restaurant
Menu Restoran:
Item Makanan: Pizza, Harga: $12,99, Jenis: Utama
Item Makanan: Salad, Harga: $8,49, Jenis: Pembuka
Item Makanan: Es Krim, Harga: $4,99, Jenis: Penutup
Item Minuman: Soda, Harga: $1,99, Volume: 0,5 L, Jenis: Soda
Item Minuman: Jus Jeruk, Harga: $2,49, Volume: 0,3 L, Jenis: Jus
Item Minuman: Bir, Harga: $3,99, Volume: 0,3 L, Jenis: Alkohol

Ditambahkan Pizza ke pesanan.
Ditambahkan Soda ke pesanan.
Pesanan Saat Ini:
Item Makanan: Pizza, Harga: $12,99, Jenis: Utama
Item Minuman: Soda, Harga: $1,99, Volume: 0,5 L, Jenis: Soda
Total Harga: $14,98

Menerapkan diskon 10% ke pesanan.
Pesanan Saat Ini:
Item Makanan: Pizza, Harga: $11,69, Jenis: Utama
Item Minuman: Soda, Harga: $1,79, Volume: 0,5 L, Jenis: Soda
Total Harga: $13,48
PS D:\A\pbo\mingguke7\restaurant> 
```

penjelasan dari source code tersebut.

Abstrak MenuItem: Ini adalah kelas dasar yang mewakili item di menu restoran, seperti makanan dan minuman. Kelas ini memiliki properti umum seperti nama dan harga. Karena abstrak, kelas ini tidak bisa diinstansiasi secara langsung dan perlu diimplementasikan oleh kelas turunan.

DapatDidiskon: Ini adalah antarmuka yang mendefinisikan perilaku diskon. Kelas yang mengimplementasikannya harus bisa menerapkan diskon terhadap harga barang berdasarkan persentase.

Makanan dan Minuman: Keduanya merupakan turunan dari MenuItem dan mengimplementasikan antarmuka DapatDidiskon. Kelas ini mewakili item makanan dan minuman yang memiliki properti spesifik, seperti jenis makanan (pembuka, utama, penutup) dan volume minuman. Selain itu, kelas ini juga menyediakan metode untuk menampilkan informasi item dan menerapkan diskon.

Restaurant: Kelas ini bertugas mengelola menu restoran dan pesanan pelanggan. Di sini, menu disimpan dalam list, pesanan dikelola, dan total harga pesanan dihitung. Kelas ini juga bisa menerapkan diskon ke seluruh pesanan yang memenuhi syarat (misalnya makanan atau minuman yang bisa diberi diskon).

main: Bagian ini adalah titik awal program. Restoran diciptakan, item ditambahkan ke menu, pesanan dibuat, dan diskon diterapkan. Program ini juga menampilkan menu dan pesanan, serta menghitung total setelah diskon.