

Teori Tugas 2

Disusun untuk memenuhi tugas mata kuliah Pemrograman Berorientasi Objek (Teori)



Disusun Oleh:

Dwika Ali Ramdhan (231511042)

2B – D3

Jurusan Teknik Komputer dan Informatika

Politeknik Negeri Bandung

2024

DAFTAR ISI

DAFTAR ISI	1
PEMBAHASAN	1
1.1. Setup Software Engine.....	1
1.2. Program dan Penjelasan.....	1
1.3. Output.....	5

PEMBAHASAN

1.1. Setup Software Engine

```
C:\Users\bushi>java -version
java version "22.0.2" 2024-07-16
Java(TM) SE Runtime Environment (build 22.0.2+9-70)
Java HotSpot(TM) 64-Bit Server VM (build 22.0.2+9-70, mixed mode, sharing)
```

1.2. Program dan Penjelasan

1.2.1. public static class Booth {

```
String namaPenjual;
ArrayList<Menu.Makanan> daftarMakanan;
ArrayList<Menu.Minuman> daftarMinuman;

public Booth(String namaPenjual) {
    this.namaPenjual = namaPenjual;
    this.daftarMakanan = new ArrayList<>();
    this.daftarMinuman = new ArrayList<>();
}

public void tambahMakanan(Menu.Makanan makanan) {
    daftarMakanan.add(makanan);
}
public void tambahMinuman(Menu.Minuman minuman) {
    daftarMinuman.add(minuman);
}
public String getNamaPenjual() {
    return namaPenjual;
}
public ArrayList<Menu.Makanan> getDaftarMakanan() {
    return daftarMakanan;
}
public ArrayList<Menu.Minuman> getDaftarMinuman() {
    return daftarMinuman;
}
}
```

Kelas ini merepresentasikan booth atau penjual di pugasera. Setiap booth memiliki nama penjual serta menu daftar makanan dan minuman yang disajikan. Metode di kelas ini memungkinkan untuk menambah makanan dan minuman ke daftar serta mendapatkan informasi tentang nama penjual dan daftar item.

1.2.2. public static class Menu {

```
public static class Makanan {
    String namaMakanan;
    Integer harga;

    public Makanan(String namaMakanan, Integer harga) {
        this.namaMakanan = namaMakanan;
        this.harga = harga;
    }
    public String getNamaMakanan() {
        return namaMakanan;
    }
    public Integer getHarga() {
        return harga;
    }
}

public static class Minuman {
    String namaMinuman;
    Integer harga;

    public Minuman(String namaMinuman, Integer harga) {
        this.namaMinuman = namaMinuman;
        this.harga = harga;
    }
    public String getNamaMinuman() {
        return namaMinuman;
    }
    public Integer getHarga() {
        return harga;
    }
}
```

Kelas ini berisi subkelas Makanan dan Minuman yang mendefinisikan item-item yang dijual. Kelas Makanan memiliki atribut nama dan harga, dan kelas minuman memiliki atribut nama dan harga pula.

1.2.3. public static class Pembeli {

```
String namaPembeli;
Booth penjualTerpilih;
Menu.Makanan makananTerpilih;
Menu.Minuman minumanTerpilih;

public Pembeli(String namaPembeli) {
    this.namaPembeli = namaPembeli;
}
public void pilihPenjual(Booth penjual) {
    this.penjualTerpilih = penjual;
}
public void pilihMakanan(Menu.Makanan makanan) {
    this.makananTerpilih = makanan;
}
public void pilihMinuman(Menu.Minuman minuman) {
    this.minumanTerpilih = minuman;
}

public void tampilkanPembelian() {
    if (penjualTerpilih == null || makananTerpilih == null || minumanTerpilih == null) {
        System.out.println("Belum ada pembelian yang dilakukan.");
    }
}
```

```

    } else {
        System.out.println("Nama Pembeli: " + namaPembeli);
        System.out.println("Penjual: " + penjualTerpilih.getNamaPenjual());
        System.out.println("Makanan: " + makananTerpilih.getNamaMakanan() + " - Harga: " + makananTerpilih.getHarga());
        System.out.println("Minuman: " + minumanTerpilih.getNamaMinuman() + " - Harga: " + minumanTerpilih.getHarga());
    }
}
}
}

}

public ArrayList<Menu.Makanan> getDaftarMakanan() {
    return daftarMakanan;
}

public ArrayList<Menu.Minuman> getDaftarMinuman() {
    return daftarMinuman;
}
}

```

1.2.4. class MainClass {

```

public static void main(String[] args) {
    Scanner scanner = new Scanner(System.in);

    Pujasera.Booth booth1 = new Pujasera.Booth("Pak Ali");
    booth1.tambahMakanan(new Pujasera.Menu.Makanan("Nasi Goreng", 10000));
    booth1.tambahMakanan(new Pujasera.Menu.Makanan("Nasi Katsu", 13000));
    booth1.tambahMakanan(new Pujasera.Menu.Makanan("Spagetti", 10000));
    booth1.tambahMinuman(new Pujasera.Menu.Minuman("Teh Manis", 5000));
    booth1.tambahMinuman(new Pujasera.Menu.Minuman("Kopi", 6000));

    Pujasera.Booth booth2 = new Pujasera.Booth("Bu Entin");
    booth2.tambahMakanan(new Pujasera.Menu.Makanan("Soto Ayam", 13000));
    booth2.tambahMakanan(new Pujasera.Menu.Makanan("Bakso", 14000));
    booth2.tambahMakanan(new Pujasera.Menu.Makanan("Mie Ayam", 12000));
    booth2.tambahMinuman(new Pujasera.Menu.Minuman("Es Jeruk", 7000));
    booth2.tambahMinuman(new Pujasera.Menu.Minuman("Es Teh", 4000));

    Pujasera.Booth booth3 = new Pujasera.Booth("Pak Nkos");
    booth3.tambahMakanan(new Pujasera.Menu.Makanan("Nasi Padang", 10000));
    booth3.tambahMakanan(new Pujasera.Menu.Makanan("Sate Padang", 22000));
    booth3.tambahMakanan(new Pujasera.Menu.Makanan("Soto Padang", 12500));
    booth3.tambahMinuman(new Pujasera.Menu.Minuman("Teh Manis", 5000));
    booth3.tambahMinuman(new Pujasera.Menu.Minuman("Teh Jeruk", 7000));

    ArrayList<Pujasera.Booth> daftarPenjual = new ArrayList<>();
    daftarPenjual.add(booth1);
    daftarPenjual.add(booth2);
    daftarPenjual.add(booth3);

    ArrayList<Pujasera.Pembeli> daftarPembeli = new ArrayList<>();

    while (true) {
        System.out.println("\n1. jajan");
        System.out.println("2. Data pembeli");
        System.out.println("3. Keluar");
        System.out.print("Pilih opsi: ");
        int opsi = scanner.nextInt();
        scanner.nextLine();

        if (opsi == 1) {
            System.out.print("Masukkan nama pembeli: ");
            String namaPembeli = scanner.nextLine();
            Pujasera.Pembeli pembeli = new Pujasera.Pembeli(namaPembeli);
            daftarPembeli.add(pembeli);
        }
    }
}

```

```

        System.out.println("\nDaftar Penjual:");

        for (int i = 0; i < daftarPenjual.size(); i++) {
            System.out.println((i + 1) + ". " + daftarPenjual.get(i).getNamaPenjual());
        }

        System.out.print("Pilih penjual (masukkan nomor): ");
        int pilihanPenjual = scanner.nextInt();
        scanner.nextLine();
        Pujasera.Booth penjualTerpilih = daftarPenjual.get(pilihanPenjual - 1);
        pembeli.pilihPenjual(penjualTerpilih);

        System.out.println("\nDaftar Makanan:");
        ArrayList<Pujasera.Menu.Makanan> daftarMakanan = penjualTerpilih.getDaftarMakanan();
        for (int i = 0; i < daftarMakanan.size(); i++) {
            System.out.println((i + 1) + ". " + daftarMakanan.get(i).getNamaMakanan() + " - Harga: " +
            daftarMakanan.get(i).getHarga());
        }

        System.out.print("Pilih makanan (masukkan nomor): ");
        int pilihanMakanan = scanner.nextInt();
        scanner.nextLine();
        Pujasera.Menu.Makanan makananTerpilih = daftarMakanan.get(pilihanMakanan - 1);
        pembeli.pilihMakanan(makananTerpilih);

        System.out.println("\nDaftar Minuman:");
        ArrayList<Pujasera.Menu.Minuman> daftarMinuman = penjualTerpilih.getDaftarMinuman();
        for (int i = 0; i < daftarMinuman.size(); i++) {
            System.out.println((i + 1) + ". " + daftarMinuman.get(i).getNamaMinuman() + " - Harga: " +
            daftarMinuman.get(i).getHarga());
        }

        System.out.print("Pilih minuman (masukkan nomor): ");
        int pilihanMinuman = scanner.nextInt();
        scanner.nextLine();
        Pujasera.Menu.Minuman minumanTerpilih = daftarMinuman.get(pilihanMinuman - 1);
        pembeli.pilihMinuman(minumanTerpilih);

    } else if (opsi == 2) {
        if (daftarPembeli.isEmpty()) {
            System.out.println("Belum ada yang jajan.");
        } else {
            for (Pujasera.Pembeli pembeli : daftarPembeli) {
                pembeli.tampilkanPembelian();
                System.out.println();
            }
        }
    } else if (opsi == 3) {
        System.out.println("Terima kasih!");
        break;
    } else {
        System.out.println("Ops! tidak valid. Silakan pilih kembali.");
    }
}
scanner.close();
}

```

Kelas main yaitu metode main. Kelas ini menginisialisasi beberapa booth, serta daftar pembeli. Program berfungsi untuk menampilkan menu, menerima input pengguna, dan memproses transaksi serta menampilkan data pembelian sesuai dengan yang dipilih oleh pengguna.

1.3. Output

```
1. jajan
2. Data pembeli
3. Keluar
Pilih opsi: 1
```

```
Daftar Penjual:
1. Pak Ali
2. Bu Entin
3. Pak Nkos
Pilih penjual (masukkan nomor): 1
```

```
Daftar Makanan:
1. Nasi Goreng - Harga: 10000
2. Nasi Katsu - Harga: 13000
3. Spagetti - Harga: 10000
Pilih makanan (masukkan nomor): 2
```

```
Daftar Minuman:
1. Teh Manis - Harga: 5000
2. Kopi - Harga: 6000
Pilih minuman (masukkan nomor): 2
```

```
1. jajan
2. Data pembeli
3. Keluar
Pilih opsi: 2
Nama Pembeli: Dwika
Penjual: Pak Ali
Makanan: Nasi Katsu - Harga: 13000
Minuman: Kopi - Harga: 6000
```