

LAPORAN PRAKTIKUM
PEMROGRAMAN BERORIENTASI OBJEK
MODUL II
PENGKONDISIAN DAN PERULANGAN



Oleh :

FACHRULLAH ZHAFRAN LISTIYONO

2211103115

SI-07-C

PROGRAM STUDI S1 SISTEM INFORMASI
FAKULTAS REKAYASA INDUSTRI
UNIVERSITAS TELKOM PURWOKERTO
2024

I. GUIDED

Menu.java

Source Code:

```
package LatihanP4FachrullahZhafranListiyono;
// Author @Fachrullah Zhafran Listiyono 2211103115

public class Menu {

    String namaItem;
    double harga;
    int kategori; // 1: Pembuka, 2: Utama, 3: Minuman

    public Menu(String namaItem, double harga, int kategori) {
        this.namaItem = namaItem;
        this.harga = harga;
        this.kategori = kategori;
    }
    // Method untuk menampilkan informasi menu

    public void tampilkanInfo() {
        String jenis = "";
        switch (kategori) {
            case 1:
                jenis = "Pembuka";
                break;
            case 2:
                jenis = "Utama";
                break;
            case 3:
                jenis = "Minuman";
                break;
        }
        System.out.println(namaItem + " - Kategori: " + jenis + " -
        Harga: Rp " + harga);
    }
}
```

Penjelasan:

Kelas Menu bertugas merepresentasikan item makanan atau minuman di restoran. Kelas ini memiliki tiga atribut utama, yaitu `namaItem` (untuk menyimpan nama item menu dalam bentuk string), `harga` (untuk menyimpan harga item dalam bentuk angka desimal), dan `kategori` (untuk menyimpan jenis kategori dari item, di mana nilai 1 menunjukkan kategori pembuka, 2 untuk makanan utama, dan 3 untuk minuman). Konstruktor kelas ini menerima ketiga parameter tersebut untuk menginisialisasi objek Menu saat dibuat. Kelas ini juga menyediakan metode `tampilkanInfo()`, yang menampilkan nama item, kategori (diubah menjadi teks yang mudah dipahami, seperti "Pembuka" untuk kategori 1), dan harga. Informasi ini ditampilkan dalam format "Nama Item - Kategori: Jenis - Harga: Rp Harga". Misalnya, jika kategori adalah 1, maka akan ditampilkan sebagai "Pembuka". Metode ini mempermudah dalam menampilkan informasi item menu.

Pesanan.java

Source Code:

```
package LatihanP4FachrullahZhafranListiyono;
// Author @Fachrullah Zhafran Listiyono 2211103115

public class Pesanan {

    Menu[] daftarPesanan;
    int jumlahPesanan;
    double totalHarga;

    public Pesanan(int kapasitas) {
        daftarPesanan = new Menu[kapasitas];
        jumlahPesanan = 0;
        totalHarga = 0;
    }
    // Method untuk menambahkan pesanan

    public void tambahPesanan(Menu item) {
        daftarPesanan[jumlahPesanan] = item;
        totalHarga += item.harga;
        jumlahPesanan++;
    }
    // Method untuk menampilkan semua pesanan

    public void tampilkanPesanan() {
        System.out.println("Daftar Pesanan:");
        for (int i = 0; i < jumlahPesanan; i++) {
            daftarPesanan[i].tampilkanInfo();
        }
        System.out.println("Total Harga: Rp " + totalHarga);
    }
    // Method untuk menghitung diskon jika total melebihi 200.000

    public double hitungDiskon() {
        if (totalHarga > 200000) {
            return totalHarga * 0.10; // Diskon 10%
        } else {
            return 0;
        }
    }
}
```

Penjelasan:

Kelas Pesanan digunakan untuk mengelola pesanan dari pelanggan di sebuah restoran. Kelas ini memiliki tiga atribut: `daftarPesanan`, sebuah array yang menyimpan objek-objek Menu, `jumlahPesanan` yang menghitung jumlah pesanan yang ditambahkan, dan `totalHarga` untuk menyimpan total harga dari semua pesanan. Konstruktor `Pesanan` menerima parameter kapasitas yang digunakan untuk menentukan ukuran maksimal array `daftarPesanan`. Metode `tambahPesanan(Menu item)` menambahkan item menu ke dalam array dan memperbarui `totalHarga` dengan menambahkan harga dari item yang baru saja ditambahkan. Kelas ini juga memiliki metode `tampilkanPesanan()` yang menampilkan semua item dalam daftar pesanan beserta total harga keseluruhannya. Selain itu, ada metode `hitungDiskon()`, yang memberikan diskon 10% jika total harga

melebihi Rp 200.000. Diskon ini dihitung dengan mengalikan total harga dengan 0.10, dan jika tidak ada diskon, metode ini akan mengembalikan nilai 0.

LatihanP4.java

Source Code:

```
package LatihanP4FachrullahZhafranListiyono;
// Author @Fachrullah Zhafran Listiyono 2211103115

import java.util.Scanner;

public class LatihanP4 {

    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        Pesanan pesanan = new Pesanan(10); // Maksimal 10 pesanan
        char tambahPesanan;
        // Perulangan untuk menambah pesanan
        do {
            System.out.print("Masukkan nama item: ");
            String namaItem = scanner.nextLine();
            System.out.print("Masukkan harga item: ");
            double hargaItem = scanner.nextDouble();
            System.out.print("Masukkan kategori (1: Pembuka, 2:
Utama, 3: Minuman): ");
            int kategoriItem = scanner.nextInt();
            scanner.nextLine(); // Bersihkan buffer
            Menu item = new Menu(namaItem, hargaItem, kategoriItem);
            pesanan.tambahPesanan(item);
            System.out.print("Apakah ingin menambah pesanan lagi?
(y/n):");
            tambahPesanan = scanner.next().charAt(0);
            scanner.nextLine(); // Bersihkan buffer
        } while (tambahPesanan == 'y' || tambahPesanan == 'Y');
        // Tampilkan pesanan dan hitung diskon
        pesanan.tampilkanPesanan();
        double diskon = pesanan.hitungDiskon();
        if (diskon > 0) {
            System.out.println("Anda mendapatkan diskon 10% sebesar
Rp " + diskon);
        }
        System.out.println("Total yang harus dibayar: Rp " +
(pesanan.totalHarga - diskon));
    }
}
```

Screenshoot Output:

```
Output - P2_kondisional (run)

run:
Masukkan nama item: Mie Ayam
Masukkan harga item: 15000
Masukkan kategori (1: Pembuka, 2: Utama, 3: Minuman): 2
Apakah ingin menambah pesanan lagi? (y/n):n
Daftar Pesanan:
Mie Ayam - Kategori: Utama - Harga: Rp 15000.0
Total Harga: Rp 15000.0
Total yang harus dibayar: Rp 15000.0
BUILD SUCCESSFUL (total time: 28 seconds)
|
```

Penjelasan:

Kelas LatihanP4 adalah kelas utama yang menjalankan seluruh program. Program ini menggunakan objek dari kelas Pesanan dan Menu untuk mengelola input pesanan dari pengguna. Dalam metode main, program menggunakan objek Scanner untuk menerima input dari pengguna, seperti nama item, harga, dan kategori item. Program ini akan meminta pengguna untuk menambah pesanan secara berulang melalui perulangan do-while, di mana pengguna dapat menambahkan item baru hingga mereka memilih untuk berhenti. Setiap kali pengguna memasukkan item baru, objek Menu dibuat dan ditambahkan ke daftar pesanan dengan menggunakan metode tambahPesanan() dari kelas Pesanan. Setelah semua pesanan selesai dimasukkan, program menampilkan seluruh daftar pesanan beserta total harga menggunakan metode tampilkanPesanan(). Jika total harga melebihi Rp 200.000, program secara otomatis menghitung dan menampilkan diskon 10% melalui metode hitungDiskon(), lalu menampilkan total akhir yang harus dibayar setelah diskon.

II. UNGUIDED

Kembangkan Sistem Manajemen Pesanan dengan menambahkan fitur berikut:

1. Tambahkan validasi input kategori. Validasi input kategori sehingga program hanya menerima angka 1, 2, atau 3, dan jika input tidak valid, program akan meminta input ulang.
2. Tambahkan perhitungan pajak 10%. Setelah diskon diberikan, tambahkan fitur untuk menghitung pajak sebesar 10% dari total harga yang sudah dikurangi diskon.

Menu.java

Source Code:

```
package LatihanP4FachrullahZhafranListiyono;
// Author @Fachrullah Zhafran Listiyono 2211103115

public class Menu {

    String namaItem;
    double harga;
    int kategori; // 1: Pembuka, 2: Utama, 3: Minuman

    public Menu(String namaItem, double harga, int kategori) {
        this.namaItem = namaItem;
        this.harga = harga;
        this.kategori = kategori;
    }
    // Method untuk menampilkan informasi menu

    public void tampilkanInfo() {
        String jenis = "";
        switch (kategori) {
            case 1:
                jenis = "Pembuka";
                break;
            case 2:
                jenis = "Utama";
                break;
            case 3:
                jenis = "Minuman";
                break;
        }
        System.out.println(namaItem + " - Kategori: " + jenis + " -
        Harga: Rp " + harga);
    }
}
```

Penjelasan:

Kelas Menu bertugas merepresentasikan item makanan atau minuman di restoran. Kelas ini memiliki tiga atribut utama, yaitu namaItem (untuk menyimpan nama item menu dalam bentuk string), harga (untuk menyimpan harga item dalam bentuk angka desimal), dan kategori (untuk menyimpan jenis kategori dari item, di mana nilai 1 menunjukkan kategori pembuka, 2 untuk makanan utama, dan 3 untuk minuman). Konstruktor kelas ini menerima ketiga parameter tersebut untuk menginisialisasi objek Menu saat dibuat. Kelas ini juga menyediakan metode tampilkanInfo(), yang menampilkan nama item, kategori (diubah menjadi teks yang mudah dipahami, seperti "Pembuka" untuk kategori 1), dan harga. Informasi ini ditampilkan dalam format "Nama Item - Kategori: Jenis - Harga: Rp Harga". Misalnya, jika kategori adalah 1,

maka akan ditampilkan sebagai "Pembuka". Metode ini mempermudah dalam menampilkan informasi item menu.

Pesanan.java

Source Code:

```
package LatihanP4FachrullahZhafranListiyono;
// Author @Fachrullah Zhafran Listiyono 2211103115

public class Pesanan {

    Menu[] daftarPesanan;
    int jumlahPesanan;
    double totalHarga;

    public Pesanan(int kapasitas) {
        daftarPesanan = new Menu[kapasitas];
        jumlahPesanan = 0;
        totalHarga = 0;
    }
    // Method untuk menambahkan pesanan

    public void tambahPesanan(Menu item) {
        daftarPesanan[jumlahPesanan] = item;
        totalHarga += item.harga;
        jumlahPesanan++;
    }
    // Method untuk menampilkan semua pesanan

    public void tampilkanPesanan() {
        System.out.println("Daftar Pesanan:");
        for (int i = 0; i < jumlahPesanan; i++) {
            daftarPesanan[i].tampilkanInfo();
        }
        System.out.println("Total Harga: Rp " + totalHarga);
    }
    // Method untuk menghitung diskon jika total melebihi 200.000

    public double hitungDiskon() {
        if (totalHarga > 200000) {
            return totalHarga * 0.10; // Diskon 10%
        } else {
            return 0;
        }
    }
}
```

Penjelasan:

Kelas Pesanan digunakan untuk mengelola pesanan dari pelanggan di sebuah restoran. Kelas ini memiliki tiga atribut: daftarPesanan, sebuah array yang menyimpan objek-objek Menu, jumlahPesanan yang menghitung jumlah pesanan yang ditambahkan, dan totalHarga untuk menyimpan total harga dari semua pesanan. Konstruktor Pesanan menerima parameter kapasitas yang digunakan untuk menentukan ukuran maksimal

array daftarPesanan. Metode tambahPesanan(Menu item) menambahkan item menu ke dalam array dan memperbarui totalHarga dengan menambahkan harga dari item yang baru saja ditambahkan. Kelas ini juga memiliki metode tampilkanPesanan() yang menampilkan semua item dalam daftar pesanan beserta total harga keseluruhannya. Selain itu, ada metode hitungDiskon(), yang memberikan diskon 10% jika total harga melebihi Rp 200.000. Diskon ini dihitung dengan mengalikan total harga dengan 0.10, dan jika tidak ada diskon, metode ini akan mengembalikan nilai 0.

LatihanP4.java

Source Code:

```
package LatihanP4FachrullahZhafranListiyono;
// Author @Fachrullah Zhafran Listiyono 2211103115

import java.util.Scanner;

public class LatihanP4 {

    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        Pesanan pesanan = new Pesanan(10); // Maksimal 10 pesanan
        char tambahPesanan;

        // Perulangan untuk menambah pesanan
        do {
            System.out.print("Masukkan nama item: ");
            String namaItem = scanner.nextLine();
            System.out.print("Masukkan harga item: ");
            double hargaItem = scanner.nextDouble();

            // Validasi input kategori
            int kategoriItem;
            do {
                System.out.print("Masukkan kategori (1: Pembuka, 2:
Utama, 3: Minuman): ");
                kategoriItem = scanner.nextInt();
                if (kategoriItem < 1 || kategoriItem > 3) {
                    System.out.println("Kategori tidak valid! Harap
masukkan angka 1, 2, atau 3.");
                }
            } while (kategoriItem < 1 || kategoriItem > 3);

            scanner.nextLine();
            Menu item = new Menu(namaItem, hargaItem, kategoriItem);
            pesanan.tambahPesanan(item);

            System.out.print("Apakah ingin menambah pesanan lagi?
(y/n): ");
            tambahPesanan = scanner.next().charAt(0);
            scanner.nextLine();
        } while (tambahPesanan == 'y' || tambahPesanan == 'Y');

        // Tampilkan pesanan dan hitung diskon
        pesanan.tampilkanPesanan();
        double diskon = pesanan.hitungDiskon();
        if (diskon > 0) {
            System.out.println("Anda mendapatkan diskon 10% sebesar
Rp " + diskon);
        }
    }
}
```



```

        // Hitung total setelah diskon
        double totalSetelahDiskon = pesanan.totalHarga - diskon;
        System.out.println("Total setelah diskon: Rp " +
totalSetelahDiskon);

        // Hitung dan tambahkan pajak 10%
        double pajak = totalSetelahDiskon * 0.10;
        System.out.println("Pajak 10%: Rp " + pajak);

        // Total akhir yang harus dibayar
        double totalAkhir = totalSetelahDiskon + pajak;
        System.out.println("Total yang harus dibayar: Rp " +
totalAkhir);
    }
}

```

Screenshoot Output:

Output - P2_kondisional (run)

```

run:
Masukkan nama item: Es Krim
Masukkan harga item: 10000
Masukkan kategori (1: Pembuka, 2: Utama, 3: Minuman): 4
Kategori tidak valid! Harap masukkan angka 1, 2, atau 3.
Masukkan kategori (1: Pembuka, 2: Utama, 3: Minuman): |

```

Output - P2_kondisional (run)

```

run:
Masukkan nama item: Mie Ayam
Masukkan harga item: 20000
Masukkan kategori (1: Pembuka, 2: Utama, 3: Minuman): 2
Apakah ingin menambah pesanan lagi? (y/n): y
Masukkan nama item: Es Teh
Masukkan harga item: 5000
Masukkan kategori (1: Pembuka, 2: Utama, 3: Minuman): 3
Apakah ingin menambah pesanan lagi? (y/n): y
Masukkan nama item: Dimsum
Masukkan harga item: 12000
Masukkan kategori (1: Pembuka, 2: Utama, 3: Minuman): 1
Apakah ingin menambah pesanan lagi? (y/n): n
Daftar Pesanan:
Mie Ayam - Kategori: Utama - Harga: Rp 20000.0
Es Teh - Kategori: Minuman - Harga: Rp 5000.0
Dimsum - Kategori: Pembuka - Harga: Rp 12000.0
Total Harga: Rp 37000.0
Total setelah diskon: Rp 37000.0
Pajak 10%: Rp 3700.0
Total yang harus dibayar: Rp 40700.0
BUILD SUCCESSFUL (total time: 35 seconds)

```

Penjelasan:

Pada kelas LatihanP4.java, program ini merupakan sistem manajemen pesanan yang meminta pengguna untuk memasukkan pesanan makanan, menghitung total harga,

diskon, dan pajak. Program dimulai dengan meminta pengguna memasukkan nama item, harga, dan kategori (1: Pembuka, 2: Utama, 3: Minuman), disertai validasi kategori agar hanya menerima input 1, 2, atau 3. Pengguna bisa terus menambah pesanan hingga memilih untuk berhenti. Setelah itu, program menampilkan daftar pesanan beserta total harga, menghitung diskon 10% jika total melebihi Rp 200.000, dan menambahkan pajak 10% pada total harga setelah diskon. Akhirnya, program menampilkan total yang harus dibayar, mencakup diskon dan pajak.