

LAPORAN PRAKTIKUM
PEMROGRAMAN BERORIENTASI OBJEK
MODUL 2
PEMILIHAN DAN PERULANGAN



Oleh :

Chris Hotasi Rajagukguk

2311103126

S1SI-07-C

PROGRAM STUDI SISTEM INFORMASI
FAKULTAS REKAYASA INDUSTRI
UNIVERSITAS TELKOM PURWOKERTO
2024

A. Studi Kasus

Manajemen Pesanan di Restoran

B. Deskripsi

Sebuah restoran ingin mengembangkan sistem manajemen pesanan berbasis objek yang dapat mengelola berbagai pesanan dari pelanggan. Sistem ini harus mampu mencatat pesanan, menghitung total harga, mengelola berbagai kategori pesanan (makanan dan minuman), dan memberikan diskon atau biaya tambahan sesuai kondisi. Dalam studi kasus ini, mahasiswa akan mempelajari penggunaan Statement IF, Statement Case, dan Statement While dalam paradigma Pemrograman Berorientasi Objek (PBO).

C. Guided

Buat project baru dengan LatihanP4Nama. Nama diganti dengan nama kalian, contoh : LatihanP4SenaWijayanto Buatlah program untuk restoran yang dapat:

1. Mencatat pesanan pelanggan dari berbagai kategori (makanan pembuka, makanan utama, minuman).
2. Menggunakan Statement CASE untuk memilih kategori pesanan.
3. Menghitung total harga berdasarkan jumlah item dan kategori.
4. Menggunakan Statement IF untuk memberikan diskon 10% jika total harga pesanan melebihi Rp 200.000.
5. Menggunakan Statement WHILE untuk menerima beberapa pesanan hingga pelanggan selesai.

Kode

Menu.java

```
/*  
  
 * Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this  
license  
  
 * Click nbfs://nbhost/SystemFileSystem/Templates/Classes/Class.java to edit this template  
  
*/  
  
package latihanp4chrishotasirajagukguk;  
  
/**  
  
 *  
  
 * @author Chris Hotasi Rajagukguk  
  
*/
```

```

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";
            case 2 -> jenis = "Utama";
            case 3 -> jenis = "Minuman";
        }

        System.out.println(namaItem + " - Kategori: " + jenis + " - Harga: Rp " + harga);
    }
}

```

Penjelasan : Kodingan di atas mendefinisikan kelas `Menu` dalam Java untuk merepresentasikan item menu restoran. Kelas ini memiliki tiga atribut: `namaItem` (String) untuk nama item, `harga` (double) untuk harga, dan `kategori` (int) yang mengklasifikasikan item sebagai Pembuka, Utama, atau Minuman. Terdapat konstruktor yang memungkinkan inisialisasi objek dengan ketiga atribut tersebut. Metode `tampilkanInfo()` berfungsi untuk menampilkan nama, kategori, dan harga item dengan menggunakan struktur kontrol switch untuk menentukan kategori. Kelas ini menyediakan cara terstruktur untuk mendefinisikan dan menampilkan informasi menu makanan.

Pesanan.java

```
/*  
 * Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this  
license  
 * Click nbfs://nbhost/SystemFileSystem/Templates/Classes/Class.java to edit this template  
 */  
  
package latihanp4chrishotasirajagukguk;  
  
/**  
 *  
 * @author Chris Hotasi Rajagukguk  
 */  
  
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 : Kodingan di atas mendefinisikan kelas `Pesanan` dalam Java untuk mengelola daftar pesanan di restoran. Kelas ini memiliki tiga atribut: `daftarPesanan` (array objek `Menu`), `jumlahPesanan` (untuk melacak jumlah item), dan `totalHarga` (untuk menghitung total biaya). Konstruktor `Pesanan(int kapasitas)` memungkinkan penentuan kapasitas maksimum pesanan. Metode `tambahPesanan(Menu item)` menambahkan item ke daftar dan memperbarui total harga serta jumlah pesanan. Metode `tampilkanPesanan()` menampilkan semua item beserta total harga, sedangkan `hitungDiskon()` menghitung diskon 10% jika total melebihi Rp 200.000. Kelas ini menyediakan fungsionalitas untuk mengelola dan menampilkan pesanan serta menghitung total biaya dengan kemungkinan diskon.

LatihanP4ChrisHotasiRajagukguk.java

/*

* Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this license

* Click nbfs://nbhost/SystemFileSystem/Templates/Classes/Main.java to edit this template

```

*/

package latihanp4chrishotasirajagukguk;

import java.util.Scanner;

/**
 *
 * @author Chris Hotasi Rajagukguk
 */

public class LatihanP4ChrisHotasiRajagukguk {

    /**
     * @param args the command line arguments
     */

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

        Pesanan pesanan = new Pesanan(10); // Maksimal 10 pesanan
        char tambahPesanan;

        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');

    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));

}

}

```

Pejelasan : Kodingan di atas adalah implementasi program yang mengelola pesanan makanan di restoran, menggunakan kelas `Pesanan` dan `Menu`. Dalam metode `main`, program menggunakan objek `Scanner` untuk menerima input dari pengguna dan membuat objek `Pesanan` dengan kapasitas maksimum 10 item. Melalui loop `do-while`, pengguna diminta untuk memasukkan nama item, harga, dan kategori untuk membuat objek `Menu`, yang kemudian ditambahkan ke daftar pesanan dengan metode `tambahPesanan()`. Setelah itu, program menampilkan semua item pesanan dan total harga. Program juga memeriksa apakah total melebihi Rp 200.000 untuk menghitung diskon 10%, yang jika ada, ditampilkan kepada pengguna. Akhirnya, total yang harus dibayar setelah diskon dihitung dan ditampilkan. Program ini memberikan antarmuka interaktif untuk mengelola dan menghitung biaya pesanan.

Output

Output - LatihanP4ChrisHotasiRajagukguk (run)

```
run:
Masukkan nama item: Sandwich
Masukkan harga item: 25000
Masukkan kategori (1: Pembuka, 2: Utama, 3: Minuman): 1
Apakah ingin menambah pesanan lagi? (y/n):y
Masukkan nama item: Nasi Ayam Bakar
Masukkan harga item: 40000
Masukkan kategori (1: Pembuka, 2: Utama, 3: Minuman): 2
Apakah ingin menambah pesanan lagi? (y/n):y
Masukkan nama item: Teh Manis
Masukkan harga item: 2000
Masukkan kategori (1: Pembuka, 2: Utama, 3: Minuman): 3
Apakah ingin menambah pesanan lagi? (y/n):n
Daftar Pesanan:
Sandwich - Kategori: Pembuka - Harga: Rp 25000.0
Nasi Ayam Bakar - Kategori: Utama - Harga: Rp 40000.0
Teh Manis - Kategori: Minuman - Harga: Rp 2000.0
Total Harga: Rp 67000.0
Total yang harus dibayar: Rp 67000.0
BUILD SUCCESSFUL (total time: 41 seconds)
```

D. 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

Kode

LatihanP4ChrisHotasiRajagukguk.java

/*

* Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this license

* Click nbfs://nbhost/SystemFileSystem/Templates/Classes/Main.java to edit this template

*/

package latihanp4chrishotasierajagukguk;

import java.util.Scanner;


```

/**
 *
 * @author Chris Hotasi Rajagukguk
 */
public class LatihanP4ChrisHotasiRajagukguk {

    /**
     * @param args the command line arguments
     */
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        Pesanan pesanan = new Pesanan(10);
        char tambahPesanan;

        do {
            System.out.print("Masukkan nama item: ");
            String namaItem = scanner.nextLine();
            System.out.print("Masukkan harga item: ");
            double hargaItem = scanner.nextDouble();
            int kategoriItem;

            while (true) {
                System.out.print("Masukkan kategori (1: Pembuka, 2: Utama, 3: Minuman): ");
                kategoriItem = scanner.nextInt();
                if (kategoriItem == 1 || kategoriItem == 2 || kategoriItem == 3) {
                    break; // Kategori valid, keluar dari loop
                } else {

```

```

        System.out.println("Input tidak valid. Silakan masukkan 1, 2, atau 3.");
    }
}

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();
double totalSetelahDiskon = pesanan.totalHarga - diskon;
// Hitung pajak
double pajak = totalSetelahDiskon * 0.10;
if (diskon > 0) {
    System.out.println("Anda mendapatkan diskon 10% sebesar Rp " + diskon);
}
System.out.println("Pajak 10% sebesar: Rp " + pajak);
System.out.println("Total yang harus dibayar: Rp " + (totalSetelahDiskon + pajak));
}
}

```

Penjelasan : Kodingan di atas adalah program untuk mengelola pesanan makanan di restoran menggunakan kelas `Pesanan` dan `Menu`. Dalam metode `main`, program menggunakan objek `Scanner` untuk menerima input dari pengguna dan membuat objek `Pesanan` dengan kapasitas maksimal 10 item. Pengguna diminta untuk memasukkan nama item, harga, dan kategori, dengan loop yang memastikan kategori yang dimasukkan valid (1, 2, atau 3). Setelah informasi lengkap,

item ditambahkan ke daftar pesanan melalui metode `tambahPesanan()`. Program kemudian menanyakan apakah pengguna ingin menambah pesanan, dan proses ini berlanjut hingga pengguna memilih tidak. Setelah semua pesanan dimasukkan, program menampilkan daftar pesanan dan menghitung diskon 10% jika total harga melebihi batas tertentu. Pajak 10% juga dihitung dari total setelah diskon, dan semua informasi tentang diskon dan total yang harus dibayar ditampilkan, memberikan gambaran jelas tentang biaya kepada pengguna.

Output

Output - LatihanP4ChrisHotasiRajagukguk (run)

```
run:
Masukkan nama item: Sandwich
Masukkan harga item: 15000
Masukkan kategori (1: Pembuka, 2: Utama, 3: Minuman): 1
Apakah ingin menambah pesanan lagi? (y/n): y
Masukkan nama item: Nasi Ayam Panggang
Masukkan harga item: 40000
Masukkan kategori (1: Pembuka, 2: Utama, 3: Minuman): 2
Apakah ingin menambah pesanan lagi? (y/n): y
Masukkan nama item: Lemon Tea
Masukkan harga item: 15000
Masukkan kategori (1: Pembuka, 2: Utama, 3: Minuman): 3
Apakah ingin menambah pesanan lagi? (y/n): n
Daftar Pesanan:
Sandwich - Kategori: Pembuka - Harga: Rp 15000.0
Nasi Ayam Panggang - Kategori: Utama - Harga: Rp 40000.0
Lemon Tea - Kategori: Minuman - Harga: Rp 15000.0
Total Harga: Rp 70000.0
Pajak 10% sebesar: Rp 7000.0
Total yang harus dibayar: Rp 77000.0
BUILD SUCCESSFUL (total time: 43 seconds)
```