

**LAPORAN PRAKTIKUM
PEMROGRAMAN BERORIENTASI OBJEK
MODUL 2
PEMILIHAN DAN PERULANGAN**



Oleh :

**Rajendra Ikmal Veorozzan
2311103122
S1SI07C**

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

➤ Guided

Buat project baru dengan LatihanP4Nama. Nama diganti dengan nama kalian, 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.

⇒ **Jawaban:**

Source Code

- Menu.java

```
package latihanp4rajendraikmalveorozzan;

/**
 *
 * @author Rajendra Ikmal Veorozzan
 * 2311103122
 * S1SI07C
 */
public class Menu {
    String namaItem;
    double harga;
    int kategori;
    public Menu(String namaItem, double harga, int kategori) {
        this.namaItem = namaItem;
        this.harga = harga;
        this.kategori = kategori;
    }
}
```

```
}

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);
}
}
```

- **Pesanan.Java**

```
package latihanp4rajendraikmalveorozzan;

/**
 *
 * @author Rajendra Ikmal Veorozzan
 * 2311103122
 * S1SI07C
 */
public class Pesanan {

    Menu[] daftarPesanan;
    int jumlahPesanan;
    double totalHarga;

    public Pesanan(int kapasitas) {
        daftarPesanan = new Menu[kapasitas];
        jumlahPesanan = 0;
        totalHarga = 0;
    }

    public void tambahPesanan(Menu item) {
        daftarPesanan[jumlahPesanan] = item;
        totalHarga += item.harga;
        jumlahPesanan++;
    }

    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);
    }

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

- **LatihanP4RajendraIkmalVeorozzan**

```
package latihanp4rajendraikmalveorozzan;

import java.util.Scanner;

/**
 *
 * @author Rajendra Ikmal Veorozzan
 * 2311103122
 * S1SI07C
 */
public class LatihanP4RajendraIkmalVeorozzan {

    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();
            System.out.print("Masukkan kategori (1: Pembuka, 2: Utama, 3: Minuman) : ");
            int kategoriItem = scanner.nextInt();
            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');

        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));
    }
}
```

- **Screenshot Hasil Program:**

```
Output - LatihanP4RajendralkmalVeorozzan (run) X
run:
Masukkan nama item : Kepiting rajungan
Masukkan harga item : 50000
Masukkan kategori (1: Pembuka, 2: Utama, 3: Minuman) : 2
Apakah ingin menambah pesanan lagi (Y/N) : N
Daftar Pesanan:
Kepiting rajungan - Kategori: Utama - Harga: Rp.50000.0
Total Harga: Rp.50000.0
Total yang harus dibayar: Rp.50000.0
BUILD SUCCESSFUL (total time: 1 minute 33 seconds)
```

```
Output - LatihanP4RajendralkmalVeorozzan (run) X
run:
Masukkan nama item : Kepiting Alaska
Masukkan harga item : 5000000
Masukkan kategori (1: Pembuka, 2: Utama, 3: Minuman) : 2
Apakah ingin menambah pesanan lagi (Y/N) : N
Daftar Pesanan:
Kepiting Alaska - Kategori: Utama - Harga: Rp.5000000.0
Total Harga: Rp.5000000.0
Anda mendapatkan diskon 10% sebesar Rp.500000.0
Total yang harus dibayar: Rp.4500000.0
BUILD SUCCESSFUL (total time: 43 seconds)
```

- **Penjelasan Hasil Program:**

Program ini merupakan sebuah aplikasi restoran yang dirancang untuk menerima input dari pengguna terkait pesanan makanan atau minuman, beserta harganya. Program ini dibangun menggunakan tiga kelas utama: Menu, Pesanan, dan LatihanP4rajendraIkmalVeorozzan sebagai file utama. Di dalam kelas Pesanan, terdapat array daftarPesanan yang menyimpan daftar item dari kelas Menu, variabel jumlahPesanan yang mencatat jumlah total item yang dipesan, serta variabel totalHarga yang berfungsi untuk menghitung total biaya dari semua pesanan. Metode tambahPesanan digunakan untuk menambahkan item ke daftar pesanan dan memperbarui total harga pesanan. Kemudian, ada metode tampilkanPesanan yang berfungsi menampilkan seluruh daftar pesanan, lengkap dengan informasi harganya dan kategori item tersebut, dengan menggunakan perulangan. Selain itu, terdapat fungsi hitungDiskon, yang akan memberikan diskon 10% jika total pesanan melebihi Rp200.000.

Di dalam kelas Pesanan, terdapat array daftarPesanan yang menyimpan daftar item dari kelas Menu, variabel jumlahPesanan yang mencatat jumlah total item yang dipesan, serta variabel totalHarga yang berfungsi untuk menghitung total biaya dari semua pesanan. Metode tambahPesanan digunakan untuk menambahkan item ke daftar pesanan dan memperbarui total harga pesanan. Kemudian, ada metode tampilkanPesanan yang berfungsi menampilkan seluruh daftar pesanan, lengkap dengan informasi harganya dan kategori item tersebut, dengan menggunakan perulangan. Selain itu, terdapat fungsi hitungDiskon, yang akan memberikan diskon 10% jika total pesanan melebihi Rp200.000.

Pada file utama LatihanP4rajendraIkmalVeorozzan, program menerima input dari pengguna mengenai item yang akan dipesan. Setelah semua pesanan dimasukkan, program akan menampilkan informasi lengkap mengenai pesanan tersebut dengan memanggil fungsi tampilkanPesanan. Jika total harga memenuhi syarat diskon (melebihi Rp200.000), program akan menghitung dan memberikan diskon 10%, serta menampilkan total harga yang harus dibayar setelah diskon diterapkan.

Pada output pertama, pengguna memasukkan pesanan "Kepiting Rajungan" dengan harga Rp50.000 dalam kategori Utama. Karena total harga tidak mencapai Rp200.000, maka diskon tidak diberikan dan total yang harus dibayar tetap Rp50.000. Di output kedua, pengguna memasukkan pesanan "Kepiting Alaska" dengan harga Rp5.000.000 dalam kategori Utama, sehingga diskon 10% sebesar Rp500.000 diberikan, dan total yang harus dibayar setelah diskon adalah Rp4.500.000.

➤ 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.

⇒ **Jawaban:**

- Source Code:

Unguided01P4Veo.Java

```
package unguided01p4veo;
import java.util.Scanner;
/**
 * @author Rajendra Veo ikmal veorozzan
 * 2311103122
 * SI07C
 */
public class Unguided01P4Veo {

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

        do {
            System.out.print("Masukkan nama Item: ");
            String namaItem = scanner.nextLine();
            System.out.print("Masukkan harga Item: ");
            double hargaItem = validasiInputHarga(scanner);

            int kategoriItem = validasiInputKategori(scanner);

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

            tambahPesanan = tanyaTambahPesanan(scanner);

        } while (tambahPesanan);

        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));
}

public static double validasiInputHarga(Scanner scanner) {
    while (!scanner.hasNextDouble()) {
        System.out.println("Harga harus berupa angka. Silakan coba lagi.");
        scanner.next();
    }
    return scanner.nextDouble();
}

public static int validasiInputKategori(Scanner scanner) {
    int kategoriItem;
    do {
        System.out.print("Masukkan kategori (1: Pembuka, 2: Utama, 3: Minuman): ");
        while (!scanner.hasNextInt()) {
            System.out.println("Input harus berupa angka. Masukkan angka 1, 2, atau 3.");
            scanner.next();
        }
        kategoriItem = scanner.nextInt();
        if (kategoriItem < 1 || kategoriItem > 3) {
            System.out.println("Kategori tidak valid. Masukkan angka 1, 2, atau 3.");
        }
    } while (kategoriItem < 1 || kategoriItem > 3);
    scanner.nextLine();
    return kategoriItem;
}

public static boolean tanyaTambahPesanan(Scanner scanner) {
    char tambahPesanan;
    do {
        System.out.print("Apakah ingin menambah pesanan lagi (Y/N): ");
        tambahPesanan = scanner.next().charAt(0);
        scanner.nextLine();
        if (tambahPesanan != 'y' && tambahPesanan != 'Y' && tambahPesanan != 'n' &&
tambahPesanan != 'N') {
            System.out.println("Input tidak valid. Harap masukkan Y atau N.");
        }
    } while (tambahPesanan != 'y' && tambahPesanan != 'Y' && tambahPesanan != 'n' &&
tambahPesanan != 'N');

    return (tambahPesanan == 'y' || tambahPesanan == 'Y');
}
}
```

Pesanan.Java

```
package unguided01p4veo;

/**
 *
 * @author Rajendra Veo ikmal veorozzan
 * 2311103122
 * SI07C
 */
public class Pesanan {
    Menu[] daftarPesanan;
    int jumlahPesanan;
    double totalHarga;

    public Pesanan(int kapasitas) {
        daftarPesanan = new Menu[kapasitas];
        jumlahPesanan = 0;
        totalHarga = 0;
    }

    public void tambahPesanan(Menu item) {
        daftarPesanan[jumlahPesanan] = item;
        totalHarga += item.harga;
        jumlahPesanan++;
    }

    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);
    }

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

Menu.Java

```
package unguided01p4veo;
/**
 * @author Rajendra Veo ikmal veorozzan
 * 2311103122
 * SI07C
 */
public class Menu {
    String namaItem;
    double harga;
    int kategori;

    public Menu(String namaItem, double harga, int kategori) {
        this.namaItem = namaItem;
        this.harga = harga;
        this.kategori = kategori;
    }

    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);
    }
}
```

- **Screenshoot Hasil Program :**

```

Output - Unguided01P4Veo (run) ×
run:
Masukkan nama Item: Bakpia patok
Masukkan harga Item: 15000
Masukkan kategori (1: Pembuka, 2: Utama, 3: Minuman): 4
Kategori tidak valid. Masukkan angka 1, 2, atau 3.
Masukkan kategori (1: Pembuka, 2: Utama, 3: Minuman): 1
Apakah ingin menambah pesanan lagi (Y/N): Y
Masukkan nama Item: Baso
Masukkan harga Item: 20000
Masukkan kategori (1: Pembuka, 2: Utama, 3: Minuman): 2
Apakah ingin menambah pesanan lagi (Y/N): N
Daftar Pesanan:
Bakpia patok - Kategori: Pembuka - Harga: Rp.15000.0
Baso - Kategori: Utama - Harga: Rp.20000.0
Total Harga: Rp.35000.0
Total yang harus dibayar: Rp.35000.0
BUILD SUCCESSFUL (total time: 1 minute 13 seconds)

```

- **Penjelasan:**

Program ini menggunakan beberapa metode untuk memvalidasi input dari pengguna dan memastikan bahwa data yang dimasukkan benar. Berikut adalah penjelasan dari alur kerja kode dan hasil output yang dihasilkan. Program meminta pengguna memasukkan nama item dan harga, kemudian meminta input kategori item. Kategori yang valid adalah 1 (Pembuka), 2 (Utama), dan 3 (Minuman). Jika pengguna memasukkan angka di luar rentang tersebut, seperti dalam contoh output yang memasukkan kategori "4" untuk item "Bakpia Patok," program menampilkan pesan kesalahan "Kategori tidak valid. Masukkan angka 1, 2, atau 3." dan meminta input ulang. Setelah pengguna berhasil memasukkan kategori yang benar (kategori 1 untuk "Pembuka"), program menanyakan apakah pengguna ingin menambah pesanan lain. Pengguna memilih "Y" untuk menambah item "Baso" dengan harga 20000 dan kategori "Utama."

Setelah semua pesanan dimasukkan, program menampilkan daftar pesanan, total harga tanpa diskon, dan total harga akhir yang harus dibayar. Pada output, daftar pesanan menunjukkan dua item yang dipesan: "Bakpia patok" dengan harga 15000 sebagai pembuka dan "Baso" dengan harga 20000 sebagai makanan utama. Total harga dari kedua item adalah 35000, dan tidak ada diskon yang diterapkan.

2. Tambahkan perhitungan pajak 10%. Setelah diskon diberikan, tambahkan fitur untuk menghitung pajak sebesar 10% dari total harga yang sudah dikurangi diskon.

⇒ **Jawaban:**

- **Sourche Code:**

Unguided02p4Veo.Java

```
package unguided02p4veo;

import java.util.Scanner;

/**
 *
 * @author Rajendra Ikmal Veorozzan
 * 2311103122
 * S1SI07C
 */
public class Unguided02P4Veo {

    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        Pesanan pesanan = new Pesanan(10);
        char tambahPesanan;
        System.out.print("=== Struk Pesanan === ");
        scanner.nextLine();
        System.out.print("=== Rajendra === ");
        scanner.nextLine();

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

            int kategoriItem = 0;
            boolean kategori = false;
            while (kategori == false) {
                System.out.print("Masukkan kategori (1: Pembuka, 2: Utama, 3: Minuman) : ");
                kategoriItem = scanner.nextInt();

                if (kategoriItem <= 3) {
                    kategori = true;
                } else {
                    System.out.println("Kategori salah input");
                }
            }
        }
    }
}
```

```
    }  
    }  
    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();  
    System.out.println("");  
    } while (tambahPesanan == 'y' || tambahPesanan == 'Y');  
  
    System.out.println("");  
    pesanan.tampilkanPesanan();  
    double diskon = pesanan.hitungDiskon();  
    if (diskon > 0) {  
        System.out.println("Anda mendapatkan diskon 10% sebesar Rp." + diskon);  
    }  
  
    double hargaDiskon = pesanan.totalHarga - diskon;  
    double pajak = pesanan.hitungPajak(hargaDiskon);  
    System.out.println("Pajak 10% Rp." + pajak);  
    System.out.println("");  
    System.out.println("Total yang harus dibayar (Sudah termasuk pajak): Rp." + (hargaDiskon +  
    pajak));  
    }  
}
```

Pesanan.Java

```
package unguided02p4veo;
/**
 * @author Rajendra Ikmal Veorozzan
 * 2311103122
 * S1SI07C
 */
public class Pesanan {
    Menu[] daftarPesanan;
    int jumlahPesanan;
    double totalHarga;

    public Pesanan(int kapasitas) {
        daftarPesanan = new Menu[kapasitas];
        jumlahPesanan = 0;
        totalHarga = 0;
    }

    public void tambahPesanan(Menu item) {
        daftarPesanan[jumlahPesanan] = item;
        totalHarga += item.harga;
        jumlahPesanan++;
    }

    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);
    }

    public double hitungDiskon() {
        if (totalHarga > 200000) {
            return totalHarga * 0.10;
        } else {
            return 0;
        }
    }

    public double hitungPajak(double hargaPajak) {
        return hargaPajak * 0.10;
    }
}
```

Menu.java

```
package unguided02p4veo;
/**
 * @author Rajendra Ikmal Veorozzan
 * 2311103122
 * S1SI07C
 */
public class Menu {
    String namaItem;
    double harga;
    int kategori;

    public Menu(String namaItem, double harga, int kategori) {
        this.namaItem = namaItem;
        this.harga = harga;
        this.kategori = kategori;
    }

    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);
    }
}
```


- **Screenshoot Hasil**

```
run:
=== Struk Pesanan ===
=== Rajendra ===
Masukkan nama item : salad
Masukkan harga item : 20000
Masukkan kategori (1: Pembuka, 2: Utama, 3: Minuman) : 1
Apakah ingin menambah pesanan lagi (Y/N) : y

Masukkan nama item : nasi goreng solaria
Masukkan harga item : 45000
Masukkan kategori (1: Pembuka, 2: Utama, 3: Minuman) : 2
Apakah ingin menambah pesanan lagi (Y/N) : n

Daftar Pesanan:
salad - Kategori: Pembuka - Harga: Rp.20000.0
nasi goreng solaria - Kategori: Utama - Harga: Rp.45000.0
Total Harga: Rp.65000.0
Pajak 10% Rp.6500.0

Total yang harus dibayar (Sudah termasuk pajak): Rp.71500.0
BUILD SUCCESSFUL (total time: 1 minute 9 seconds)
```

- **Penjelasan :**

Program ini bertujuan untuk mencatat dan menghitung total pesanan makanan, termasuk kategori, harga, serta pajak. Program dimulai dengan meminta pengguna memasukkan nama item, harga, dan kategori (dengan pilihan kategori 1 untuk pembuka, 2 untuk utama, dan 3 untuk minuman). Setelah setiap item dimasukkan, program menanyakan apakah pengguna ingin menambahkan item lagi. Jika ya, pengguna kembali memasukkan pesanan baru, jika tidak, program melanjutkan ke perhitungan dan menampilkan hasil. Program ini menyimpan setiap pesanan ke dalam daftar yang dikelola oleh objek Pesanan. Setelah selesai mencatat semua pesanan, program menampilkan daftar item, kategori, dan harga masing-masing. Selain itu, program menghitung total harga dari semua pesanan. Pada output, terlihat dua item yang dipesan, yaitu salad (kategori pembuka) dengan harga Rp. 20,000 dan nasi goreng Solaria (kategori utama) dengan harga Rp. 45,000. Total harga kedua pesanan adalah Rp. 65,000. Selanjutnya, program menghitung pajak sebesar 10% dari total harga, yakni Rp. 6,500, sehingga total yang harus dibayar adalah Rp. 71,500. Program berjalan dengan sukses dan menampilkan hasil akhir yang sesuai