

MODULE 4

INTERFACE

Interface adalah suatu kontrak atau blueprint dalam pemrograman berorientasi objek yang menyatakan metode-metode yang harus diimplementasikan oleh kelas-kelas yang menggunakannya. Dalam interface, hanya deklarasi metodenya yang disediakan tanpa adanya implementasi konkret. Kelas yang mengimplementasikan suatu interface diharuskan untuk memberikan implementasi dari semua metode yang dideklarasikan dalam interface tersebut.



Anda merupakan seorang pengembang perangkat lunak untuk sistem manajemen restoran. Dalam pengembangan perangkat lunak ini, Anda ingin mengimplementasikan konsep interface pada beberapa komponen untuk mempermudah manajemen pesanan dan tampilan menu. Berikut adalah deskripsi singkat tentang program yang Anda kembangkan:

- Terdapat dua jenis menu, yaitu makanan dan minuman, yang diwakili oleh kelas **Makanan** dan **Minuman** yang merupakan turunan dari kelas abstrak **Menu** yang berisi objek nama dan harga.
- Setiap menu dapat dipesan, sehingga diberikan interface **Pesanan** yang berisi metode pesan().
- Selain dapat dipesan, menu juga dapat ditampilkan dengan interface **Tampilan** yang berisi metode tampilkan().
- Program memiliki kelas **PesananPelanggan** yang bertanggung jawab untuk mengelola pesanan dari pelanggan. Pesanan-pesanan ini disimpan dalam sebuah array list.
- Ada metode tambahPesanan() di kelas **PesananPelanggan** yang memungkinkan penambahan pesanan baru.

- Terdapat metode `tampilkanPesanan()` untuk menampilkan pesanan-pesanan yang telah diterima dari pelanggan.
- Program juga memiliki metode `hitungTotalPembayaran()` di kelas **PesananPelanggan** untuk menghitung total pembayaran dari seluruh pesanan.
- Pada kelas **Restoran**, terdapat contoh penggunaan program dengan membuat beberapa menu, membuat pesanan pelanggan, menampilkan pesanan, menghitung total pembayaran, dan mencetak struk pembayaran.

Question

Implementasikan program di atas dalam bahasa pemrograman Java dan tunjukkan hasil eksekusi programnya. Jelaskan juga bagaimana konsep interface digunakan dalam program tersebut.

Contoh Main Class (Restoran):

```
package week4_interface;

public class Restoran {

    public static void main(String[] args) {
        // Membuat menu
        Makanan makanan1 = new Makanan("Nasi Goreng", 25000);
        Makanan makanan2 = new Makanan("Ayam Bakar", 30000);
        Minuman minuman1 = new Minuman("Es Teh", 5000);
        Minuman minuman2 = new Minuman("Es Jeruk", 6000);

        // Membuat pesanan pelanggan
        PesananPelanggan pesananPelanggan = new PesananPelanggan();
        pesananPelanggan.tambahPesanan(makanan1);
        pesananPelanggan.tambahPesanan(minuman2);

        // Menampilkan pesanan pelanggan
        pesananPelanggan.tampilkanPesanan();

        // Menghitung total pembayaran
        double totalPembayaran = pesananPelanggan.hitungTotalPembayaran();

        // Mencetak struk pembayaran
        System.out.println("\nStruk Pembayaran:");
        pesananPelanggan.tampilkanPesanan();
        System.out.println("Total Pembayaran: " + totalPembayaran);
    }
}
```

Contoh Output:

```
Pesanan Pelanggan:  
Pesanan makanan 'Nasi Goreng' telah diterima.  
Pesanan minuman 'Es Jeruk' telah diterima.  
  
Struk Pembayaran:  
Pesanan Pelanggan:  
Pesanan makanan 'Nasi Goreng' telah diterima.  
Pesanan minuman 'Es Jeruk' telah diterima.  
Total Pembayaran: 31000.0
```