

# Encapsulation

## Pokok Pembahasan

- Encapsulation (Pembungkusan)
- Access Modifier

## Tujuan Belajar

- Agar mengetahui konsep Enkapsulasi secara jelas
- Mengetahui penggunaan dari Access Modifier

Encapsulation adalah mekanisme pemrograman yang membungkus kode dan data yang dimanipulasi dan menjaganya supaya terhindar dari interferensi dan penggunaan yang tidak perlu. Salah satu caranya dengan membentuk objek. konsep ini memiliki mekanisme untuk membungkus sebuah data/variable, data tersebut tidak bisa diakses oleh class lain, dengan menggunakan konsep Encapsulation, semua data harus kita set menjadi private, agar class lain tidak dapat mengaksesnya.

Jika diantara kalian ada yang belum paham apa itu private, kata kunci private adalah sebuah modifier untuk membatasi hak akses pada variabel, class dan method, jika data-data di set hak aksesnya menjadi private, maka data tersebut hanya bisa digunakan oleh class itu sendiri dan tidak bisa digunakan oleh class luar manapun.

Kita akan membuat sebuah program sederhana dengan menggunakan konsep encapsulation, pertama buatlah 2 buah class, misalnya kita akan memberi nama restoran dan makanan.

Data-data yang disimpan secara private akan diletakan pada class restoran, lalu data tersebut akan kita akses melalui method public Setter dan Getter pada method main di class makanan.

```
public class restoran {  
  
    //Variable (Private)  
    private String menu;  
    private double harga;  
    private boolean spesial;  
  
    //Method Setter (Public) dengan Parameter  
    public void setMenu(String menu) {  
        this.menu = menu;  
    }  
    public void setharga(double harga) {  
        this.harga = harga;  
    }  
    public void setSpesial(boolean spesial) {  
        this.spesial = spesial;  
    }  
    //Method Getter (Public)  
    public String getMenu() {  
        return menu;  
    }  
    public double getHarga() {  
        return harga;  
    }  
}
```

```
public boolean getSpesial() {  
    return spesial;  
}  
}
```

Data-data tersebut akan kita jalankan pada method main di class makanan, kita tidak bisa memanggil variabelnya secara langsung, tapi kita bisa mengakses variable/datanya melalui method-method tersebut, seperti berikut ini:

```
public class makanan {  
    public static void main(String[] args) {  
        //Membuat Instance/Objek dari Class restoran  
        restoran data = new restoran();  
  
        //Memasukan Data pada Variable  
        data.setMenu("Ayam Goreng");  
        data.setharga(17000.0);  
        data.setSpesial(true);  
  
        //Memanggil Method Get dari Class restoran dan Menampilkannya  
        System.out.println("Menu Makanan: " + data.getMenu());  
        System.out.println("Harga Makanan: Rp." + data.getHarga());  
        System.out.println("Menu Spesial: " + data.getSpesial());  
    }  
}
```

Output :

```
/home/code-maniac/.sdkman/candidates/java/8.0.292.j9-adpt/bin/java ...  
Menu Makanan: Ayam Goreng  
Harga Makanan: Rp.17000.0  
Menu Spesial: true  
  
Process finished with exit code 0  
|
```