



**Kampus
Merdeka**
INDONESIA JAYA

UIN

**PROGRAM STUDI
TEKNIK INFORMATIKA**

Mata Kuliah : OBJECT ORIENTED PROGRAMMING

Minggu ke 6

**FAKULTAS SAINS DAN TEKNOLOGI
UIN MAULANA MALIK IBRAHIM MALANG**



informatika.uin-malang.ac.id



[informatikauinmalang1841](https://www.youtube.com/channel/UCinformatikauinmalang1841)



ti@uin-malang.ac.id



08814381300

Tujuan perkuliahan

Mampu memahami dan mengimplementasikan konsep Encapsulation pada Object Oriented Programming

Apa itu Enkapsulasi

- **Pengemasan data dan perilaku** ke dalam satu entitas tunggal yang disebut kelas.
- **Mengatur, mengamankan, dan membatasi** akses ke data serta perilaku objek.

Manfaat Enkapsulasi

- Keamanan Data
- Pengontrolan Akses
- Isolasi Implementasi
- Pencegahan Konflik Nama
- Abstraksi
- Perbaikan Kode dan Pengembangan
- Kode yang Lebih Bersih





Komponen Penting Encapsulation

- Access Modifier
 - Method Setter & Getter
- 

Akses Identifier

- **Kata kunci atau perintah** yang digunakan dalam pemrograman berorientasi objek untuk mengontrol **tingkat akses**
- **Terdapat beberapa akses identifier** yang digunakan untuk menentukan siapa yang dapat mengakses atau memodifikasi elemen-elemen kelas.

Beberapa akses identifier

- Public (dapat diakses dari mana saja)
- Private (hanya dapat diakses dari dalam kelas tersebut)
- Protected (dapat diakses dari dalam kelas itu sendiri dan dari kelas-kelas turunannya)
- Default (hanya dapat diakses oleh kelas-kelas dalam paket yang sama)



Contoh access modifier

```
1 public class MyClass {
2     public int publicVariable; // Variabel publik
3     public void publicMethod() {
4         // Metode publik
5     }
6     private int privateVariable; // Variabel pribadi
7     private void privateMethod() {
8         // Metode private
9     }
10    protected int protectedVariable; // Variabel dilindungi
11    protected void protectedMethod() {
12        // Metode protected
13    }
14    int defaultVariable; // Variabel dengan akses default
15    void defaultMethod() {
16        // Metode default
17    }
18 }
```

Method Setter

- **Mengubah atau mengisi** nilai pada property
- Bersifat tidak mengembalikan nilai atau **void**, karena hanya mengisi nilai pada property

Contoh Method Setter

```
1  ✓ public class Person {  
2      // Variabel pribadi (private)  
3      private String name;  
4      private int age;  
5  
6      // Method setter untuk mengatur nilai nama  
7  ✓  public void setName(String name) {  
8      |      this.name = name;  
9      |  }  
10 }  
11
```

Method Getter

- **Mengambil isi atau nilai** dari property
- Bersifat mengembalikan nilai atau **Return**, karena mengambil nilai pada property

Contoh Method Getter

```
1 public class Person {  
2     // Variabel pribadi (private)  
3     private String name;  
4     private int age;  
5  
6     // Method getter untuk mengambil nilai nama  
7     public String getName() {  
8         return name;  
9     }  
10 }
```



Kampus
Merdeka
INDONESIA JAYA

THANK YOU
FOR YOUR ATTENTION

