

Modul 3

1. Capaian Pembelajaran Matakuliah (CPMK)

CPMK142:

Mampu menerapkan teknik pemrograman terstruktur, modular, dan berorientasi objek untuk menyelesaikan permasalahan komputasi sesuai kebutuhan aplikasi sistem komputer.

CPMK143:

Mampu menggunakan bahasa pemrograman, pustaka, framework, IDE, version control, debugging tools, dan tools pendukung lainnya secara tepat dalam proses pengembangan perangkat lunak.

Indikator Ketercapaian:

Mahasiswa mampu menerapkan encapsulation.

2. Dasar Teori

Enkapsulasi

Arti dari **enkapsulasi** adalah untuk memastikan bahwa data "sensitif" disembunyikan dari pengguna.

Untuk mencapai hal ini, Anda harus mendeklarasikan variabel/atribut kelas sebagai **private** (tidak dapat diakses dari luar kelas).

Dalam C++, aksesibilitas anggota kelas dikontrol menggunakan tiga jenis *access modifier*:

- **public:** Anggota kelas dapat diakses dari bagian mana pun di dalam program, termasuk dari luar kelas tersebut.
- **private:** Anggota kelas **hanya** dapat diakses dari dalam kelas itu sendiri. Bagian luar kelas (termasuk fungsi main()) tidak memiliki akses langsung.
- **protected:** Anggota kelas bersifat privat, namun tetap dapat diakses oleh kelas turunan (*derived class*).

Mengapa Enkapsulasi?

- Sebaiknya atribut kelas Anda dideklarasikan sebagai private (sebisanya mungkin). Enkapsulasi memastikan kontrol yang lebih baik atas data Anda, karena Anda (atau orang lain) dapat mengubah satu bagian kode tanpa memengaruhi bagian lainnya.
- Peningkatan keamanan data

Contoh Kehidupan Nyata

Bayangkan gaji seorang karyawan:

- Gaji bersifat rahasia - karyawan tidak dapat mengubahnya secara langsung.
- Hanya manajer mereka yang dapat memperbarui atau membagikannya jika dianggap perlu.

Enkapsulasi bekerja dengan cara yang sama. Data disembunyikan, dan hanya metode tepercaya yang dapat mengakses atau memodifikasinya.

3. Alat dan Bahan

Hardware :

1. Laptop/PC
2. LCD/proyektor
3. Jaringan internet

Software :

1. Sistem operasi Windows/Linux/macOS
2. Java Development Kit (JDK)
3. IDE pemrograman: NetBeans / IntelliJ IDEA / Visual Studio Code
4. LMS untuk materi dan aktivitas pembelajaran

4. Langkah Kerja / Prosedur Praktikum

Enkapsulasi Pribadi

Untuk mengakses atribut privat, gunakan metode publik "get" dan "set"

Contoh

```
#include <iostream>
using namespace std;

class Employee {
private:
    // Private attribute
    int salary;

public:
    // Setter
    void setSalary(int s) {
        salary = s;
    }
}
```

```

// Getter
int getSalary() {
    return salary;
}
};

int main() {
    Employee myObj;
    myObj.setSalary(50000);
    cout << myObj.getSalary();
    return 0;
}

```

Penjelasan

- `salary` bersifat **pribadi** - tidak dapat diakses secara langsung.
- `setSalary()` menetapkan nilai
- `getSalary()` mengembalikan nilai

Kita menggunakannya `myObj.setSalary(50000)` untuk menetapkan nilai, dan `myObj.getSalary()` untuk mencetaknya.

5. Lembar Kerja

Latihan 3

Buat sebuah program dengan contoh **Enkapsulasi** dalam kehidupan sehari-hari dan lakukan coding program seperti dalam modul 3.

HASIL PERCOBAAN:

Screenshot Hasil LATIHAN 3 dan Berikan Analisis.
(UPLOAD ke LMS → TUGAS PRAKTIKUM 3)

Tugas Mandiri 3

(Kerjakan Tugas Mandiri 3 Upload ke LMS → **TUGAS TEORI 3**)

1. Apa yang dimaksud dengan enkapsulasi (encapsulation) dalam Pemrograman Berorientasi Objek, dan apa keuntungan utamanya dalam pengembangan perangkat lunak skala besar?
2. Jelaskan tujuan utama mengapa sebuah bahasa pemrograman berorientasi objek menyediakan fitur *enkapsulasi*?
3. Salah satu manfaat utama enkapsulasi adalah **Validasi Data**. Buatlah sebuah contoh logika/pseudocode fungsi *Setter* untuk atribut umur di dalam class *Pengguna*.?
4. Mengapa enkapsulasi dikatakan dapat meningkatkan keamanan dan kemudahan dalam melakukan **perawatan kode (maintenance)** jangka panjang pada aplikasi yang besar? Jelaskan alasan logisnya
5. Jelaskan perbedaan mendasar antara penentu akses *private*, *protected*, and *public*. Mengapa atribut objek umumnya diatur sebagai *private* sementara metodenya diatur sebagai *public*?



Program Studi
SISTEM KOMPUTER