

LAPORAN PRAKTIKUM PEMROGRAMAN BERORIENTASI OBJEK
MODUL 3 ENCAPSULATION AND FILE



Nama : Mahran Radifan Zhafir

Nim : 105219001

PROGRAM STUDI ILMU KOMPUTER
FAKULTAS SAINS DAN ILMU KOMPUTER
UNIVERSITAS PERTAMINA

2021

1. Pendahuluan

Pada praktikum ini, praktikan diminta untuk dapat memahami bagaimana cara menerapkan enkapsulasi dan metode menggunakan file handling pada pemrograman berbasis objek menggunakan Bahasa Java.

Enkapsulasi adalah sebuah metode untuk mengatur hak akses pada class dengan alasan keamanan dan juga mempertahankan kevalidan data, karena tidak dapat diakses dari method yang tidak sesuai dan diizinkan.

File Handling adalah sebuah metode untuk menyimpan inputan dari user dan semua keluaran dapat disimpan pada sebuah memori eksternal sehingga dapat digunakan lagi walaupun memori sudah dihapus karena memory reset saat kita menjalankan program untuk iterasi selanjutnya.

Rumusan masalah dari praktikum kali ini adalah bagaimana cara enkapsulasi data pada file mahasiswa dan bagaimana cara melakukan write dan read pada file handling.

2. Pembahasan

```
import java.util.Scanner;  
import java.io.*;
```

Gambar 2.1. Penggunaan Library

2 library ini digunakan untuk membantu kita dalam menerima inputan dari user dan juga menghandle file. jika tidak menggunakan 2 library ini, source code tidak bisa menerima inputan dari user dan juga tidak bisa menghandle file yang kita inginkan

```
mahasiswa mah1 = new mahasiswa();  
mahasiswa mah2 = new mahasiswa();
```

Gambar 2.2 Inisiasi objek

```
String tempStr;  
int tempInt;
```

Gambar 2.3 Variabel menampung sementara inputan user

```
System.out.println("=====");
System.out.print("Masukkan Nama : ");
tempStr = input.next();
mah1.setNama(tempStr);
System.out.print("Masukkan NIM : ");
tempStr = input.next();
mah1.setNim(tempStr);
System.out.print("Masukkan tinggi : ");
tempInt = input.nextInt();
mah1.setTinggi(tempInt);
System.out.print("Masukkan Berat : ");
tempInt = input.nextInt();
mah1.setBerat(tempInt);
mah1.setBmi();
```

Gambar 2.4 Penerimaan inputan untuk data bmi

pada saat menerima inputan dari user, saya memerlukan sebuah variabel penampung untuk menampung inputan dari user dan menggunakan metode setter untuk memberikan data kepada ada file mahasiswa yang variabelnya bertipe data privat yang di mana pengaksesan tipe data private harus dari method kelas tersebut. nanti, untuk mendapatkan nilai yang sudah kita simpan di dalam file mahasiswa, kita akan menggunakan metode getter untuk mengirim data dari file mahasiswa ke file main.java

```

PrintWriter output;
try {
    output = new PrintWriter("bmi.txt");
    output.println("Mahasiswa 1 :");
    output.println("Nama : " + mah1.getNama());
    output.println("NIM : " + mah1.getNim());
    output.println("tinggi : " + mah1.getTinggi());
    output.println("berat : " + mah1.getBerat());
    output.println("bmi : " + mah1.getBmi());
    output.println("Mahasiswa 2 :");
    output.println("Nama : " + mah2.getNama());
    output.println("NIM : " + mah2.getNim());
    output.println("tinggi : " + mah2.getTinggi());
    output.println("berat : " + mah2.getBerat());
    PrintWriter output - App.main(String[])
    output.close();
} catch (FileNotFoundException e) {
    // TODO Auto-generated catch block
    e.printStackTrace();
}

```

Gambar 2.5 Pengisian File bmi.txt

file handling adalah salah satu metode untuk membuat dan mengedit file melalui pemrograman java titik data didapatkan melalui metode get yang kita dapatkan dari file mahasiswa titik pertama sekali kita harus membuka terlebih dahulu, seperti pada pemograman dasar CPP yang saya dapatkan di semester 1. dan juga tidak lupa harus menutup program file sehingga file dapat ditulis kedalam BMI.txt. penggunaan try dan catch wajib dicantumkan karena ini adalah sebuah metode apabila file yang kita buat tidak ditemukan, atau tidak dapat dibuat, maka try dan catch akan melakukan exception dan mencetak kesalahan yang dibuat oleh sistem.

```
try {  
    File inputFile = new File("bmi.txt");  
    Scanner scanner = new Scanner(inputFile);  
    while (scanner.hasNextLine()) {  
        String line = scanner.nextLine();  
        System.out.println(line);  
    }  
    scanner.close();  
} catch (FileNotFoundException e) {  
    e.printStackTrace();  
}
```

Gambar 2.6 Reda File untuk menampilkan hasil perhitungan di konsol

3. Kesimpulan

enkapsulasi adalah sebuah metode yang memungkinkan program untuk tidak mencampuri file yang ada di antara satu sama lain dan hanya dapat diakses jika diberikan izin tertentu. dengan menggunakan metode ini, kemungkinan untuk terjadi persilangan data yang tidak diinginkan menjadi menurun.

lalu File Handling adalah salah satu metode untuk menuliskan dan membaca file yang ada pada pemrograman Java.

untuk membuat enkapsulasi, file pada mahasiswa perlu dicatat sebagai private. apa guna private? ini berguna untuk menjaga data supaya hanya bisa diakses oleh kelas tersebut.

lalu bagaimana cara kita mengakses data pada enkapsulasi yang sudah kita buat di file mahasiswa? kita bisa menggunakan metode setter dan getter

lalu untuk menuliskan data ke dalam file, kita memerlukan library dari java.io untuk membantu kita dalam penulisan file. perlu diperhatikan bahwa file harus ditutup setelah dibuka supaya file dapat tertulis oleh pemrograman.