

LAPORAN PRAKTIKUM ALGORITMA PEMROGRAMAN

DATA PRIMITIF DAN VARIABEL



Oleh :

Fawwaz Khalid

NIM 2511532004

MATA KULIAH ALGORITMA PEMROGRAMAN

DOSEN PENGAMPU : DR. WAHYUDI, S.T, M.T

ASISTEN LABOR : AUFFAN TAUFIQURRAHMAN

FAKULTAS TEKNOLOGI INFORMASI

DEPARTEMEN INFORMATIKA

UNIVERSITAS ANDALAS

PADANG, 14 november 2025

KATA PENGANTAR

Puji Syukur saya ke tuhan yang maha esa, karena nya saya bisa menyelesaikan laporan praktikum ini dengan baik Penyusunan laporan praktikum ini bertujuan untuk memenuhi tugas laporan alpro saya, saya menulis ini dengan bantuan banyak pihak Saya ucapkan terimakasih sebesar-besarnya kepada berbagai pihak yang membantu saya membuat laporan ini

Padang, 2025

Tim Penyusun

Fawwaz Khalid

| | |
|---------------------------------|-----------|
| KATA PENGANTAR | i |
| DAFTAR ISI | ii |
| BAB I PENDAHULUAN | 1 |
| 1.1 Latar belakang..... | 1 |
| BAB II PEMBAHASAN | 2 |
| 2.1 Alat dan Bahan | 2 |
| 2.2 Landasan Teori | 2 |
| 2.3 Langkah Kerja | 2 & 3 |
| BAB III KESIMPULAN | 4 |
| 3.1 Ringkasan | 4 |
| 3.2 Saran | 4 |
| DAFTAR PUSTAKA | 5 |

BAB I

PENDAHULUAN

1.1 Latar Belakang

Pada praktikum pekan ini kami belajar tentang class, object, dan pemanggilan method dengan class berbeda pada java

Materi ini mengajarkan bagaimana data bisa disimoan, diproses, dan di tampilkan pada strukur program di java

Pada praktikum pekan ini,saya membuat class yang berhubungan,satu untuk menyimpan data, dan satunya untuk pengujian, dan saya untuk memberi input dan output

1.2 Rumusan Masalah

1. Bagaimana cara membuat program konsep oop
2. Apa itu class?
3. Apakah kamu bisa membuat program dengan class lebih dari satu

1.3 Tujuan Praktikum

1. Bisa membuat program dengan lebih dari satu class
2. Mengerti logika program OOP
3. Bisa membuat operasi dengan lebih dari satu class

BAB II

PEMBAHASAN

2.1 Alat dan Bahan

1. Perangkat komputer atau laptop
2. Teks editor atau *IDE* (misalnya NetBeans, Eclipse, atau IntelliJ IDEA)

2.2 Landasan Teori

Apa kamu sudah tahu kepanjangan dari OOP? OOP adalah merupakan kepanjangan dari *Object Oriented Programming*. OOP merupakan suatu metode pemrograman yang berorientasi kepada objek. Dalam bahasa Indonesia OOP dikenal dengan PBO (Pemrograman Berorientasi Objek).

Lalu apa tujuannya? OOP bertujuan untuk mempermudah pengembangan sebuah program. Ia memiliki variabel dan fungsi yang dibungkus ke dalam objek ataupun *class*. Keduanya dapat saling berinteraksi sehingga membentuk sebuah program

Membuat project java

1. Membuat file baru
2. Menuliskan kode yang ingin di jalankan dan run kode jika kode gagal maka perbaiki
- 3 contoh kode yang dilakukan pekan 7:

```

1 package pekan7_2511532004;
2
3 public class Mahasiswa_2511532004 {
4     private int nim;
5     private String nama, nim2;
6
7     public void setNim(int nim) {
8         this.nim = nim;
9     }
10
11     public void setNim2(String nim2) {
12         this.nim2 = nim2;
13     }
14
15     public void setNama(String nama) {
16         this.nama = nama;
17     }
18
19     public int getNim() {
20         return nim;
21     }
22
23     public String getNim2() {
24         return nim2;
25     }
26
27     public String getNama() {
28         return nama;
29     }
30
31     public void Cetak() {
32         System.out.println("NIM : " + nim);
33         System.out.println("Nama: " + nama);
34     }
35     public void Cetak2() {
36         System.out.println("NIM : " + nim2);
37         System.out.println("Nama: " + nama);
38     }
39 }
40

```

Pada class ini, sama membuat class untuk menyimpan data mahasiswa dengan konsep *enkapsulasi*, class ini menyimpan nim nim2 dan nama di keyword private, lalu menyediakan setter getter untuk mengisi dan mengambil nilai tersebut, dengan cetak untuk menampilkan data nya lalu 2 kode di bawah di buat untuk menampilkan datanya

2 contoh lainnya:

```

1 package pekan7_2511532004;
2
3 import java.util.Scanner;
4
5 public class PanggilMahasiswa2 {
6
7     public static void main(String[] args) {
8         Scanner input = new Scanner(System.in);
9
10        System.out.print("NIM: ");
11        String x = input.nextLine();
12
13        System.out.print("Nama: ");
14        String y = input.nextLine();
15
16        Mahasiswa_2511532004 a = new Mahasiswa_2511532004();
17        a.setNim2(x);
18        a.setNama(y);
19
20        if (x.startsWith("25")) {
21            System.out.println(y + " anda angkatan 2025");
22        }
23
24        if (x.contains("1153")) {
25            System.out.println("Anda mahasiswa informatika");
26        }
27
28        a.Cetak2();
29
30        input.close();
31    }
32 }

```

```

1 package pekan7_2511532004;
2
3 public class PanggilMahasiswa_2511532004 {
4
5     public static void main(String[] args) {
6         Mahasiswa_2511532004 a = new Mahasiswa_2511532004();
7         a.setNim(23532);
8         a.setNama("Rahmat");
9         System.out.println(a.getNim());
10        System.out.println(a.getNama());
11        a.Cetak();
12    }
13 }
14 }
15

```

Setelah kode berjalan seperti yang kita inginkan:

4. Menyimpan file
5. Menjalankan program menggunakan *compiler* Java
6. Mencatat hasil output

BAB III

KESIMPULAN

3.1 Ringkasan

Berdasarkan praktikum yang dilakukan, dapat disimpulkan bahwa:

1. Fungsi suatu class bisa di gunakan di class lainnya
2. OOP bisa di pahami

3.2 Saran

Praktikum serupa dapat dikembangkan dengan:

1. Lebih menjelaskan perihal oop

DAFTAR PUSTAKA

1. *Apa itu OOP pada Java? Beserta Contohnya*. Dicoding Blog. Diakses dari <https://www.dicoding.com/blog/apa-itu-oop-pada-java-beserta-contohnya/>

