

LAPORAN PRAKTIKUM
ALGORITMA DAN PEMROGRAMAN
OPERATOR (BANGUN RUANG)
TUGAS ALPRO PEKAN 3

Disusun Oleh :

KINAYA NOVRYA MANDA
(2511531016)

Dosen Pengampu :

DR. WAHYUDI, S.T, M.T

Asisten Praktikum :

MUHAMMAD ZAKI AL HAFIZ



DEPARTEMEN INFORMATIKA
FAKULTAS TEKNOLOGI INFORMASI
UNIVERSITAS ANDALAS

2025

KATA PENGANTAR

Puji syukur penulis sampaikan kehadirat Allah SWT. Salawat dan salam disampaikan kepada Nabi Muhammad SAW. Karena thaufik dan hidayah-Nya, laporan praktikum Java ini dapat diselesaikan dengan baik dan tepat waktu. Laporan ini disusun sebagai salah satu bentuk pertanggungjawaban dari kegiatan praktikum yang telah dilaksanakan, sekaligus sebagai sarana untuk memperdalam pemahaman mengenai konsep dasar serta penerapan bahasa pemrograman Java.

Melalui praktikum ini, penulis memperoleh pengalaman langsung dalam memahami operator yang terdapat pada java dan implementasi logika pemrograman menggunakan Java. Diharapkan laporan ini dapat memberikan gambaran yang jelas mengenai materi yang telah dipelajari serta hasil dari percobaan yang dilakukan selama praktikum berlangsung.

Penulis menyadari bahwa laporan ini masih jauh dari sempurna, baik dari segi isi maupun penyajiannya. Oleh karena itu, kritik dan saran yang membangun sangat diharapkan demi perbaikan di masa yang akan datang.

Akhir kata, penulis mengucapkan terima kasih kepada dosen pengampu, asisten laboratorium, serta semua pihak yang telah memberikan bimbingan, arahan, dan dukungan sehingga laporan praktikum ini dapat tersusun. Semoga laporan ini dapat bermanfaat bagi pembaca, khususnya dalam memahami dasar-dasar pemrograman Java.

Padang, September 2025

Penulis

DAFTAR ISI

KATA PENGANTAR.....	i
DAFTAR ISI	ii
DAFTAR LAMPIRAN	iii
BAB 1 PENDAHULUAN	
1.1 Latar Belakang.....	1
1.2 Tujuan Praktikum.....	1
1.3 Manfaat Praktikum.....	2
BAB 2 PEMBAHASAN	
2.1 Kode Program.....	3
2.2 Langkah Kerja	4
2.3 Analisis Hasil.....	6
BAB 3 KESIMPULAN	
3.1 Hasil Praktikum	10
3.2 Saran Pengembangan	10
DAFTAR KEPUSTAKAAN	12

DAFTAR LAMPIRAN

Kode Program 2.1 (Kode Program HitungVolume)	3
Gambar 2.1 (Langkah 1)	4
Gambar 2.2 (Langkah 2)	5
Gambar 2.3 (Langkah 3)	5
Gambar 2.4 (Langkah 4)	5
Gambar 2.5 (Langkah 5)	5
Gambar 2.6 (Langkah 6)	6
Gambar 2.7 (Hasil)	6
Gambar 2.8 (Flowchart)	7
Gambar 3.1 (Hasil Praktikum HitungVolume)	10
Tabel 2.1 (Pseudocode)	8

BAB 1

PENDAHULUAN

1.1 Latar Belakang

Pemrograman Java merupakan salah satu bahasa pemrograman berorientasi objek yang banyak digunakan untuk mempelajari dasar-dasar logika algoritma, termasuk dalam penyelesaian perhitungan matematis. Salah satu penerapan yang sederhana adalah menghitung volume bangun ruang, seperti tabung. Perhitungan volume tabung dengan rumus $V = \pi \times r^2 \times t$ dapat diimplementasikan dalam program Java menggunakan kelas *Scanner* untuk menerima input dari pengguna, operator aritmatika untuk perhitungan, serta menampilkan hasil output. Dengan praktikum ini, mahasiswa dapat memahami penerapan rumus matematika ke dalam program komputer sekaligus melatih keterampilan dalam penggunaan *input-output* dasar di Java.

1.2 Tujuan Praktikum

Tujuan dilakukannya praktikum ini adalah sebagai berikut :

1. Melatih mahasiswa dalam menuliskan algoritma berupa pseudocode dan flowchart sebelum mengimplementasikan ke dalam bahasa pemrograman Java.
2. Mempelajari penggunaan kelas *Scanner* dalam mengambil input dari pengguna.
3. Memahami cara menggunakan operator aritmatika (+, -, *, /) untuk melakukan perhitungan matematis.
4. Mengimplementasikan rumus volume tabung ke dalam program Java secara sederhana dan sistematis.

1.3 Manfaat Praktikum

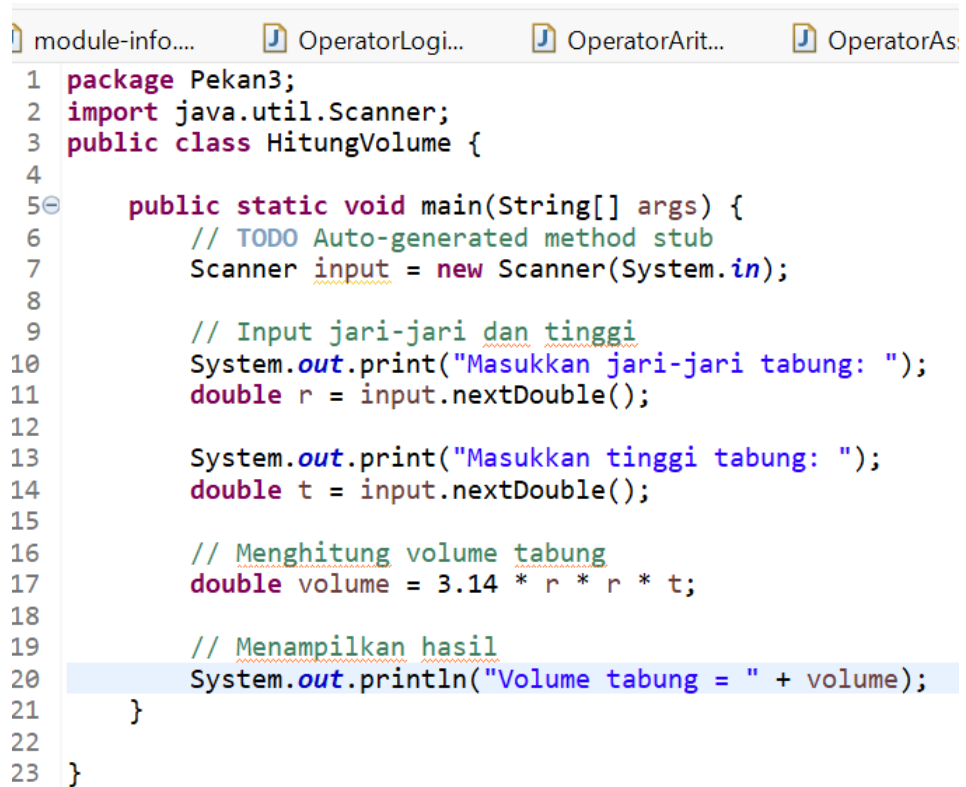
Manfaat dilakukannya praktikum ini adalah sebagai berikut :

1. Mahasiswa mampu menghubungkan teori matematika dengan penerapannya dalam pemrograman komputer.
2. Memberikan pemahaman dasar tentang cara kerja input, proses, dan output dalam program Java.
3. Menjadi latihan awal untuk membangun logika pemrograman yang lebih kompleks tanpa harus menggunakan percabangan atau perulangan.
4. Menambah keterampilan praktis dalam menyusun pseudocode, flowchart, dan mengembangkan kode program yang sesuai dengan permasalahan.

BAB II

PEMBAHASAN

2.1 Kode Program



```

1 package Pekan3;
2 import java.util.Scanner;
3 public class HitungVolume {
4
5     public static void main(String[] args) {
6         // TODO Auto-generated method stub
7         Scanner input = new Scanner(System.in);
8
9         // Input jari-jari dan tinggi
10        System.out.print("Masukkan jari-jari tabung: ");
11        double r = input.nextDouble();
12
13        System.out.print("Masukkan tinggi tabung: ");
14        double t = input.nextDouble();
15
16        // Menghitung volume tabung
17        double volume = 3.14 * r * r * t;
18
19        // Menampilkan hasil
20        System.out.println("Volume tabung = " + volume);
21    }
22
23 }

```

Kode Program 2.1 (Kode Program HitungVolume)

2.1.1 Uraian Kode Program

1. *import java.util.Scanner;* digunakan agar program bisa membaca input dari keyboard.
2. *public class VolumeKerucut* dan *public static void main* adalah struktur utama program Java.
3. *Scanner input = new Scanner(System.in);* untuk mengambil input dari pengguna.

4. Program meminta pengguna memasukkan nilai r (jari-jari) dan t (tinggi).
5. Rumus: $\text{volume} = 3.14 * r * r * t$;
6. Program mencetak hasil volume kerucut ke layar dengan `System.out.println`

2.1.2 Langkah Kerja

1. Membuat kelas baru bernama *HitungVolume* dalam package *Pekan3*.

Source folder:

Package:

☐ Enclosing type:

Name:

Modifiers: ☒ public ☐ package ☐ private ☐ protected
☐ abstract ☐ final ☐ static
☒ none ☐ sealed ☐ non-sealed ☐ final

Superclass:

Interfaces:

Which method stubs would you like to create?
☒ public static void main(String[] args)
☐ Constructors from superclass
☐ Inherited abstract methods

Do you want to add comments? (Configure templates and default value [here](#))
☐ Generate comments

Gambar 2.1 (Langkah 1)

2. Program dimulai dari method *main()*, yang merupakan titik masuk (*entry point*) program Java.


```
public static void main(String[] args) {
```

Gambar 2.2 (Langkah 2)

2. Impor dan buat objek *Scanner* untuk membaca input dari keyboard.

```
6 // TODO Auto-generated method stub
7 Scanner input = new Scanner(System.in);
8
```

Gambar 2.3 (Langkah 3)

3. Minta input jari-jari (r) tabung dari pengguna, lalu simpan ke variabel r.

Minta input tinggi (t) tabung dari pengguna, lalu simpan ke variabel t.

```
9 // Input jari-jari dan tinggi
10 System.out.print("Masukkan jari-jari tabung: ");
11 double r = input.nextDouble();
12
13 System.out.print("Masukkan tinggi tabung: ");
14 double t = input.nextDouble();
15
```

Gambar 2.4 (Langkah 4)

4. Hitung volume tabung dengan rumus: $V = 3.14 \times r^2 \times t$, lalu hasil disimpan dalam variabel volume.

```
15
16 // Menghitung volume tabung
17 double volume = 3.14 * r * r * t;
18
```

Gambar 2.5 (Langkah 5)

5. Tampilkan hasil perhitungan volume tabung ke layar.

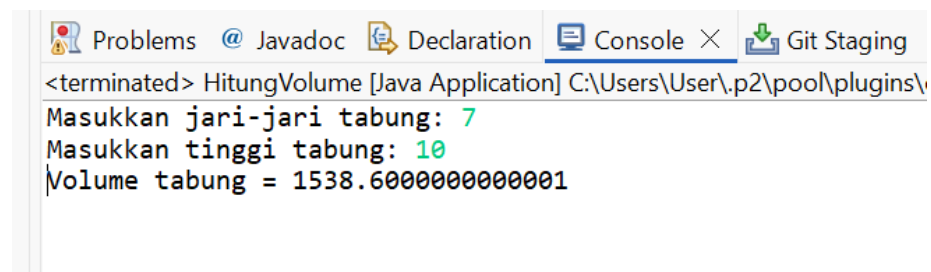
```

18
19 // Menampilkan hasil
20 System.out.println("Volume tabung = " + volume);
21 }
22
23 }

```

Gambar 2.6 (Langkah 6)

6. Hasil akhirnya menampilkan volume tabung yang ditampilkan dengan format yang lebih rapi.



```

Problems @ Javadoc Declaration Console X Git Staging
<terminated> HitungVolume [Java Application] C:\Users\User\p2\pool\plugins\
Masukkan jari-jari tabung: 7
Masukkan tinggi tabung: 10
Volume tabung = 1538.6000000000001

```

Gambar 2.7 (Hasil)

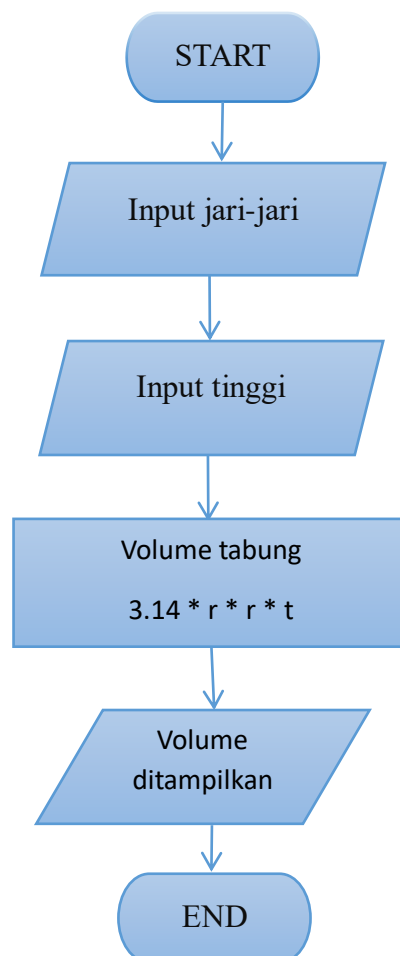
2.1.3 Analisis Hasil

Program perhitungan volume tabung dengan Java berhasil dijalankan sesuai dengan rumus matematis $V = \pi \times r^2 \times t$, di mana r adalah jari-jari alas tabung dan t adalah tinggi tabung. Hasil uji coba menunjukkan bahwa program mampu menerima input dari pengguna melalui kelas *Scanner*, kemudian memproses data menggunakan operator aritmatika perkalian (*) dan pangkat dua dengan perkalian berulang ($r * r$). Hal ini sejalan dengan konsep pemrograman dasar dalam Java yang menjelaskan bahwa setiap operasi matematis dapat diimplementasikan menggunakan operator aritmatika sederhana tanpa memerlukan struktur kendali (Sun & Microsystems, 1999).

Selain itu, penggunaan tipe data *double* memungkinkan hasil perhitungan berupa bilangan pecahan yang lebih presisi dibandingkan *int* (Oracle, 2023). Secara teoritis, tabung merupakan salah satu bangun ruang yang volumenya ditentukan oleh luas alas berbentuk lingkaran dikalikan dengan tinggi (Purcell, Varberg, & Rigdon, 2007).

Dengan demikian, program ini tidak hanya memberikan pemahaman praktis dalam implementasi rumus matematika ke dalam kode, tetapi juga memperkuat keterampilan mahasiswa dalam menghubungkan teori geometri dengan aplikasi komputasi.

2.2 Flowchart



Gambar 2.8 (Flowchart)

Flowchart ini menggambarkan alur sederhana dari program menghitung volume tabung. Program diawali dengan *START*, kemudian pengguna (atau program) melakukan *input* jari-jari dan tinggi tabung, lalu program akan menghitung volume tabung menggunakan rumus yang telah di tentukan. Setelah itu, program akan mengeksekusi perintah untuk menampilkan volume tabung ke layar. Akhirnya, program ditutup dengan simbol *END*. Flowchart ini termasuk *sekuensial (linear)* karena hanya memiliki satu jalur alur tanpa percabangan maupun perulangan.

2.3 Pseudocode

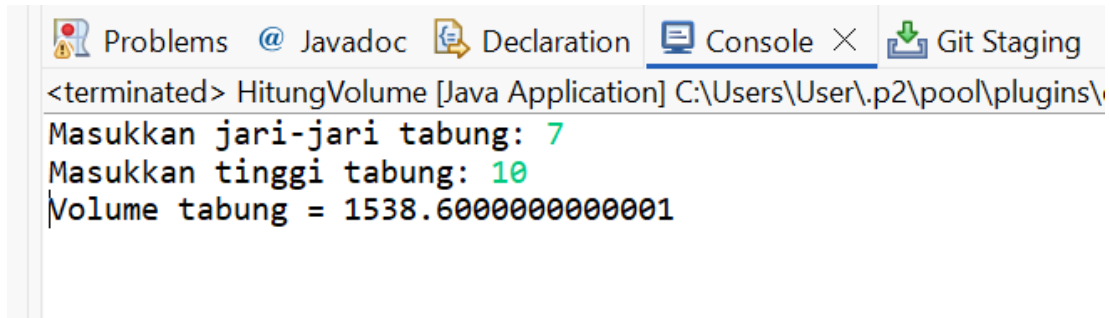
<p>Judul:</p> <p>Program Hitung Volume</p> <p>{Program yang menghitung Volume Tabung}</p>
<p>Deklarasi:</p> <p>Jari-jari tabung : double r;</p> <p>Tinggi tabung : double t;</p> <p>Volume tabung : double volume;</p>
<p>Pseudocode:</p> <ol style="list-style-type: none"> 1. Start 2. Input r 3. Input t 4. $\text{volume} \leftarrow 3.14 * r * r * t$ 5. Output "Volume tabung = ", volume 6. End

Tabel 2.1 (Pseudocode)

Pseudocode tersebut menjelaskan langkah-langkah logis untuk menghitung volume tabung. Pertama, dideklarasikan variabel dengan tipe data *double* yaitu *r* untuk jari-jari, *t* untuk tinggi, dan *volume* untuk menyimpan hasil perhitungan. Program dimulai dengan meminta input dari pengguna berupa nilai jari-jari dan tinggi tabung. Setelah itu, dilakukan proses perhitungan menggunakan rumus volume tabung $V = 3.14 \times r^2 \times t$. Hasil perhitungan kemudian ditampilkan ke layar sebagai output, dan program diakhiri setelah menampilkan hasil.

BAB III KESIMPULAN

3.1 Hasil Praktikum



```
<terminated> HitungVolume [Java Application] C:\Users\User\.p2\pool\plugins\
Masukkan jari-jari tabung: 7
Masukkan tinggi tabung: 10
Volume tabung = 1538.6000000000001
```

Gambar 3.1 (Hasil Praktikum HitungVolume)

Pada program menghitung volume tabung, pengguna memasukkan jari-jari tabung sebesar 7 dan tinggi tabung sebesar 10. Program kemudian menghitung volume tabung menggunakan rumus $V = 3.14 \times r^2 \times t$. Perhitungan dilakukan dengan langkah: $3.14 \times 7^2 \times 10 = 3.14 \times 49 \times 10 = 1538.63.14$. Hasil perhitungan tersebut ditampilkan oleh program sebagai Volume tabung = 1538.6. Hal ini menunjukkan bahwa program dapat menerima input dengan benar, memprosesnya sesuai rumus geometri, dan menghasilkan output yang sesuai dengan teori matematika mengenai volume tabung.

3.2 Saran Pengembangan

Program perhitungan volume tabung yang telah dibuat masih sederhana, sehingga dapat dikembangkan lebih lanjut agar lebih interaktif dan fleksibel. Dari sisi kode program, pengembangan dapat dilakukan dengan menambahkan validasi input agar nilai jari-jari dan tinggi tidak boleh bernilai nol atau negatif, serta menambahkan format pembulatan hasil menggunakan *DecimalFormat* agar output terlihat lebih rapi.

Dari sisi *pseudocode*, pengembangan dilakukan dengan menambahkan langkah-langkah validasi input serta percabangan sederhana untuk menampilkan menu pilihan bangun ruang. Sementara itu, pada *flowchart*, pengembangan dapat dilakukan dengan menambahkan simbol *decision* (percabangan) untuk memilih jenis bangun ruang yang akan dihitung, serta simbol proses tambahan untuk mengecek validitas input.

Dengan pengembangan tersebut, program, *flowchart*, dan *pseudocode* tidak hanya melatih pemahaman dasar mahasiswa tentang perhitungan matematis, tetapi juga melatih logika dalam menangani kasus yang lebih kompleks, sehingga program menjadi lebih bermanfaat, mudah digunakan, dan mendekati aplikasi praktis yang nyata.

DAFTAR PUSTAKA

- [1] J. G. Sun and Microsystems, *The Java Language Specification*. Addison-Wesley, 1999.
- [2] Oracle, "Java SE Documentation," 2023. [Online]. Available: <https://docs.oracle.com/javase/>
- [3] E. J. Purcell, D. Varberg, and S. E. Rigdon, *Calculus with Analytic Geometry*, 9th ed. Prentice Hall, 2007.