LAPORAN PRAKTIKUM

ALGORITMA DAN PEMROGRAMAN

TIPE DATA DAN VARIABEL DI JAVA

Disusun Oleh:

Nama: Rifki Maulana

NIM: 2511533007

Dosen Pengampu: Dr. Wahyudi, S.T.,M.T.

Asisten Praktikum: Jovantri Immanuel Gulo



DEPARTEMEN INFORMATIKA

FAKULTAS TEKNOLOGI INFORMASI

UNIVERSITAS ANDALAS

TAHUN 2025

**BAB I PENDAHULUAN**

**1.1 Latar Belakang**

Pemrograman merupakan keterampilan dasar yang penting dalam bidang informatika. Salah satu konsep dasar yang harus dipahami oleh mahasiswa adalah tipe data dan variabel, karena keduanya menjadi dasar bagaimana program menyimpan, memanipulasi, dan menampilkan informasi. Bahasa pemrograman Java banyak digunakan di dunia akademik dan industri; Java memiliki tipe data primitif seperti int, double, char, dan boolean yang sering dipakai untuk menyelesaikan masalah sehari-hari dalam pemrograman.

**1.2 Tujuan Praktikum**

Praktikum ini memiliki tujuan sebagai berikut:

1. Mahasiswa dapat memahami tipe data dan variabel dalam bahasa Java.

2. Mahasiswa mampu melakukan deklarasi dan inisialisasi variabel dengan tipe int, double, char, dan boolean.

3. Mahasiswa dapat menulis program Java sederhana yang menampilkan nilai dari variabel dengan baik.

**1.3 Manfaat Praktikum**

meningkatkan pemahaman konsep dasar pemrograman (tipe data dan variabel), meningkatkan keterampilan menulis kode Java yang benar, dan mengembangkan kemampuan dokumentasi (pseudocode dan flowchart).

**BAB 2 PEMBAHASAN**

**2.1 Uraian Kode Program**

**package** Pekan2;

**public** **class** ContohChar {

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

// Deklarasi variabel char

**char** huruf1 = 'A';

**char** huruf2 = 'B';

**char** angka = '7';

**char** simbol = '#';

// Menampilkan nilai variabel char

System.***out***.println("Contoh variabel char:");

System.***out***.println("Huruf pertama: "+ huruf1);

System.***out***.println("Huruf kedua: " + huruf2);

System.***out***.println("Angka: " + angka);

System.***out***.println("Simbol: " + simbol);

//Operasi dengan char (berdasarkan kode Unicode/ASCII)

**char** huruf3 = (**char**) (huruf1 + 1); // A (65) + 1 = B (66)

System.***out***.println("Huruf1 + 1 = " + huruf3);

// Char juga bisa disimpan dalam integer (ASCII/Unicode value)

**int** kodeHuruf = huruf1;

String biner1 = String.*format*("%8s", Integer.*toBinaryString*(huruf1)).replace(' ', '0');

System.***out***.println("Kode ASCII dari " + huruf1 + " = " + kodeHuruf);

System.***out***.println("Kode BINER dari " + huruf1 + " = " + biner1);

// Menggabungkan char menjadi string

String kata = "" + huruf1 + huruf2 + angka + simbol;

System.***out***.println("Gabungan char menjadi string: " + kata);

}

}

Kode Program 2.1 Contoh Char

**package** Pekan2;

**public** **class** DeklarasiVariabel {

/\* program java

\*latihan

\*tentang pendeklarasian variabel

\*/

**static** **int** *umur*=25; /\* variabel dapat langsung diinisiasi \*/

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

**int** kode;

**boolean** isDibawahUmur; /\* perhatikan penulisan nama variabel \*/

kode = 1234; /\* pengisian variabel (assignment) \*/

**double** gaji; /\* deklarasi variabel dapat diamana saja \*/

gaji = 5500000.23;

isDibawahUmur = **true**;

System.***out***.println("status: "+isDibawahUmur);

System.***out***.println("kode:"+kode);

System.***out***.println("Umur:"+*umur*);

System.***out***.println("Gaji:"+gaji);

}

}

Kode Program 2.2 Deklarasi Variabel

**package** Pekan2;

**public** **class** KelilingLingkaran {

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

**final** **double** PI = 3.14; /\* Definisi konstanta \*/

**double** radius = 30; /\* Deklarasi variabel \*/

System.***out***.println("Keliling = " + 2 \* PI \* radius);

}

}

Kode Program 2.3 Keliling Lingkaran

**2.2 Langkah Kerja**

Langkah-langkah menyusun program:

1. Buka Eclipse IDE

2. Membuat file Java baru.

3. Menentukan nama public class dan menyesuaikannya dengan nama file.

4. Deklarasi variabel dengan tipe data dasar (int, float, char, boolean, double).

5. Memberi nilai awal (inisialisasi).

6. Menampilkan hasil dengan System.out.println.

7. Kompilasi dan jalankan program.

**2.3 Analisis Hasil**

1. Kesesuaian tipe data dengan nilai yang disimpan

Variabel yang menggunakan int, sehingga hanya menyimpan bilangan bulat tanpa pecahan; perilaku ini sesuai dengan definisi tipe integral di Java [1][2].

2. Format output dan pengaruh format terhadap tampilan nilai

Program menggunakan System.out.printf/System.out.format untuk menampilkan nilai dengan format tertentu; mekanisme ini bergantung pada kelas java.util.Formatter sehingga format spesifier (mis. %.2f) berpengaruh langsung pada pembulatan dan representasi nilai floating-point pada output [3].

4. Potensi kesalahan dan perilaku tidak terduga

Pembulatan/Presisi: karena float hanya mempunyai presisi terbatas (sekitar 6–9 digit desimal), operasi aritmetika berturut-turut dapat menghasilkan kesalahan pembulatan yang kumulatif; bila presisi lebih tinggi dibutuhkan, pertimbangkan double atau tipe khusus untuk keuangan seperti BigDecimal [4][1].

5. Kesesuaian terhadap tujuan praktikum

Program memenuhi tujuan praktikum: memperagakan deklarasi, inisialisasi, dan penggunaan empat tipe data primitif dasar di Java serta menampilkan hasilnya dengan format yang rapi. Pengamatan praktikum menguatkan teori bahwa pemilihan tipe data harus disesuaikan dengan sifat data yang akan disimpan dan tujuan penggunaannya [1][2][4].

**BAB III KESIMPULAN**

**3.1 Kesimpulan**

1. Program berhasil menggunakan dan menampilkan 4 tipe data dasar Java: int, char,double, dan boolean.

2. Perbedaan perilaku tiap tipe data terlihat jelas: int hanya menampung bilangan bulat, float menampung bilangan desimal (dengan batas presisi), char menyimpan satu karakter tunggal, dan boolean hanya true/false.

3. Penulisan variabel, inisialisasi, dan penampilan output menggunakan System.out.println/printf sesuai tujuan praktikum dan menghasilkan output yang sesuai.

**3.2 Saran Pengembangan**

1. Menangani Banyak Data

Kembangkan agar dapat menampung dan menampilkan data beberapa siswa sekaligus (menggunakan array atau ArrayList).

2. Gunakan Tipe Data yang Tepat

Pertimbangkan double bila presisi desimal lebih tinggi diperlukan; waspadai pembulatan saat menampilkan nilai (formatting).

**DAFTAR PUSTAKA**

[1] Oracle, “Primitive Data Types,” The Java™ Tutorials, [Daring]. Tersedia pada: <https://docs.oracle.com/javase/tutorial/java/nutsandbolts/datatypes.html> [Diakses: 21-Sep-2025]

[2] Oracle, Java Language Specification, ch. 4 “Types, Values, and Variables”, Java SE 12 Edition, [Daring]. Tersedia pada: <https://docs.oracle.com/javase/specs/jls/se12/html/jls-4.html> [Diakses: 21-Sep-2025]

[3] Oracle, “Formatting Numeric Print Output,” The Java™ Tutorials, [Daring]. Tersedia pada: <https://docs.oracle.com/javase/tutorial/java/data/numberformat.html> [Diakses: 21-Sep-2025]

[4] D. Darcy, “What Every Java Programmer Should Know About Floating Point,” OracleDevLive presentation (OpenJDK), 2021. [Daring]. Tersedia pada: <https://cr.openjdk.org/~darcy/Presentations/OracleDevLive/OracleDevLive-2021-09-FloatingPoint.pdf> [Diakses: 21-Sep-2025]