

LAPORAN PRAKTIKUM
PEMROGRAMAN ALGORITMA DAN PEMROGRAMAN

“TYPE DATA DAN VARIABEL DI JAVA”

disusun oleh:

Dennis Shauqi Akbar

2511533020

Dosen pengampu: Dr. Wahyudi. S.T.M.T

Asisten Pratikum: Rahmad DRO



DEPARTEMEN INFORMATIKA
FAKULTAS TEKNOLOGI INFORMASI
UNIVERSITAS ANDALAS
TAHUN 2025

KATA PENGANTAR

Puji dan syukur kami panjatkan ke hadirat Allah SWT atas segala rahmat dan karunia-Nya, sehingga kami dapat menyelesaikan tugas kelompok ini dengan baik. Shalawat serta salam semoga senantiasa tercurah kepada junjungan kita Nabi Muhammad SAW, keluarga, sahabat, serta para pengikutnya hingga akhir zaman.

Tugas Individu ini kami susun dengan tujuan untuk memenuhi salah satu tugas mata kuliah Praktikum Algoritma dan Pemrograman, dengan Judul “Type Data Dan Variabel Di Java” yang mencakup Contoh Char, , Deklarasi Variable, dan Keliling Lingkaran.

Kami menyadari bahwa penyusunan tugas ini masih jauh dari sempurna, baik dari segi materi maupun penyajiannya. Oleh karena itu, kami sangat mengharapkan kritik dan saran yang membangun dari semua pihak demi kesempurnaan tugas kami di masa mendatang.

Akhir kata, kami mengucapkan terima kasih kepada dosen pembimbing yang telah memberikan arahan sehingga tugas ini dapat terselesaikan tepat waktu. Semoga tugas ini dapat bermanfaat bagi kami khususnya, dan bagi pembaca pada umumnya.

Padang, 27 September 2025

Dennis Shauqi Akbar

DAFTAR ISI

KATA PENGANTAR.....	i
DAFTAR ISI.....	ii
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang.....	1
1.2 Tujuan	1
1.3 Manfaat Pratikum.....	2
BAB II PEMBAHASAN	3
2.1 Contoh Char	3
2.2 Deklarasi Variabel	4
2.3 Keliling Lingkaran	6
BAB III KESIMPULAN.....	3
3.1 Kesimpulan	3
3.2 Saran Pengembangan.....	3
DAFTAR PUSTAKA.....	3

BAB I

PENDAHULUAN

1.1 Latar Belakang

Dalam pemrograman pada Bahasa Java, tipe data dan variabel merupakan fondasi utama yang tidak dapat dipisahkan. Java merupakan bahasa pemrograman berorientasi objek yang bersifat *strongly typed*, sehingga setiap variabel harus dideklarasikan dengan tipe data tertentu sebelum digunakan. Hal ini dimaksudkan untuk menjaga konsistensi, keamanan data, serta meminimalisir kesalahan saat program dijalankan.

Variabel merupakan wadah yang digunakan untuk menyimpan data di dalam memori komputer agar dapat diakses dan dimanipulasi selama program berjalan. Untuk mendefinisikan variabel di Java, programmer harus menentukan nama variabel serta tipe datanya.

Tipe data adalah kategori atau jenis nilai yang dapat disimpan dalam variabel. Java memiliki dua jenis tipe data utama, yaitu

1. Tipe data primitif : byte, short, int, long, float, double, char, dan boolean
2. Tipe data non-primitif : *String*, *Array*, *Class*, dan *Object*

Dengan adanya tipe data, programmer dapat mengatur memori secara efisien serta melakukan operasi sesuai jenis data. Misalnya, operasi aritmatika hanya dapat dilakukan pada tipe numerik, sementara logika digunakan untuk tipe boolean.

Oleh karena itu, pemahaman yang baik tentang tipe data dan variabel di Java sangatlah penting. Tanpa dasar ini, programmer akan kesulitan dalam menulis kode yang benar, efisien, dan bebas dari error.

1.2 Tujuan

1. Memahami konsep dasar variabel dalam bahasa pemrograman Java.
2. Mengetahui perbedaan antara tipe data primitif dan non-primitif.
3. Mampu mendeklarasikan, menginisialisasi, serta menggunakan variabel sesuai tipe data yang tepat.

1.3 Manfaat Pratikum

1. Mahasiswa memperoleh pemahaman mendasar mengenai peran variabel sebagai wadah penyimpanan data dalam program
2. Mahasiswa dapat memilih tipe data yang tepat sesuai kebutuhan program
3. Mahasiswa terampil dalam menulis kode Java yang benar dan bebas error terkait tipe data.

BAB II

PEMBAHASAN

2.1 Contoh Char

```
1 package pekan2;
2
3 public class ContohChar {
4
5     public static void main(String[] args) {
6         // TODO Auto-generated method stub
7         // Deklarasi Variabel char
8         char huruf1 = 'A';
9         char huruf2 = 'B';
10        char angka = '7';
11        char simbol = '#';
12
13        // Menampilkan nilai variabel char
14        System.out.println("Contoh variabel char:");
15        System.out.println("Huruf pertama: " + huruf1);
16        System.out.println("Huruf kedua: " + huruf2 );
17        System.out.println("Angka: " + angka);
18        System.out.println("Simbol: " + simbol);
19
20        // Operasi dengan char (berdasarkan kode Unicode/ASCII)
21        char huruf3 = (char) (huruf1 + 1); // A (65) + 1 = 66 = B [66]
22        System.out.println("Huruf1 + 1 = " + huruf3);
23
24        //Char juga bisa disimpan dalam integer (ASCII/Unicode value)
25        int kodeHuruf = huruf1;
26        String biner1 = String.format("%8s", Integer.toBinaryString(kodeHuruf)).replace(' ', '0');
27        System.out.println("Kode ASCII dari " + huruf1 + " = " + kodeHuruf);
28        System.out.println("Kode Biner dari " + huruf1 + " = " + biner1);
29
30        //Menggabungkan char menjadi string
31        String kata = "" + huruf1 + huruf2 + angka + simbol;
32        System.out.println("Gabungan char menjadi string: " + kata);
33    }
34 }
35
36 }
37
```

Kode Program 2.1

Dalam program Contoh Char, tahap awal yang dilakukan Adalah mendeklarasikan sejumlah variable bertipe char seperti huruf, angka, dan symbol. Variabel-variabel ini kemudian ditampilkan ke layer untuk memperlihatkan bahwa tipe data char hanya dapat menampung satu karakter. Setelah itu, dilakukan operasi aritmatika terhadap char dengan memanfaatkan nilai Unicode/ASCII. Sebagai contoh, karakter ‘A’ yang memiliki nilai 65 ditambahkan dengan 1 sehingga menghasilkan karakter ‘B’ dengan nilai 66. Program juga memperlihatkan bahwa karakter dapat dikonversi ke dalam bentuk bilangan bulat (int) untuk mengetahui nilai ASCII-nya, bahkan ditampilkan pula representasi biner dari karakter tersebut yang ditampilkan dalam output berikut:

```
Kontoh variabel char:
Huruf pertama: A
Huruf kedua: B
Angka: 7
Simbol: #
Huruf1 + 1 = B
Kode ASCII dari A = 65
Kode Biner dari A = 01000001
Gabungan char menjadi string: AB7#
```

Gambar 2.1

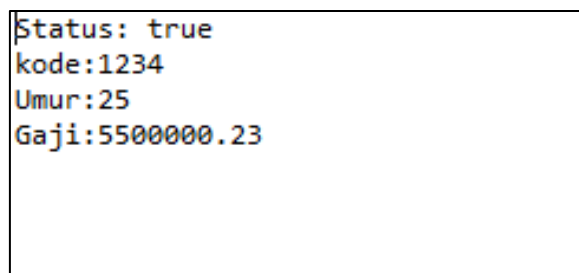
Dari output diatas variable-variabel char yang telah dideklarasikan berhasil memunjukan karakter sesuai nilai yang diberikan: huruf, angka, simbol. Ini membuktikan bahwa tipe data char mampu menyimpan satu karakter tunggal secara tepat. Selanjutnya, Ketika operasi penjumlahan dilakukan pada karakter 'A' dengan menambah satu unit. muncul karakter 'B': hal ini mengindikasikan bahwa cahr di Java dapat dioperasikan seperti bilangan berdasarkan kode Unicode/ASCII. Di mana 'A' memiliki nilai 65 dan hasilnya menjadi 66 untuk 'B'. program juga memperlihatkan konversi char ke nilai integer untuk mengetahui kode ASCII, serta representasinya dalam bentuk biner, yakni 01000001 untuk 'A'. pada bagian akhir, beberapa variable char disatukan menjadi sebuah string yang menghasilkan teks "AB7#", yang menunjukan bahwa karakter-karakter dapat digabung untuk membentuk string. Secara keseluruhan, hasil ini memperkuat bahwa tiper data char di Java tidak hanya untuk emnyimpan karakter, melainkan juga dapat dimanipulasi secara numerik dan dikombinasikan menjadi teks.

2.2 Deklarasi Variabel

```
1 package pekan2;
2
3 public class DeklarasiVariabel {
4     /* program java
5      * latihan
6      * tentang pendeklarasian variabel
7      */
8     static int umur=25; /* variabel dapat langsung diinisiasi */
9     public static void main(String[] args) {
10         int kode;
11         boolean isDibawahUmur; /* perhatikan penulisan nama variabel */
12         kode = 1234; /* pengisian variabel (assignment)*/
13         double gaji; /* deklarasi variabel dapat dimana saja */
14         gaji = 5500000.23;
15         isDibawahUmur = true;
16         System.out.println("Status: "+isDibawahUmur);
17         System.out.println("kode:"+kode);
18         System.out.println("Umure:"+umur);
19         System.out.println("Gaji:"+gaji);
20
21     }
22
23 }
24
25
```

Kode Program 2.2

Program Deklarasi Variabel diawali dengan mendeklarasikan variabel bertipe int bernama umur yang langsung diinisialisasi dengan nilai 25. Selanjutnya di dalam metode main, dibuat beberapa variabel tambahan, yaitu kode bertipe int, isDibawahUmur bertipe Boolean, serta gaji bertipe double. Variabel kode kemudian diberi nilai 1234, sedangkan gaji diinisialisasi dengan nilai 5500000,23. nilai isDibawahUmur menjadi true untuk menunjukkan status logis dari kondisi tertentu. Setelah semua variabel dideklarasikan dan diberi nilai, program menggunakan perintah `System.out.println` untuk menampilkan hasil ke layar berupa status Boolean, kode angka, umur, dan gaji. Dari Langkah-langkah ini, program memperlihatkan bahwa variabel di java dapat dideklarasikan baik dengan langsung diberi nilai maupun dipisahkan dari proses inisialisasi, serta dapat menyimpan berbagai jenis data seperti bilangan bulat, bilangan decimal, maupun nilai logika.



```
Status: true
kode:1234
Umur:25
Gaji:5500000.23
```

Gambar 2.2

Berdasarkan hasil output diatas, terlihat bahwa setiap variabel berhasil menyimpan dan menampilkan nilai sesuai dengan tipe data masing-masing. Variabel `isDiBawahUmur` yang bertipe boolean menampilkan nilai `true`, menunjukkan bahwa tipe data logika hanya menyimpan dua kemungkinan, yaitu benar (*true*) atau salah (*false*). Variabel `kode` yang bertipe int menampilkan nilai 1234 sebagai bilangan bulat, sedangkan variabel `umur` yang juga bertipe int menampilkan nilai 25 yang sebelumnya sudah diinisialisasi langsung. Adapun variabel `gaji` yang menggunakan tipe data double menampilkan nilai 5500000.23, menunjukkan bahwa tipe data ini mampu menyimpan bilangan desimal dengan presisi tinggi. Analisis ini membuktikan bahwa Java menyediakan berbagai tipe data sesuai kebutuhan penyimpanan nilai, serta mendukung fleksibilitas deklarasi dan inisialisasi variabel baik secara langsung maupun terpisah. Dengan demikian, program ini memperlihatkan perbedaan fungsi setiap tipe data dan pentingnya memilih tipe yang sesuai agar data dapat diolah dengan tepat.

2.3 Keliling Lingkaran

```

1 package pekan2;
2
3 public class KelilingLingkaran {
4
5     public static void main(String[] args) {
6         // TODO Auto-generated method stub
7         final double PI=3.1414; /* Definisi konstanta */
8         double radius = 30; /* Deklarasi variabel */
9         System.out.println("Keliling = " + 2 * radius);
10
11     }
12
13 }

```

Kode Program 2.3

Program **diatas** diawali dengan mendefinisikan sebuah konstanta PI menggunakan kata kunci `final` bertipe `double` dengan nilai 3.1414, yang merepresentasikan nilai π dalam perhitungan matematika. Selanjutnya dideklarasikan variabel `radius` dengan nilai 30 sebagai jari-jari lingkaran. Setelah itu, program menggunakan perintah `System.out.println` untuk menghitung keliling lingkaran dengan rumus $2 * PI * radius$ dan menampilkannya ke layar. Dari langkah-langkah tersebut, dapat dipahami bahwa konstanta digunakan untuk menyimpan nilai tetap yang tidak dapat diubah selama program berjalan, sementara variabel digunakan untuk menyimpan data yang bisa disesuaikan. Program ini menunjukkan bagaimana Java mengimplementasikan konsep konstanta dan variabel dalam operasi perhitungan sederhana untuk memperoleh nilai keliling lingkaran.



Keliling = 188.484

Gambar 2.3

Berdasarkan hasil output diatas, nilai konstanta PI ditetapkan sebesar 3.1414 dan variabel `radius` bernilai 30. Ketika program dijalankan, perhitungan keliling lingkaran dilakukan dengan menggunakan rumus $2 * PI * radius$. Substitusi nilai ke dalam rumus menghasilkan perhitungan $2 * 3.1414 * 30 = 188.484$. Hasil ini kemudian ditampilkan pada layar dengan output berupa kalimat **"Keliling = 188.484"**. Analisis ini memperlihatkan bahwa program berhasil menerapkan konsep konstanta dan variabel dalam perhitungan matematis sederhana.

Penggunaan final pada PI memastikan bahwa nilai konstanta π tidak dapat diubah, sedangkan variabel radius dapat diubah sesuai kebutuhan untuk menghitung keliling lingkaran dengan jari-jari berbeda. Dengan demikian, program ini tidak hanya menegaskan pentingnya konstanta dalam menjaga nilai tetap, tetapi juga menunjukkan fleksibilitas variabel dalam pemrosesan data numerik.

BAB III

KESIMPULAN

3.1 Kesimpulan

Dari serangkaian praktikum yang dilakukan, dapat disimpulkan bahwa variabel dan tipe data dalam Java memiliki peran penting dalam menyimpan serta mengolah data sesuai dengan kebutuhan. Program ContohChar menunjukkan bahwa tipe data char tidak hanya menyimpan karakter tunggal tetapi juga dapat digunakan dalam operasi numerik, representasi ASCII, hingga digabungkan menjadi string. Program DeklarasiVariabel memperlihatkan bahwa Java mendukung berbagai tipe data seperti int, double, dan boolean, serta memungkinkan deklarasi variabel baik langsung dengan inisialisasi maupun terpisah. Sedangkan program KelilingLingkaran menegaskan perbedaan antara variabel dan konstanta, di mana final digunakan untuk nilai tetap seperti π , sementara variabel dapat diubah sesuai kebutuhan. Secara keseluruhan, pemahaman mengenai variabel, konstanta, dan tipe data dasar merupakan landasan penting dalam pemrograman Java karena menentukan keakuratan dan efisiensi dalam pengolahan data.

3.2 Saran Pengembangan

Untuk pengembangan selanjutnya, praktikum dapat diperluas dengan menggunakan tipe data non-primitif seperti String, Array, atau Class agar pemahaman lebih komprehensif. Selain itu, pengolahan variabel dapat dikombinasikan dengan logika kontrol (*if*, *loop*) maupun input dari pengguna sehingga program menjadi lebih interaktif dan aplikatif.

DAFTAR PUSTAKA

- [1] Oracle, *The Java Language Specification, Java SE 11 Edition, Chapter 4: Types, Values, and Variables*. Oracle, 2018. [Online]. Available: <https://docs.oracle.com/javase/specs/jls/se11/html/jls-4.html>
- [2] D. J. Eck, *Javanotes 9: Notes on Programming in Java, Section 2.2 — Variables and the Primitive Types*. Hobart and William Smith Colleges, 2022. [Online]. Available: <https://math.hws.edu/javanotes/c2/s2.html>
- [3] Java Development Journal, "Java Data Types," *Java Development Journal*, 2020. [Online]. Available: <https://javadevjournal.com/java/java-data-types/>
- [4] M. Alghamdi, L. Moreno, and E. Shihab, "Characterising the Knowledge about Primitive Variables in Java Code Comments," *arXiv preprint arXiv:2103.12291*, 2021. [Online]. Available: <https://arxiv.org/abs/2103.12291>

