

**LAPORAN PRATIKUM
ALGORITMA PEMROGRAMAN**

**TIPE DATA PRIMITIF, VARIABEL, DAN KONSTANTA
PEMROGRAMAN JAVA**

disusun oleh:

Khairunnisa M.

NIM 2511532005

Dosen Pengampu:

DR. Wahyudi, S. T., M. T.

Asisten Pratikum:

Aufan Taufiqurrahman



**DEPARTEMEN INFORMATIKA
FAKULTAS TEKNOLOGI INFORMASI
UNIVERSITAS ANDALAS**

2025

KATA PENGANTAR

Puji syukur penulis panjatkan ke hadirat Tuhan Yang Maha Esa karena berkat rahmat dan karunia-Nya, laporan praktikum dengan judul **“Tipe Data Primitif, Variabel, dan Konstanta pada Pemrograman Java”** dapat diselesaikan dengan baik dan tepat waktu. Laporan ini disusun untuk memenuhi salah satu tugas pada mata kuliah Algoritma Pemrograman sekaligus memberikan pemahaman praktis mengenai konsep dasar pemrograman, khususnya penggunaan tipe data primitif, variabel, dan konstanta dalam bahasa Java. Penulis menyadari laporan ini masih jauh dari sempurna, sehingga kritik dan saran yang membangun sangat diharapkan demi penyempurnaan di masa mendatang. Ucapan terima kasih penulis sampaikan kepada dosen pengampu, asisten praktikum, serta semua pihak yang telah membantu hingga laporan ini dapat bermanfaat bagi penulis maupun pembaca.

Padang, 17 September 2025

Khairunnisa M

DAFTAR ISI

HALAMAN JUDUL.....	i
KATA PENGANTAR.....	ii
DAFTAR ISI.....	iii
 BAB I PENDAHULUAN.....	 4
1.1 Latar Belakang	4
1.2 Tujuan.....	4
1.3 Manfaat.....	5
 BAB II PEMBAHASAN.....	 6
2.1 Tipe Data Primitif	6
2.2 Variabel.....	9
2.3 Konstanta.....	11
 BAB III KESIMPULAN.....	 13
 DAFTAR PUSTAKA.....	 14
LAMPIRAN.....	15

BAB I

PENDAHULUAN

1.1 Latar Belakang

Algoritma dan pemrograman merupakan dasar utama dalam mempelajari ilmu komputer. Pemahaman mengenai tipe data, variabel, dan konstanta menjadi hal yang sangat penting karena ketiganya merupakan fondasi dalam membangun sebuah program. Bahasa pemrograman Java, sebagai salah satu bahasa pemrograman populer yang bersifat berorientasi objek, menyediakan berbagai tipe data primitif yang digunakan untuk menyimpan nilai sederhana seperti bilangan bulat, bilangan pecahan, karakter, dan nilai logika.

Variabel berfungsi sebagai wadah penyimpanan data yang nilainya dapat berubah selama program dijalankan, sedangkan konstanta digunakan untuk menyimpan nilai yang tetap. Dengan memahami konsep tipe data, variabel, dan konstanta, seorang programmer dapat menulis program yang lebih efisien, terstruktur, serta mudah dipahami dan dikelola.

Pratikum mengenai tipe data primitif, variabel, dan konstanta dalam pemrograman Java bertujuan agar mahasiswa mampu mengimplementasikan konsep dasar tersebut ke dalam program sederhana. Melalui pemahaman ini, mahasiswa diharapkan dapat mengembangkan kemampuan logika dan berfikir sistematis dalam menyelesaikan permasalahan menggunakan bahasa pemrograman Java.

1.2 Tujuan

1. Memahami konsep dan karakteristik tipe data primitif dalam bahasa pemrograman Java.
2. Mengetahui cara mendeklarasikan dan menggunakan variabel sesuai tipe data yang dibutuhkan.

3. Memahami penggunaan konstanta dengan kata kunci *final* serta perbedaannya dengan variabel.
4. Meningkatkan keterampilan dalam menuliskan kode program sederhana dengan memanfaatkan tipe data primitif, variabel, dan konstanta.
5. Melatih kemampuan menganalisis kebutuhan tipe data dan konstanta untuk menyelesaikan permasalahan dalam program.
6. Menumbuhkan pemahaman mengenai pentingnya pemilihan tipe data, variabel, dan konstanta untuk menghasilkan program yang efisien, mudah dipahami, dan terstruktur.

1.3 Manfaat Pratikum

1. Memahami dasar pemrograman – mahasiswa lebih mudah memahami logika dasar dalam membangun program melalui penggunaan tipe data, variabel, dan konstanta.
2. Meningkatkan keterampilan coding – Melatih kemampuan menulis kode Java dengan sintaks yang benar, rapi, dan sesuai standar.
3. Membiasakan berpikir logis dan sistematis – karena penggunaan tipe data, variabel, dan konstanta menuntut mahasiswa berpikir terstruktur.
4. Membangun pondasi untuk materi lanjutan – Konsep ini menjadi dasar penting sebelum mempelajari struktur data, fungsi, maupun pemrograman berorientasi objek.
5. Mendukung pemecahan masalah – membantu mahasiswa menyusun program sederhana untuk menyelesaikan persoalan sehari-hari dengan pendekatan algoritmik.

BAB II

PEMBAHASAN

Pada pratikum ini dilakukan implementasi dasar pemrograman Java dengan focus pada tipe data primitif, variabel, dan konstanta. Pratikum bertujuan untuk melatih pemahaman mahasiswa dalam mendeklarasikan variabel, menggunakan konstanta, serta memanfaatkan tipe data primitive sesuai kebutuhan program.

2.1. Tipe Data Primitif

Dalam bahasa pemrograman Java, tipe data primitif adalah tipe data dasar yang sudah tersedia secara bawaan (*built-in*) oleh bahasa Java, bukan berasal dari objek atau *class*. Tipe data ini digunakan untuk menyimpan nilai sederhana dan memiliki ukuran memori tetap. Java memiliki delapan tipe data primitif, yaitu *byte*, *short*, *int*, *long*, *float*, *double*, *char*, dan *boolean*.

- a) *Int*, merupakan tipe data bilangan bulat yang paling sering di gunakan. *Int* memiliki ukuran 32 bit (4 *byte*) dan rentang nilai dari -2,147,483,648 sampai 2,147,483,647.
- b) *Byte*, digunakan untuk menghemat memori jika nilai yang disimpan kecil. *Byte* memiliki ukuran 8 bit (1 *byte*) dan rentang nilai dari -128 sampai 127.
- c) *Short*, berguna jika data lebih besar dari *byte* tetapi masih kecil untuk *int*. *Short* memiliki ukuran 16 bit (2 *byte*) dan rentang nilai dari -32,768 sampai 32,767.
- d) *Long*, tipe data ini dipakai untuk angka yang sangat besar. *Long* memiliki ukuran 64 bit (8 *byte*).

- e) *Float*, digunakan untuk angka desimal yang tidak terlalu presisi. *Float* memiliki ukuran 32 bit (4 *byte*) dan presisi sekitar 7 digit desimal.
- f) *Double*, cocok untuk perhitungan desimal yang membutuhkan presisi lebih tinggi. *Double* memiliki ukuran 64 bit (8 *byte*) dan presisi sekitar 15-16 digit desimal.
- g) *Char*, menyimpan 1 karakter, termasuk huruf, angka, atau simbol. *Char* memiliki ukuran 16 bit (2 *byte*, *Unicode*).

Pada praktikum, mahasiswa mencoba mendeklarasikan beberapa variabel dengan tipe data tersebut serta menampilkan hasilnya melalui perintah `system.out.println()` . untuk lebih memahami penggunaan tipe data primitif dalam bahasa Java, berikut ditunjukkan sebuah contoh program yang berisi deklarasi variabel *char*, operasi pada karakter berdasarkan kode *ASCII/Unicode*, serta penggabungan beberapa karakter menjadi sebuah *string*. Program ini juga menampilkan bagaimana karakter dapat direpresentasikan sebagai kode *ASCII* dan *biner*. Potongan kode dapat dilihat pada gambar 2.1 dibawah ini.

```

1 package Pekan2;
2
3 public class ContohChar {
4     public static void main(String[] args) {
5         // Deklarasi variabel char
6         char huruf1 = 'A';
7         char huruf2 = 'B';
8         char angka = '7';
9         char simbol = '#';
10
11         // Menampilkan nilai variabel char
12         System.out.println("Contoh variabel char:");
13         System.out.println("Huruf pertama: " + huruf1);
14         System.out.println("Huruf kedua: " + huruf2);
15         System.out.println("Angka: " + angka);
16         System.out.println("Simbol: " + simbol);
17
18         // Operasi dengan char (berdasarkan kode Unicode/ASCII)
19         char huruf3 = (char) (huruf1 + 1); // A (65) + 1 = B (66)
20         System.out.println("huruf1 + 1 = " + huruf3);
21
22         // char juga bisa disimpan dalam integer (ASCII/Unicode value)
23         int kodeHuruf = huruf1;
24         String biner1 = String.format("%8s", Integer.toBinaryString(kodeHuruf)).replace(' ', '0');
25         System.out.println("Kode ASCII dari " + huruf1 + " = " + kodeHuruf);
26         System.out.println("Kode BINER dari " + huruf1 + " = " + biner1);
27
28         // Menggabungkan char menjadi string
29         String kata = "" + huruf1 + huruf2 + angka + simbol;
30         System.out.println("Gabungkan char menjadi string: " + kata);
31
32     }
33 }
34
35 }
36

```

Gambar 2.1. Program Java dengan Tipe Data *char*.

Langkah kerja:

1. Membuat project baru.
 - a) Buka *Eclipse* Java
 - b) Buat sebuah *package* dengan nama *Pekan2*
 - c) Buat file baru dengan nama *ContohChar.java*
2. Menuliskan kode program.
 - a) Deklarasikan variabel *char* dengan beberapa nilai karakter, seperti huruf, angka, dan simbol.
 - b) Tambahkan perintah *System.out.println()* untuk menampilkan nilai variabel *char* tersebut.
3. Melakukan operasi pada *char*:
 - a) Tambahkan operasi aritmatika pada variabel *char* dengan memanfaatkan nilai *Unicode/ASCII*.
4. Menyimpan *char* dalam bentuk *intenger*:
 - a) Konversi *char* ke dalam bentuk kode *ASCII* dengan menyimpannya ke variabel bertipe *int*.
 - b) Tampilkan nilai *ASCII* dan bentuk *biner* dengan *Integer.toBinaryString()* .
5. Menggabungkan beberapa *char* menjadi *string*.
 - a) Buat variabel *string* yang berisi gabungan dari beberapa *char* (huruf1 + huruf2 + angka + simbol).
 - b) Tampilkan hasil penggabungan tersebut ke layar.
6. Menjalankan program.
 - a) Compile dan jalankan program *ContohChar.java* .
 - b) Amati hasil *output* yang menampilkan nilai variabel *char*, hasil operasi *Unicode*, kode *ASCII*, kode *biner*, serta penggabungan karakter menjadi *string*.

Dari hasil percobaan program *ContohChar* meunjukkan penggunaan tipe data primitif *char* dalam Java yang dapat menyimpan satu karakter,

baik berupa huruf, angka, maupun simbol. Melalui praktikum ini terlihat bahwa nilai *char* tidak hanya bisa ditampilkan langsung, tetapi juga dapat dioalah menggunakan kode *ASCII/Unicode*, dikonversi ke bilangan bulat atau *biner*, serta digabungkan menjadi *string*. Hal ini membuktikan bahwa *char* memiliki fleksibilitas tinggi dalam pengolahan data karakter, sehingga pemahaman tipe data ini sangat penting sebagai dasar untuk manipulasi teks, pengkodean, maupun pemrosesan simbol dalam pemrograman.

2.2. Variabel

Variabel merupakan wadah untuk penyimpanan data dalam memori komputer yang diberi nama, sehingga data tersebut dapat diakses dan digunakan kembali di dalam program. Namun variabel ini nilainya dapat berubah selama program dijalankan, sesuai dengan tipe data yang digunakan.

Pada praktikum, dilakukan percobaan dengan mendeklarasikan variabel, memberikan nilai awal, serta menampilkan pada layar, digunakan contoh program sederhana dalam bahasa Java. Pada program ini terdapat beberapa variabel dengan tipe data berbeda, yaitu *int*, *boolean*, dan *double*. Selain itu, terdapat pula variabel *static* bertipe *int* yang langsung diinisialisasi. Potongan kode program dapat dilihat pada gambar 2.2 di bawah ini.

```

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("Umur:"+umur);
19         System.out.println("Gaji:"+gaji);
20     }
21 }
22
23

```

Gambar 2.2. Program Deklarasi Variabel

Langkah kerja:

1. Membuat *package* dengan nama `Pekan2` sebagai tempat penyimpanan program.
2. Mendeklarasikan kelas dengan nama `DeklarasiVariabel` yang bersifat *public*.
3. Menambahkan komentar pada program untuk menjelaskan tujuan program dan bagian-bagian kode.
4. Mendeklarasikan variabel *static* umur dengan tipe data *int* dan langsung diberi nilai 25.
5. Membuat method utama `main(String[] args)` sebagai titik awal eksekusi program.
6. Didalam method main, mendeklarasikan beberapa variabel:
 - a) *Int* kode; untuk menyimpan nilai bilangan bulat.
 - b) *Boolean* `isDibawahUmur`; untuk menyimpan nilai logika (*true/false*).
 - c) *Double* gaji; untuk menyimpan nilai pecahan/desimal.
7. Memberikan nilai pada variabel-variabel tersebut:
 - a) `Kode = 1234;`
 - b) `Gaji = 550000.23;`
 - c) `IsDibawahUmur = true;`
8. Menampilkam isi dari masing-masing menggunakan perintah `System.out.println()` , yaitu menampilkan status, kode, umur, dan gaji ke layar.
9. Menjalankan program untuk melihat hasil *output* sesuai nilai variabel yang telah diisi.

Dari hasil program `DeklarasiVariabel` memperlihatkan bagaimana variabel digunakan untuk menyimpan dan menampilkan data dengan berbagai tipe dasar di Java, yaitu *int*, *boolean*, dan *double*. Program ini

membuktikan bahwa nilai variabel dapat ditentukan saat deklarasi maupun diubah kemudian melalui *assignment*, serta dapat ditampilkan ke layar menggunakan perintah *System.out.println*. dengan demikian, pratikum ini menegaskan bahwa pemahaman tentang variabel sangat penting sebagai dasar dalam pemrograman karena hampir seluruh proses komputasi melibatkan penyimpanan, pengolahan, dan penampilan data melalui variabel.

2.3. Konstanta.

Konstanta digunakan untuk menyimpan nilai tetap yang tidak boleh diubah selama program berjalan. Pada Java, konstanta dibuat dengan menambahkan kata kunci *final*. Konstanta berguna untuk menyimpan nilai yang sifatnya tetap, seperti nilai π , bilangan tetap, atau konfigurasi program.

Program berikut digunakan untuk menghitung keliling lingkaran dengan memanfaatkan konstanta π (PI) dan variabel radius. Dalam program ini, konstanta pi didefinisikan dengan nilai 3.14, sedangkan radius lingkaran ditentukan dengan nilai 30. Perhitungan keliling dilakukan dengan rumus $2 * \pi * \text{radius}$, kemudian hasilnya ditampilkan ke layar. Gambar dibawah ini menunjukkan kode program lengkapnya.

```

1 package Pekan2;
2
3 public class KelilingLingkaran {
4     public static void main (String[] args) {
5         final double PI = 3.14; /* Definisi konstanta */
6         double radius = 30; /* Deklarasi variabel */
7         System.out.println("keliling = " + 2 * PI * radius);
8     }
9 }
10
11

```

Gambar 2.3. Program Keliling Lingkaran

Langkah kerja:

1. Membuat *package* dengan nama Pekan2 sebagai tempat penyimpanan program.

2. Membuat kelas Java Bernama KelilingLingkaran sebagai wadah program.
3. Mendefinisikan method *main* sebagai titik awal eksekusi program.
4. Mendeklarasikan konstanta PI bertipe *double* dengan nilai 3.14 menggunakan kata kunci *final*, sehingga nilainya tidak dapat diubah selama program berjalan.
5. Mendeklarasikan variabel radius bertipe *double* dengan nilai 30 sebagai jari-jari lingkaran.
6. Melakukan perhitungan keliling lingkaran dengan rumus $2 * PI * radius$.
7. Menampilkan hasil perhitungan ke layar menggunakan *System.out.println()*.
8. Program selesai dijalankan dan hasil keliling lingkaran ditampilkan.

Dari analisis ini dapat disimpulkan bahwa program berhasil mengimplementasikan penggunaan konstanta dan variabel, serta memperlihatkan bagaimana operasi aritmatika sederhana dalam Java dapat digunakan untuk menyelesaikan perhitungan matematis.

BAB III

KESIMPULAN

Berdasarkan hasil pratikum pemrograman Java mengenai tipe data primitive, variabel, dan konstanta, dapat disimpulkan bahwa praktikan telah memahami cara mendeklarasikan, menginisialisasi, serta menggunakan variabel dan konstanta dalam sebuah program. Program yang dibuat berhasil menampilkan nilai variabel bertipe *int*, *double*, *boolean*, dan *char*, serta memperlihatkan bagaimana konstanta digunakan dalam perhitungan matematis, seperti menghitung keliling lingkaran. Praktikum ini juga menunjukkan pentingnya pemahaman tipe data primitif untuk memastikan data disimpan dan diolah dengan benar sesuai kebutuhan program.

Adapun saran untuk pengembangan praktikum selanjutnya adalah agar materi diperluas dengan melibatkan operasi yang lebih kompleks, seperti penggunaan array, tipe data non-primitif, serta penerapan fungsi atau method untuk meningkatkan modularitas program. Selain itu, penggunaan *library* Java yang lebih lanjut dapat membantu memperkaya pemahaman praktikan mengenai penerapan konsep variabel, konstanta, dan operasi aritmatika dalam kasus nyata.

DAFTAR PUSTAKA

- [1] C. S. Team and C. S. Team, "Tipe Data Primitif dalam Dunia Pemrograman," 07 Juni 2021. [Online]. Available: <https://codingstudio.id/blog/tipe-data-primitif/>.
- [2] Muhardian, Ahmad; Muhardian, Ahmad, "Belajar Pemrograman Java: Variabel dan Tipe Data," PetaniKode, 09 November 2015. [Online]. Available: <https://www.petanikode.com/java-variabel-dan-tipe-data/>.
- [3] Andre, "Tutorial Belajar Java Part 12: Pengertian dan Cara Pembuatan Konstanta Bahasa Java," Dunia Ilmu Komputer, 15 November 2019. [Online]. Available: https://www.duniailkom.com/tutorial-belajar-java-pengertian-dan-cara-pembuatan-konstanta-bahasa-java/#google_vignette.
- [4] S. S. Putro, D. R. Anamisa and F. A. Mufarroha, "Algoritma Pemrograman," MNC Publishing, November 2019. [Online]. Available: https://books.google.co.id/books?hl=en&lr=&id=diCeEAAAQBAJ&oi=fnd&pg=PP1&dq=algoritma+pemrograman&ots=1Y4rCL5OLu&sig=S41zxj2N5OM_yCvMPTHfrkbs_iQ&redir_esc=y#v=onepage&q=algoritma%20pemrograman&f=false.
- [5] J. Enterprise, "Java Komplet," PT Elex Media Komputindo, 2017. [Online]. Available: https://books.google.co.id/books?hl=en&lr=&id=G0NGDwAAQBAJ&oi=fnd&pg=PP1&dq=TIPE+DATA+PRIMITIF,+VARIABEL,+DAN+KONSTANTA+PADA+PEMR OGRAMAN+JAVA&ots=VcBGm3mF_c&sig=wJQsvvjBorKjA2wmt6kuAarNujU&redir_esc=y#v=onepage&q&f=false.