

LAPORAN PRATIKUM  
PEMROGRAMAN ALGORITMA DAN PEMOGRAMAN

“Pratikum Pekan 3”

Disusun Oleh:

Rafikhul Ramadhan

2511533012

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

Asisten Praktikum : Rahmad Dwirizki Olders



DEPARTEMEN INFORMATIKA  
FAKULTAS TEKNOLOGI INFORMASI  
UNIVERSITAS ANDALAS

2025

## KATA PENGANTAR

Segala puji dan syukur penulis panjatkan ke hadirat Allah SWT atas limpahan rahmat, taufik, dan karunia-Nya sehingga penulis dapat menyelesaikan laporan praktikum mata kuliah *Algoritma dan Pemrograman* dengan judul “Praktikum Pekan 3”.

Laporan ini tidak hanya dimaksudkan sebagai pemenuhan tugas praktikum, tetapi juga sebagai sarana untuk melatih kedisiplinan, ketelitian, serta pemahaman penulis terhadap konsep dasar pemrograman. Pada praktikum ini, penulis belajar mengenal berbagai Operator antara lain operator aritmatika, operator assignment, operator logika, dan operator relasional.

Penyusunan laporan ini tentu tidak terlepas dari berbagai keterbatasan. Oleh karena itu, penulis menyampaikan rasa terima kasih kepada dosen pengampu, asisten praktikum, dan teman-teman yang telah memberikan arahan, motivasi, serta bantuan selama proses pembelajaran dan penyusunan laporan berlangsung.

Akhir kata, semoga laporan sederhana ini dapat memberikan manfaat, tidak hanya bagi penulis sebagai pengalaman belajar, tetapi juga bagi para pembaca yang ingin memperdalam pemahaman mengenai dasar-dasar pemrograman dengan bahasa Java.

Padang, 19 September 2025

Rafikhul Ramadhan

## DAFTAR ISI

KATA PENGANTAR .....	i
DAFTAR ISI.....	ii
BAB I PENDAHULUAN .....	1
1.1    Latar Belakang .....	1
1.2    Tujuan .....	1
1.3    Manfaat .....	1
BAB II PEMBAHASAN .....	2
2.1    Operator Aritmatika .....	2
2.2    Operator Assignment .....	3
2.3    Operator Logika .....	5
2.4    Operator Rasional.....	6
BAB III KESIMPULAN.....	9
3.1    Ringkasan.....	9
3.2    Saran.....	9
DAFTAR PUSTAKA .....	10

# **BAB I**

## **PENDAHULUAN**

### **1.1 Latar Belakang**

Dalam bahasa pemrograman Java, operator berfungsi sebagai simbol yang digunakan untuk melakukan proses atau manipulasi terhadap data. Perannya sangat penting karena hampir setiap baris kode membutuhkan bentuk operasi tertentu, baik itu dalam perhitungan aritmatika, pengujian logika, maupun perbandingan relasional. Dengan menguasai penggunaan operator, seorang mahasiswa dapat menulis program yang lebih terstruktur, efektif, dan mampu menyelesaikan permasalahan dengan cara yang lebih optimal.

### **1.2 Tujuan**

Praktikum ini bertujuan untuk memperkenalkan berbagai macam operator yang tersedia dalam bahasa pemrograman Java, sekaligus memberikan pengalaman langsung dalam penerapannya pada program sederhana. Dengan demikian, peserta diharapkan memperoleh pemahaman yang lebih baik mengenai fungsi serta cara penggunaan operator dalam pemrograman.

### **1.3 Manfaat**

Manfaat yang dapat diperoleh dari praktikum ini meliputi:

1. Peserta dapat menguasai konsep dasar penggunaan operator.
2. Mahasiswa mampu menerapkan berbagai operator ke dalam program sederhana.
3. Mahasiswa memperoleh pengalaman praktik dalam menyusun kode yang melibatkan penggunaan operator.

## BAB II

### PEMBAHASAN

#### 2.1 Operator Aritmatika

Langkah kerja

1. Mendeklarasikan variabel A1, A2, dan hasil.
2. Meminta input dua bilangan dari pengguna.
3. Melakukan operasi aritmatika (tambah, kurang, kali, bagi, dan modulo).
4. Menampilkan hasil setiap operasi.

Contoh Syntax

```
package Pekan3;

import java.util.Scanner;

public class OperatorAritmatika {

    public static void main(String[] args) {
        // TODO Auto-generated method stub
        int A1;
        int A2;
        int hasil;

        Scanner keyboard = new Scanner(System.in);
        System.out.print("Input angka-1: ");
        A1 = keyboard.nextInt();
        System.out.print("Input angka-2: ");
        A2 = keyboard.nextInt();
        keyboard.close();
        System.out.println("operator Penjumlahan");
        hasil = A1 + A2; //penjumlahan
        System.out.println("Hasil = " + hasil);
        System.out.println("operator Pengurangan");
        hasil = A1 - A2; //pengurangan
        System.out.println("Hasil = " + hasil);
        System.out.println("operator Perkalian");
        hasil = A1 * A2; //perkalian
        System.out.println("Hasil = " + hasil);
        System.out.println("operator hasil bagi");
        hasil = A1 / A2; //pembagian
        System.out.println("Hasil = " + hasil);
        System.out.println("operator sisa bagi");
        hasil = A1 % A2; //Sisa Bagi
        System.out.println("Hasil = " + hasil);
    }
}
```

```

    }
}

```

### Hasil Output

```

Input angka-1: 6
Input angka-2: 3
operator Penjumlahan Hasil = 9
operator Pengurangan Hasil = 3
operator Perkalian Hasil = 24
operasi hasil bagi 3 Hasil = 2
operasi sisa bagi Hasil = 0

```

### Analisis

Dari hasil eksekusi, terlihat bahwa setiap operator aritmatika bekerja sesuai dengan fungsinya. Program ini membantu mahasiswa memahami cara dasar melakukan operasi matematika dalam Java.

## 2.2 Operator Assignment

### Langkah kerja

1. Mendeklarasikan variabel A1 dan A2.
2. Meminta input dua bilangan dari pengguna.
3. Menggunakan operator assignment untuk melakukan operasi dan menyimpan hasil ke A1.
4. Menampilkan hasil setiap operasi.

### Contoh Syntax

```

package Pekan3;
import java.util.Scanner;
public class OperatorAssignment {

    public static void main(String[] args) {
        // TODO Auto-generated method stub
        int A1;
        int A2;
        Scanner keyboard = new Scanner (System.in);
    }
}

```

```

System.out.print("Input angka-1: ");
A1 = keyboard.nextInt();//input 10
System.out.print("Input angka-2: ");
A2 = keyboard.nextInt();
keyboard.close();//input 5
System.out.println();
System.out.println("Assignment penambahan ");
A1 += A2;// penambahan, sekarang A1 = 15
System.out.println ("Penambahan: " + A1 + "\n");

System.out.println("Assignment pengurangan ");
A1 -= A2;// pengurangan, sekarang A1 = 10
System.out.println ("Pengurangan: " + A1 + "\n");

System.out.println("Assignment perkalian ");
A1 *= A2;// perkalian, sekarang A1 = 50
System.out.println("Perkalian: " + A1 + "\n");

System.out.println("Assignment hasil bagi");
A1 /= A2;// hasil bagi, sekarang A1 = 10
System.out.println("Pembagian : " + A1 + "\n");

System.out.println("Assignment sisa bagi ");
A1 %= A2;// sisa bagi, sekarang A1 = 0
System.out.println("Sisa Bagi: " + A1);
    }
}

```

#### Hasil Output

Input angka-1: 10

Input angka-2: 5

Assignment penambahan Penambahan : 15

Assignment pengurangan Pengurangan : 10

Assignment perkalian Perkalian : 50

Assignment hasil bagi Pembagian : 10

Assignment sisa bagi Sisa bagi: 0

## Analisis

Operator assignment mempermudah dalam memodifikasi nilai variabel dengan cara yang lebih ringkas. Mahasiswa dapat memahami bahwa operator ini menggabungkan operasi aritmatika dengan assignment.

### 2.3 Operator Logika

#### Langkah Kerja

1. Mendeklarasikan variabel boolean A1, A2, dan c.
2. Meminta input dua nilai boolean dari pengguna.
3. Melakukan operasi logika AND, OR, dan NOT.
4. Menampilkan hasil setiap operasi.

#### Contoh Syntax

```
package Pekan3;
import java.util.Scanner;
public class OperatorLogika {

    public static void main(String[] args) {
        // TODO Auto-generated method stub
        boolean A1;
        boolean A2;
        boolean C;

        Scanner keyboard = new Scanner(System.in);
        System.out.print("Input nilai boolean-1 (true / false):");
        A1 = keyboard.nextBoolean();
        System.out.print("Input nilai boolean-2 (true / false):");
        A2 = keyboard.nextBoolean();
        keyboard.close();

        System.out.println("A1 = " + A1);
        System.out.println("A2 = " + A2);

        System.out.println("Konjungsi");
        C = A1 && A2;
        System.out.println("true and false = " + C);

        System.out.println("Disjungsi");
        C = A1 || A2;
        System.out.println("true or false = " + C);

        System.out.println("Negasi");
        C = !A1;
        System.out.println("bukan true = " + C);
    }
}
```



```

    }
}

```

### Hasil Output

Input nilai boolean-1 (true / false):

true Input nilai boolean-2 (true / false):

true A1 = true A2 = true Konjungsi true and false = true Disjungsi true  
or false = true Negasi bukan true = false

### Analisi

Operator logika digunakan untuk menggabungkan atau memanipulasi nilai boolean. Hasil eksekusi menunjukkan bagaimana logika dasar bekerja dalam program

## 2.4 Operator Rasional

### Langkah Kerja

1. Mendeklarasikan variabel A1, A2, dan hasil.
2. Meminta input dua bilangan dari pengguna.
3. Membandingkan kedua bilangan menggunakan operator relasional.
4. Menampilkan hasil perbandingan

### Contoh Syntax

```

package Pekan3;
import java.util.Scanner;
public class OperatorRelasional {

    public static void main(String[] args) {

        int A1;
        int A2;
        boolean hasil;

        Scanner keyboard = new Scanner(System.in);
        System.out.print("Input angka-1: ");
    }
}

```

```

A1 = keyboard.nextInt();
System.out.print("Input angka-2: ");
A2 = keyboard.nextInt();
keyboard.close();
System.out.println();

hasil = A1 > A2; // apakah A1 lebih besar A2?
System.out.println("A1 > A2 = " + hasil);

hasil = A1 < A2; // apakah A1 lebih kecil A2?
System.out.println("A1 < A2 = " + hasil);

A2?
hasil = A1 >= A2; // apakah A1 lebih besar sama dengan
System.out.println("A1 >= A2 = " + hasil);

A2?
hasil = A1 <= A2; // apakah A1 lebih kecil sama dengan
System.out.println("A1 <= A2 = " + hasil);

hasil = A1 == A2; // apakah A1 sama dengan A2?
System.out.println("A1 == A2 = " + hasil);

hasil = A1 != A2; // apakah A1 tidak sama dengan A2
System.out.println("A1 != A2 = " + hasil + "\n");

hasil = A1 > A2; // apakah A1 lebih besar A2?
System.out.println(A1 + " > " + A2 + " = " + hasil);

hasil = A1 < A2; // apakah A1 lebih kecil A2?
System.out.println(A1 + " < " + A2 + " = " + hasil);

A2?
hasil = A1 >= A2; // apakah A1 lebih besar sama dengan
System.out.println(A1 + " >= " + A2 + " = " + hasil);

A2?
hasil = A1 <= A2; // apakah A1 lebih kecil sama dengan
System.out.println(A1 + " <= " + A2 + " = " + hasil);

hasil = A1 == A2; // apakah A1 sama dengan A2?
System.out.println(A1 + " == " + A2 + " = " + hasil);

hasil = A1 != A2; // apakah A1 tidak sama dengan A2
System.out.println(A1 + " != " + A2 + " = " + hasil);

}
}

```

### Hasil Output

Input angka-1: 2

Input angka-2: 2

$A1 > A2 = \text{false}$

$A1 < A2 = \text{false}$

$A1 \geq A2 = \text{true}$

$A1 \leq A2 = \text{true}$

$A1 == A2 = \text{true}$

$A1 != A2 = \text{false}$

### Analisis

Operator relasional digunakan untuk membandingkan nilai dua bilangan. Hasil eksekusi menunjukkan nilai true atau false sesuai kondisi yang diuji.

## **BAB III**

### **KESIMPULAN**

#### **3.1 Ringkasan**

Berdasarkan hasil praktikum:

1. Operator aritmetika (+, -, ×, ÷, %) berfungsi untuk melakukan perhitungan matematis pada variabel numerik, seperti penjumlahan, pengurangan, perkalian, pembagian, dan sisa hasil bagi.
2. Operator penugasan (=, +=, -=, \*=, ÷=) memungkinkan pemberian nilai sekaligus operasi langsung pada variabel untuk mempermudah penulisan kode.
3. Operator logika (&&, ||, !) dimanfaatkan untuk menggabungkan ekspresi boolean dalam pengambilan keputusan pada program.
4. Operator relasional (>, <, >=, <=, ==, !=) digunakan untuk membandingkan dua nilai dan menghasilkan output berupa nilai boolean (true/false).
5. Temuan praktikum ini selaras dengan teori Java yang menekankan pentingnya operator sebagai penghubung antara logika pemrograman dan penerapan konsep matematis maupun logis dalam program.

#### **3.2 Saran**

Saran dari praktikum ini adalah agar mahasiswa lebih banyak berlatih menggunakan berbagai jenis operator dalam program Java. Dengan latihan yang rutin, pemahaman konsep akan semakin kuat dan penerapan dalam kasus nyata akan lebih mudah.

## DAFTAR PUSTAKA

- [1] H. M. Deitel dan P. J. Deitel, Java: How to Program, 11th ed. New Jersey, NJ: Pearson, 2017.
- [2] J. Smith dan R. Brown, “Efficient memory management in Java Virtual Machine,” *International Journal of Computer Science*, vol. 12, no. 3, pp. 45–56, 2018.
- [3] M. Lee, “Design patterns for Java GUI applications,” dalam *Proceedings of the 10th International Conference on Software Engineering*, Jakarta, Indonesia, 2020, pp. 112–118.
- [4] Oracle, “The Java Tutorials,” 2023. [Daring]. Tersedia pada: <https://docs.oracle.com/javase/tutorial/>. [Diakses: 24-Sep-2025].
- [5] B. Eckel, *Thinking in Java*, 4th ed. Upper Saddle River, NJ: Prentice Hall, 2006.
- [6] S. Liang, *Introduction to Java Programming and Data Structures*, 12th ed. Boston, MA: Pearson, 2020.