

LAPORAN PRAKTIKUM
PEMROGRAMAN ALGORITMA DAN PEMROGRAMAN
“PENERAPAN OPERATOR DASAR DALAM PEMROGRAMAN JAVA”

disusun oleh
Hafizh Habibullah
2511531002

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

Asisten Praktikum: Muhammad Zaki Al Hafiz



DEPARTEMEN INFORMATIKA
FAKULTAS TEKNOLOGI INFORMASI
UNIVERSITAS ANDALAS
TAHUN 2025

KATA PENGANTAR

Puji syukur dipanjatkan ke hadirat Allah SWT. karena atas rahmat dan karunia-Nya, laporan praktikum ini dapat diselesaikan dengan baik. Laporan ini dibuat sebagai salah satu kewajiban dari kegiatan praktikum yang telah dilakukan, dengan penekanan utama pada materi tentang penerapan Operator Dasar dalam Bahasa Pemrograman Java.

Penulis mengucapkan terima kasih kepada Dosen dan Asisten Praktikum yang sudah membantu dalam proses pelaksanaan praktikum ini. Semoga laporan ini bermanfaat bagi penulis maupun pembacanya.

Padang, 26 September 2025

Hafizh Habibullah

DAFTAR ISI

KATA PENGANTAR.....	i
DAFTAR ISI.....	ii
DAFTAR LAMPIRAN	iii
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang.....	1
1.2 Tujuan.....	1
1.3 Manfaat.....	1
BAB II PEMBAHASAN.....	2
2.1 Program OperatorAritmatika.....	2
2.2 Program OperatorAssignment	4
2.3 Kode Program OperatorLogika	6
2.4 Kode Program OperatorRelasional.....	8
BAB III KESIMPULAN.....	10
3.1 Kesimpulan.....	10
3.2 Saran Pengembangan.....	10
DAFTAR PUSTAKA	11

DAFTAR LAMPIRAN

Gambar 2. 1 - Kode Program OperatorAritmatika.....	2
Gambar 2. 2 - Kode Program OperatorAssignment.....	4
Gambar 2. 3 - Kode Program OperatorLogika	6
Gambar 2. 4 - Kode Program OperatorRelasional	8
Tabel 2. 1 - Hasil Output dari OperatorAritmatika	3
Tabel 2. 2 - Hasil Output dari OperatorAssignment	5
Tabel 2. 3 - Hasil Output dari OperatorLogika	7

BAB I

PENDAHULUAN

1.1 Latar Belakang

Dalam pemrograman, operator merupakan elemen dasar yang tidak bisa dipisahkan. Operator berperan untuk melaksanakan berbagai proses seperti penghitungan aritmetika, penugasan nilai (*assignment*), pengambilan keputusan dengan logika, serta perbandingan melalui relasional. Penguasaan yang mendalam terhadap operator-operator ini akan sangat mendukung dalam merancang program yang efisien dan mudah dimengerti.

Dengan demikian, pembahasan mengenai penerapan operator aritmatika, penugasan, logika, dan relasional di Java merupakan langkah penting yang pertama dalam memahami dasar-dasar alur pemrograman.

1.2 Tujuan

Tujuan dari praktikum ini adalah:

1. Mempelajari penerapan operator aritmatika, penugasan, logika, dan relasional dalam bahasa pemrograman Java.
2. Mengaplikasikan operator-operator itu ke dalam program yang sederhana.
3. Mengetahui perbedaan peran dan output dari setiap tipe operator.
4. Mengembangkan keterampilan dalam menyusun program yang terstruktur dengan memanfaatkan operator dasar

1.3 Manfaat

Praktikum ini diharapkan memberikan beberapa manfaat, di antaranya:

1. Memberi penjelasan mendasar tentang fungsi operator dalam Java.
2. Mendukung penerapan operator guna menyelesaikan masalah logika dan matematika.
3. Menjadi dasar untuk mengembangkan program yang lebih rumit di tahap berikutnya.
4. Mengembangkan kemampuan analisis untuk menyelesaikan masalah dengan menggunakan bahasa pemrograman

BAB II

PEMBAHASAN

Dalam praktikum ini, eksperimen dilakukan dengan mengembangkan program sederhana menggunakan bahasa pemrograman Java. Program yang dikembangkan berfokus pada pemanfaatan berbagai jenis operator, yakni operator aritmetika, operator penugasan, operator logika, dan operator relasional.

Setiap program dibuat untuk memperlihatkan bagaimana setiap operator berfungsi dalam proses pengolahan data. Dengan demikian, perbedaan hasil dan penerapannya dapat dilihat sesuai dengan kebutuhan logika dalam pemrograman

2.1 Program OperatorAritmatika

```
1 package pekan3;
2 import java.util.Scanner;
3 public class OperatorAritmatika {
4     public static void main(String[] args) {
5         int A1;
6         int A2;
7         int hasil;
8         Scanner keyboard = new Scanner(System.in);
9         System.out.print("Input angka-1: ");
10        A1 = keyboard.nextInt();
11        System.out.print("Input angka-2: ");
12        A2 = keyboard.nextInt();
13        keyboard.close();
14        System.out.println("operator Penjumlahan");
15        hasil = A1 + A2; // penjumlahan
16        System.out.println("Hasil = " + hasil);
17        System.out.println("operator Pengurangan");
18        hasil = A1 - A2; // pengurangan
19        System.out.println("Hasil = " + hasil);
20        System.out.println("operator Perkalian");
21        hasil = A1 * A2; // perkalian
22        System.out.println("Hasil = " + hasil);
23        System.out.println("operator hasil bagi");
24        hasil = A1 / A2; // Pembagian
25        System.out.println("Hasil = " + hasil);
26        System.out.println("operator sisa bagi");
27        hasil = A1 % A2; // Sisa Bagi
28        System.out.println("Hasil = " + hasil);
29    }
30 }
```

Gambar 2. 1 - Kode Program OperatorAritmatika

Program **OperatorAritmatika** dirancang untuk menunjukkan penerapan berbagai operator aritmatika dalam Java. Program ini diawali dengan mendeklarasikan tiga variabel jenis int, yaitu **A1**, **A2**, dan **hasil**. Selanjutnya, aplikasi meminta pengguna untuk mengetik dua angka bulat melalui keyboard, kemudian menyimpannya dalam variabel **A1** dan **A2**.

Setelah data didapat, program menunjukkan hasil dari beberapa operasi matematika, yaitu penjumlahan, pengurangan, perkalian, pembagian, dan sisa bagi. Setiap hasil kalkulasi disimpan dalam variabel hasil dan ditampilkan menggunakan perintah **System.out.println()**. Operasi pembagian di sini menghasilkan angka bulat karena tipe data yang digunakan adalah **int**, sedangkan operasi sisa bagi menunjukkan nilai modulus dari kedua bilangan itu.

Dengan rencana ini, program **OperatorAritmatika** memperlihatkan cara operator dasar di Java dapat digunakan untuk memproses data angka, serta bagaimana hasil perhitungan tersebut dapat ditampilkan langsung kepada pengguna.

Hasil Output

Input angka-1: 2

Input angka-2: 5

operator Penjumlahan

Hasil = 7

operator Pengurangan

Hasil = -3

operator Perkalian

Hasil = 10

operator hasil bagi

Hasil = 0

operator sisa bagi

Hasil = 2

Tabel 2. 1 - Hasil Output dari OperatorAritmatika

2.2 Program OperatorAssignment

```
1 package pekan3;
2 import java.util.Scanner;
3 public class OperatorAssignment {
4     public static void main(String[] args) {
5         int A1;
6         int A2;
7         Scanner keyboard = new Scanner(System.in);
8         System.out.print("Input angka-1: ");
9         A1 = keyboard.nextInt();
10        System.out.print("Input angka-2: ");
11        A2 = keyboard.nextInt();
12        keyboard.close();
13        System.out.println("Assignment penambahan ");
14        A1 += A2; // penambahan, sekarang A1 = 15
15        System.out.println("Penambahan : " + A1);
16        System.out.println("Assignment pengurangan ");
17        A1 -= A2; // pengurangan, sekarang A1 = 10
18        System.out.println("Pengurangan : " + A1);
19        System.out.println("Assignment perkalian ");
20        A1 *= A2; // perkalian, sekarang A1 = 50
21        System.out.println("Perkalian : " + A1);
22        System.out.println("Assignment hasil bagi ");
23        A1 /= A2; // hasil bagi, sekarang A1 = 10
24        System.out.println("Pembagian : " + A1);
25        System.out.println("Assignment sisa bagi ");
26        A1 %= A2; // sisa bagi, sekarang A1 = 0
27        System.out.println("Sisa bagi : " + A1);
28    }
29 }
```

Gambar 2. 2 - Kode Program OperatorAssignment

Program **OperatorAssignment** dibuat untuk menunjukkan pemanfaatan berbagai operator assignment dalam bahasa pemrograman Java. Program ini dimulai dengan mendeklarasikan dua variabel yang bertipe int, yaitu **A1** dan **A2**. Selanjutnya, pengguna diminta untuk mengetikkan dua angka bulat melalui keyboard, yang selanjutnya disimpan ke dalam variabel tersebut.

Setelah nilai didapat, program melaksanakan serangkaian operasi penugasan, yaitu **+=**, **-=**, ***=**, **/=**, dan **%**. Setiap tindakan dilakukan dengan mengubah nilai variabel **A1** sesuai dengan nilai **A2**, kemudian hasil kalkulasi ditampilkan dengan perintah **System.out.println()**. Contohnya, dalam operasi **A1 += A2**, nilai **A1** akan ditambahkan dengan **A2** dan segera disimpan kembali dalam variabel **A1**. Prinsip yang sama diterapkan pada pengurangan, perkalian, pembagian, dan sisa bagi.

Dengan rancangan ini, program **OperatorAssignment** memperlihatkan cara penggunaan *Operator Assignment* untuk memperbarui nilai variabel secara langsung tanpa harus menuliskan kembali ekspresi secara keseluruhan. Hal ini menjadikan penulisan kode lebih singkat dan efisien, serta mempermudah dalam pengelolaan nilai variabel selama proses perhitungan.

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

Tabel 2. 2 - Hasil Output dari OperatorAssignment

2.3 Kode Program OperatorLogika

```
1 package pekan3;
2 import java.util.Scanner;
3 public class OperatorLogika {
4     public static void main(String[] args) {
5         boolean A1;
6         boolean A2;
7         boolean c;
8         Scanner keyboard = new Scanner(System.in);
9         System.out.print("Input nilai boolean-1 (true / false): ");
10        A1 = keyboard.nextBoolean();//input 10
11        System.out.print("Input nilai boolean-2 (true / false): ");
12        A2 = keyboard.nextBoolean();
13        keyboard.close();
14        System.out.println("A1 = " +A1);
15        System.out.println("A2 = " +A2);
16        System.out.println("Konjungsi" );
17        c= A1&&A2;
18        System.out.println("true and false = "+c );
19        System.out.println("Disjungsi" );
20        c= A1||A2;
21        System.out.println("true or false = "+c );
22        System.out.println("Negasi " );
23        c= !A1;
24        System.out.println("bukan true = "+c);
25    }
26 }
```

Gambar 2. 3 - Kode Program OperatorLogika

Program **OperatorLogika** dirancang untuk menunjukkan penggunaan operator logika dalam bahasa pemrograman Java. Program ini memakai tiga variabel dengan tipe *boolean*, yaitu **A1**, **A2**, dan **c**. Melalui penggunaan keyboard, pengguna diminta untuk memasukkan dua nilai *boolean* (benar atau salah) yang disimpan dalam variabel **A1** dan **A2**.

Setelah angka dimasukkan, program menunjukkan hasil dari berbagai operasi logika, yaitu **konjungsi (AND)**, **disjungsi (OR)**, dan **negasi (NOT)**. Operasi **A1 && A2** akan memberikan hasil true hanya jika kedua variabel bernilai true. Operasi **A1 || A2** menghasilkan nilai true jika salah satu atau kedua variabel bernilai benar. Operasi **!A1** berfungsi untuk membalik nilai logika A1, sehingga jika awalnya true, akan menjadi false, dan sebaliknya.

Dengan desain ini, program **OperatorLogika** menggambarkan cara kerja operator logika dalam pengolahan nilai *boolean*. Implementasi ini berguna untuk memahami dasar-dasar pengambilan keputusan dalam pemrograman, karena operator logika sering kali dipakai dalam struktur kontrol seperti kondisi dan pengulangan.

Hasil Output

Input nilai boolean-1 (true / false): true

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

A1 = true

A2 = false

Konjungsi

true and false = false

Disjungsi

true or false = true

Negasi

bukan true = false

Tabel 2. 3 - Hasil Output dari OperatorLogika

2.4 Kode Program OperatorRelasional

```
1 package pekan3;
2 import java.util.Scanner;
3 public class OperatorRelasional {
4     public static void main(String[] args) {
5         int A1;
6         int A2;
7         boolean hasil;
8         Scanner keyboard = new Scanner(System.in);
9         System.out.print("Input angka-1: ");
10        A1 = keyboard.nextInt();
11        System.out.print("Input angka-2: ");
12        A2 = keyboard.nextInt();
13        keyboard.close();
14        hasil = A1 > A2; // apakah A1 lebih besar A2?
15        System.out.println("A1 > A2 = " + hasil);
16        hasil = A1 < A2; // apakah A1 lebih kecil A2?
17        System.out.println("A1 < A2 = " + hasil);
18        hasil = A1 >= A2; // apakah A1 lebih besar samadengan A2?
19        System.out.println("A1 >= A2 = " + hasil);
20        hasil = A1 <= A2; // apakah A1 lebih kecil samadengan A2?
21        System.out.println("A1 <= A2 = " + hasil);
22        hasil = A1 == A2; // apakah A1 samadengan A2?
23        System.out.println("A1 == A2 = " + hasil);
24        hasil = A1 != A2; // apakah A1 tidak samadengan A2?
25        System.out.println("A1 != A2 = " + hasil);
26    }
27 }
```

Gambar 2. 4 - Kode Program OperatorRelasional

Program **OperatorRelasional** dimulai dengan mendeklarasikan dua variabel integer (**A1** dan **A2**) serta sebuah variabel *boolean* (**hasil**) untuk menyimpan hasil dari perbandingan. Setelah itu, program memanfaatkan **Scanner** untuk mengambil input dua bilangan dari pengguna. Nilai pertama disimpan dalam variabel **A1**, sedangkan nilai kedua disimpan dalam variabel **A2**.

Selanjutnya, program melaksanakan beberapa operasi perbandingan dengan menggunakan operator relasional, yaitu: **>**, **<**, **>=**, **<=**, **==**, dan **!=**. Setiap hasil dari perbandingan akan disimpan dalam variabel **hasil**, kemudian ditampilkan di layar dengan menggunakan **System.out.println()**. Dengan demikian, pengguna dapat segera mengetahui apakah pernyataan perbandingan itu bernilai **true** atau **false** berdasarkan input yang diberikan.

Hasil Output

Input angka-1: 10

Input angka-2: 5

A1 > A2 = true

A1 < A2 = false

A1 >= A2 = true

A1 <= A2 = false

A1 == A2 = false

A1 != A2 = true

Tabel 2. 4 - Hasil Output dari Operator Relasional

BAB III

KESIMPULAN

3.1 Kesimpulan

Secara umum, praktikum ini memberikan wawasan mendasar tentang tipe data, operator, dan struktur program yang sederhana dalam bahasa Java. Setiap program yang dikembangkan memiliki tujuan tertentu, meliputi penggunaan variabel berjenis char, operasi matematika, hingga penggunaan operator relasional untuk membandingkan angka. Melalui eksperimen ini, dapat disimpulkan bahwa prinsip-prinsip dasar tersebut adalah fondasi penting dalam pemrograman, karena akan menjadi landasan untuk mengembangkan program yang lebih rumit.

3.2 Saran Pengembangan

Untuk pengembangan di masa mendatang, praktikum bisa diperluas dengan menyertakan studi kasus nyata sehingga penerapan konsep menjadi lebih relevan. Penggunaan antarmuka grafis atau integrasi dengan struktur data sederhana juga dapat meningkatkan pemahaman mahasiswa dalam menerapkan konsep dasar pemrograman Java.

DAFTAR PUSTAKA

- [1] H. Schildt, *Java: The Complete Reference*, 11th ed. New York, NY, USA: McGraw-Hill, 2019.
- [2] J. Bloch, *Effective Java*, 3rd ed. Boston, MA, USA: Addison-Wesley, 2018.
- [3] Oracle, “The Java™ Tutorials,” Oracle, 2025. [Online]. Available: <https://docs.oracle.com/javase/tutorial/> [Diakses: 26-Sep-2025].
- [4] R. Fathansyah, *Basis Data*, Bandung, Indonesia: Informatika, 2015.
- [5] W. S. Nugroho, *Algoritma dan Pemrograman dengan Java*, Yogyakarta, Indonesia: Andi, 2016.