

LAPORAN PRAKTIKUM
ALGORITMA DAN PEMOGRAMAN
“CLASS SCANNER DAN MACAM-MACAM OPERATOR”

DISUSUN OLEH:

NOFRI ILHAM

2511531013

DOSEN PENGAMPU:

Dr. WAHYUDI, S.T, M.T

ASISTEN PRAKTIKUM:

MUHAMMAD ZAKI AL HAFIZ



DEPARTEMEN INFORMATIKA
FAKULTAS TEKNOLOGI INFORMASI
UNIVERSITAS ANDALAS

2025

KATA PENGANTAR

Assalamualaikum Warahmatullahi Wabarakatuh

Puji syukur penulis panjatkan ke hadirat Tuhan Yang Maha Esa atas segala rahmat dan karunia-Nya, sehingga laporan praktikum alghoritma pemograman dengan judul “**Class Scanner dan Macam-Macam Operator**” dapat diselesaikan dengan baik. Laporan ini disusun sebagai salah satu bentuk pelaksanaan praktikum pada mata kuliah Pemrograman Berorientasi Objek dengan bahasa pemrograman Java.

Pada praktikum kali ini, penulis mempelajari tentang penggunaan **Class Scanner** untuk menangani input dari pengguna serta pembahasan mengenai **macam-macam operator** yang ada dalam bahasa Java, seperti operator aritmatika, relasional, logika, dan penugasan. Pemahaman terhadap kedua hal ini sangat penting dalam pengembangan program yang dinamis dan interaktif.

Penulis ingin menyampaikan terima kasih yang sebesar-besarnya kepada dosen pengampu, asisten praktikum, serta semua pihak yang telah membantu selama proses praktikum berlangsung.

Dalam penyusunan laporan ini, penulis menyadari masih banyak kekurangan dan keterbatasan. Oleh karena itu, kritik dan saran yang membangun sangat diharapkan untuk perbaikan di masa yang akan datang.

Padang, 26 September 2025

Nofri Ilham

DAFTAR ISI

KATA PENGANTAR	1
BAB I PENDAHULUAN	1
I.1 Latar Belakang	1
I.2 Tujuan	2
I.3 Manfaat	2
BAB II PEMBAHASAN	1
II.1 Class Scanner	1
II.1.1 Pengertian Class Scanner	1
II.1.2 Penggunaan Class Scanner	1
II.1.3 Keunggulan Class Scanner	2
II.1.4 Metode Utama Class Scanner	2
II.2 Operator pada Java	3
II.2.1 Operator Aritmatika	3
II.2.2 Operator Penugasan	4
II.2.3 Operator Perbandingan	4
II.2.4 Operator Logika	4
II.3 Kode Program Pekan 3	4
II.3.1 Kode program Operator Aritmatika	5
II.3.2 Kode Program Operator Assignment	6
II.3.3 Kode Program Operator Relasional	6
II.3.4 Kode Program Operator Logika	7
BAB III PENUTUP	1
III.1 Kesimpulan	1
III.2 Saran	1
DAFTAR PUSTAKA	2

BAB I

PENDAHULUAN

I.1 Latar Belakang

Dalam dunia pemrograman, terutama pada bahasa Java, pengelolaan input dan operasi pada data merupakan dasar dari hampir semua aplikasi yang dikembangkan. Proses menerima masukan dari pengguna (input) dan melakukan manipulasi terhadap data yang dimasukkan sangat penting untuk keberhasilan dan interaktivitas suatu program. Salah satu cara untuk menangani input dari pengguna dalam Java adalah dengan menggunakan kelas Scanner. Kelas ini memfasilitasi pembacaan data yang dimasukkan oleh pengguna melalui berbagai jenis format, seperti teks, angka bulat, angka desimal, dan lain-lain.

Selain itu, dalam pemrograman, pengolahan data tidak dapat lepas dari penggunaan operator. Operator adalah simbol atau tanda yang digunakan untuk melakukan operasi tertentu pada data. Dalam Java, terdapat berbagai jenis operator, seperti operator aritmatika yang digunakan untuk melakukan perhitungan, operator perbandingan yang digunakan untuk membandingkan dua nilai, operator logika yang digunakan untuk menggabungkan kondisi, dan operator penugasan yang digunakan untuk memberikan nilai pada variabel. Pemahaman terhadap operator ini sangat penting agar program dapat melakukan proses perhitungan dan pengambilan keputusan yang tepat.

Dalam rangka memahami dan mempraktikkan penggunaan kelas Scanner dan berbagai macam operator, dilakukan praktikum ini. Praktikum ini bertujuan untuk memberikan pemahaman yang lebih mendalam mengenai cara menggunakan Scanner dalam memperoleh input dari pengguna serta bagaimana berbagai operator dapat digunakan untuk melakukan pengolahan data sesuai kebutuhan dalam pemrograman.

Melalui pemahaman dasar tentang penggunaan Scanner dan berbagai jenis operator, mahasiswa diharapkan dapat membangun logika pemrograman yang baik dan efisien. Materi ini menjadi langkah awal yang penting dalam menyusun

program interaktif serta menjadi fondasi untuk mempelajari konsep pemrograman Java yang lebih kompleks di tahap selanjutnya.

I.2 Tujuan

1. Mempelajari berbagai jenis operator dalam Java, seperti operator aritmatika, relasional, logika, dan penugasan.
2. Mengimplementasikan penggunaan Class Scanner dan operator dalam pembuatan program sederhana.
3. Melatih kemampuan dalam menyusun logika dasar pemrograman yang melibatkan input, proses, dan output.
4. Mengetahui dan memahami cara kerja kelas Scanner dalam membaca input dari pengguna pada bahasa pemrograman Java.

I.3 Manfaat

1. Mahasiswa dapat memahami dan menggunakan kelas Scanner dengan tepat dalam pengambilan input dari pengguna, sehingga program yang dibuat menjadi lebih interaktif dan dinamis.
2. Mahasiswa mampu mengenal, memahami, dan mengaplikasikan berbagai jenis operator dalam bahasa Java sesuai dengan fungsinya masing-masing, sehingga dapat mengolah data dengan benar dan efisien.
3. Melatih kemampuan analisis dan logika pemrograman mahasiswa dalam menyusun kode program yang melibatkan input, proses, dan output secara terstruktur dan sistematis.
4. Meningkatkan keterampilan dalam pemecahan masalah yang berkaitan dengan pengolahan data serta pengambilan keputusan dalam pemrograman.
5. Memberikan dasar yang kuat sebagai bekal dalam mempelajari konsep pemrograman Java yang lebih kompleks dan penerapannya dalam pembuatan aplikasi nyata di masa depan.

BAB II

PEMBAHASAN

II.1 Class Scanner

II.1.1 Pengertian Class Scanner

Di Java, Class Scanner terdapat dalam paket `java.util` dan digunakan untuk mendapatkan input untuk tipe primitif seperti `int`, `double`, dll., dan `string`. Kita dapat menggunakan kelas ini untuk membaca input dari pengguna atau berkas.

II.1.2 Penggunaan Class Scanner

1. Pertama, impor paket `java.util.Scanner` di bagian atas file program. Tanpa mengimpor paket ini, kita tidak dapat menggunakan kelas `Scanner`. Kita juga dapat mengimpor `java.util.*`. Dengan mengimpor paket ini, kita dapat menggunakan semua kelas yang ada dalam paket `util`.
2. Buat objek kelas `Scanner`. Di sini, "sc" adalah objek dari kelas `Scanner`. Kita dapat memberinya nama yang berbeda untuk memudahkan kita, seperti `in`, `var`, atau `obj`, dll.
3. Buat variabel dan gunakan objek kelas pemindai, panggil metode yang sesuai untuk mengambil nilai input.

Berikut contoh penggunaan Class Scanner

```

1 package pekan3;
2
3 import java.util.Scanner;
4
5 public class OperatorAritmatika {
6
7     public static void main(String[] args) {
8         int A1;
9         int A2;
10        int hasil;
11        Scanner keyboard = new Scanner (System.in);
12        System.out.print("input angka-1: ");
13        A1 = keyboard.nextInt();
14        System.out.print("input angka-2: ");
15        A2 = keyboard.nextInt();
16        keyboard.close();
17        System.out.println("operator penjumlahan");
18        hasil = A1 + A2; //penjumlahan
19        System.out.println("Hasil = " + hasil);
20        System.out.println("operator pengurangan");
21        hasil = A1 - A2; //pengurangan
22        System.out.println("Hasil = " + hasil);
23        System.out.println("operator perkalian");
24        hasil = A1 * A2; //perkalian
25        System.out.println("Hasil = " + hasil);
26        System.out.println("operator hasil bagi");
27        hasil = A1 / A2; //pembagian
28        System.out.println("Hasil = " + hasil);
29        System.out.println("operator sisa bagi");
30        hasil = A1 % A2; //pembagian
31        System.out.println("Hasil = " + hasil);
32        // TODO Auto-generated method stub
33    }
34
35 }
36

```

Kode Program 2.1

II.1.3 Keunggulan Class Scanner

1. Kemudahan dalam membaca input pengguna
2. Berguna untuk menginput dari sumber
3. Mendukung berbagai tipe data
4. Mengurangi potensi kesalahan pada input
5. Mudah digunakan dan dipahami

II.1.4 Metode Utama Class Scanner

Kelas pemindai membantu mengambil aliran input standar di Java. Oleh karena itu, kita memerlukan beberapa metode untuk mengekstrak data dari aliran tersebut. Metode yang digunakan untuk mengekstrak data dijelaskan di bawah ini:

Metode	Keterangan
nextBoolean().	Digunakan untuk membaca nilai Boolean
nextByte()	Digunakan untuk membaca nilai Byte
nextDouble()	Digunakan untuk membaca nilai Ganda
nextFloat()	Digunakan untuk membaca nilai Float
nextInt()	Digunakan untuk membaca nilai Int
nextLine()	Digunakan untuk membaca nilai Baris
nextLong()	Digunakan untuk membaca nilai Long
nextShort()	Digunakan untuk membaca nilai Pendek

II.2 Operator pada Java

Dalam Java, operator merupakan tulang punggung yang memfasilitasi interaksi antara variabel dan nilai, yang memungkinkan implementasi logika dan eksekusi berbagai operasi. Operator ini pada dasarnya adalah simbol khusus atau rangkaian simbol yang dikenali oleh kompiler Java untuk melakukan operasi spesifik pada satu, dua, atau bahkan tiga operan, yang dapat berupa variabel, literal, atau ekspresi.

Pada intinya, Java dirancang untuk menjadi bahasa pemrograman yang intuitif dan andal, cocok untuk berbagai macam aplikasi, mulai dari pengembangan web hingga aplikasi seluler dan sistem perusahaan berskala besar. Efisiensi dan keterbacaan kode Java sangat bergantung pada penggunaan operator yang tepat. Operator memungkinkan pengembang untuk memanipulasi data, membuat keputusan, dan mengontrol alur eksekusi dalam program, menjadikannya alat yang sangat diperlukan dalam perangkat programmer.

II.2.1 Operator Aritmatika

Operator ini digunakan untuk operasi matematika dasar seperti penjumlahan, pengurangan, perkalian, dan pembagian. Operator ini penting dalam aplikasi komputasi berat atau bahkan program sederhana yang memerlukan perhitungan aritmatika dasar.

Operator Fungsi	Contoh Hasil
+ Penjumlahan	7+8 15
- Pengurangan	8-7 1
* Perkalian	7*8 56
/ Pembagian	7*3 2
% Modulus(sisa bagi)	7%3 1

II.2.2 Operator Penugasan

Operator penugasan digunakan untuk menetapkan nilai ke variabel. Operator penugasan paling dasar adalah =, tetapi Java juga menawarkan operator penugasan gabungan yang menggabungkan operasi aritmatika dengan penugasan, membuat kode lebih ringkas dan mudah dibaca.

II.2.3 Operator Perbandingan

Juga dikenal sebagai operator relasional, operator ini membandingkan dua nilai atau ekspresi dan menentukan hubungan di antara keduanya, seperti persamaan, lebih besar dari, atau lebih kecil dari. Operator ini penting dalam pernyataan alur kontrol seperti if, for, dan whileloop.

Operator Fungsi	Contoh
== Sama Dengan	a==b
!= Tidak sama dengan	a!=b
> Lebih besar	a>b
< Lebih kecil	a= Lebih besar atau sama dengan	a>=b
<= Lebih kecil atau sama dengan	a<=b

II.2.4 Operator Logika

Operator logika bekerja dengan nilai boolean dan sangat penting dalam proses pengambilan keputusan dan logika kondisional. Operator ini membantu dalam membangun ekspresi boolean kompleks yang dapat bernilai salah satu true atau keduanya false.

Operator	Keterangan
&&	Logika AND: mengembalikan true ketika kedua kondisi benar.
	Logika ATAU: mengembalikan benar jika setidaknya satu kondisi benar.
!	Logika NOT: mengembalikan true ketika suatu kondisi salah dan sebaliknya

II.3 Kode Program Pekan 3

Pada praktikum pekan 3, terdapat empat program yang dibuat, yaitu kode program operator aritmatika, kode program operator penugasan, kode program operator perbandingan, dan kode program operator logika.

II.3.1 Kode program Operator Aritmatika

```

1 package pekan3;
2
3 import java.util.Scanner;
4
5 public class OperatorAritmatika {
6
7     public static void main(String[] args) {
8         int A1;
9         int A2;
10        int hasil;
11        Scanner keyboard = new Scanner (System.in);
12        System.out.print("input angka-1: ");
13        A1 = keyboard.nextInt();
14        System.out.print("input angka-2: ");
15        A2 = keyboard.nextInt();
16        keyboard.close();
17        System.out.println("operator penjumlahan");
18        hasil = A1 + A2; //penjumlahan
19        System.out.println("Hasil = " + hasil);
20        System.out.println("operator pengurangan");
21        hasil = A1 - A2; //pengurangan
22        System.out.println("Hasil = " + hasil);
23        System.out.println("operator perkalian");
24        hasil = A1 * A2; //perkalian
25        System.out.println("Hasil = " + hasil);
26        System.out.println("operator hasil bagi");
27        hasil = A1 / A2; //pembagian
28        System.out.println("Hasil = " + hasil);
29        System.out.println("operator sisa bagi");
30        hasil = A1 % A2; //pembagian
31        System.out.println("Hasil = " + hasil);
32        // TODO Auto-generated method stub
33    }
34 }

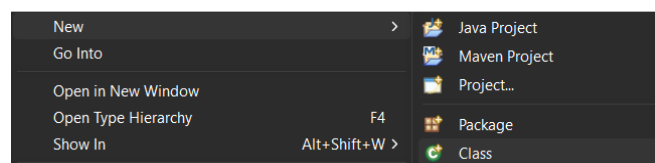
```

Kode Program 2.2

Kode yang pertama dibuat pada pekan ke 3 adalah contoh pendeklarasian variabel. Beberapa tipe data yang digunakan pada kode program ini yaitu char, string dan int.

Langkah pengerjaan kode program di atas adalah:

1. Buatlah package dan class yang baru untuk memulai menulis sebuah program.



2. Deklarasikan Variabel
 - a. int A1, A2; → untuk menyimpan angka input dari user.
 - b. Int hasil ; → untuk mehyimpan hasil operasi
3. Jalankan kode program
4. Cobalah untuk menginputkan data untuk mengecek jalan atau tidaknya program.

II.3.2 Kode Program Operator Assignment

```

1 package pekan3;
2 import java.util.Scanner;
3 public class OperatorAssignment {
4
5     public static void main(String[] args) {
6         int A1;
7         int A2;
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("Assignment penambahan");
15        A1 += A2; //penambahan, sekarang A1 = 15
16        System.out.println("penambahan = " + A1);
17        System.out.println("Assignment pengurangan");
18        A1 -= A2; //pengurangan, sekarang A1 = 10
19        System.out.println("pengurangan = " + A1);
20        System.out.println("Assignment perkalian");
21        A1 *= A2; //perkalian, sekarang A1 = 50
22        System.out.println("perkalian = " + A1);
23        System.out.println("Assignment hasil bagi");
24        A1 /= A2; //hasil bagi, sekarang A1 = 10
25        System.out.println("pembagian = " + A1);
26        System.out.println("Assignment sisa bagi");
27        A1 %= A2; //sisa bagi, sekarang A1 =
28        System.out.println("Sisa bagi = " + A1);
29        // TODO Auto-generated method stub
30
31    }
32
33 }
34

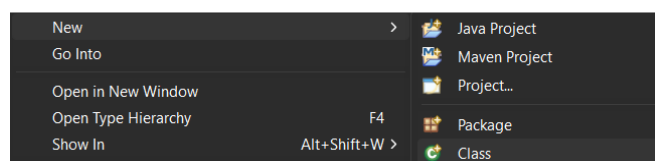
```

Kode Program 2.3

Kode program yang kedua mirip dengan kode program yang pertama, namun di kode program kedua nilai akan terus berubah sesuai running an program

Langkah pengerjaan kode diatas adalah sebagai berikut:

1. Buatlah package dan class yang baru untuk memulai menulis sebuah program



2. Deklarasikan Variabel
3. Jalankan kode program
4. Cobalah untuk menginputkan data untuk mengecek jalan atau tidaknya program.

II.3.3 Kode Program Operator Relasional

```

1 package pekan3;
2 import java.util.Scanner;
3 public class OperatorRelasional {
4
5     public static void main(String[] args) {
6         int A1;
7         int A2;
8         boolean hasil;
9         Scanner keyboard = new Scanner(System.in);
10        System.out.print("input angka-1: ");
11        A1 = keyboard.nextInt();
12        System.out.print("input angka-2: ");
13        A2 = keyboard.nextInt();
14        keyboard.close();
15        hasil = A1 > A2;
16        System.out.println("A1 > A2 = " + hasil);
17        hasil = A1 < A2;
18        System.out.println("A1 < A2 = " + hasil);
19        hasil = A1 >= A2;
20        System.out.println("A1 >= A2 = " + hasil);
21        hasil = A1 <= A2;
22        System.out.println("A1 <= A2 = " + hasil);
23        hasil = A1 == A2;
24        System.out.println("A1 == A2 = " + hasil);
25        hasil = A1 != A2;
26        System.out.println("A1 != A2 = " + hasil);
27        // TODO Auto-generated method stub
28    }
29
30 }
31 }

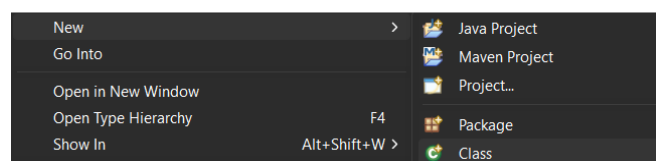
```

Kode Program 2.4

Pada kode program ketiga yaitu sebuah kode program yang menggunakan operator relasional. Sebuah kode program yang akan menghasilkan nilai true or false.

Langkah pengerjaan kode diatas adalah sebagai berikut:

1. Buatlah package dan class yang baru untuk memulai menulis sebuah program



2. Deklarasikan Variabel
3. Jalankan kode program ,serta cek apakah ada kesalahan
4. Cobalah untuk menginputkan data agar kodingan berjalan

II.3.4 Kode Program Operator Logika

```

1 package pekan3;
2 import java.util.Scanner;
3 public class OperatorRelasional {
4
5     public static void main(String[] args) {
6         int A1;
7         int A2;
8         boolean hasil;
9         Scanner keyboard = new Scanner(System.in);
10        System.out.print("input angka-1: ");
11        A1 = keyboard.nextInt();
12        System.out.print("input angka-2: ");
13        A2 = keyboard.nextInt();
14        keyboard.close();
15        hasil = A1 > A2;
16        System.out.println("A1 > A2 = " + hasil);
17        hasil = A1 < A2;
18        System.out.println("A1 < A2 = " + hasil);
19        hasil = A1 >= A2;
20        System.out.println("A1 >= A2 = " + hasil);
21        hasil = A1 <= A2;
22        System.out.println("A1 <= A2 = " + hasil);
23        hasil = A1 == A2;
24        System.out.println("A1 == A2 = " + hasil);
25        hasil = A1 != A2;
26        System.out.println("A1 != A2 = " + hasil);
27        // TODO Auto-generated method stub
28    }
29
30 }
31 }

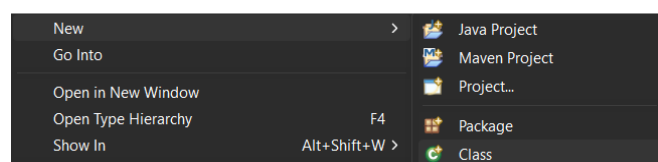
```

Kode Program 2.5

Kode program terakhir pada pekan ke 3 adalah program yang menggunakan operator logika. Operator logika nantinya akan memeriksa sebuah nilai dan nantinya akan menghasilkan nilai true or false.

Langkah pengerjaan kode diatas adalah sebagai berikut:

1. Buatlah package dan class yang baru untuk memulai menulis sebuah program



2. Deklarasikan Variabel
3. Jalankan kode program ,serta cek apakah ada kesalahan
4. Cobalah untuk menginputkan data agar kodingan berjalan

BAB III PENUTUP

III.1 Kesimpulan

Berdasarkan hasil praktikum yang telah dilaksanakan dengan topik *Class Scanner dan Macam-Macam Operator* pada bahasa pemrograman Java, dapat diambil beberapa kesimpulan sebagai berikut:

1. **Class Scanner** berperan penting dalam membaca masukan dari pengguna dengan berbagai tipe data, seperti int, double, float, dan String, sehingga memungkinkan program menjadi lebih interaktif dan dinamis.
2. **Operator dalam Java** terdiri dari beberapa jenis, yaitu operator aritmatika, penugasan, perbandingan, dan logika, yang masing-masing memiliki fungsi untuk memproses data serta mendukung pengambilan keputusan dalam program.
3. Melalui implementasi program sederhana menggunakan *Scanner* dan operator, mahasiswa memperoleh pemahaman mengenai konsep dasar alur pemrograman yang meliputi input–proses–output.
4. Praktikum ini juga memperkuat keterampilan mahasiswa dalam membangun logika pemrograman serta memberikan dasar yang bermanfaat untuk mempelajari konsep Java yang lebih kompleks pada tahap selanjutnya.

III.2 Saran

Adapun saran yang dapat diberikan terkait pelaksanaan praktikum ini adalah sebagai berikut:

1. Pada praktikum selanjutnya sebaiknya diberikan **contoh kasus yang lebih bervariasi dan mendekati aplikasi nyata**, sehingga pemahaman mahasiswa terhadap penggunaan *Scanner* dan operator menjadi lebih komprehensif.
2. Mahasiswa diharapkan dapat **lebih sering berlatih membuat program secara mandiri** dengan menggabungkan beberapa jenis operator sekaligus, sehingga kemampuan logika dan keterampilan pemecahan masalah semakin terasah.
3. Dokumentasi kode program sebaiknya dilengkapi dengan **komentar yang jelas dan sistematis**, agar memudahkan pembaca dalam memahami maksud dari setiap bagian kode.

DAFTAR PUSTAKA

[1] GeeksforGeeks, “Scanner Class in Java,”. [Daring]. Tersedia pada:
<https://www.geeksforgeeks.org/java/scanner-class-in-java/> [Diakses:
26-Sep-2025]

[2] Divisidev, “Tutorial Java 7 : Jenis Operator Part 2,”. [Daring]. Tersedia pada:
<https://divisidev.com/post/tutorial-java-7-jenis-operator-part-2> [Diakses:
27-Sep-2025]