**LAPORAN PRAKTIKUM 3**



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

Nama: Muhammad Rayhan Anugrah  
NIM: 231153319

**FAKULTAS TEKNOLOGI INFORMASI**

**DEPARTEMEN INFORMATIKA**

**UNIVERSITAS ANDALAS**

**2024**

1. **Pendahuluan**
2. Tujuan Praktikum

 Memahami penggunaan operator dasar dalam pemrograman Java, seperti operator aritmatika, assignment, logika, dan relasional.

 Mengenal implementasi operator aritmatika untuk memanipulasi data numerik.

 Mengaplikasikan operator assignment untuk melakukan perubahan nilai variabel secara efektif.

 Memahami logika dasar dalam pengambilan keputusan menggunakan operator logika.

 Menggunakan operator relasional untuk melakukan perbandingan data.

1. Kajian Teori
2. Operator Aritmatika

Operator aritmatika adalah operator yang digunakan untuk melakukan operasi matematika dasar pada data numerik, seperti penjumlahan (+), pengurangan (-), perkalian (\*), pembagian (/), dan sisa bagi atau modulus (%). Operator ini memungkinkan pemrogram untuk melakukan perhitungan dan manipulasi angka dalam program. Misalnya, a + b akan menjumlahkan nilai a dan b.

1. Operator Assignment

Operator assignment digunakan untuk memberikan nilai ke suatu variabel. Bentuk dasar operator ini adalah = yang langsung memberikan nilai ke variabel. Selain itu, terdapat operator assignment gabungan, seperti +=, -=, \*=, /=, dan %=, yang memungkinkan pemrogram untuk melakukan operasi aritmatika sekaligus mengubah nilai variabel. Sebagai contoh, a += b setara dengan a = a + b, yang berarti nilai a akan bertambah sebesar b.

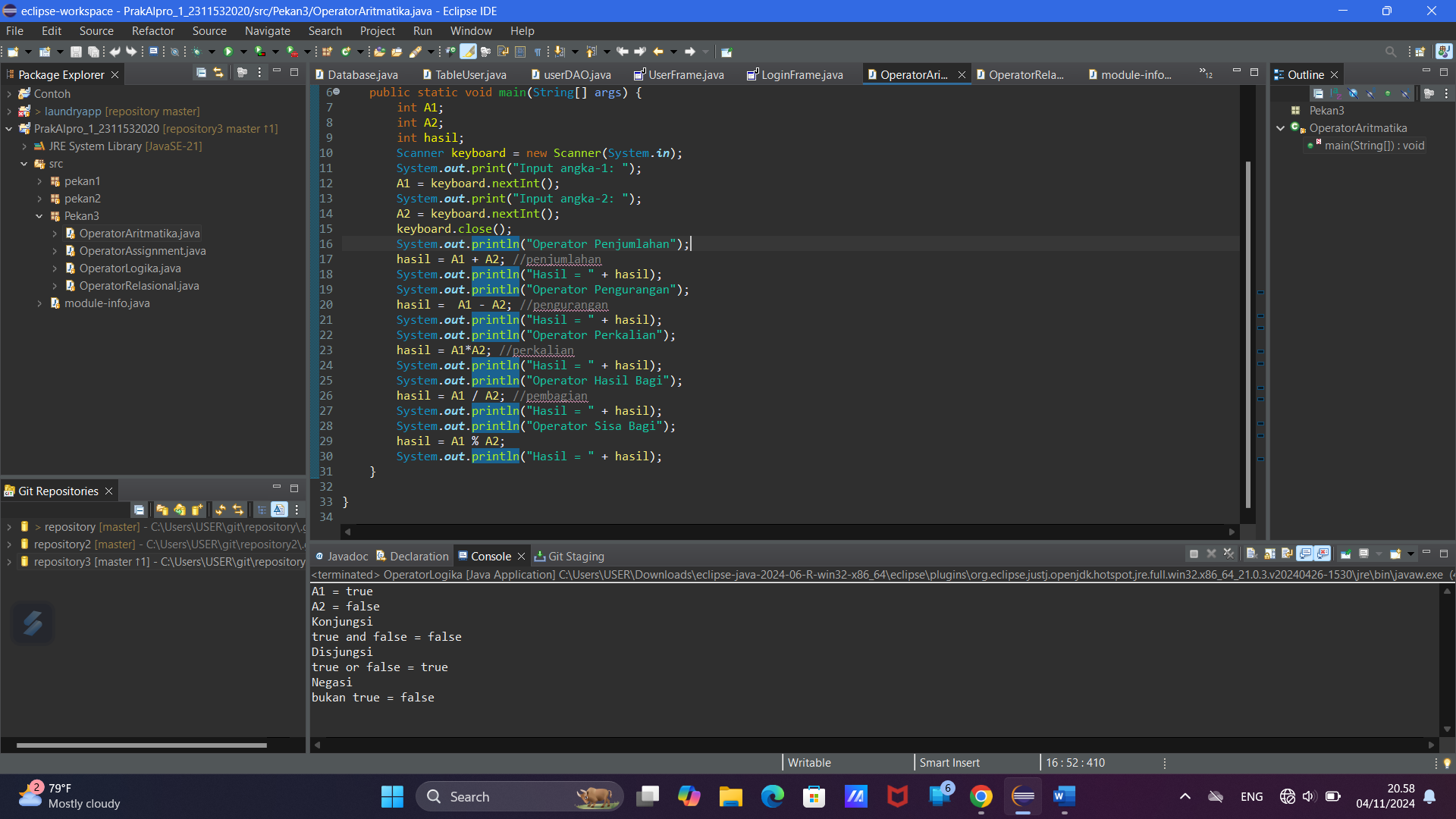
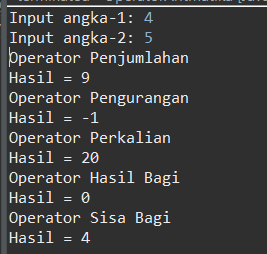
1. Operator Logika

Operator logika digunakan dalam pernyataan kondisi untuk menggabungkan beberapa kondisi boolean. Terdapat tiga operator logika utama, yaitu AND (&&), OR (||), dan NOT (!). Operator AND menghasilkan nilai true jika semua kondisi bernilai true, sedangkan OR menghasilkan true jika salah satu kondisi bernilai true. Operator NOT digunakan untuk membalik nilai boolean; jika suatu kondisi true, NOT akan mengubahnya menjadi false, dan sebaliknya. Operator logika sering digunakan dalam pengambilan keputusan di dalam program.

1. Operator Relasional

Operator relasional digunakan untuk membandingkan dua nilai. Operator ini mencakup >, <, >=, <=, ==, dan !=, yang masing-masing berarti lebih besar, lebih kecil, lebih besar atau sama dengan, lebih kecil atau sama dengan, sama dengan, dan tidak sama dengan. Hasil dari operasi ini adalah nilai boolean true atau false, tergantung pada apakah kondisi perbandingan terpenuhi. Operator relasional sangat penting dalam pengendalian alur program, terutama saat digunakan dalam kondisi if atau perulangan.

1. **Pembahasan**
2. Class OperasiAritmatika



Class ini dibuat untuk melakukan operasi aritmatika sederhana seperti penjumlahan, pengurangan, perkalian, pembagian, dan sisa bagi yang mana menggunakan dua angka yang dimasukkan oleh pengguna.

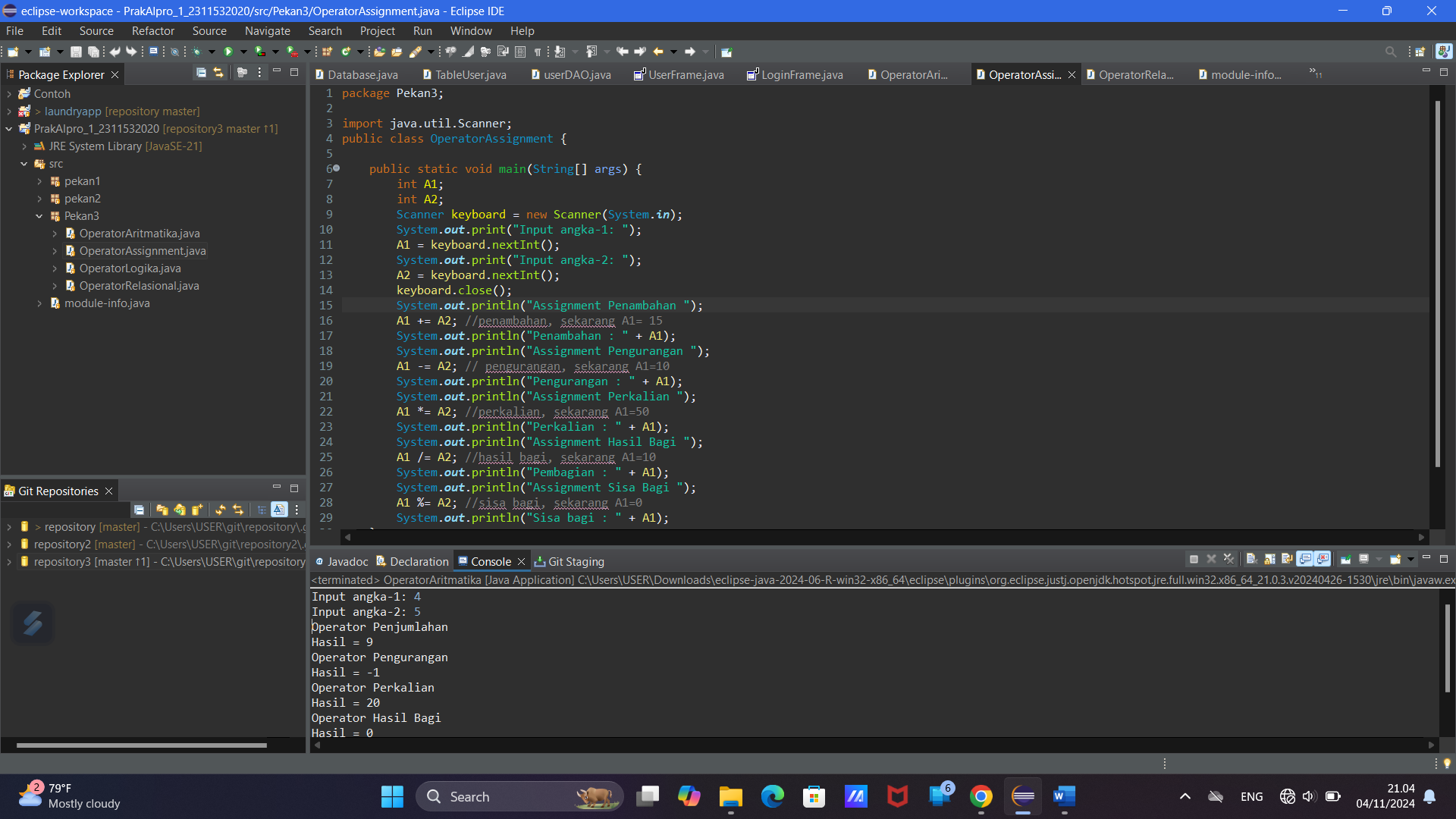
Di awal kode, terdapat deklarasi tiga variabel bertipe int, yaitu A1, A2, dan hasil. Variabel A1 dan A2 berfungsi untuk menyimpan angka yang dimasukkan oleh pengguna, sedangkan variabel “hasil” akan menyimpan hasil dari setiap operasi aritmatika yang dilakukan.

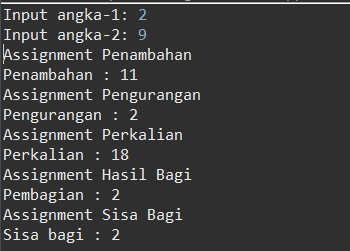
Selanjutnya, program membuat objek Scanner bernama keyboard yang digunakan untuk menerima input dari pengguna melalui keyboard. Kemudian, program meminta pengguna untuk memasukkan dua input berupa angka, yaitu A1 dan A2, lalu dilanjutkan dengan menggunakan keyboard.nextInt() yang akan menyimpan input tersebut dalam variabel A1 dan A2. Setelah input diterima, Scanner tersebut ditutup dengan keyboard.close() untuk menghemat sumber daya.

Setelah memperoleh dua angka dari pengguna, program melanjutkan ke bagian operasi aritmatika. Pertama, program melakukan operasi penjumlahan antara A1 dan A2 dengan A1 + A2, kemudian hasilnya disimpan dalam variabel hasil dan ditampilkan dengan System.out.println. Operasi ini diikuti dengan operasi pengurangan, di mana A1 - A2 dilakukan dan hasilnya juga disimpan dalam hasil, kemudian ditampilkan. Setelah pengurangan, program melakukan operasi perkalian dengan A1 \* A2, dan hasilnya disimpan dalam hasil serta ditampilkan.

Kemudian, program melakukan operasi pembagian menggunakan A1 / A2, dan hasilnya disimpan dalam hasil serta ditampilkan. Terakhir, program melakukan operasi sisa bagi dengan A1 % A2, yang berfungsi untuk mendapatkan sisa dari pembagian A1 oleh A2. Hasil dari operasi ini disimpan dalam hasil dan ditampilkan sebagai output.

1. Class OperatorAssignment





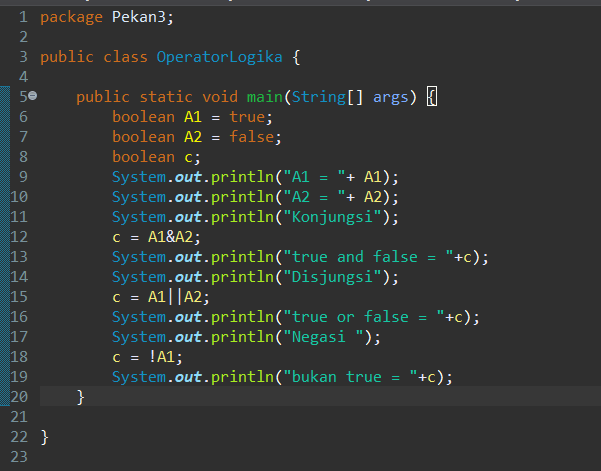
Class `OperatorAssignment` dibuat untuk memperlihatkan cara kerja operator assignment dalam operasi aritmatika di Java, yaitu operator `+=`, `-=`, `\*=`, `/=`, dan `%=`. Program ini meminta pengguna untuk memasukkan dua angka, kemudian menerapkan berbagai operasi assignment pada angka-angka tersebut dan menampilkan hasilnya.

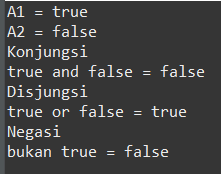
Pada awal program, terdapat deklarasi dua variabel bertipe `int`, yaitu `A1` dan `A2`, yang akan digunakan untuk menyimpan angka yang dimasukkan oleh pengguna. Selanjutnya, objek `Scanner` dengan nama `keyboard` dibuat untuk menerima input dari pengguna. Program meminta pengguna untuk memasukkan dua angka melalui keyboard, yang kemudian disimpan ke dalam variabel `A1` dan `A2`. Setelah input selesai diterima, `Scanner` ditutup menggunakan `keyboard.close()` untuk menghemat sumber daya.

Bagian berikutnya dari program adalah penerapan berbagai operasi assignment. Pertama, program melakukan operasi Assignment Penambahan dengan `A1 += A2`. Operator `+=` menambahkan nilai `A2` ke `A1` dan menyimpan hasilnya kembali ke `A1`. Hasil penambahan ini kemudian ditampilkan ke layar. Selanjutnya, program melakukan operasi Assignment Pengurangan dengan `A1 -= A2`, yang mengurangi nilai `A2` dari `A1` dan menyimpan hasilnya kembali ke `A1`. Hasil pengurangan ini juga ditampilkan ke layar.

Setelah itu, program melakukan Assignment Perkalian dengan `A1 \*= A2`, yang mengalikan nilai `A1` dengan `A2` dan menyimpan hasilnya kembali ke `A1`. Hasil perkalian ini ditampilkan ke layar. Kemudian, program melakukan Assignment Hasil Bagi dengan `A1 /= A2`, yang membagi nilai `A1` dengan `A2` dan menyimpan hasilnya kembali ke `A1`. Hasil pembagian ini juga ditampilkan. Terakhir, program melakukan Assignment Sisa Bagi dengan `A1 %= A2`, yang menghitung sisa dari pembagian `A1` oleh `A2` dan menyimpan hasilnya kembali ke `A1`. Hasil sisa bagi ini ditampilkan sebagai output.

Secara keseluruhan, class `OperatorAssignment` ini memperlihatkan bagaimana operator assignment dapat digunakan untuk memperbarui nilai suatu variabel dengan melakukan operasi aritmatika sekaligus. Penggunaan operator-operator ini membuat kode lebih ringkas dan efisien dibandingkan menuliskan operasi aritmatika secara terpisah

1. Class OperatorLogika



Class OperatorLogika digunakan untuk mendemonstrasikan penggunaan operator logika di Java, yaitu operator AND (&), OR (||), dan NOT (!). Operator-operator ini digunakan untuk melakukan operasi pada nilai-nilai boolean, yang mana dalam contoh ini diwakili oleh dua variabel A1 dan A2.

Di awal program, terdapat deklarasi dua variabel bertipe boolean, yaitu A1 dan A2, yang masing-masing diinisialisasi dengan nilai true dan false. Selain itu, ada variabel c yang digunakan untuk menyimpan hasil dari setiap operasi logika. Program pertama kali menampilkan nilai A1 dan A2 untuk menunjukkan kondisi awal dari kedua variabel tersebut.

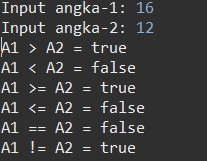
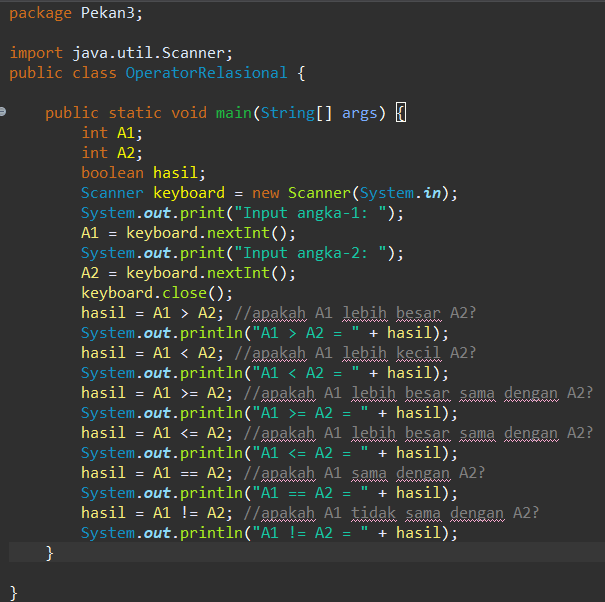
Selanjutnya, program menampilkan operasi logika Konjungsi (AND) menggunakan A1 & A2. Operator & akan menghasilkan true hanya jika kedua operandnya bernilai true. Karena dalam contoh ini A1 bernilai true dan A2 bernilai false, hasil dari A1 & A2 akan false, yang kemudian disimpan dalam c dan ditampilkan ke layar.

Kemudian, program melakukan operasi Disjungsi (OR) menggunakan A1 || A2. Operator || akan menghasilkan true jika salah satu atau kedua operandnya bernilai true. Karena A1 bernilai true dan A2 bernilai false, hasil dari A1 || A2 akan true, yang disimpan dalam c dan ditampilkan.

Terakhir, program menunjukkan operasi Negasi (NOT) menggunakan !A1. Operator ! membalikkan nilai boolean dari operandnya, sehingga jika operandnya true, hasilnya akan false, dan sebaliknya. Karena A1 bernilai true, maka !A1 akan menghasilkan false. Hasil ini disimpan dalam c dan ditampilkan.

Secara keseluruhan, class OperatorLogika ini menunjukkan cara kerja dari tiga operator logika dasar di Java, yaitu AND, OR, dan NOT, dengan contoh praktis yang melibatkan nilai-nilai boolean. Program ini membantu dalam memahami bagaimana kombinasi dari nilai true dan false akan mempengaruhi hasil dari operasi logika tersebut, yang sangat penting dalam pemrograman, terutama saat bekerja dengan kondisi dan pernyataan kontrol.

1. Class OperatorRelasional



Class OperatorRelasional bertujuan untuk memperlihatkan cara kerja operator relasional di Java, yaitu operator yang digunakan untuk membandingkan dua nilai dan menghasilkan hasil dalam bentuk boolean (true atau false). Program ini akan meminta pengguna untuk memasukkan dua angka, kemudian melakukan berbagai perbandingan antara kedua angka tersebut dan menampilkan hasilnya.

Pada awal program, dua variabel bertipe int, yaitu A1 dan A2, dideklarasikan. Selain itu, variabel hasil bertipe boolean digunakan untuk menyimpan hasil dari setiap perbandingan. Objek Scanner bernama keyboard dibuat untuk menerima input angka dari pengguna. Program meminta pengguna untuk memasukkan dua angka yang akan disimpan di A1 dan A2, kemudian menutup Scanner dengan keyboard.close() setelah input selesai untuk menghemat sumber daya.

Bagian berikutnya dari program adalah penerapan berbagai operator relasional. Pertama, program melakukan perbandingan lebih besar (>) dengan A1 > A2. Operator > mengecek apakah A1 lebih besar dari A2. Hasil perbandingan ini akan disimpan dalam hasil dan ditampilkan ke layar. Selanjutnya, program melakukan perbandingan lebih kecil (<) dengan A1 < A2, untuk memeriksa apakah A1 lebih kecil dari A2, dan hasilnya ditampilkan.

Program kemudian melanjutkan dengan perbandingan lebih besar atau sama dengan (>=) menggunakan A1 >= A2, yang akan menghasilkan true jika A1 lebih besar atau sama dengan A2. Hasilnya disimpan dalam hasil dan ditampilkan. Begitu pula, program melakukan perbandingan lebih kecil atau sama dengan (<=) dengan A1 <= A2 untuk mengecek apakah A1 lebih kecil atau sama dengan A2, dan hasilnya juga ditampilkan.

Selanjutnya, program mengecek kesamaan antara A1 dan A2 menggunakan sama dengan (==) dengan A1 == A2. Operator == akan menghasilkan true jika A1 sama dengan A2. Hasil ini disimpan dalam hasil dan ditampilkan. Terakhir, program melakukan perbandingan tidak sama (!=) menggunakan A1 != A2, yang akan menghasilkan true jika A1 tidak sama dengan A2, dan hasilnya ditampilkan.

Secara keseluruhan, class OperatorRelasional ini memberikan gambaran tentang bagaimana operator relasional bekerja di Java. Operator-operator ini sering digunakan dalam pernyataan kondisi untuk mengontrol aliran program berdasarkan perbandingan dua nilai, yang sangat penting dalam pengambilan keputusan di dalam program. Program ini memudahkan pemahaman konsep perbandingan nilai-nilai numerik dan bagaimana hasilnya bisa mempengaruhi proses logika dalam pemrograman.

1. **Kesimpulan**

Dalam praktikum ini, pemahaman mengenai operator aritmatika, assignment, logika, dan relasional telah diaplikasikan melalui berbagai contoh implementasi di dalam bahasa pemrograman Java. Setiap jenis operator memainkan peran yang berbeda dalam membangun logika dan struktur program yang baik, sekaligus memperkuat kemampuan dalam menyusun algoritma yang efisien. Melalui percobaan ini, beberapa kesimpulan utama dapat ditarik terkait pemahaman terhadap masing-masing operator.

Pertama, operator aritmatika sangat penting untuk melakukan manipulasi data numerik. Operator seperti penjumlahan (+), pengurangan (-), perkalian (\*), pembagian (/), dan modulus (%) tidak hanya bermanfaat dalam operasi matematis, tetapi juga berguna dalam berbagai skenario pemrograman, seperti perhitungan looping atau pemrosesan data yang melibatkan angka. Penggunaan operator ini memperlihatkan bagaimana operasi dasar matematika dapat diterapkan dalam program untuk menghasilkan output yang diinginkan. Selain itu, percobaan ini memperkuat pemahaman bahwa operator aritmatika juga dapat dipadukan dengan variabel bertipe data numerik, sehingga memungkinkan manipulasi angka yang lebih dinamis.

Selanjutnya, operator assignment merupakan dasar dalam pemberian nilai pada variabel. Selain = sebagai bentuk dasar dari assignment, penggunaan operator assignment gabungan, seperti +=, -=, \*=, /=, dan %=, memperlihatkan kepraktisan dalam melakukan operasi yang lebih ringkas. Pemahaman ini berguna untuk efisiensi kode, karena programmer dapat menghindari penulisan kode berulang yang panjang dengan menggunakan operator assignment gabungan. Praktikum ini juga memperlihatkan bahwa operator assignment tidak hanya berguna untuk inisialisasi, tetapi juga dalam meng-update nilai variabel berdasarkan kondisi tertentu di dalam program.

Di sisi lain, operator logika berfungsi sebagai penghubung antara pernyataan kondisi yang lebih kompleks. Penggunaan operator AND (&&), OR (||), dan NOT (!) membantu dalam menentukan hasil dari beberapa kondisi yang digabungkan. Melalui praktikum ini, dapat dipahami bahwa operator logika memungkinkan program membuat keputusan berdasarkan lebih dari satu kondisi, sehingga menciptakan alur logika yang lebih kaya. Operator logika sangat penting dalam pengambilan keputusan, terutama saat digunakan dalam pernyataan if-else atau perulangan yang membutuhkan kondisi tertentu untuk dieksekusi. Percobaan dengan operator logika juga memperlihatkan bagaimana kombinasi nilai boolean true dan false dalam berbagai kondisi dapat menghasilkan hasil yang berbeda.

Terakhir, operator relasional digunakan untuk membandingkan dua nilai dan menentukan apakah suatu kondisi terpenuhi atau tidak. Operator ini penting dalam evaluasi kondisi yang mengarahkan alur program. Misalnya, operator >, <, >=, <=, ==, dan != memungkinkan program untuk mengeksekusi tindakan tertentu berdasarkan hasil perbandingan antara dua nilai. Praktikum ini menunjukkan bahwa operator relasional sangat berguna dalam kondisi kontrol, seperti pernyataan if, yang memberikan fleksibilitas dalam menjalankan kode sesuai dengan kondisi tertentu. Dengan memahami cara kerja operator relasional, pemrogram dapat dengan mudah menyusun logika yang membutuhkan perbandingan nilai.

Secara keseluruhan, praktikum ini menunjukkan bahwa pemahaman mendalam terhadap operator-operator dasar dalam Java sangatlah penting dalam mengembangkan program yang efektif dan efisien. Masing-masing operator berfungsi untuk membantu dalam menyusun logika, mengelola data, dan mengatur alur kontrol dalam program. Penguasaan terhadap operator-operator ini akan menjadi fondasi yang kuat bagi mahasiswa dalam menyusun program yang lebih kompleks di masa mendatang. Dengan latihan berkelanjutan, mahasiswa akan semakin terampil dalam memilih dan menggunakan operator yang tepat sesuai kebutuhan pemrograman mereka. Pengalaman ini juga mengajarkan pentingnya efisiensi dalam penulisan kode, di mana setiap operator memiliki peran dalam menghemat waktu komputasi dan menyederhanakan struktur program.