Tugas I Makassar, 21 Februari 2025

**PRAKTIKUM**

**PEMROGRAMAN BERORINTASI OBJEK**



|  |  |
| --- | --- |
| Nama | : Sitti Lutfia |
| Stambuk | : 13020230255 |
| Dosen | : Mardiyah Hasnawi, S.Kom., M.T., MTA. |

**PROGRAM STUDI TEKNIK INFORMATIKA**

**FAKULTAS ILMU KOMPUTER**

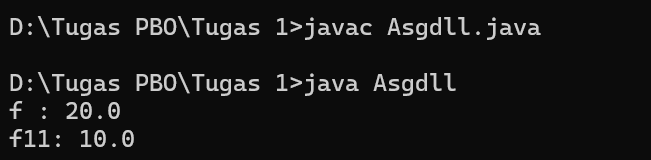
**UNIVERSITAS MUSLIM INDONESIA**

**MAKASSAR**

**2025**

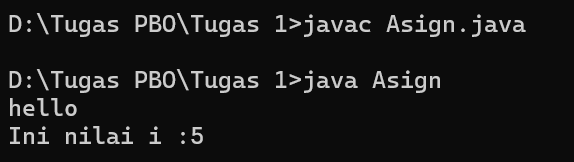
Tugas Program

1. Agsdll



Penjelasannya: Program di atas adalah program Java sederhana yang mendeklarasikan dan menginisialisasi dua variabel bertipe float dan double, kemudian mencetak nilai kedua variabel tersebut ke layar. Program diawali dengan deklarasi kelas Asgdll dan metode main(), yang merupakan titik awal eksekusi program. Di dalam metode main(), variabel f bertipe float diinisialisasi dengan nilai 20.0f, di mana akhiran f menunjukkan bahwa nilai tersebut adalah tipe float, bukan double. Selanjutnya, variabel fll bertipe double dideklarasikan tetapi belum langsung diberikan nilai. Kemudian, fll diinisialisasi dengan 10.0f, yang sebenarnya bertipe float, namun secara otomatis dikonversi ke double karena double memiliki presisi lebih tinggi dibandingkan float. Akhirnya, perintah System.out.println digunakan untuk mencetak nilai dari variabel f dan fll, dengan hasil ditampilkan dalam format multiline karena adanya karakter escape \n, sehingga nilai variabel ditampilkan pada baris yang berbeda.

1. Asign

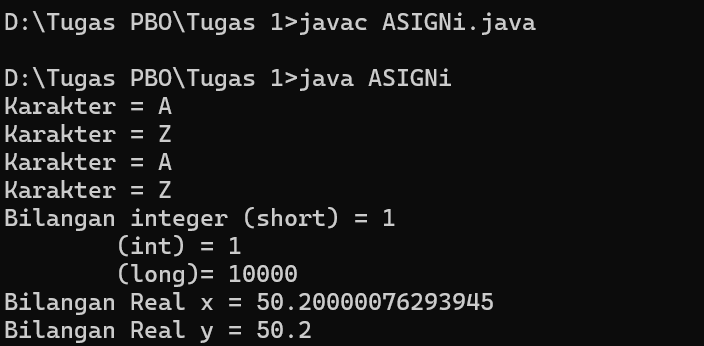


Penelasannya: Program di atas merupakan program Java sederhana yang menunjukkan penggunaan variabel integer dan fungsi pencetakan teks ke layar. Program diawali dengan deklarasi kelas Asign, yang berisi metode main() sebagai titik awal eksekusi.

Di dalam main(), variabel i bertipe int dideklarasikan tetapi belum langsung diberikan nilai. Kemudian, perintah System.out.print("hello\n"); digunakan untuk mencetak kata "hello" diikuti oleh karakter \n, yang menyebabkan kursor berpindah ke baris baru setelah mencetak teks.

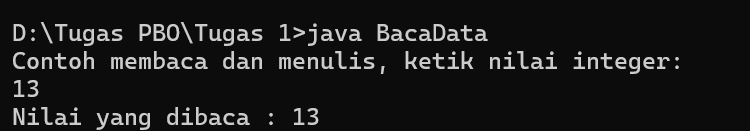
Setelah itu, variabel i diberikan nilai 5. Perintah System.out.println("Ini nilai i :" + i); digunakan untuk mencetak teks "Ini nilai i :" diikuti oleh nilai i yang sudah diinisialisasi.

1. ASIGNi



Penjelasannya: Program diatas adalah program Java sederhana yang mendemonstrasikan penggunaan berbagai tipe data primitif, seperti short, int, long, char, double, dan float, serta cara mencetaknya ke layar menggunakan System.out.println(). Program dimulai dengan deklarasi dan inisialisasi beberapa variabel. Variabel ks bertipe short diinisialisasi dengan nilai 1, ki bertipe int juga diberikan nilai 1, dan kl bertipe long dengan nilai 10000. Selain itu, variabel c bertipe char diinisialisasi dengan nilai 65, yang dalam kode ASCII merepresentasikan huruf 'A', sedangkan c1 langsung diberikan karakter 'Z'. Program juga menggunakan variabel x bertipe double yang diberi nilai 50.2f, dan y bertipe float dengan nilai yang sama.

Setelah mendeklarasikan dan menginisialisasi variabel, program mencetak nilai dari masing-masing variabel ke layar. Pada bagian pertama, program mencetak karakter c yang berisi 'A' dan c1 yang berisi 'Z' dua kali berturut-turut. Kemudian, program mencetak nilai dari variabel bertipe integer (short, int, dan long). Variabel ks ditampilkan sebagai "Bilangan integer (short) = 1", ki sebagai "int = 1", dan kl sebagai "long = 10000". Program menggunakan karakter \t (tab) untuk merapikan output. Selanjutnya, program mencetak nilai variabel bertipe double dan float, di mana x bertipe double dan y bertipe float, keduanya memiliki nilai 50.2. Program ini menunjukkan bagaimana Java menangani berbagai tipe data primitif serta bagaimana konversi tipe data bekerja, seperti dalam kasus variabel x, yang meskipun diberi nilai bertipe float, tetap dikonversi secara otomatis ke double karena presisi yang lebih tinggi.

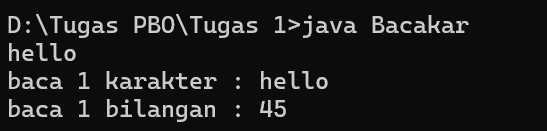
1. Bac Data

Penjelasannya: Program BacaData adalah program Java sederhana yang menggunakan kelas Scanner untuk membaca input dari pengguna melalui keyboard. Program ini dimulai dengan deklarasi variabel a bertipe int yang akan digunakan untuk menyimpan nilai integer yang dimasukkan oleh pengguna. Selain itu, sebuah objek Scanner bernama masukan dideklarasikan untuk menangani input.

Dalam eksekusinya, program pertama-tama mencetak pesan "Contoh membaca dan menulis, ketik nilai integer:" menggunakan System.out.print(), yang meminta pengguna untuk memasukkan angka. Selanjutnya, objek Scanner diinisialisasi dengan new Scanner(System.in);, memungkinkan program membaca input dari keyboard. Kemudian, metode nextInt() digunakan untuk membaca bilangan bulat yang diketik oleh pengguna dan menyimpannya dalam variabel a.

Jika pengguna mengetik dua kali masukan.nextInt(); secara berurutan, maka program akan meminta input dua kali sebelum mencetak hasilnya. Namun, jika tidak ada input kedua, program akan mengalami error atau menunggu input lebih lanjut. Setelah membaca nilai, program mencetak hasilnya dengan pesan "Nilai yang dibaca :" diikuti oleh nilai yang dimasukkan pengguna. Program ini menunjukkan cara dasar interaksi pengguna dengan Java Console menggunakan Scanner untuk membaca data dari input standar.

1. Bacakar



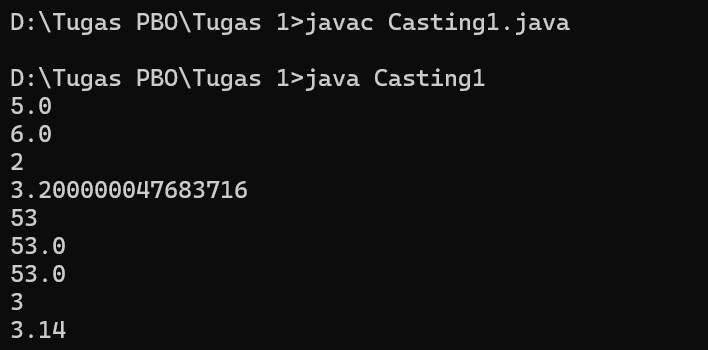
Penjelasannya: Program Bacakar adalah program Java yang menggunakan kelas BufferedReader untuk membaca input dari pengguna melalui keyboard. Program ini menunjukkan cara membaca karakter dan bilangan integer menggunakan metode readLine() serta cara mengonversi string menjadi tipe data numerik.

Di dalam metode main(), program mendeklarasikan variabel cc bertipe char untuk menyimpan satu karakter dan bil bertipe int untuk menyimpan bilangan bulat. Selanjutnya, program membuat objek InputStreamReader bernama isr, yang digunakan untuk membaca input dari System.in (keyboard). Kemudian, dua objek BufferedReader dideklarasikan: dataIn yang menggunakan isr, dan datAIn yang langsung dibuat dengan InputStreamReader dalam satu baris kode. Kedua objek ini berfungsi untuk membaca input dalam bentuk string yang nantinya akan diproses lebih lanjut.

Program pertama-tama mencetak "hello", lalu meminta pengguna untuk memasukkan satu karakter dengan pesan "baca 1 karakter :". Pengguna harus mengetik satu karakter, yang kemudian dibaca menggunakan dataIn.readLine().charAt(0), di mana readLine() membaca seluruh baris input, dan charAt(0) mengambil karakter pertama dari string yang dimasukkan. Setelah itu, program meminta pengguna memasukkan satu bilangan dengan pesan "baca 1 bilangan :". Nilai yang dimasukkan dibaca oleh datAIn.readLine(), yang mengembalikan string, lalu dikonversi menjadi integer menggunakan Integer.parseInt().

Bagian kode yang dikomentari (JOptionPane.showInputDialog() dan JOptionPane.showMessageDialog()) menunjukkan alternatif penggunaan GUI dialog box dari pustaka javax.swing, namun tidak dieksekusi dalam program ini. Akhirnya, program mencetak nilai cc dan bil, lalu menampilkan "bye" sebagai output terakhir. Program ini menunjukkan teknik membaca input menggunakan BufferedReader, yang lebih efisien dibandingkan Scanner untuk membaca data dalam jumlah besar.

1. Casting1



Penjelasannya: Program Casting1 dalam Java mendemonstrasikan konsep type casting, yaitu proses mengonversi satu tipe data ke tipe lain secara eksplisit menggunakan casting operator (tipe\_data). Program ini mengubah nilai variabel dari satu tipe ke tipe lainnya dan mencetak hasilnya ke layar.

Di dalam metode main(), beberapa variabel dideklarasikan dengan berbagai tipe data:

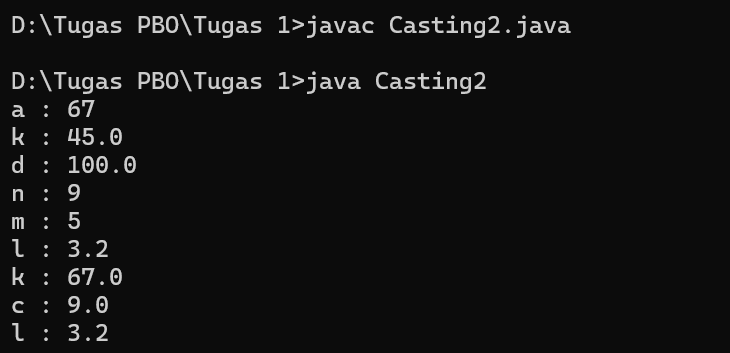
* a dan b bertipe int dengan nilai 5 dan 6.
* d dan e bertipe float dengan nilai 2.0 dan 3.2.
* g bertipe char dengan karakter '5'.
* k bertipe double dengan nilai 3.14.

Program kemudian mencetak hasil konversi berbagai variabel menggunakan casting operator:

1. (float)a mengonversi a dari int ke float, yang tetap bernilai 5.0.
2. (double)b mengonversi b dari int ke double, yang tetap bernilai 6.0.
3. (int)d mengonversi d dari float ke int, sehingga desimalnya dihilangkan dan menjadi 2.
4. (double)e mengonversi e dari float ke double, tanpa perubahan nilai.
5. (int)g mengonversi g dari char ke int, yang menghasilkan nilai 53 karena karakter '5' memiliki kode ASCII 53.
6. (float)g dan (double)g mengonversi g ke tipe float dan double, tetap menghasilkan nilai 53.0 karena karakter '5' tetap dikonversi berdasarkan kode ASCII.
7. (int)k mengonversi k dari double ke int, yang membulatkan angka desimal ke bawah menjadi 3.
8. (float)k mengonversi k dari double ke float, tetap bernilai 3.14.

Program ini menunjukkan bagaimana konversi eksplisit bekerja di Java, terutama ketika mengonversi antara int, float, double, dan char. Salah satu poin penting yang diperlihatkan adalah bahwa konversi dari char ke int menggunakan kode ASCII, dan konversi dari float atau double ke int menyebabkan hilangnya angka desimal.

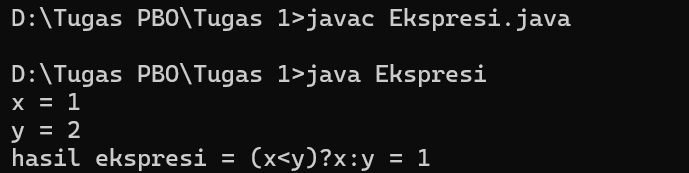
1. Casting2



Penjelasannya: Program ini menunjukkan bagaimana Java menangani konversi tipe data melalui kelas wrapper (Integer, Double, Float, dan String). Dengan menggunakan metode seperti parseInt(), parseDouble(), dan parseFloat(), kita dapat mengonversi String ke angka. Sebaliknya, dengan String.valueOf(), kita dapat mengonversi angka ke String. Selain itu, metode intValue() dan doubleValue() digunakan untuk konversi antar tipe numerik.

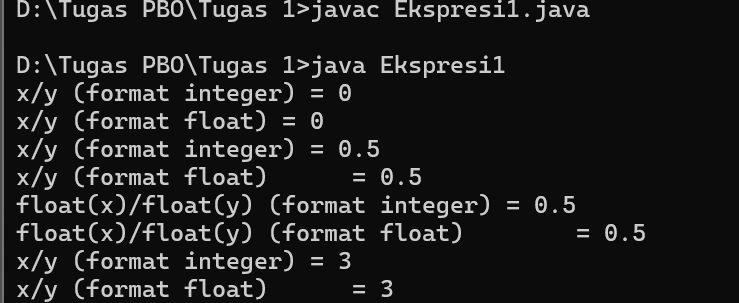
Program ini penting untuk memahami bagaimana menangani input dalam bentuk string serta mengonversinya ke bentuk numerik agar bisa digunakan dalam operasi matematika.

1. Ekspresi



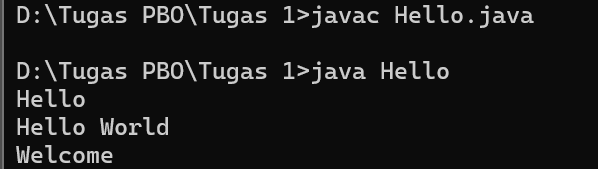
Penjelasannya: Program ini menunjukkan bagaimana operator ternary bekerja sebagai alternatif if-else dalam situasi sederhana. Dalam hal ini, operator ?: memilih nilai yang lebih kecil antara x dan y berdasarkan kondisi x < y. Operator ini berguna untuk membuat kode lebih ringkas dan mudah dibaca dalam kasus pemilihan sederhana.

1. Ekspresi 1



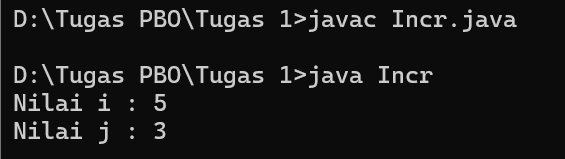
Penjelasannya: program ini adalah bahwa pembagian antara dua bilangan bertipe integer (int) akan selalu menghasilkan integer, sedangkan pembagian menggunakan float atau double akan menyertakan angka desimal. Untuk memastikan hasil memiliki desimal, salah satu atau kedua operand harus bertipe float atau double, baik dengan cara konversi langsung (fx = x) atau menggunakan casting ((float)x).

1. Hello



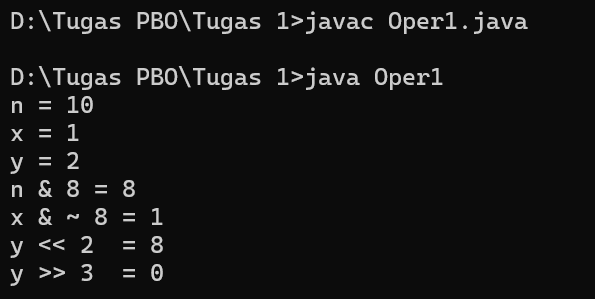
Penjelasannya: Program Hello dalam Java ini berfungsi untuk mencetak teks ke layar menggunakan metode System.out.print() dan System.out.println(). Program ini dimulai dengan mencetak kata "Hello" tanpa berpindah ke baris baru. Selanjutnya, perintah berikutnya mencetak "Hello " dengan tambahan "\n", yang berfungsi sebagai karakter newline, sehingga teks ini muncul di baris baru. Kemudian, perintah System.out.println("World") mencetak kata "World" dan secara otomatis berpindah ke baris baru. Terakhir, perintah System.out.println("Welcome") mencetak kata "Welcome" dan berpindah ke baris berikutnya. Perbedaan utama antara print() dan println() dalam program ini adalah bahwa print() hanya mencetak teks tanpa pindah baris setelahnya, sedangkan println() mencetak teks dan langsung berpindah ke baris baru. Hasil akhir dari program ini adalah tampilan teks yang terformat dengan baik, di mana setiap bagian muncul sesuai urutan yang telah ditentukan dalam kode.

1. Incr



Penjelasannya: Program Incr dalam Java ini bertujuan untuk mendemonstrasikan penggunaan operator increment (++) dalam operasi penugasan. Program dimulai dengan mendeklarasikan dua variabel i dan j bertipe int. Variabel i diinisialisasi dengan nilai 3, kemudian nilai i disalin ke variabel j dengan operasi j = i++. Dalam operasi ini, i++ merupakan post-increment, yang berarti nilai i akan digunakan terlebih dahulu untuk diberikan ke j, kemudian i bertambah 1 setelahnya, sehingga nilai j tetap 3, sedangkan i menjadi 4. Selanjutnya, perintah System.out.println() mencetak nilai i setelah dilakukan pre-increment (++i), yang berarti nilai i akan bertambah 1 sebelum digunakan dalam ekspresi, sehingga nilai yang dicetak untuk i adalah 5. Sementara itu, nilai j tetap 3 karena telah disalin sebelum i bertambah.

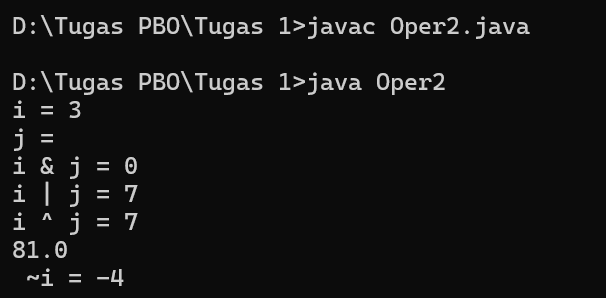
1. Oper1



Penjelasan: Program Oper1 dalam Java bertujuan untuk mendemonstrasikan penggunaan operator bitwise pada beberapa operasi bilangan bulat. Program ini mendeklarasikan tiga variabel bertipe int, yaitu n dengan nilai 10, x dengan nilai 1, dan y dengan nilai 2. Setelah mencetak nilai awal ketiga variabel tersebut, program menjalankan beberapa operasi bitwise.

Pertama, ekspresi n & 8 menggunakan operator AND bitwise (&) yang membandingkan setiap bit dari n (10 = 1010₂) dengan 8 (1000₂), menghasilkan 1000₂ = 8.  
Kedua, ekspresi x & ~8 menggunakan NOT bitwise (~) pada 8, yang mengubah 8 (1000₂) menjadi -9 (1111...0111₂ dalam representasi komplemen dua), lalu dilakukan AND bitwise (&) dengan x (0001₂), menghasilkan 0.  
Ketiga, ekspresi y << 2 menggunakan left shift (<<), yaitu menggeser bit y (2 = 0010₂) dua kali ke kiri, menghasilkan 8 (1000₂).  
Keempat, ekspresi y >> 3 menggunakan right shift (>>), yaitu menggeser y (2 = 0010₂) tiga kali ke kanan, yang dalam representasi biner akan menghasilkan 0 karena semua bit bergeser keluar.

1. Oper2

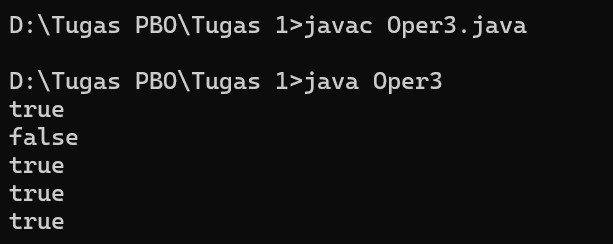


Penjelasan: Program Oper2 dalam Java bertujuan untuk mendemonstrasikan penggunaan operator bitwise pada variabel bertipe char. Dalam program ini, dua variabel i dan j bertipe char diberikan nilai 3 dan 4. Karena char di Java direpresentasikan sebagai angka Unicode, operasi bitwise akan bekerja pada representasi numerik dari karakter tersebut.

Pertama, nilai i dan j ditampilkan dengan (int) i untuk memastikan bahwa i ditampilkan sebagai angka dan bukan karakter.  
Kemudian, beberapa operasi bitwise dilakukan:

* i & j (bitwise AND) membandingkan setiap bit dari 3 (0011₂) dengan 4 (0100₂) dan menghasilkan 0 (0000₂).
* i | j (bitwise OR) membandingkan setiap bit dan menghasilkan 7 (0111₂).
* i ^ j (bitwise XOR) membandingkan bit yang berbeda menghasilkan 7 (0111₂).
* Math.pow(i, j) menghitung i pangkat j, yaitu 3^4 = 81.0.
* ~i (bitwise NOT) melakukan operasi komplemen dua, yang berarti ~3 menghasilkan -4 dalam representasi integer.

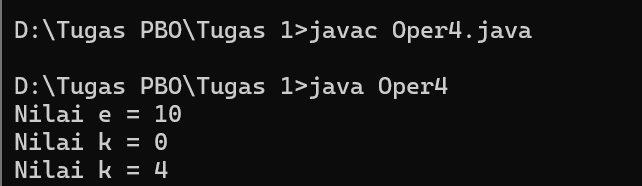
1. Oper3



Penjeasan: Program Oper3 dalam Java bertujuan untuk mendemonstrasikan penggunaan operator logika dan bitwise pada kondisi if. Program ini menggunakan berbagai kombinasi operator &&, &, ||, dan | untuk mengevaluasi ekspresi boolean dan mencetak hasilnya.

Berikut adalah penjelasan kode baris per baris:

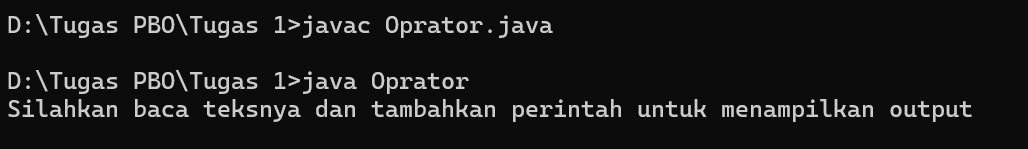
1. if (true && true) { System.out.println(true && true); }
   * && adalah operator logika AND yang hanya mengevaluasi ekspresi kedua jika ekspresi pertama bernilai true.
   * Karena kedua operand bernilai true, maka ekspresi menghasilkan true dan mencetak true.
2. if (true & true) { System.out.println(true & false); }
   * & adalah operator bitwise AND yang mengevaluasi kedua operand tanpa memperhatikan kondisi awal.
   * Karena ekspresi pertama (true & true) bernilai true, maka blok kode dijalankan.
   * Namun, ekspresi dalam System.out.println(true & false) mengevaluasi true & false, yang menghasilkan false, sehingga yang dicetak adalah false.
3. if (true) { System.out.println(true); }
   * Karena kondisi true, maka perintah dalam blok if dieksekusi, dan true dicetak.
4. if (true || true) { System.out.println(true); }
   * || adalah operator logika OR yang mengevaluasi ekspresi pertama terlebih dahulu dan langsung berhenti jika sudah true.
   * Karena kondisi true || true pasti true, maka program mencetak true.
5. if (true | false) { System.out.println(true | false); }
   * | adalah operator bitwise OR yang mengevaluasi kedua operand, berbeda dengan || yang hanya mengevaluasi bagian pertama jika sudah true.
   * Karena true | false bernilai true, maka blok dieksekusi dan hasilnya true dicetak.
6. Oper4



Penjelasan: Program Oper4 dalam Java menunjukkan penggunaan operator ternary (?:) untuk memilih nilai berdasarkan kondisi tertentu. Pertama, variabel i dan j diinisialisasi dengan nilai 0, sedangkan c dan d bertipe char dengan nilai 8 dan 10. Variabel e diberikan nilai menggunakan ekspresi (((int)c > (int)d) ? c : d), yang membandingkan nilai numerik c dan d. Karena 8 < 10, maka e bernilai 10. Selanjutnya, variabel k diberikan nilai berdasarkan perbandingan i > j, yang dalam kasus ini menghasilkan 0 karena kedua variabel memiliki nilai yang sama.

Setelah menampilkan hasil, nilai i diubah menjadi 2 dan j menjadi 3. Operator ternary kembali digunakan dengan ekspresi ((i++ > j++) ? i : j). Evaluasi dilakukan dengan membandingkan nilai awal i dan j, yaitu 2 > 3, yang bernilai false, sehingga nilai j setelah ditingkatkan (j = 4) diberikan ke k. Akhirnya, program mencetak hasil nilai e, k awal, dan k setelah evaluasi kedua.

1. Oprator



Penjelasan: Program Oprator dalam Java mendemonstrasikan penggunaan berbagai jenis operator, termasuk operator logika, aritmetika, dan perbandingan. Namun, dalam kode ini tidak terdapat perintah untuk menampilkan output ke layar, sehingga meskipun operasi dilakukan, hasilnya tidak akan terlihat.

Pertama, program menginisialisasi tiga variabel Bool1, Bool2, dan TF untuk operasi logika. Operator && (AND), || (OR), dan ^ (XOR) digunakan untuk mengevaluasi kombinasi nilai boolean. Selanjutnya, variabel i dan j bertipe int diberikan nilai 5 dan 2, kemudian dilakukan operasi aritmetika seperti penjumlahan, pengurangan, pembagian, perkalian, dan modulus (%), tetapi hanya hasil akhir yang tersimpan dalam hsl.

Kemudian, variabel x dan y bertipe float diinisialisasi dengan nilai 5, dan dilakukan operasi aritmetika yang serupa dengan variabel int. Program juga menggunakan operator perbandingan seperti == (sama dengan), != (tidak sama dengan), < (kurang dari), > (lebih dari), <= (kurang dari atau sama dengan), dan >= (lebih dari atau sama dengan) pada variabel bertipe int dan float.

Agar program ini menampilkan hasil ke layar, perlu ditambahkan perintah System.out.println() pada setiap operasi, sehingga pengguna dapat melihat hasil evaluasi ekspresi secara langsung.