Tugas 2 Makassar, 13 Maret 2025

TUGAS

PEMROGRAMAN BERORIENTASI OBJEK



Nama : Wahyuni

Stambuk : 13020230110

Kelas : B2

Frekuensi : Pemrograman Berorientasi objek

Dosen : : Mardiyyah Hasnawi, S.Kom., M.T.,MTA

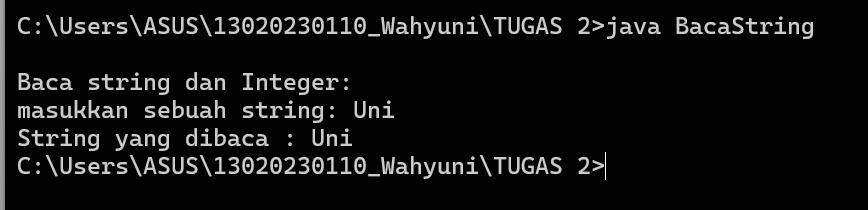
PROGRAM STUDI TEKNIK INFORMATIKA

FAKULTAS ILMU KOMPUTER

UNIVERSITAS MUSLIM INDONESIA

MAKASSAR

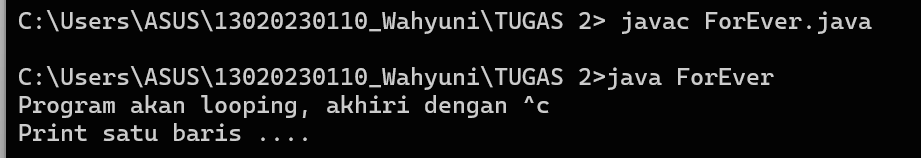
2025



Penjelasan:

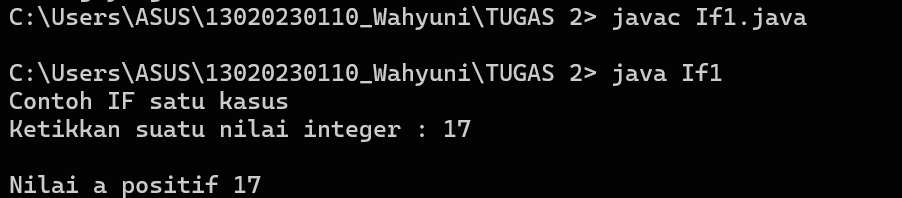
Output dari inputan saat dijalankan, program akan meminta kita untuk memberikan sebuah string melalui konsol. Tahapan ini dimulai dengan menampilkan sebuah pesan yang menyarankan kita untuk memasukkan string. Setelah pengguna mengetik stringnya dan menekan Enter, program akan memanfaatkan BufferedReader untuk menangkap input tersebut

jika kita mengetik "Uni" sebagai input, maka program akan menampilkan output "String yang dibaca:”Uni". Program ini dirancang dengan kesederhanaan dan hanya fokus pada pengambilan serta penampilan string, tanpa melakukan tugas lain seperti pengolahan data atau perubahan tipe data.



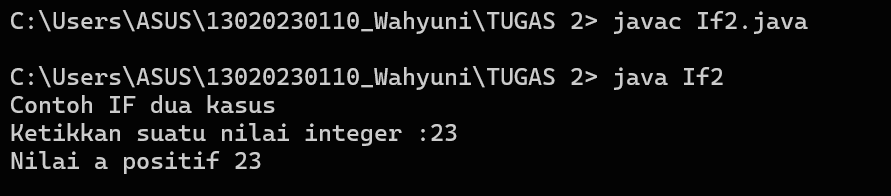
Penjelasan:

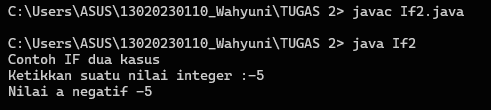
Program dari output di atas disajikan ini dirancang untuk mencetak pesan berulang kali tanpa henti Saat program diaktifkan, ia akan menunjukkan pesan "Program akan looping, akhiri dengan ^c" di layar, yang menginformasikan pengguna bahwa program akan mulai melakukan loop tak terbatas. Selanjutnya, program tersebut akan memasuki blok while (true), yang menandakan bahwa perintah di dalam blok itu akan terus dijalankan tanpa adanya syarat untuk menghentikannya. Nah jika ingin menghentikan sebuah program maka silahkan tambahkan kata break di akhir program maka akan berehnti secara otomatis tanpa perlu di berikan perintah pada outputnya



Penjelasannya:

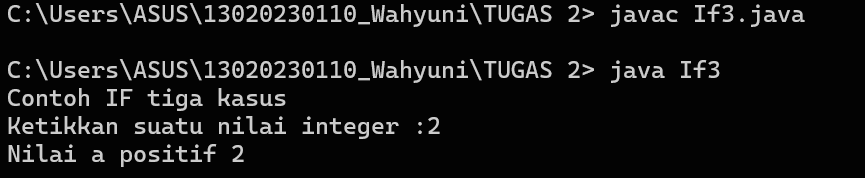
Program dari output yang disajikan memiliki tujuan untuk mengambil input integer dari pengguna dan menunjukkan nilai tersebut jika itu adalah positif. Saat program dijalankan, pengguna akan melihat pesan "Contoh IF satu kasus" di layar konsol, disusul dengan permintaan untuk memasukkan nilai integer. Program ini menggunakan kelas Scanner untuk menerima input dari pengguna.  
  
Setelah pengguna mengetikkan nilai integer dan menekan tombol Enter, nilai tersebut akan disimpan dalam variabel a. Program kemudian akan memverifikasi apakah nilai a adalah positif (artinya a lebih besar dari atau sama dengan 0) melalui pernyataan if. Apabila kondisi ini terpenuhi, program akan menampilkan pesan yang mengatakan "Nilai a positif" diikuti oleh nilai yang dimasukkan oleh pengguna.  
  
Sebagai ilustrasi, jika pengguna menginput nilai 16, hasil yang akan muncul adalah "Nilai a positif 17". Namun, jika pengguna menginput nilai negatif, seperti -3, program tidak akan memberikan output karena kondisi dalam pernyataan if tidak terpenuhi. Oleh karena itu, program ini hanya akan mencetak nilai jika itu positif dan tidak memberikan reaksi pada nilai negatif atau nol.





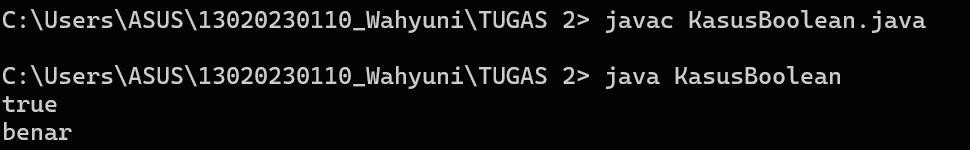
Penjelasannya:

Program Java yang diberikan bertugas untuk mengumpulkan sebuah angka bulat dari pengguna dan memberikan reaksi berdasarkan apakah angka itu bernilai positif atau negatif. Saat program diluncurkan, pengguna akan melihat teks “Contoh IF dua kasus” di layar konsol, lalu diikuti dengan instruksi untuk memasukkan nilai integer. Program ini memanfaatkan kelas Scanner untuk mengambil masukan dari pengguna.  
  
Setelah pengguna mengetik angka bulat dan menekan tombol Enter, angka tersebut disimpan dalam variabel a. Program selanjutnya menggunakan pernyataan if untuk mengevaluasi apakah angka a tersebut positif (yaitu, apakah a lebih besar atau sama dengan 0). Jika syarat ini terpenuhi, program akan menampilkan pesan “Nilai a positif” diikuti dengan angka yang dimasukkan. Sebaliknya, jika angka a kurang dari 0, program akan melaksanakan blok else dan menampilkan pesan “Nilai a negatif” beserta angka yang dimasukkan.  
  
Contohnya, jika pengguna mengetikkan angka 23, hasil yang muncul akan berupa “Nilai a positif 23”. Namun, bila pengguna menginput angka -5, hasil yang akan ditampilkan adalah “Nilai a negatif -5”. Dengan demikian, program ini memberikan respons yang jelas kepada pengguna berdasarkan angka yang dimasukkan, dengan dua kemungkinan hasil yang saling melengkapi.



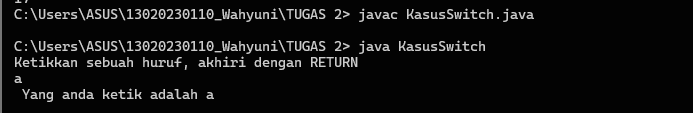
Penjelasannya:

Program hampir sama dengan program sebelumnya namun berbeda, karena pada outputnya Sebagai ilustrasi, apabila pengguna memasukkan angka 2, output yang akan muncul adalah "Nilai a positif 2". Jika pengguna menginput 0, outputnya akan menjadi "Nilai Nol 0". Dan jika pengguna mengisi -3, output yang dihasilkan akan berbunyi "Nilai a negatif -3". Dengan cara ini, program ini memberikan umpan balik yang jelas kepada pengguna berdasarkan angka yang dimasukkan, dengan tiga kemungkinan keluaran yang mencakup semua variasi nilai integer.



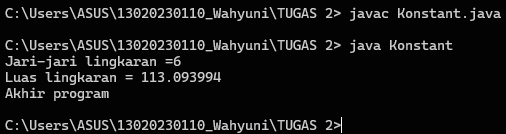
Penjelasanya:

rogram ini dibuat untuk memperlihatkan penggunaan ekspresi kondisional yang berkaitan dengan tipe data boolean. Saat dijalankan, variabel boolean bernama bool diatur dengan nilai true. Program ini kemudian memanfaatkan pernyataan if untuk mengecek nilai dari variabel bool.  
  
Di awal, program akan mengevaluasi apakah bool sama dengan true. Karena bool sudah diset ke true, maka program akan menampilkan "true" di konsol. Selanjutnya, program memeriksa nilai negasi dari bool lewat !bool. Mengingat bool bernilai true, negasinya menjadi false, sehingga program akan mencetak "benar" di konsol.  
  
Dengan kata lain, hasil keluaran dari program ini terdiri dari dua baris: yang pertama adalah "true" dan yang kedua adalah "benar." Program ini tidak meminta interaksi dari pengguna, melainkan hanya menunjukkan output berdasarkan nilai yang ditentukan dalam kode. Ini mencontohkan bagaimana ekspresi boolean dapat ditempatkan untuk mengatur jalannya program dan menciptakan output yang sesuai dengan kondisi yang ada.



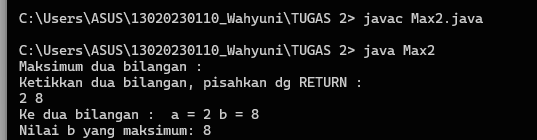
Penjelasannya:

Program ini dirancang untuk menerima input karakter dari pengguna dan memberikan reaksi sesuai dengan huruf vokal yang diterima. Saat program dijalankan, pengguna akan dihadapkan pada pesan "Silakan masukkan sebuah huruf, diakhiri dengan menekan ENTER" yang muncul di layar konsol, meminta mereka untuk memberikan satu huruf. Program ini memanfaatkan kelas Scanner untuk mendapatkan input dari pengguna.  
  
Sebagai ilustrasi, jika pengguna mengetikkan huruf 'e', maka output yang akan muncul adalah "Huruf yang Anda masukkan adalah e". Di sisi lain, jika pengguna memasukkan huruf 'b', maka hasilnya akan menunjukkan "Huruf yang Anda masukkan adalah konsonan". Dengan demikian, program ini memberikan umpan balik yang jelas berdasarkan huruf yang diinputkan, dengan penekanan pada identifikasi huruf vokal dan penanganan huruf yang bukan vokal.



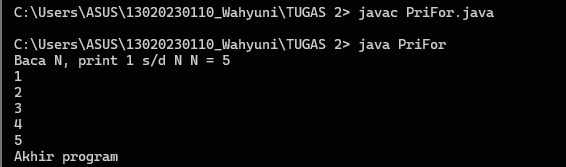
Penjelasannya:

Program Java yang disajikan memiliki tujuan untuk menghitung area sebuah lingkaran berdasarkan nilai jari-jari yang dimasukkan oleh user. Saat program ini dijalankan, user akan melihat pesan "Jari-jari lingkaran =" di konsol, yang meminta mereka untuk memberikan angka jari-jari lingkaran. Program ini memanfaatkan kelas Scanner untuk mengambil input dari user.  
  
Sebagai ilustrasi, jika user memberikan nilai jari-jari 6, output yang akan muncul adalah "Luas lingkaran = 113.093994", yang merupakan hasil dari perhitungan ( 3.1415 × 6 × 6 ). Setelah hasil ditampilkan, program juga mencetak "Akhir program" untuk menunjukkan bahwa proses eksekusi program telah selesai. Dengan begitu, program ini menyediakan metode yang sederhana dan efisien untuk menghitung area lingkaran berdasarkan nilai jari-jari dari user.



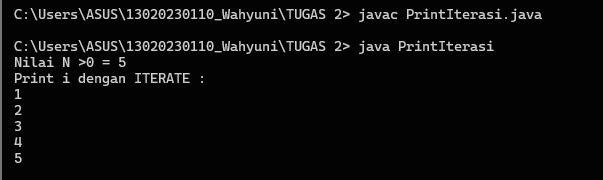
Penjelasannya:

Setelah pengguna memasukkan dua angka dan menekan Enter, program akan menyimpan angka pertama ke dalam variabel a dan angka kedua ke dalam variabel b. Selanjutnya, program mencetak kedua angka itu dengan pesan "Kedua angka: a = [nilai a] b = [nilai b]". Program kemudian memanfaatkan pernyataan if untuk membandingkan nilai a dan b. Apabila a lebih besar dari atau sama dengan b, program akan menampilkan pesan "Nilai a yang maksimum [nilai a]". Jika tidak, program akan menunjukkan pesan "Nilai b yang maksimum:[nilaib]".  
  
Sebagai contoh, jika pengguna memasukkan angka 2 dan 8 , output yang akan muncul seperti diatas



Penjelasannya:

Program ini dirancang untuk menerima sebuah angka bulat (N) dari pengguna dan menampilkan semua angka bulat mulai dari 1 hingga (N) dengan menggunakan loop for. Ketika dijalankan, pengguna akan melihat pesan "Masukkan N, cetak 1 sampai N" di layar, kemudian diminta untuk memberikan input nilai (N). Kelas Scanner digunakan oleh program untuk mengumpulkan inputdari pihak pengguna.   
Setelah pengguna memberikan nilai (N) dan menekan tombol Enter, program akan menyimpan input tersebut ke dalam variabel (N). Selanjutnya, program akan memanfaatkan loop for untuk menampilkan angka dari 1 sampai (N). Dalam setiap siklus, nilai (i) (yang awalnya bernilai 1) akan ditampilkan di layar, dan nilai (i) akan bertambah satu hingga mencapai nilai N  
Sebagai ilustrasi, jika pengguna menginput angka 5, hasil yang akan terlihat adalah seperti diatas.



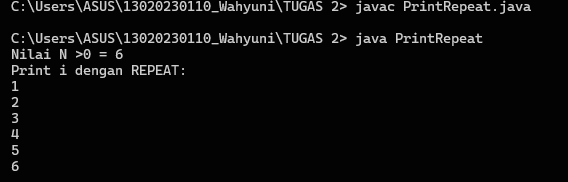
Penjelasannya:

Program Java yang diberikan adalah contoh sederhana menggunakan struktur iteratif untuk mengevaluasi angka dari 1 hingga (N), yang merupakan bilangan bulat positif yang dimasukkan oleh pengguna. Setelah program menyatakan variabel n dan i, pengguna akan meminta Anda untuk memasukkan nilai yang harus lebih besar dari 0.

Program kemudian menggunakan loop tidak terbatas ((;;)) untuk mencetak nilai I dari 1 Jika tidak, nilai yang saya tambahkan diulang dan proses ini berulang hingga semua angka dari 1 hingga 1 hingga (N) dicetak.

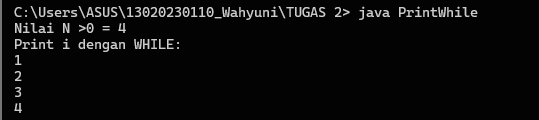
Misalnya, ketika pengguna masuk 5, program mencetak angka 1, 2, 3, 4, dan 5.

Program ini menunjukkan metode dasar untuk membuat biaya berdasarkan input pengguna menggunakan loop di java.



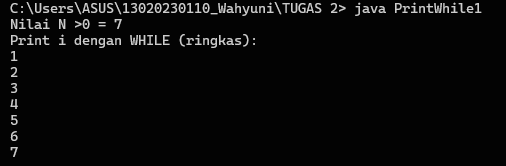
Penjelasannya:

Program ini adalah contoh penggunaan struktur pengulangan do-while untuk mencetak angka dari 1 hingga ( N ), di mana ( N ) adalah bilangan bulat positif yang dimasukkan oleh pengguna. Program dimulai dengan mengimpor kelasScanner untuk membaca input dari konsol. Setelah mendeklarasikan variable N dan I, program meminta pengguna untuk memasukkan nilai ( N ) yang harus lebih besar dari 0. Setelah nilai ( N ) diinput, program menginisialisasi variable i dengan 1 dan kemudian memasuki blok do. Di dalam blok ini, program mencetak nilai i dan kemudian menambah nilai i dengan 1. Proses ini akan terus berlanjut hingga nilai i melebihi ( N ). Sebagai contoh, jika pengguna memasukkan 6, program akan mencetak angka 1, 2, 3, 4, 5,dan 6 secara berurutan. Program ini menunjukkan cara menggunakan pengulangan do- whie dalam Java, di mana blok kode dieksekusi setidaknya satu kali sebelum memeriksa kondisi pengulangan.



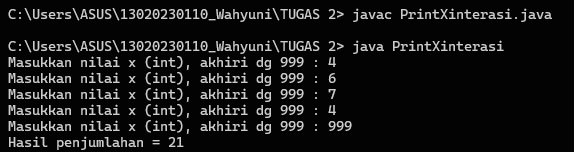
Penjelasannya:

Program ini adalah contoh penggunaan struktur pengulangan While untuk mencetak angka dari 1 hingga ( N ), di mana ( N ) adalah bilangan bulat positif yang dimasukkan oleh pengguna. Program ini dimulai dengan mengimpor kelas Scanner untuk memungkinkan pembacaan input dari konsol. Setelah mendeklarasikan variable N dan i, program meminta pengguna untuk memasukkan nilai ( N ) yang harus lebih besar dari 0. Setelah nilai ( N ) diinput, program menginisialisasi variable i dengan 1 dan kemudian memasuki loop while. Di dalam loop ini, program memeriksa apakah nilai i kurang dari atau sama dengan ( N ). Jika kondisi tersebut terpenuhi, program mencetak nilai i dan kemudian menambah nilai i dengan 1. Proses ini akan berlanjut hingga nilai i melebihi ( N ). Sebagai contoh, jika pengguna memasukkan 4, program akan mencetak angka 1, 2, 3, dan 4 secara berurutan. Program ini menunjukkan cara menggunakan pengulangan while dalam Java, di mana loop akan terus berjalan selama kondisi yang ditentukan tetap benar.



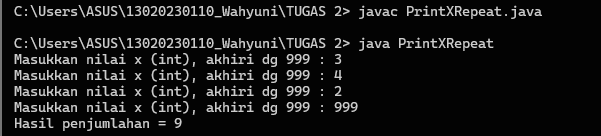
Penjelasannya:

Program Java yang diberikan adalah contoh penggunaan struktur pengulangan while yang lebih ringkas untuk mencetak angka dari 1 hingga ( N ), di mana ( N ) adalah bilangan bulat positif yang dimasukkan oleh pengguna. Program ini dimulai dengan mengimpor kelas Scanner untuk membaca input dari konsol. Setelah mendeklarasikan variable N dan menginisialisasi variable i dengan 1, program meminta pengguna untuk memasukkan nilai ( N ) yang harus lebih besar dari 0. Setelah nilai ( N ) diinput, program mencetak pesan yang menunjukkan bahwa proses pencetakan angka akan dimulai. Selanjutnya, program memasuki loop while di mana selama nilai i kurang dari atau sama dengan ( N ), program akan mencetak nilai i  dan secara bersamaan menambah nilai i dengan 1 menggunakan operator post-increment(i++). Proses ini akan berlanjut hingga nilai i melebihi ( N ). Sebagai contoh, jika pengguna memasukkan 5, program akan mencetak angka 1, 2, 3, 4, 5, 6, dan 7 secara berurutan. Program ini menunjukkan cara yang efisien untuk menggunakan pengulangan while dalam Java, dengan menggabungkan pencetakan dan peningkatan nilai dalam satu baris kode.



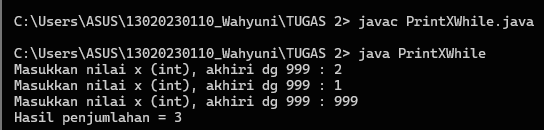
Penjelasannya:

Program ini adalah contoh penggunaan iterasi untuk menjumlahkan serangkaian nilai integer yang dimasukkan oleh pengguna, dengan pengakhiran input menggunakan angka 999. Program ini dimulai dengan mengimpor kelas Scanner untuk membaca input dari konsol. Setelah mendeklarasikan variable Sum untuk menyimpan total penjumlahan dan variable x untuk menyimpan nilai yang dimasukkan, program meminta pengguna untuk memasukkan nilai integer. Jika pengguna langsung memasukkan 999, program akan mencetak pesan "Kasus kosong" dan tidak melakukan penjumlahan. Namun, jika pengguna memasukkan nilai lain, program akan menginisialisasi Sum dengan nilai tersebut dan kemudian memasuki loop tak terbatas. Di dalam loop, program terus meminta pengguna untuk memasukkan nilai baru. Jika nilai yang dimasukkan adalah 999, loop akan dihentikan. Jika tidak, nilai tersebut akan



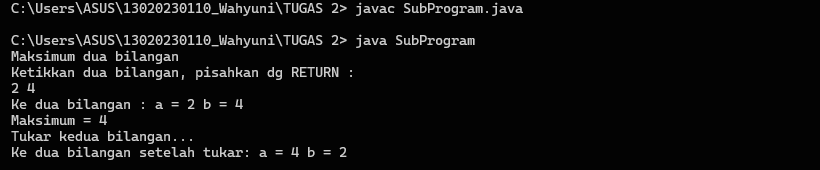
Penjelasannya:

program ini berfokus pada interaksi pengguna untuk menjumlahkan nilai integer yang dimasukkan. Program dimulai dengan meminta pengguna untuk memasukkan nilai integer melalui konsol. Jika pengguna memasukkan angka 999 pada input pertama, program akan segera menghentikan proses dan menampilkan pesan "Kasus kosong", yang menunjukkan bahwa tidak ada nilai yang dijumlahkan. Namun, jika nilai yang dimasukkan adalah angka lain, program akan menginisialisasi variable sum dengan nilai 0 dan kemudian memasuki loop di-while. Di dalam loop ini, program akan terus meminta pengguna untuk memasukkan nilai baru, menambahkan setiap nilai yang dimasukkan ke dalam sum, hingga pengguna memasukkan 999. Setelah pengguna memasukkan 999, loop akan berhenti, dan program akan mencetak hasil penjumlahan dari semua nilai yang telah dimasukkan sebelumnya dalam format "Hasil penjumlahan = [nilai]", di mana [nilai] adalah total dari semua angka yang dimasukkan. Dengan demikian, input berupa serangkaian angka integer dan output berupa total penjumlahan memberikan gambaran yang jelas tentang fungsi program, serta cara interaktif untuk mengumpulkan dan memproses data dari pengguna.



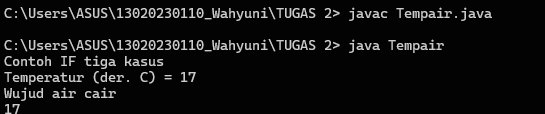
Penjelasannya:

program ini berfokus pada interaksi pengguna dengan aplikasi untuk menjumlahkan nilai integer. Pengguna diminta untuk memasukkan nilai integer satu per satu melalui konsol. Proses dimulai dengan permintaan untuk memasukkan nilai pertama, di mana jika pengguna memasukkan angka 999, program akan segera menghentikan proses dan menampilkan pesan "Kasus kosong", menandakan bahwa tidak ada nilai yang dijumlahkan. Jika nilai yang dimasukkan adalah angka lain, program akan menyimpan nilai tersebut dan melanjutkan untuk meminta input tambahan. Pengguna dapat terus memasukkan nilai integer hingga mereka memutuskan untuk mengakhiri proses dengan memasukkan 999. Setelah pengguna memasukkan 999, program akan menghentikan pengambilan input dan mencetak hasil penjumlahan dari semua nilai yang telah dimasukkan sebelumnya, menampilkan hasil dalam format "Hasil penjumlahan = [nilai]", di mana [nilai] adalah total dari semua angka yang dimasukkan. Dengan demikian, input berupa serangkaian angka integer dan output berupa total penjumlahan dari angka-angka tersebut memberikan gambaran yang jelas tentang fungsi program.



Pennjelasannya:

program ini berfokus pada interaksi pengguna untuk menentukan nilai maksimum dari dua bilangan bulat dan menukar nilai-nilai tersebut. Program dimulai dengan meminta pengguna untuk memasukkan dua bilangan integer secara berurutan, yang dipisahkan dengan menekan tombol "Return" setelah masing-masing bilangan. Setelah pengguna memasukkan kedua bilangan, program akan menampilkan nilai kedua bilangan tersebut. Selanjutnya, program memanggil fungsi max untuk menghitung dan menampilkan nilai maksimum dari kedua bilangan yang dimasukkan. Setelah itu, program akan memanggil prosedur tukar untuk menukar nilai kedua bilangan. Meskipun prosedur ini tidak mengubah nilai variable a dan b di dalam main karena Java menggunakan passing by value, program tetap mencetak hasil pertukaran dengan menampilkan nilai a dan b setelah proses pertukaran. Output akhir mencakup nilai maksimum dari kedua bilangan dan hasil pertukaran, memberikan gambaran yang jelas tentang fungsi program dalam memproses dan menampilkan data yang dimasukkan oleh pengguna.



Penjelasannya :

program ini berfokus pada menentukan wujud air berdasarkan suhu yang dimasukkan oleh pengguna. Program dimulai dengan meminta pengguna untuk memasukkan nilai temperatur dalam derajat Celsius. Setelah pengguna memasukkan nilai tersebut, program akan menggunakan struktur control if-else- if untuk mengevaluasi suhu yang dimasukkan. Jika nilai temperatur kurang dari 0, program akan mencetak "Wujud air beku" diikuti dengan nilai temperatur yang dimasukkan. Jika temperatur berada dalam rentang 0 hingga 100 derajat Celsius, program akan mencetak "Wujud air cair" beserta nilai temperatur. Terakhir, jika temperatur lebih dari 100 derajat Celsius, program akan mencetak "Wujud air uap/gas" diikuti dengan nilai temperatur. Dengan demikian, input berupa satu nilai integer yang mewakili temperatur, dan output berupa deskripsi wujud air berdasarkan nilai tersebut, memberikan gambaran yang jelas tentang bagaimana program ini berfungsi dalam mengklasifikasikan keadaan air berdasarkan suhu.