**LAPORAN PRAKTIKUM 5**

**Algoritma Pemrograman**



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

Nama: Muhammad Rayhan Anugrah  
NIM: 2311533019

**FAKULTAS TEKNOLOGI INFORMASI**

**DEPARTEMEN INFORMATIKA**

**UNIVERSITAS ANDALAS**

**2024**

1. **Pendahuluan**
2. Tujuan Praktikum

* Mengenal struktur dan sintaks perulangan for
* Memahami cara kerja perulangan for dalam pemrograman
* Menjelaskan perbedaan perulangan for dengan jenis perulangan lainnya
* Mampu membuat program sederhana menggunakan perulangan for
* Mengimplementasikan perulangan for untuk menyelesaikan berbagai macam persoalan algoritma
* Menerapkan konsep perulangan pada kasus nyata dalam pemrograman

1. Kajian Teori

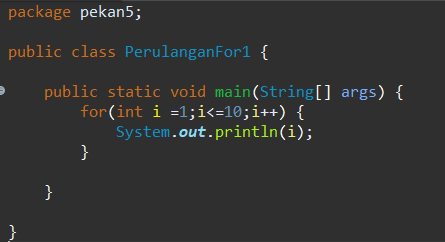
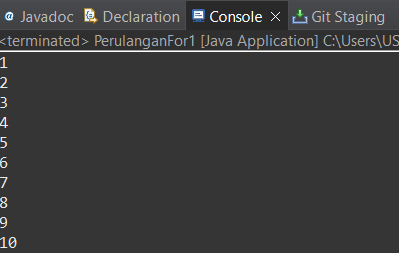
Perulangan (looping) adalah konsep dasar dalam pemrograman yang memungkinkan eksekusi perintah atau blok kode berulang kali. Salah satu jenis perulangan yang paling umum digunakan adalah perulangan **for**.

Perulangan ini sangat berguna ketika jumlah iterasi sudah diketahui atau dapat ditentukan pada awal program. Struktur dasar dari perulangan **for** terdiri dari tiga bagian utama: inisialisasi, kondisi, dan iterasi. Inisialisasi digunakan untuk mendeklarasikan dan memberikan nilai awal pada variabel yang akan digunakan dalam perulangan. Kondisi adalah ekspresi boolean yang diuji sebelum setiap iterasi; selama kondisi bernilai true, perulangan akan terus berlangsung. Setelah setiap iterasi, bagian iterasi akan mengubah nilai variabel untuk menuju kondisi penghentian.

Sebagai contoh, dalam bahasa pemrograman Python, perulangan **for** dapat digunakan untuk mencetak angka dari 1 hingga 5 dengan kode berikut: for i in range(1, 6): print(i). Pada contoh ini, nilai i dimulai dari 1 dan akan bertambah setiap kali iterasi hingga mencapai 5. Keuntungan utama dari perulangan **for** adalah efisiensinya, terutama ketika jumlah iterasi sudah diketahui, serta kemudahan dalam pengendalian variabel iterasi dan kondisi penghentian.

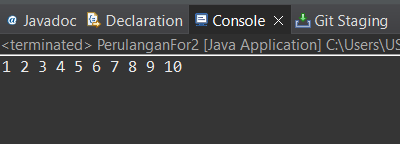
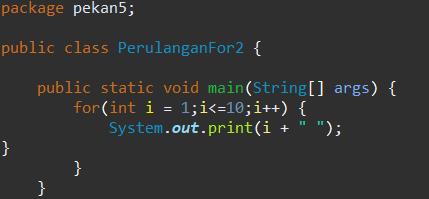
Perulangan ini juga mudah dibaca dan dipahami, menjadikannya pilihan yang populer dalam banyak aplikasi pemrograman, seperti mengiterasi elemen dalam array, menghitung jumlah elemen yang memenuhi kriteria tertentu, atau melakukan perhitungan berulang seperti dalam algoritma pengurutan atau pencarian.

1. **Pembahasan**
2. Class PerulanganFor1



Kode di atas adalah contoh implementasi perulangan for dalam bahasa pemrograman Java. Program ini bertujuan untuk mencetak angka dari 1 hingga 10. Pertama, kelas PerulanganFor1 dideklarasikan dengan metode utama main, yang merupakan titik awal eksekusi program. Di dalam metode main, terdapat perulangan for dengan tiga bagian utama: inisialisasi, kondisi, dan iterasi. Pada bagian inisialisasi, variabel i diatur dengan nilai awal 1. Kondisi i <= 10 memastikan perulangan akan terus berlangsung selama nilai i kurang dari atau sama dengan 10. Setiap kali perulangan dijalankan, nilai i akan bertambah 1, sesuai dengan iterasi i++. Di dalam blok perulangan, perintah System.out.println(i); akan mencetak nilai dari variabel i pada setiap iterasi. Hasil dari program ini adalah angka-angka 1 hingga 10 yang dicetak satu per satu di layar, dengan setiap angka tampil pada baris baru. Program ini menunjukkan cara dasar menggunakan perulangan for untuk mengiterasi sejumlah nilai yang telah ditentukan sebelumnya.

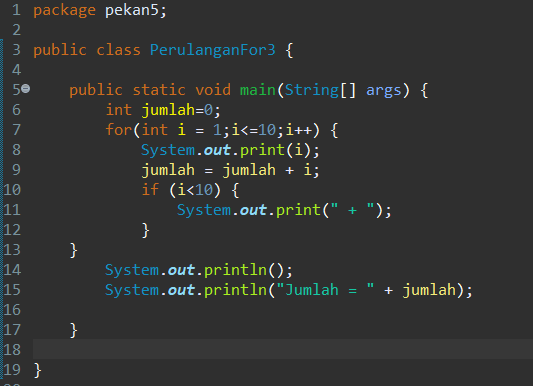
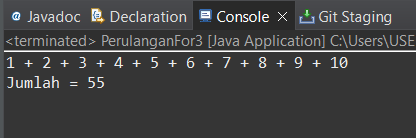
1. Class PerulanganFor2



Kode di atas adalah contoh implementasi perulangan for dalam bahasa pemrograman Java yang digunakan untuk mencetak angka dari 1 hingga 10. Program ini dimulai dengan mendeklarasikan kelas PerulanganFor2 dan metode utama main, yang berfungsi sebagai titik awal eksekusi program. Di dalam metode main, terdapat perulangan for dengan tiga bagian utama: inisialisasi, kondisi, dan iterasi.

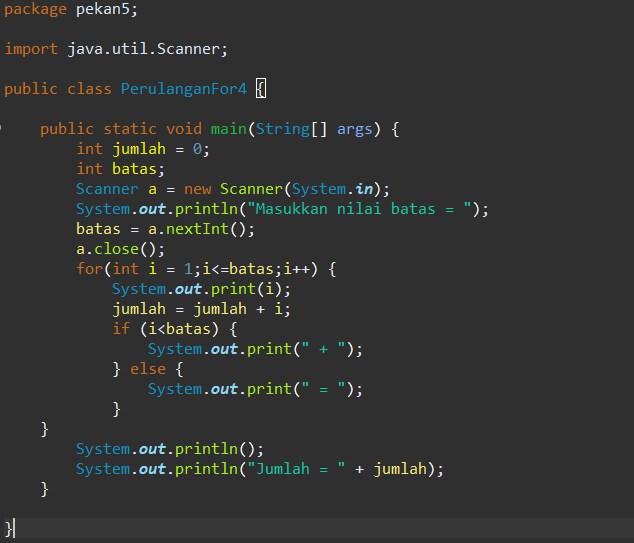
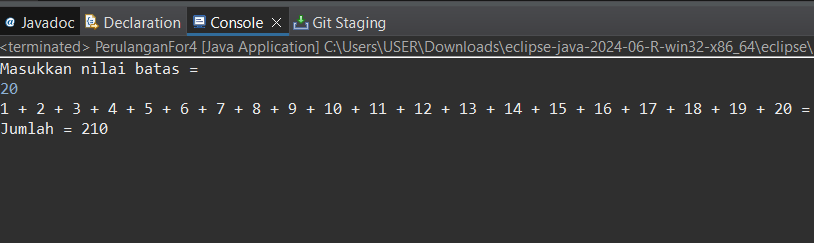
Pada bagian inisialisasi, variabel i diatur dengan nilai awal 1. Kondisi i <= 10 memastikan perulangan akan berlangsung selama nilai i kurang dari atau sama dengan 10. Setiap iterasi, nilai i akan bertambah 1 sesuai dengan bagian iterasi i++. Berbeda dengan program sebelumnya yang menggunakan System.out.println, kali ini perintah yang digunakan adalah System.out.print(i + " "). Perintah ini mencetak nilai i pada satu baris yang sama dengan sebuah spasi di antara angka-angka yang dicetak, sehingga hasilnya akan menjadi satu deretan angka yang dipisahkan oleh spasi. Program ini mencetak angka dari 1 hingga 10 secara horizontal, seperti yang terlihat pada gambar di atas sebelah kanan.

Kode ini menunjukkan cara menggunakan perulangan for untuk mencetak output dalam satu baris, berbeda dengan contoh sebelumnya yang mencetak setiap angka pada baris baru.

1. Class PerulanganFor3

Kode di atas merupakan contoh penggunaan perulangan for dalam bahasa pemrograman Java untuk menghitung dan menampilkan jumlah dari angka 1 hingga 10. Program ini dimulai dengan mendeklarasikan kelas PerulanganFor3 dan metode utama main. Di dalam metode main, variabel jumlah diinisialisasi dengan nilai 0 yang nantinya akan digunakan untuk menyimpan hasil penjumlahan angka. Perulangan for berjalan dengan inisialisasi i = 1, kondisi i <= 10, dan iterasi i++, yang berarti nilai i akan dimulai dari 1 dan bertambah 1 hingga mencapai 10. Dalam setiap iterasi perulangan, nilai i akan dicetak dengan menggunakan System.out.print(i).

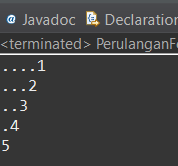
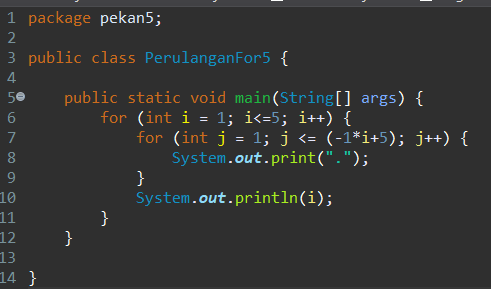
Selain itu, setiap angka yang dicetak akan ditambahkan ke dalam variabel jumlah menggunakan operasi jumlah = jumlah + i. Setelah mencetak angka, program memeriksa apakah i kurang dari 10. Jika benar, program akan mencetak tanda plus (" + ") untuk memisahkan angka-angka tersebut dalam format penjumlahan. Begitu perulangan selesai, program mencetak hasil akhir penjumlahan melalui perintah System.out.println("Jumlah = " + jumlah). Program ini menunjukkan cara menghitung jumlah dari angka-angka dalam sebuah urutan menggunakan perulangan for, serta menampilkan hasilnya dalam format yang lebih jelas.

1. Class PerulanganFor4

Kode di atas merupakan contoh penggunaan perulangan for dalam bahasa pemrograman Java yang melibatkan input dari pengguna untuk menentukan batas akhir penjumlahan. Program ini dimulai dengan mendeklarasikan kelas PerulanganFor4 dan metode utama main. Pertama, variabel jumlah diinisialisasi dengan nilai 0, yang digunakan untuk menyimpan hasil penjumlahan.

Variabel batas dideklarasikan untuk menampung input dari pengguna, yang akan menentukan hingga angka berapa perulangan dilakukan. Program menggunakan Scanner untuk membaca input dari pengguna melalui konsol. Pengguna diminta untuk memasukkan nilai batas, dan nilai tersebut kemudian disimpan dalam variabel batas. Setelah itu, perulangan for dimulai dengan inisialisasi i = 1 dan berlanjut selama i <= batas. Dalam setiap iterasi, nilai i akan dicetak dengan perintah System.out.print(i) dan nilai i akan ditambahkan ke variabel jumlah menggunakan operasi jumlah = jumlah + i. Program juga memeriksa apakah i kurang dari nilai batas. Jika benar, program akan mencetak tanda plus (" + ") setelah setiap angka. Jika i mencapai batas, program akan mencetak tanda sama dengan (" = ") untuk menunjukkan akhir dari deret penjumlahan. Setelah perulangan selesai, hasil akhir penjumlahan dicetak dengan perintah System.out.println("Jumlah = " + jumlah). Program ini menunjukkan bagaimana cara menggunakan input dari pengguna untuk menentukan jumlah angka yang akan dijumlahkan dalam perulangan for.

1. Class PerulanganFor5



Kode di atas adalah contoh penggunaan perulangan for yang bersarang (nested loop) dalam bahasa pemrograman Java. Program ini mencetak pola berupa angka yang diikuti oleh titik (".") di setiap barisnya.

Program ini dimulai dengan mendeklarasikan kelas PerulanganFor5 dan metode utama main. Pada bagian pertama, terdapat perulangan for yang mengatur variabel i dimulai dari 1 hingga 5. Perulangan pertama ini menentukan baris yang akan dicetak, di mana setiap baris akan mencetak angka yang sesuai dengan nilai i.

Di dalam perulangan pertama, terdapat perulangan for kedua yang menggunakan variabel j, yang bertanggung jawab untuk mencetak titik (".") di setiap baris. Kondisi perulangan kedua adalah j <= (-1\*i + 5), yang berarti jumlah titik yang dicetak berkurang setiap kali nilai i bertambah. Secara spesifik, pada iterasi pertama (ketika i = 1), program akan mencetak 4 titik sebelum angka 1, pada iterasi kedua (ketika i = 2), program akan mencetak 3 titik sebelum angka 2, dan seterusnya, hingga pada iterasi terakhir (ketika i = 5), program mencetak 0 titik sebelum angka 5. Perulangan pertama memastikan bahwa setiap baris akan berisi angka yang sesuai dengan nilai i setelah sejumlah titik yang dihitung oleh perulangan kedua. Setelah perulangan kedua selesai, perintah System.out.println(i) akan mencetak angka yang sesuai pada baris baru.

1. **Kesimpulan**

Dalam praktikum ini, saya telah mempelajari penerapan berbagai jenis perulangan for dalam bahasa pemrograman Java, serta cara penggunaannya untuk menyelesaikan berbagai permasalahan komputasi. Program pertama, PerulanganFor1, menunjukkan penggunaan perulangan for untuk mencetak angka dari 1 hingga 10. Perulangan ini sangat sederhana dan digunakan untuk memahami dasar-dasar perulangan, di mana angka yang dicetak muncul satu per baris. Pada program kedua, PerulanganFor2, saya memodifikasi perulangan untuk mencetak angka yang dipisahkan oleh spasi dalam satu baris. Hal ini menunjukkan bahwa output dari perulangan dapat diubah sesuai dengan format yang diinginkan tanpa mengubah alur perulangan itu sendiri.

Pada program ketiga, PerulanganFor3, saya memperkenalkan penjumlahan berturut-turut dengan menggunakan perulangan for. Setiap angka yang dicetak ditambahkan ke dalam variabel jumlah, yang akhirnya menghasilkan total penjumlahan dari angka-angka tersebut. Program ini memperlihatkan bagaimana perulangan dapat digunakan tidak hanya untuk mencetak nilai, tetapi juga untuk memanipulasi data secara akurat, seperti dalam hal penjumlahan berturut-turut. Program selanjutnya, PerulanganFor4, memanfaatkan input dari pengguna untuk menentukan batas perulangan. Dengan cara ini, program menjadi lebih fleksibel karena dapat menyesuaikan jumlah iterasi berdasarkan input yang diberikan. Program ini juga menunjukkan penggunaan kondisi dalam perulangan untuk mencetak tanda "+" antar angka dan "=" setelah angka terakhir untuk menunjukkan akhir dari deret penjumlahan.

Akhirnya, pada program kelima, PerulanganFor5, saya mempelajari penggunaan perulangan bersarang (nested loop) untuk mencetak pola yang lebih kompleks. Program ini menghasilkan pola angka yang diikuti oleh titik (".") dengan jumlah titik yang berkurang setiap barisnya. Hal ini memberikan gambaran mengenai bagaimana perulangan for dapat digunakan untuk mencetak pola dengan aturan tertentu dan memberikan tantangan dalam menentukan jumlah iterasi dalam perulangan bersarang.

Secara keseluruhan, praktikum ini memperkenalkan konsep dasar perulangan dalam pemrograman, serta variasi penggunaannya dalam menyelesaikan masalah-masalah komputasi sederhana. Melalui program-program yang diuji, dapat disimpulkan bahwa perulangan for merupakan alat yang sangat berguna dan fleksibel dalam berbagai aplikasi pemrograman, dari mencetak deret angka hingga manipulasi data lebih lanjut. Setiap program yang diujikan memperlihatkan bagaimana pengaturan kondisi dan pengulangan dapat mempengaruhi hasil output, yang juga memberikan pemahaman yang lebih baik tentang bagaimana perulangan dapat diimplementasikan dalam berbagai situasi.