NAMA : ANDI RAODA TUL JANNA

NIM : 190250501010

KELAS : SISTEM INFORMASI.A

**PEMBAHASAN LOOPING (PERULANGAN)**

Looping atau Perulangan Pada Java :

Setiap kondisi loop merupakan suatu ekspresi boolean yang mengendalikan tubuh loop. Kondisi tersebut dievaluasi untuk menentukan apakah tubuh loop di eksekusi atau tidak. Jika hasil evaluasi bernilai true, maka tubuh loop dieksekusi, sedangkan jika bernilai false, maka keseluruhan loop berhenti. Pernyataan (statement) dalam tubuh loop akan dieksekusi secara berulang.

1. **Perulangan While**

Pernyataan while adalah pernyataan atau blok pernyataan yang diulang-ulang selama kondisi -loop bernilai true. Sintaks untuk perulangan while adalah sebagai berikut:

while (kondisi-loop) {

//tubuh loop

statement-statement;

}

Perhatikan contoh source code berikut ini:

int hitung = 1; //variable hitung diinisialisasikan dengan nilai 1

while ( hitung <= 10 ) {

//kondisi-loop (hitung <= 10) dievaluasi, jika bernilai true

System.out.println ("Mudah Belajar Java");

//menampilkan kalimat "Mudah Belajar Java"

hitung ++;

//menambahkan variable hitung dengan 1

} //jika kondisi bernilai false maka perulangan akan berhenti

Kesalahan pemrograman yang umum terjadi adalah loop tidak bisa berhenti atau yang biasa disebut dengan infinite loop. Yaitu kesalahan yang mengakibatkan program tidak bisa menghentikan loop karena kondisi loop selalu bernilai true.

Contoh source code di atas merupakan kondisi pengulangan yang tidak akan pernah bisa berhenti (infinite loop). Hal tersebut dikarenakan nilai hitung akan selalu bernilai <=10 (true). Penasaran? teman-teman bisa coba sendiri untuk membuktikannya.

1. **Perulangan Do While**

Struktur perulangan do-while sebenarnya mirip dengan struktur perulangan while. Namun bedanya pada perulangan do-while ini, pemeriksaan kondisi-loop dilakukan pada akhir blok perulangan dan proses perulangan akan dilakukan minimal sekali meskipun ternyata kondisinya tidak terpenuhi (bernilai false).

Untuk memahami perbedaan antara struktur while dengan do-while, coba perhatikan dua buah potongan source code program berikut:

Source Code While:

int i = 6;

while (i < 5) {

System.out.println ("Belajar Java Itu Mudah");

i++;

}

Pada contoh diatas, blok pengulangan while tidak akan dieksekusi. Hal ini disebabkan nilai variabel i lebih besar dari 5 yang menyebabkan kondisi-loop tidak terpenuhi (bernilai false).

Source Code Do-While:

int i = 6;

do {

System.out.println ("Belajar Java itu Mudah");

i++;

} while (i < 5);

Do-while melakukan pengecekan kondisi loop setelah mengeksekusi statement yang berada pada tubuh loop. Sehingga walaupun pada pengecekan kondisi loop bernilai salah (false) statement di dalam tubuh loop tetap dikerjakan.

Perhatikan contoh berikut ini:

int hitung =1;

//variable hitung diinisialisasikan dengan nilai 1

do {

//Tubuh Loop

System.out.println (hitung + " ");

//Statement yang berapada pada tubuh loop akan dieksekusi (menampilkan nilai variable hitung)

hitung++;

//Kemudian menambahkan dengan 1 (hitung++)

} while (hitung <= 10);

//Kondisi-loop dievaluasi setelah tubuh loop dieksekusi, jika

bernilai true maka tubuh loop akan kembali dieksekusi.

Jika bernilai false maka perulangan akan berhenti.

1. **Perulangan For**

Struktur for pada umumnya digunakan untuk melakukan pengulangan yang banyaknya sudah pasti atau sudah diketahui sebelumnya. Dalam pengulangan for kita harus menentukan nilai awal pengulangan dan nilai akhir pengulangan. Pengulangan for tidak membutuhkan counter untuk menaikan variabel karena sudah disebutkan pada salah satu parameter pengulangan. Bentuk umum pengulangan for adalah sebagai berikut:

for (nilai inisialisasi awal; kondisi loop; iterasi;) {

//tubuh loop, statement yang akan diulang

}

Proses pengulangan akan terus dilakukan selama kondisi loop bernilai true. Dengan kata lain, proses pengulangan hanya akan dihentikan apabila kondisi telah bernilai false atau sudah tidak terpenuhi lagi. Pengulangan for biasanya menggunakan suatu variabel untuk mengendalikan berapa kali tubuh loop akan dieksekusi dan menentukan kapan loop akan berhenti. Variabel ini disebut juga dengan variabel kontrol.

Nilai inisialisasi awal merupakan variabel kontrol, proses inisialisasi nilai awal hanya akan dilakukan sekali saja. Iterasi berfungsi menaikan (increment) nilai variabel kontrol dan kondisi loop mengevaluasi apakah kondisi perulangan bernilai true atau false.

Contoh Program Pengulangan For:

public class contoh\_for{

public static void main (String[] args) {

System.out.println ("Pengulangan Dengan For Arah Indeks Naik");

for (int i = 1; i <= 5; ++i)

System.out.println ("Indeks Naik ke = "+i);

System.out.println ("Pengulangan Dengan For Arah Indeks Turun");

for (int i = 5; i > 0; --i)

System.out.println ("Indeks Naik ke = "+i);

}

}

Hasil (Output) dari contoh program diatas adalah:

Pengulangan Dengan For Arah Indeks Naik

Indeks Naik ke = 1

Indeks Naik ke = 2

Indeks Naik ke = 3

Indeks Naik ke = 4

Indeks Naik ke = 5

Pengulangan Dengan For Arah Indeks Turun

Indeks Naik ke = 5

Indeks Naik ke = 4

Indeks Naik ke = 3

Indeks Naik ke = 2

Indeks Naik ke = 1