




Back End Development 1 ○

+

Sesi 3



Fundamental⁺ Programming with Looping

Fundamental Programming with Looping - Sesi 3

Looping Introduction

Perulangan atau Loop adalah sebuah hal yang digunakan untuk mengeksekusi sebuah set dari suatu instruksi atau fungsi secara berulang ketika beberapa kondisi menjadi nyata (true).

Ada dua jenis atau tipe perulangan pada Java.

1. perulangan java for,
2. perulangan java while.

Namun perulangan ini memiliki turunan yaitu, perulangan java do-while.

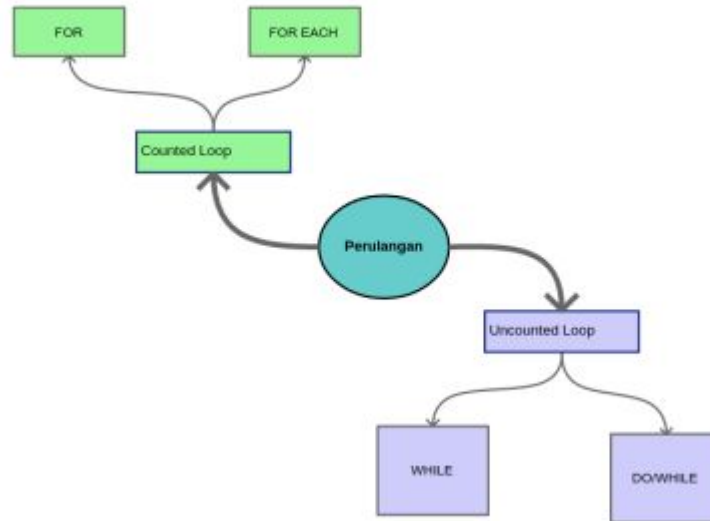
Berikut kami jelaskan perbedaannya:

1. Definisi masing-masing perulangan.
 - **Perulangan for** adalah pernyataan aliran kontrol yang mengulang bagian dari program beberapa kali.
 - **Perulangan while** adalah pernyataan aliran kontrol yang mengeksekusi bagian dari program berulang kali berdasarkan kondisi boolean yang diberikan.
 - **Perulangan do-while** adalah pernyataan aliran kontrol yang mengeksekusi bagian dari program setidaknya satu kali dan eksekusi selanjutnya bergantung pada kondisi boolean yang diberikan.



2. Waktu Penggunaan.

- Jika jumlah iterasi tetap, disarankan untuk menggunakan **perulangan for**.
- Jika jumlah iterasi tidak tetap, disarankan untuk menggunakan **perulangan while**.
- Jika jumlah iterasi tidak tetap dan Anda harus menjalankan loop sekali, disarankan untuk menggunakan **perulangan do-while**.



Perulangan terbagi menjadi 2 jenis :

- **Counted Loop** : Perulangan yang jumlah pengulangannya terhitung atau tentu, terdiri dari perulangan for dan for-each
- **Uncounted Loop** : Perulangan yang jumlah pengulangannya tidak tentu, terdiri dari perulangan do dan do/while

Fundamental Programming with Looping - Sesi 3

A. Perulangan For

Perulangan ini digunakan untuk mengiterasi bagian dari program beberapa kali. Jika angka dari iterasi sudah pasti, maka direkomendasikan untuk menggunakan perulangan for.

```
1  for( int hitungan = 0; hitungan <= 10; hitungan++ ){  
2      // blok kode yang akan diulang  
3  }
```

- Variabel hitungan tugasnya untuk menyimpan hitungan pengulangan.
- Hitungan <= 10 artinya selama nilai hitungannya lebih kecil atau sama dengan 10, maka pengulangan akan terus dilakukan. Dengan kata lain, perulangan ini akan mengulang sebanyak 10 kali.
- Hitungan++ fungsinya untuk menambah satu(+1) nilai hitungan pada setiap pengulangan.
- Blok kode For dimulai dengan tanda '{' dan diakhiri dengan '}'.



Contoh :

```
for (int i = 0; i < 2; i++) {}
```

Perulangan for memiliki tiga bagian:

Bagian pertama menjalankan sekali ketika Anda memasukan perulangan.

Bagian kedua adalah *gate keeper*, jika mengembalikan nilai true, maka Anda akan menjalankan pernyataan di loop, jika mengembalikan false, maka Anda akan keluar dari loop.

Hal ini pun berjalan tepat setelah bagian pertama untuk pertama kalinya, kemudian setiap kali loop selesai dan bagian ketiga dijalankan.

Bagian ketiga adalah pernyataan terakhir yang akan dijalankan setiap kali loop berjalan.

```
int i = 0;
i < 2 // 0 < 2 = true
    // perulangan lanjut
i++ // i bernilai 1 sekarang
i < 2 // 1 < 2 = true
    // perulangan lanjut
i++ // i bernilai 2 sekarang
i < 2 // 2 < 2 = false
    // Perulangan selesai
```

Kita dapat menghilangkan bagian pertama dan ketiga dari perulangan (meskipun hal ini akan membuatnya terlihat aneh), dan perulangan akan tetap bekerja:

```
for (;i < 3;) {}
```



Fundamental Programming with Looping - Sesi 3

Latihan

Buatlah script code dalam java menggunakan perulangan for untuk menampilkan output berikut :

```
12 *****  
13 *****  
14 *****  
15 *****  
16 *****  
17 *****
```

A. Tampilkan Bintang

```
31  
32 0 1 2 3 4 5 6 7 8 9 10
```

B. Tampilkan nilai 1 - 10

```
44 Hasil output:  
45  
46 1 3 5 7 9 11 13 15 17 19
```

C. Tampilkan Bilangan Ganjil Saja dari 1 - 20

Fundamental Programming with Looping - Sesi 3

Perulangan For-Each

Perulangan FOR-EACH digunakan untuk menampilkan isi array. Perulangan juga dilakukan dengan kata kunci For.

```
31  for ( int item : dataArray ) {  
32      // blok kode yang diulang  
33  }
```

Penjelasan :

- Variabel item akan menyimpan nilai dari array.
- Untuk setiap item dalam dataArray, maka lakukan perulangan.

```
1  public class PerulanganForeach {  
2      public static void main(String[] args) {  
3  
4          // membuat array  
5          int angka[] = {3,1,42,24,12};  
6  
7          // menggunakan perulangan For each untuk menampilkan angka  
8          for( int x : angka ){  
9              System.out.print(x + " ");  
10         }  
11  
12     }  
13 }
```



Fundamental Programming with Looping - Sesi 3

Break & Continue

Kata kunci ini membantu Anda untuk mengontrol perulangan di dalamnya. Break akan menyebabkan loop berhenti dan akan segera beralih ke pernyataan berikutnya setelah perulangan.

```
int i;
for (i = 0; i < 4; i++) {
    if (i >= 3) {
        break;
    }
    System.out.println("Selamat Belajar!");
}
System.out.println(i);
// Output:
// Selamat Belajar!
// Selamat Belajar!
// Selamat Belajar!
// 3
```

Kata kunci 'continue' akan menghentikan iterasi saat ini dan akan pindah ke yang berikutnya. Perhatikan bahwa di dalam perulangan for, hal ini masih akan menjalankan bagian ketiga.

```
int i;
for (i = 0; i < 4; i++) {
    if (i >= 3) {
        break;
    }
    System.out.println("Selamat Belajar!");
    if (i >= 2) {
        continue;
    }
    System.out.println("JAVA");
}
System.out.println(i);
// Output
// Selamat Belajar!
// JAVA
// Selamat Belajar!
// JAVA
// Selamat Belajar!
// 3
```

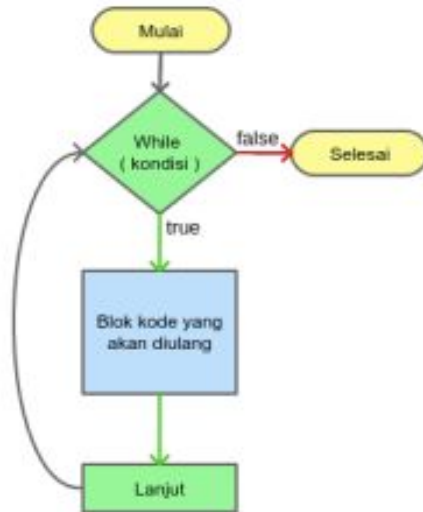


Fundamental Programming with Looping - Sesi 3

Uncounted Loop

Cara kerja perulangan ini seperti percabangan, ia akan melakukan perulangan selama kondisinya bernilai true.

Perulangan While



Perulangan Do/While



```
12  while ( kondisi ) {  
13      // blok kode yang akan diulang  
14  }
```

Struktur Penulisan While

Kondisi ini hanya memiliki nilai true & false, perulangan while akan berhenti sampai kondisi bernilai false.

Perulangan while digunakan untuk mengiterasi bagian dari program beberapa kali. Sama seperti perulangan for, perulangan while direkomendasikan untuk Anda gunakan ketika angka dari iterasi belum pasti.

Hal inilah yang menyebabkan syntax dari perulangan ini sangat mirip dengan perulangan For.

```
//while loop
int i=1;
while (i<=4) {
    System.out.println(i);
    i++;
}
```

Kemudian, perulangan while memiliki turunan yakni perulangan Java do while yang digunakan untuk mengiterasi bagian dari program beberapa kali.

Gunakan perulangan ini angka dari perulangan belum pasti dan Anda diharuskan untuk mengeksekusi perulangan setidaknya satu kali.



```

1  import java.util.Scanner;
2
3  public class PerulanganWhile {
4      public static void main(String[] args) {
5
6          // membuat variabel dan scanner
7          boolean running = true;
8          int counter = 0;
9          String jawab;
10         Scanner scan = new Scanner(System.in);
11
12         while( running ) {
13             System.out.println("Apakah anda ingin keluar?");
14             System.out.print("Jawab [ya/tidak]> ");
15
16             jawab = scan.nextLine();
17
18             // cek jawabannya, kalau ya maka berhenti mengulang
19             if( jawab.equalsIgnoreCase("ya") ){
20                 running = false;
21             }
22
23             counter++;
24         }
25
26         System.out.println("Anda sudah melakukan perulangan sebanyak " + counter + " kali");
27     }
28 }
29

```

Pada script diatas telah dilakukan perulangan sebanyak 4 kali, bisa saja terjadi 10 kali. Kalau nilai variabel running bernilai false, maka perulangan berhenti

// Contoh kode while di atas dapat kita baca seperti ini:
 “Lakukan perulangan selama nilai running bernilai true.”



Fundamental Programming with Looping - Sesi 3

Latihan

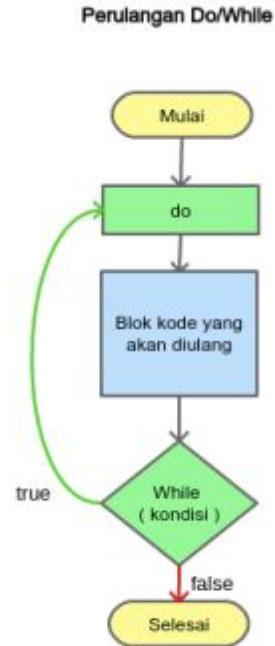
Buat lah script code dalam java menggunakan perulangan while untuk menampilkan output seperti dibawah ini : .

```
54 Hasil Output :  
55 Perulangan ke-0  
56 Perulangan ke-1  
57 Perulangan ke-2  
58 Perulangan ke-3  
59 Perulangan ke-4  
60 Perulangan ke-5  
61 Perulangan ke-6  
62 Perulangan ke-7  
63 Perulangan ke-8  
64 Perulangan ke-9  
65 Perulangan ke-10
```

Fundamental Programming with Looping - Sesi 3

Perulangan Do - While

Cara kerja perulangan ini seperti percabangan, ia akan melakukan perulangan selama kondisinya bernilai true.



Cara kerja perulangan Do/While sebenarnya sama seperti perulangan While. Bedanya, Do/While melakukan satu kali perulangan dulu. Kemudian mengecek kondisinya.

```
1  do {  
2      // blok kode yang akan diulang  
3  } while (kondisi);
```

Jadi kerjakan dulu do baru di cek kondisi while-nya.

Kalo kondisi bernilai true, maka lanjut perulangan.

Buat lah script code dalam java menggunakan perulangan do - while untuk menampilkan output seperti dibawah ini : .

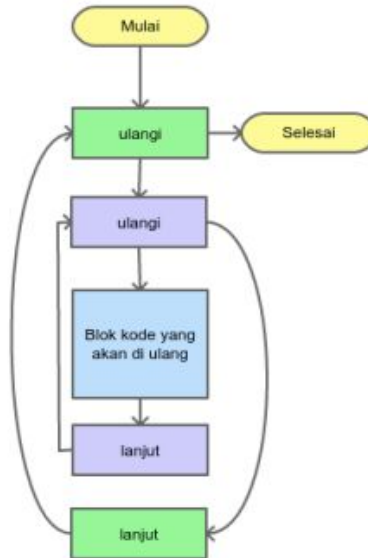
```
54  Hasil Output :  
55  Perulangan ke-0  
56  Perulangan ke-1  
57  Perulangan ke-2  
58  Perulangan ke-3  
59  Perulangan ke-4  
60  Perulangan ke-5  
61  Perulangan ke-6  
62  Perulangan ke-7  
63  Perulangan ke-8  
64  Perulangan ke-9  
65  Perulangan ke-10
```

Fundamental Programming with Looping - Sesi 3

Nested Loop

Perulangan juga dapat bersarang. Perulangan bersarang maksudnya, perulangan dalam perulangan atau disebut juga nested loop.

Perulangan Bersarang



```
1 public class Nestedloop {
2     public static void main(String[] args) {
3
4         // membuat variabel
5         int x, y;
6
7         // melakukan parulang sebanyak x dan y kali
8         for (x = 0; x <= 5; x++){
9             for( y = 0; y <= 3; y++){
10                 System.out.format("Perulangan [x=%d, y=%d] %n", x, y);
11             }
12         }
13     }
14 }
```

Penjelasan

- Perulangan bersarang sering digunakan pada array multi dimensi.
- Jenis perulangan di dalam perulangan bisa berbeda, misalnya di dalam perulangan while ada perulangan for.

