

MODUL

PEMOGRAMAN JAVA

Oleh:

Chalifa Chazar

Pertemuan 5

Looping Statement

Tujuan:

- » Mahasiswa mampu memahami konsep struktur kondisi FOR, WHILE dan DO-WHILE
- » Mahasiswa mampu menggunakan struktur kondisi FOR, WHILE dan DO-WHILE

Pustaka:

- » Bambang Hariyanto (2014): **Esensi-Esensi Bahasa Pemograman Java (Revisi Keempat)**. Informatika. Bandung.

Pada suatu kondisi tertentu, kadang kita ingin memerintahkan komputer untuk dapat menjalankan perintah secara berulang-ulang. Pengulangan juga dikenal dengan istilah "looping", adalah sebuah kondisi dimana suatu alur program akan dijalankan secara terus-menerus sampai memenuhi suatu kondisi akhir. Pada Java terdapat beberapa jenis pernyataan untuk menyatakan suatu pengulangan, yaitu dengan menggunakan pernyataan FOR, pernyataan WHILE dan pernyataan DO-WHILE.

1. Pernyataan FOR

Pernyataan FOR di Java menyediakan sarana mengulang kode sejumlah tertentu. Pengulangan ini terstruktur untuk mengulangi kode sampai tercapai batas. Berikut ini adalah bentuk sintaks penulisan dari pernyataan FOR.

```
for (int Angka = 0; Angka <= 10; Angka++) {  
    //kode yang akan diulang  
}
```

Berikut ini adalah penjelasan dari sintaks di atas:

- » Variabel Angka tugasnya untuk menyimpan hitungan pengulangan.

- » Angka ≤ 10 artinya selama nilai angkanya lebih kecil atau sama dengan 10, maka pengulangan akan terus dilakukan. Artinya pengulangan ini akan dilakukan sebanyak 10 kali.
- » Angka++ fungsinya untuk menambah satu (+1) nilai Angka pada setiap pengulangan.
- » Blok kode FOR dimulai dengan tanda '{' dan diakhiri dengan tanda '}'.

Perhatikan contoh berikut ini.

```
public class Pengulangan {  
    public static void main(String[] args) {  
        for (int i=0; i<=10; i++) {  
            System.out.println("script.id");  
        }  
    }  
}
```

2. Pernyataan WHILE

Pengulangan WHILE mempunyai kondisi pengendali pernyataan berulang. Berikut ini adalah bentuk sintaks penulisan dari pernyataan WHILE.

```
while (kondisi) {  
    //kode yang akan diulang  
}
```

Berikut ini adalah penjelasan dari sintaks di atas:

- » Kondisi dapat diisi dengan perbandingan maupun variabel boolean. Kondisi ini hanya memiliki nilai TRUE dan FALSE.
- » Pengulangan WHILE akan berhenti sampai kondisi bernilai FALSE.

Perhatikan contoh berikut ini.

```
public static void main(String[] args) {  
    int angka = 1;  
    while (angka <=5) {  
        System.out.println("pengulangan ke-" + angka);  
  
        angka++;  
    }  
}
```

3. Pernyataan DO-WHILE

Pengulangan DO-WHILE mirip dengan pengulangan WHILE hanya saja terdapat pemeriksaan kondisi terlebih dahulu sebelum melakukan pengulangan berikutnya. Berikut ini adalah bentuk sintaks penulisan dari pernyataan DO-WHILE.

```
do {  
    //kode yang akan diulang  
} while (kondisi);
```

Berikut ini adalah penjelasan dari sintaks di atas:

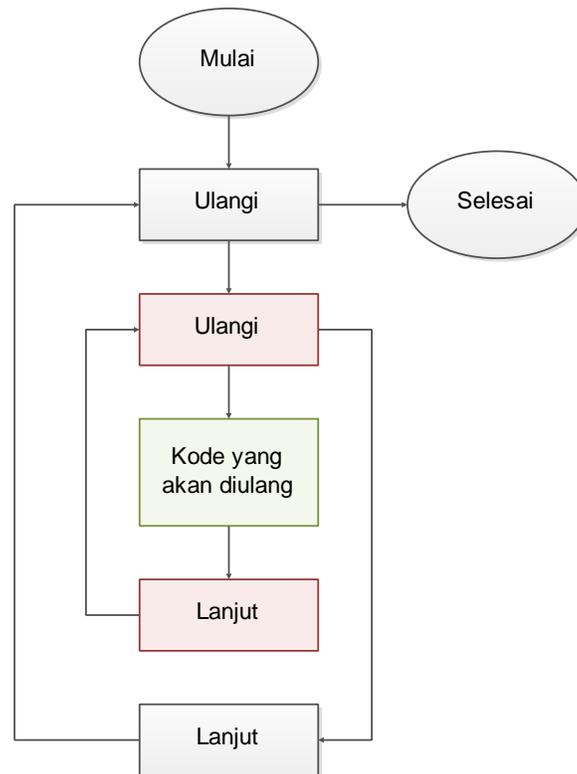
- » Komputer akan menjalankan DO terlebih dahulu, selanjutnya mengecek WHILE (kondisi).
- » Apabila kondisi bernilai TRUE, maka pengulangan akan terus dilakukan

Perhatikan contoh berikut ini.

```
public static void main(String[] args) {  
    int angka = 1;  
    do {  
        System.out.println ("pengulangan ke-" + angka);  
        angka++;  
    } while (angka <= 5);  
}
```

4. Pengulangan Bersarang

Java memungkinkan adanya pengulangan di dalam pengulangan lainnya atau dikenal dengan pengulangan bersarang (nested loop). Berikut ini flowchat dari bentuk nested loop.



Berikut ini adalah contoh nested looping.

```

public static void main(String[] args) {
    int x, y;
    for (x = 0; x <= 3; x++){
        for (y = 1; y <= 3; y++) {
            System.out.format("pengulangan [x=%d, y=%d] %n", x, y);
        }
    }
}
    
```

Catatan!

- » Pengulangan bersarang sering digunakan pada array multidimensi

- » Jenis nested looping dapat dilakukan dalam pernyataan pengulangan yang berbeda, misalnya di dalam pernyataan WHILE terdapat pernyataan FOR.

5. Pernyataan FOR-EACH

Terdapat pula jenis pengulangan dengan menggunakan pernyataan FOR-EACH. Umumnya pernyataan FOR-EACH digunakan untuk menampilkan isi dari suatu array (akan dibahas pada modul selanjutnya). Intinya array adalah suatu variabel yang dapat menyimpan lebih dari satu nilai dan memiliki suatu index.

Berikut ini adalah bentuk sintaks penulisan dari pernyataan FOR-EACH.

```
for (int item: dataArray) {  
    //kode yang akan diulang  
}
```

Berikut ini adalah penjelasan dari sintaks di atas:

- » Variabel item akan menyimpan nilai dari array.
- » Sintaks di atas dapat dibaca "untuk setiap item dalam dataArray, maka lakukan pengulangan".

Perhatikan contoh berikut ini.

```
public static void main(String[] args) {  
    int angka[] = {1,3,5,7,9};  
    for (int x : angka) {  
        System.out.println(x + " ");  
    }  
}
```

Latihan!

1. Buatlah sebuah aplikasi untuk menghitung kelipatan pangkat sampai dengan 1000.
2. buatlah sebuah aplikasi untuk menampilkan hasil sebagai berikut:

```
*  
**  
***  
****  
*****
```

----- GOOD LUCK -----