

# **PENGULANGAN (LOOPING)**

Dosen : Sulistyowati, ST., M.Kom.

# STATEMENT PENGULANGAN

- Pengulangan Digunakan untuk melakukan sekelompok langkah/instruksi secara berulang-ulang. Sekelompok langkah/instruksi yang diulang ini disebut *loop body*.
- Ada 2 macam bentuk perulangan, yaitu :
  1. Perulangan Deterministik  
Adalah bentuk perulangan dimana jumlah perulangan sudah dapat diketahui pasti.  
Bentuk : for
  2. Perulangan Non Deterministik  
Adalah bentuk perulangan dimana jumlah perulangan berdasarkan kondisi (belum dapat diketahui pasti).  
Bentuk : while dan do-while

# STATEMENT PENGULANGAN - WHILE

Statement while adalah statement pengulangan yang jumlah perulangannya tergantung pada suatu kondisi logika tertentu. Bentuk umum konstruksi while adalah sebagai berikut :

```
while (boolean_expression)
{
    code_block;
}
```

# STATEMENT PENGULANGAN - WHILE

Pada konstruksi while, langkah-langkah proses perulangannya adalah sebagai berikut :

- Sistem memeriksa *boolean\_expression*.
- Jika nilai *boolean\_expression* adalah *true*, maka *code\_block* akan dieksekusi. Jika tidak, maka *code\_block* tidak dieksekusi.
- Jika di dalam *code\_block* terdapat pernyataan kondisi yang menyebabkan proses harus keluar dari blok while, maka proses akan keluar dari pengulangan, meskipun *boolean\_expression* masih bernilai *true*.

# CONTOH WHILE

```
public class Contoh1{  
    public static void main(String[] args){  
        int num1 = 0;  
        int num2 = 23;  
        int num3 = num1+num2;  
        while(num3 > num1){  
            num2-=3;  
            num1+=2;  
            num3 = num1+num2;  
            System.out.println("nilai 1 = "+num1+"  
            ,nilai 2 = "+num2+" , nilai 3 = "+num3);  
        }  
    }  
}
```

# CONTOH WHILE

```
public class Contoh2 {  
    public static void main(String[] args) {  
        int vari = 20;  
        while(true) {  
            System.out.println("Nilai variable = " +vari);  
            --vari;  
            if(vari<10) {  
                break;  
            }  
        }  
    }  
}
```

# STATEMENT PENGULANGAN - FOR

Bentuk umum dari statement *for* adalah :

```
for (initialize[, initialize] ;  
    boolean_expression; update[, update] )  
{  
    code_block;  
}
```

- bagian inisialisasi, yang berisi pernyataan pemberian nilai awal untuk suatu variabel parameter;
- bagian *boolean\_expression*, yang berisi pernyataan logika yang akan diperiksa, sebagai syarat pengulangan terus dilanjutkan. Pengulangan akan dilanjutkan jika nilai ekspresi boolean pada segmen ini bernilai *true*.
- bagian *update*, yang berisi pernyataan *updating* parameter ketika satu putaran pada pengulangan selesai dieksekusi;

# STATEMENT PENGULANGAN - FOR

- bagian inisialisasi dapat diisi dengan lebih dari 1 pernyataan inisialisasi;
- bagian *boolean\_expression* hanya dapat diisi oleh 1 pernyataan logika;
- bagian *update* dapat diisi dengan lebih dari 1 pernyataan update.



# CONTOH FOR

```
public class Contoh3{  
    public static void main(String[] args){  
        for(int vari=20; vari>=10;vari--){  
            System.out.println("Nilai variable =" + vari);  
        }  
    }  
}
```

# CONTOH FOR

```
public class Contoh4{  
    public static void main(String[] args){  
        for(int A=20,B=0;A>=10&&B<=5;A--,B++){  
            System.out.println("Nilai  
variable1= " + A);  
            System.out.println("Nilai  
variable2= " + B);  
        }  
    }  
}
```

# CONTOH FOR

```
public class Contoh5{  
    public static void main(String[] args){  
        int var = 20;  
        for( ; ; ){  
            System.out.println("Nilai variabel= "+var);  
            var--;  
            if(var<10)break;  
        }  
    }  
}
```

# STATEMENT PENGULANGAN – DO..WHILE

Statement pengulangan do/while mirip dengan statement while. Perbedaannya adalah pada urutan prosesnya, yaitu :

- Looping dijalankan terlebih dahulu;
- Dilakukan pemeriksaan kondisi.

# CONTOH DO..WHILE

```
public class Contoh6{  
    public static void main(String[] args){  
        int variable = 20;  
        do{  
            System.out.println("Nilai variable1= "+  
variable);  
            variable--;  
        }while(variable>0);  
    }  
}
```