

LOOPING & CONDITIONAL

Disusun Oleh:
Reza Budiawan

Untuk:
Tim Dosen Algoritma & Pemrograman Lanjut

Hanya dipergunakan untuk kepentingan pengajaran di lingkungan Fakultas Ilmu Terapan, Universitas Telkom

Pengkondisian (if-else)

- Bentuk Umum 1

```
if(kondisi_benar) {  
    statement;  
}
```

- Hal yang harus diperhatikan:
 - Statemen dalam kurung kerawal if hanya dieksekusi jika kondisi benar terpenuhi
 - Kurung kerawal bersifat optional jika statemen yang berada pada kurung kerawal hanya 1.

Pengkondisian (if-else)

- Bentuk Umum 2 & 3

```
if(kondisi_benar) {  
    statement_1;  
}else{  
    statement_2;  
}
```

```
if(kondisi_benar_1){  
    statement_1;  
}else if(kondisi_benar_2){  
    statement_2;  
}else{  
    statement_3  
}
```

Pengkondisian (switch-case)

- Bentuk Umum:

```
switch(x) {  
    case 1: statement_1_1;  
           statement_1_2;  
           break;  
    case 2: statement_2_1;  
           statement_2_2;  
           break;  
    .....  
    default: statement;  
}
```

Perulangan (while)

- Bentuk Umum

```
while (kondisi_benar) {  
  
}
```

- Hal yang harus diperhatikan:

- Kondisi_benar harus memiliki moment menjadi 'salah' agar perulangan berhenti (tidak mencapai endless looping)
- Perulangan dapat dihentikan walau tidak mencapai kondisi salah, menggunakan "break"

Contoh While

```
int x = 5;
while(x>0) {
    System.out.println("x: "+x);
    x--;
    if(x==2) {
        break;
    }
}
```

```
Output - AlproLanjutGenap1516 (run) ✖
run:
x: 5
x: 4
x: 3
BUILD SUCCESSFUL (total time: 0 seconds)
```

```
int x = 5;
while(x>0) {
    System.out.println("x: "+x);
    x--;
}
```

```
Output - AlproLanjutGenap1516 (run) ✖
run:
x: 5
x: 4
x: 3
x: 2
x: 1
BUILD SUCCESSFUL (total time: 0 seconds)
```

Perulangan (do-while)

- Bentuk Umum

```
do{  
  
}while(kondisi_benar);
```

- Hal yang harus diperhatikan:

- Kondisi_benar harus memiliki moment menjadi 'salah' agar perulangan berhenti (tidak mencapai endless looping)
- Perulangan dapat dihentikan walau tidak mencapai kondisi salah, menggunakan "break"

Contoh do-while

```
int x = 5;
do{
    System.out.println("x: "+x);
    x--;
}while(x>0);
```

```
Output - AlproLanjutGenap1516 (run) ✖
run:
x: 5
x: 4
x: 3
x: 2
x: 1
BUILD SUCCESSFUL (total time: 0 seconds)
```

```
int x = 5;
do{
    System.out.println("x: "+x);
    x--;
    if(x==2){
        break;
    }
}while(x>0);
```

```
Output - AlproLanjutGenap1516 (run) ✖
run:
x: 5
x: 4
x: 3
BUILD SUCCESSFUL (total time: 0 seconds)
```


Perulangan (for)

- Bentuk Umum 1

```
for(inisialisasi;kondisi_berhenti;kontrol_var) {  
  
}
```

- Hal yang harus diperhatikan:
 - Kondisi_berhenti harus terpenuhi untuk menghentikan perulangan.
 - Perulangan dapat dihentikan walau tidak mencapai kondisi salah, menggunakan “break”

Contoh perulangan for

```
for(int x=5;x>0;x--){  
    System.out.println("x: "+x);  
}
```

```
Output - AlproLanjutGenap1516 (run) ✖  
run:  
x: 5  
x: 4  
x: 3  
x: 2  
x: 1  
BUILD SUCCESSFUL (total time: 0 seconds)
```

```
for(int x=5;x>0;x--){  
    System.out.println("x: "+x);  
    if(x==2){  
        break;  
    }  
}
```

```
Output - AlproLanjutGenap1516 (run) ✖  
run:  
x: 5  
x: 4  
x: 3  
x: 2  
BUILD SUCCESSFUL (total time: 0 seconds)
```

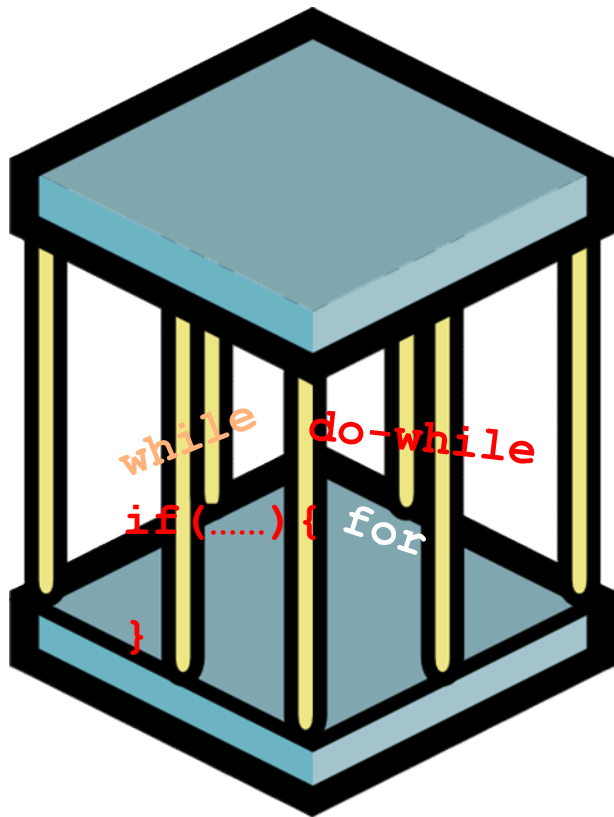
Perulangan (for)

- Bentuk Umum 2

```
for(tipeData var_temp : list){  
  
}
```

- Catatan: Bentuk ini dipelajari saat pembahasan array/array list

Perulangan & Pengkondisian Bersarang



END OF SLIDE...
