



Java Fundamental | Bootcamp

Nested Loop

Subjects : Loop Condition

1. While (If-Else)
2. Do-While (If-Else)
3. For (If-Else)
4. Pengenalan Nested Loop

While (If-Else)

Syntax kondisi (If-Else) juga bisa digunakan bersama dengan looping, jika kita ingin membuat looping untuk iterasi yang tidak pasti namun memiliki kondisi didalamnya, kita bisa menggabungkan while dan condition.

While (If-Else)

```
public class KelasWhileVariasi {  
    Run | Debug  
    public static void main(String[] args) {  
        int noAntrian = 1;  
        while (noAntrian < 5) {  
            if (noAntrian > 3) {  
                System.out.println(noAntrian + " Tidak Dapat Hadiah");  
            } else {  
                System.out.println(noAntrian + " Dapat Hadiah");  
            }  
            noAntrian++;  
        }  
    }  
}
```

1 Dapat Hadiah
2 Dapat Hadiah
3 Dapat Hadiah
4 Tidak Dapat Hadiah

Do-While (If-Else)

Begitu juga dengan Do-While, jika kita ingin melakukan perulangan yang block code nya akan dieksekusi terlebih dahulu dan dalam block code tersebut memerlukan suatu kondisi tertentu, maka kita bisa gunakan if-else didalam do-while.

Do-While (If-Else)

```
public class KelasDoWhileVariasi {  
    Run | Debug  
    public static void main(String[] args) {  
        int noAntrian = 1;  
        do {  
            if (noAntrian > 3) {  
                System.out.println(noAntrian + " Tidak Dapat Hadiah");  
            } else {  
                System.out.println(noAntrian + " Dapat Hadiah");  
            }  
            noAntrian++;  
        } while (noAntrian < 5);  
    }  
}
```


1 Dapat Hadiah
2 Dapat Hadiah
3 Dapat Hadiah
4 Tidak Dapat Hadiah

For (If-Else)

Selain While dan Do-While, kita juga dapat memasukkan kondisi ke dalam for, jika kita akan melakukan looping dengan iterasi yang sudah ditentukan, maka kita dapat melakukannya menggunakan for, kita dapat menambahkan kondisi if-else didalamnya jika diperlukan.

For (If-Else)


```
public class KelasForVariasi {  
    Run | Debug  
    public static void main(String[] args) {  
        int noAntrian = 1;  
        for (int i = 0; i < 4; i++) {  
            if (noAntrian > 3) {  
                System.out.println(noAntrian + " Tidak Dapat Hadiah");  
            } else {  
                System.out.println(noAntrian + " Dapat Hadiah");  
            }  
        }  
    }  
}
```



```
1 Dapat Hadiah  
1 Dapat Hadiah  
1 Dapat Hadiah  
1 Dapat Hadiah
```

For (If-Else)

```
public class KelasForVariasi {  
    Run | Debug  
    public static void main(String[] args) {  
        int noAntrian = 1;  
        for (int i = 0; i < 4; i++) {  
            if (noAntrian > 3) {  
                System.out.println(noAntrian + " Tidak Dapat Hadiah");  
            } else {  
                System.out.println(noAntrian + " Dapat Hadiah");  
            }  
            noAntrian++;  
        }  
    }  
}
```



1 Dapat Hadiah
2 Dapat Hadiah
3 Dapat Hadiah
4 Tidak Dapat Hadiah

Tabel Nested Loop



		for 2 (dalam for 1) J		
		j	j	j
for 1 (I)	1	0	1	2
	2	0	1	2
	3	0	1	2
	4	0	1	2
	5	0	1	2

```
for (int i = 1; i <= 5; i++) {  
    System.out.println("loop luar");  
    for (int j = 0; j < 3; j++) {  
        System.out.println("loop dalam");  
    }  
}
```

i = baris, j = kolom.

Kita bisa lihat setiap 'i' dieksekusi maka otomatis 'j' akan di jalankan hingga tuntas / hingga iterasi terakhir dari 'j'.

Jika ganti baris (i+1) maka 'j' akan diulangi lagi dari awal hingga expression false.