

BAB V PENYELEKSIAN KONDISI DAN PERULANGAN

Untuk menghasilkan suatu program, sangat penting untuk mengatur agar program dapat berjalan dengan aliran atau susunan yang baik, sehingga dapat memecahkan masalah atau menciptakan solusi dari suatu permasalahan. Ada dua macam kontrol alur (*control flow*) yang digunakan dalam pemrograman Java, yaitu pengkondisian (*conditional*) dan pengulangan (*looping*).

A. Pengkondisian (*Conditional*)

Conditional merupakan suatu pengaturan alur program berdasar kondisi boolean yang dijadikan patokan.

1. Pengkondisian Dengan *if*

Pengaturan alur program seringkali berdasar dari ekspresi tertentu yang menghasilkan nilai boolean. Apabila boolean bernilai *true*, maka blok pernyataan akan dijalankan. Pengkondisian semacam ini dapat dilakukan dengan *if*.

Sintaks :

if (<ekspresi boolean>)

{

 <pernyataan>

}

Pernyataan di antara tanda { } akan dijalankan apabila ekspresi boolean bernilai true atau benar, sedangkan untuk ekspresi boolean terdapat di dalam tanda ().

Latihan 14. SeleksiIf.java

```
class SeleksiIf {  
    public static void main(String[] args) {  
        int a,b;  
        a = 20;  
        b = a/2;  
        if(b>5){  
            System.out.println("Pernyataan1 Test Seleksi If dieksekusi");  
        }  
        System.out.println("Pernyataan2 Test Seleksi If dieksekusi");  
    }  
}
```

```

Command Prompt
G:\JAVUA>javac SeleksiIf.java
G:\JAVUA>java SeleksiIf
Pernyataan1 Test Seleksi If dieksekusi
Pernyataan2 Test Seleksi If dieksekusi

```

2. Pengkondisian Dengan *if* dan *else*

Pada pengkondisian dengan menggunakan *if*, tidak jelas apa yang akan dilakukan apabila ekspresi boolean bernilai false. Apabila ingin memproses suatu ekspresi boolean yang bernilai false, dapat menambahkan *else*.

Sintaks :

```

if (<ekspresi boolean>)
{
    <pernyataan>
}
else
{
    <pernyataan>
}

```

Pernyataan pada blok *if* sebelum *else* akan dijalankan apabila ekspresi boolean bernilai true, sedangkan apabila ekspresi boolean bernilai false, maka akan dijalankan pernyataan yang ada setelah else.

Latihan 15. SeleksiIfElse.java

```

class SeleksiIfElse {
    public static void main(String[] args) {
        int x;
        x = 10;
        if(x>5) {
            System.out.println("Pernyataan1a    Test    Seleksi    IfElse
                                dieksekusi");
            System.out.println("Pernyataan1b    Test    Seleksi    IfElse
                                dieksekusi");
        }
        else {
            System.out.println("Pernyataan2a    Test    Seleksi    IfElse
                                dieksekusi");
        }
    }
}

```

```

        System.out.println("Pernyataan2b    Test    Seleksi    IfElse
        dieksekusi");
    }
}
}

```

```

G:\JAVUA>javac SelekstIfElse.java
G:\JAVUA>java SelekstIfElse
Pernyataan1a Test Seleksi IfElse dieksekusi
Pernyataan1b Test Seleksi IfElse dieksekusi

```

3. Pengkondisian Dengan *if* dan *else if*

Selain *if*, *if* dan *else*, dapat juga menggunakan kombinasi *if else* secara berkelanjutan. Misalnya apabila ingin melakukan pengkondisian yang melakukan pengecekan nilai boolean sebanyak tiga kali, maka kita dapat melakukan seperti berikut :

```

if (<ekspresi boolean>)
{
    <pernyataan>
}
else if (<ekspresi boolean>)
{
    <pernyataan>
}
else if (<ekspresi boolean>)
{
    <pernyataan>
}
else
{
    <pernyataan>
}

```

Latihan 16. SelekstIfElseIf.java

```

class SelekstIfElseIf{
    public static void main(String[] args) {
        int bulan = 12;
        if(bulan<=3)

```

```

        System.out.println("Kuartal 1 Test Seleksi IfElseIf");
    else if(bulan<=6)
        System.out.println("Kuartal 2 Test Seleksi IfElseIf");
    else if(bulan<=9)
        System.out.println("Kuartal 3 Test Seleksi IfElseIf");
    else
        System.out.println("Kuartal 4 Test Seleksi IfElseIf");
    }
}

```



```

G:\JAWA>javac SeleksiIfElseIf.java
G:\JAWA>java SeleksiIfElseIf
Kuartal 4 Test Seleksi IfElseIf

```

4. Pengkondisian Dengan Switch

Switch digunakan untuk memilih berdasar banyak pilihan, dimana dari masing-masing pilihan akan menghasilkan pernyataan atau aksi yang berlainan sesuai pilihan yang dilakukan. Biasanya penggunaan switch berkaitan dengan menu, dimana apabila menu pertama dipilih, maka switch akan menjalankan blok dari case 1, lalu apabila menu kedua dipilih, maka switch akan menjalankan blok dari case 2, demikian seterusnya. Bila tidak memenuhi kriteria menu case manapun, switch akan menjalankan blok default.

Sintaks penggunaan switch adalah :

```

Switch (<ekspresiinteger>)
{
    Case <nilaivariabel>: <blok pernyataan>
                        Break;
    Case <nilaivariabel>: <blok pernyataan>
                        Break;
    Case <nilaivariabel>: <blok pernyataan>
                        Break;
    default: <blok pernyataan>
            break;
}

```

Latihan 17. SeleksiSwitch.java

```
class SeleksiSwitch {  
    public static void main(String[] args) {  
        int bulan, year;  
        bulan = 1;  
        year = 2005;  
        switch(bulan) {  
            case 1: switch(year) {  
                case 2004 : System.out.println("Bulan 1 tahun 2004");  
                    break;  
                case 2005 : System.out.println("Bulan 1 tahun 2005");  
                    break;  
            }  
            break;  
            case 2: switch(year) {  
                case 2004 : System.out.println("Bulan 2 tahun 2004");  
                    break;  
                case 2005 : System.out.println("Bulan 2 tahun 2005");  
                    break;  
            }  
            break;  
            case 3: switch(year) {  
                case 2004 : System.out.println("Bulan 3 tahun 2004");  
                    break;  
                case 2005 : System.out.println("Bulan 3 tahun 2005");  
                    break;  
            }  
            break;  
            default:  
                System.out.println("Bulan dan Tahun yang dicari tidak ada  
pada pilihan");  
                break;  
        }  
        System.out.println("Sudah keluar dari switch");  
    }  
}
```



```
G:\JAVUA>javac SeleksiSwitch.java
G:\JAVUA>java SeleksiSwitch
Bulan 1 tahun 2005
Sudah keluar dari switch
```

B. Pengulangan (*Looping*)

Looping adalah perulangan suatu blok kode program berdasar kondisi yang ditentukan sampai tercapai kondisi untuk menghentikannya (*terminasi*). Setiap perulangan memiliki empat bagian, yaitu *inisialisasi*, *badan program*, *iterasi*, dan *terminasi*. Inisialisasi adalah program yang menyiapkan keadaan awal perulangan. Badan program adalah pernyataan yang ingin kita ulangi. Iterasi adalah program yang kita jalankan setelah badan program, tetapi sebelum bagian tersebut dijalankan lagi. Terminasi adalah pernyataan boolean yang diperiksa setiap kali selama perulangan untuk melihat apakah sudah waktunya menghentikan eksekusi. Pada Java dikenal tiga macam bentuk perulangan, yaitu : *while*, *do-while*, dan *for*.

1. Pengulangan Dengan *for*

Pengulangan dapat dilakukan dengan menggunakan pernyataan *for*. Loop dengan *for* melakukan inisialisasi sebelum iterasi/loop pertama, kemudian melakukan tes kondisional dan pada akhir dari tiap iterasi melakukan penambahan atau pengurangan pada variabel (*step*) yang digunakan sebagai basis pengkondisian untuk perulangan.

Sintaks :

```
for (inisialisasi ; ekspresi boolean ; step)
{
    <pernyataan>
}
```

Latihan 18. For.java

```
class For {
    public static void main(String[] args) {
        int c;
        System.out.println("Sebelum for");
        for(c=0;c<10;c++)
            System.out.println("Nilai c: " + c);
        System.out.println("Setelah for");
    }
}
```

```
Command Prompt
G:\JAVUA>javac For.java
G:\JAVUA>java For
Sebelum for
Nilai c: 0
Nilai c: 1
Nilai c: 2
Nilai c: 3
Nilai c: 4
Nilai c: 5
Nilai c: 6
Nilai c: 7
Nilai c: 8
Nilai c: 9
Setelah for
```

2. **Pengulangan Dengan *while***

Selain for, kita juga dapat menggunakan *while* untuk melakukan perulangan. Perulangan akan terus berjalan selama ekspresi boolean bernilai true atau benar.

Sintaks :

```
while (ekspresi boolean)
    {pernyataan}
```

Latihan 19. While.java

```
class While {
    public static void main(String[] args) {
        int a = 10 ;
        System.out.println("Sebelum while");
        while(a>=10) {
            System.out.println("Nilai a : "+a);
            a--;
        }
        System.out.println("Setelah while");
    }
}
```

```
Command Prompt
G:\JAVUA>javac While.java
G:\JAVUA>java While
Sebelum while
Nilai a : 10
Setelah while
```

3. Pengulangan Dengan *do-while*

Do-while memiliki urutan yang terbalik dibandingkan dengan *while* dan memiliki perbedaan utama, yaitu : bahwa pada *do-while* pernyataan dijalankan terlebih dahulu minimal satu kali, sedangkan pada *while* memungkinkan pernyataan sama sekali tidak dijalankan apabila kondisi boolean pada saat pertama kali sudah salah.

Sintaks :

```
do
{
    <pernyataan>
}
while (ekspresi boolean);
```

Latihan 20. DoWhile.java

```
class DoWhile {
    public static void main(String[] args) {
        int b = 9 ;
        System.out.println("Sebelum do");
        do {
            System.out.println("Nilai b : "+b);
            b--;
        } while (b>=10);
        System.out.println("Setelah do");
    }
}
```



```
Command Prompt
G:\JAVUA>javac DoWhile.java
G:\JAVUA>java DoWhile
Sebelum do
Nilai b : 9
Setelah do
```

C. Penggunaan Break

Pada pengulangan, bisa menggunakan *break* untuk menghentikan pengulangan pada suatu kondisi tertentu. Fungsi *break* tersebut adalah untuk menghentikan eksekusi sebuah blok program, misalnya untuk memaksa keluar dari sebuah looping.

D. Penggunaan Continue

Pada pengulangan bisa juga menggunakan continue, dimana continue akan melanjutkan proses pengulangan berikutnya tanpa menjalankan statemen sesudahnya.

E. Penggunaan Label

Penggunaan break maupun continue dapat dikombinasikan dengan label, yang fungsinya sama dengan goto pada pemrograman BASIC.

F. Penggunaan Return

Return biasanya digunakan di dalam method. Dengan menggunakan perintah return, alur eksekusi dikembalikan ke bagian dari program yang memanggil method tersebut.

LATIHAN

- 1. **Buatlah program yang memakai Break, Continue, Label, Return !**
- 2. **Buatlah program yang menghasilkan output seperti di bawah ini !**

```

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

- 2. **Buatlah program untuk mendapatkan bilangan prima dari suatu range bilangan !**
- 3. **Buatlah program untuk pemberian nilai hasil ujian, dengan syarat sebagai berikut.**
A : 85 – 100
B : 70 – 84
C : 55 – 79
D : 30 – 54
E : 0 – 29