



Modul Praktikum Pemrograman Berorientasi Objek

Jurusan Ilmu Komputer
Fakultas Matematika dan Ilmu Pengetahuan Alam
Universitas Lampung

2022
Semester Ganjil

Modul 3. Struktur Kontrol Java

Struktur kontrol merupakan sebuah blok program yang menganalisis variabel untuk menentukan alur program. Pada modul kali ini anda akan mempelajari dua jenis Struktur Kontrol yang sangat sering digunakan yaitu struktur kontrol pernyataan keputusan dan perulangan.

Capaian Pembelajaran

1. Mahasiswa mampu memahami dan mengimplementasikan struktur kontrol pernyataan keputusan
2. Mahasiswa mampu memahami dan mengimplementasikan struktur kontrol perulangan
3. Mahasiswa mampu mengimplementasikan pernyataan *branching* yang mana memperbolehkan *redirection* dari alur sebuah program

Materi

Percabangan

Percabangan adalah pernyataan yang akan menjalankan suatu fungsi ataupun program yang telah ditentukan pada kondisi tertentu. Percabangan dapat membantu anda untuk membuat beberapa alur untuk kondisi yang berbeda-beda.

Terdapat dua jenis percabangan yang biasa digunakan, yaitu :

1. If .. Else

Pernyataan ini dapat menjalankan suatu fungsi di dalamnya dengan kondisi yang dapat kita sesuaikan. Misalnya sama dengan, kurang dari, lebih dari, dsb.

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

Pernyataan if memiliki tiga bentuk:

- a. If
- b. If .. Else
- c. If .. Else If

2. Switch

Pernyataan ini dapat menjalankan suatu fungsi di dalamnya namun hanya jika kondisinya sama dengan nilai tertentu.

```
switch(value){  
    case 1:  
        statement1;  
  
    default:  
        statementDefault;  
}
```

Case menyatakan kondisi jika value sama dengan case tersebut. Sedangkan default menyatakan statement yang akan dieksekusi jika tidak ada case yang terpenuhi.

Perulangan

Perulangan adalah pernyataan yang akan mengeksekusi suatu blok program berulang-ulang hingga kondisi tertentu. Dalam pemrograman java terdapat tiga jenis perulangan yang paling sering digunakan.

1. While

Perulangan While merupakan perulangan yang hanya memerlukan kondisi untuk mengeksekusi program pada penulisan syntaxnya.

```
while(condition){  
    statement;  
}
```

Dalam perulangan ini program akan memeriksa terlebih dahulu apakah kondisinya terpenuhi atau tidak, jika terpenuhi maka akan mengeksekusi statement yang ada di dalamnya.

2. For

Perulangan For memiliki 3 parameter dalam kondisinya. Yaitu deklarasi variabel iterasi, kondisi untuk mengeksekusi statement, dan perubahan nilai variabel iterasi.

```
for(variable = nilai_awal; kondisi; variable++/--){
    statement;
}
```

Sama seperti while, perulangan for memeriksa terlebih dahulu apakah kondisinya terpenuhi atau tidak, kemudian jika terpenuhi maka akan mengeksekusi statement.

3. Do While

Perulangan Do While merupakan perulangan yang cukup berbeda dari perulangan lainnya. Perulangan ini akan mengeksekusi statement setidaknya 1x terlebih dahulu baru kemudian kondisinya akan diperiksa.

```
do{
    statement;
}while(condition);
```

Pernyataan Branching

Branching merupakan pernyataan kontrol yang dapat membantu anda untuk mengatur alur program. Branching biasa digunakan bersamaan dengan percabangan, looping, dan fungsi. Terdapat tiga jenis branching yang ada di Java, yaitu :

1. Break

Break merupakan pernyataan yang bisa digunakan untuk memotong alur program yang sedang berjalan. Misalnya pada sebuah looping, atau switch. Hal ini dilakukan agar program mengeksekusi blok yang tidak perlu.

2. Continue

Continue merupakan pernyataan yang bisa digunakan untuk melanjutkan alur program ke langkah berikutnya dan melewati blok program yang ada di bawahnya.

3. Return

Return bisa digunakan untuk menghentikan eksekusi suatu fungsi dengan atau tanpa mengembalikan suatu nilai.

Langkah Praktikum

Membuat Project

1. Buka NetBeans
2. Buatlah project baru dengan nama **nama_Praktikum3**

Membuat Percabangan dengan If Else

1. Di dalam fungsi main buatlah sebuah variabel **nilai** dengan tipe data **integer** yang akan menerima input user melalui console.
2. Buatlah sebuah percabangan **if** yang menyatakan jika **nilai \geq 76** maka cetak "Anda Lulus!"!

```
public static void main(String[] args) {  
    Scanner input = new Scanner(System.in);  
  
    System.out.print("Masukkan nilai : ");  
    int nilai = input.nextInt();  
  
    if(nilai >= 76){  
        System.out.println("Anda Lulus!");  
    }  
}
```

3. Buat ketentuan lainnya menggunakan **else** dan cetak "Nilai anda tidak memenuhi!"

```
public static void main(String[] args) {  
    Scanner input = new Scanner(System.in);  
  
    System.out.print("Masukkan nilai : ");  
    int nilai = input.nextInt();  
  
    if(nilai >= 76){  
        System.out.println("Anda Lulus!");  
    }else{  
        System.out.println("Nilai anda tidak memenuhi!");  
    }  
}
```

4. Selanjutnya buatlah **if** kedua yang berada di antara if dan else sebelumnya yang menyatakan jika **nilai \geq 71** maka cetak "Nilai anda B+"

```

public static void main(String[] args) {
    Scanner input = new Scanner(System.in);

    System.out.print("Masukkan nilai : ");
    int nilai = input.nextInt();

    if(nilai >= 76){
        System.out.println("Anda Lulus!");
    }if(nilai >= 71){
        System.out.println("Nilai anda B+");
    }else{
        System.out.println("Nilai anda tidak memenuhi!");
    }
}

```

Dari program tersebut maka akan didapatkan hasil sebagai berikut :

Run :
 Masukkan nilai : 80
 Anda Lulus!
 Nilai anda B+

5. Langkah sebelumnya akan menghasilkan output yang salah karena kondisi pertama dan kedua terpenuhi. Oleh karena itu diperlukan sebuah kondisi dimana kondisi kedua hanya akan dijalankan jika kondisi pertama tidak terpenuhi.
6. Tambahkan **else** di depan **if** kedua.

```

public static void main(String[] args) {
    Scanner input = new Scanner(System.in);

    System.out.print("Masukkan nilai : ");
    int nilai = input.nextInt();

    if(nilai >= 76){
        System.out.println("Anda Lulus!");
    }else if(nilai >= 71){
        System.out.println("Nilai anda B+");
    }else{
        System.out.println("Nilai anda tidak memenuhi!");
    }
}

```

Maka hasil dari program tersebut akan menjadi :

Run :

Masukkan nilai : 80
Anda Lulus!

Membuat Percabangan dengan switch

1. Lanjutkan program sebelumnya dengan menambahkan sebuah variabel **huruf_mutu** dengan tipe data **char**.
2. Ubah statement pada pernyataan **if else**. Dengan ketentuan :

Nilai	Huruf Mutu
≥ 76	A
$71 \leq \text{Nilai} < 76$	B
$66 \leq \text{Nilai} < 71$	C
Nilai < 66	D

```
public static void main(String[] args) {
    Scanner input = new Scanner(System.in);

    System.out.print("Masukkan nilai : ");
    int nilai = input.nextInt();
    char huruf_mutu;

    if(nilai >= 76){
        huruf_mutu = 'A';
    }else if(nilai >= 71){
        huruf_mutu = 'B';
    }else if(nilai >= 66){
        huruf_mutu = 'C';
    }else{
        huruf_mutu = 'D';
    }
}
```

3. Tambahkan **switch** setelah pernyataan **if else**. Dengan ketentuan :

Huruf Mutu	Output
A	Sangat Baik
B	Baik
C	Cukup

D	Kurang
default	Huruf Mutu Tidak Sesuai

```

public static void main(String[] args) {
    Scanner input = new Scanner(System.in);

    System.out.print("Masukkan nilai : ");
    int nilai = input.nextInt();
    char huruf_mutu;

    if(nilai >= 76){
        huruf_mutu = 'A';
    }else if(nilai >= 71){
        huruf_mutu = 'B';
    }else if(nilai >= 66){
        huruf_mutu = 'C';
    }else{
        huruf_mutu = 'D';
    }

    switch(huruf_mutu){
        case 'A':
            System.out.println("Sangat Baik");

        case 'B':
            System.out.println("Baik");

        case 'C':
            System.out.println("Cukup");

        case 'D':
            System.out.println("Kurang");

        default:
            System.out.println("Huruf Mutu tidak sesuai");
    }
}

```

Hasil dari program tersebut jika diberi input 78 adalah :

Run :
 Masukkan nilai : 78
 Sangat Baik
 Baik
 Cukup
 Kurang
 Huruf Mutu tidak sesuai

4. Untuk mengatasi error pada code sebelumnya anda bisa menambahkan **break** pada setiap case.


```
switch(huruf_mutu){  
    case 'A':  
        System.out.println("Sangat Baik");  
        break;  
    case 'B':  
        System.out.println("Baik");  
        break;  
    case 'C':  
        System.out.println("Cukup");  
        break;  
    case 'D':  
        System.out.println("Kurang");  
        break;  
    default:  
        System.out.println("Huruf Mutu tidak sesuai");  
}
```

Hasil dari program tersebut jika diberi input 78 adalah :

Run :
Masukkan nilai : 78
Sangat Baik

Membuat Perulangan

1. Lanjutkan program sebelumnya, buat program tersebut dapat menerima input beberapa kali menggunakan perulangan.
2. Anda dapat menggunakan perulangan **while**, **for**, maupun **do..while**.

Menggunakan Branching

1. Di dalam program perulangan yang sudah anda buat sebelumnya tambahkan sebuah pernyataan **if** yang menyatakan apabila input yang diberikan **lebih dari 100** maka tampilkan **"Input tidak sesuai"** lalu hentikan perulangan menggunakan **break**.

```

...
System.out.print("Masukkan nilai : ");
nilai = input.nextInt();

if(nilai > 100){
    System.out.println("Input tidak sesuai");
    break;
}

if(nilai >= 76){
    huruf_mutu = 'A';
}else if(nilai >= 71){
    huruf_mutu = 'B';
}else if(nilai >= 66){
    huruf_mutu = 'C';
}else{
    huruf_mutu = 'D';
}
...

```

2. Kemudian tambahkan sebuah pernyataan **if** lagi yang menyatakan apabila input yang diberikan merupakan **kelipatan 10**, maka lanjutkan program ke input berikutnya tanpa mencetak apa-apa menggunakan **continue**.

```

...
System.out.print("Masukkan nilai : ");
nilai = input.nextInt();

if(nilai > 100){
    System.out.println("Input tidak sesuai");
    break;
}

if((nilai%10) == 0){
    continue;
}

if(nilai >= 76){
    huruf_mutu = 'A';
}else if(nilai >= 71){
    huruf_mutu = 'B';
}else if(nilai >= 66){
    huruf_mutu = 'C';
}else{
    huruf_mutu = 'D';
}
...

```

Tugas

1. Modifikasi program yang telah dibuat, sesuaikan nilai huruf mutu dengan standard yang ada di Universitas Lampung.
2. Kumpulkan hasil akhirnya dengan cara **Copy** isi dari file utama, lalu letakkan pada Online Text di Vclass.

