

Nama : Hikari AuFa Yafi

Nim : A11.2021.13217

Tugas Merangkum PBO

➤ Percabangan

- Menggunakan statement if – else atau switch.
- Syntax penulisan if - else :
if (kondisi)
 statement1
else
 statement2
- if - else dapat berada lagi di dalam if - else (nested if - else) :
if (kondisi)
 if (kondisi)
 statement1
 else
 statement2
else
 statement3

contoh

```
Void hitung ()  
{  
nAngka=(0.24*tgs) + (0.35*mid)+(0.4*uas);  
System.out.println ("Nilai Akhir : " + nAngka);  
If (nAngka >85)  
{  
nHuruf = 'A';  
}  
Else if ((nAngka>55)&&(nAngka<=70))  
{  
nHuruf ='C';  
}  
Else if ((nAngka>39)&&(nAngka<=55))  
{  
nHuruf ='D';  
}  
Else {nHuruf = 'E';}  
System.out.pritnln("Nilai Huruf ; " + nHuruf);  
}
```

Jika menggunakan = itu berarti menunjukan assignement atau pemberian value

Jika menggunakan == itu berarti compile atau pembandingan

➤ Switch

Syntax penulisan switch:

```
switch (ekspresi) {  
  case Constant1 :  
    statement1; break;  
  case Constant2 :  
    statement2; break;  
  ...  
  default :  
    defaultStatement  
}
```

- Syntax penulisan switch diatas, dapat di ekuivalen-kan dengan penulisan if-else yang nested, menjadi :

```
nilaiSwitch = ekspresi;  
if (nilaiSwitch == nilai_1)  
  statement_1  
else if ( nilaiSwitch == nilai_2)  
  statement_2  
.....  
Else if (nilaiSwitch == nilai_i)  
  Statement_i  
Else  
  Statement_(i+1)
```

➤ While

- Mengulang badan looping selama kondisi adalah benar
- Badan looping adalah statement-statement yang berada di antara tanda kurung kurawal
- Syntax :
While (kondisi){
 Statement1;
 Statement2;
}

Contoh

```
Public class MyWhile {  
  
  Public class void main (String args[] {  
  
    Int c=0;  
  
    While (c<10) {  
  
      System.out.println ( "While – " +c);  
  
      C++;  
  
    }  
  
  }  
  
}
```

Note :

1. Dalam java perulangan itu ada 3 (do while,while,for) kalau di python ada 2 (for dan while)
2. While di sini itu artinya mengulang tapi kita tidak tahu berapa kali perulangan tersebut terjadi
3. For itu kita tau kalau perulangan yang terjadi itu berapa.

➤ Perulangan dengan For

```
Syntax perulangan dengan for
for (statement1; kondisi; statement2) {
    statement3;
}
```

Hal ini sama dengan perintah dengan while :

```
statement1;

while (kondisi) {

    statement3;

    statement2;

}
```

Contoh perulangan FOR

```
For (i = 1; i <=20; i = i+1)
{

    System.out.println(i);

}
```

Penjelasan : i = 1 itu adalah nilai awalnya

i <=20 itu adalah kondisinya apa

i = i+1 itu adalah counternya

yang akan menambah nanti itu yang ada di kondisinya. Jadi perulangan (i) nya itu tidak boleh stuck, kenapa? Karena bisa terjadi looping forever sehingga akan membebani processor dan ram.

➤ Perulangan dengan Do-While

Do...while digunakan untuk mengulang statement sebanyak satu kali terlebih dahulu, kemudian akan mengecek statement didalam while apakah bernilai benar, jika bernilai benar maka akan diulang kembali. Jika statement di dalam while bernilai salah maka perulangan akan berakhir.

➤ Break

Menghentikan looping secara paksa; artinya sebelum kondisi looping terpenuhi, looping dapat dihentikan.

- Contoh

```
Public class Mywhilebreak {  
  
Public static void main (String args []) {  
  
Int c=0;  
  
While (c<10) {  
  
C++;  
  
If (c==7) {break;} else {  
System.out.println ("While - " + c;}  
}  
}  
}
```

Note :looping yang dikerjakan pada contoh di atas yaitu cukup 1-6 saja dikarenakan pada angka 7 ke atas itu ada break dimana kondisi looping sudah tidak bisa di eksekusi lagi.

➤ Continue

Menghentikan looping yang terjadi saat itu dan Kembali ke looping selanjutnya (dari awal).

Contoh

```
Public class Mywhilebreak {  
  
Public static void main (String args []) {  
  
Int c=0;  
  
While (c<10) {  
  
C++;  
  
If (c==7) {continue;} else {  
System.out.println ("While - " + c;}  
}  
}  
}
```

➤ **Nested class**

- Merupakan class di dalam class; artinya dalam satu source code terdiri dari banyak class.
- Contoh

```
Class hewan{
  Class herbivora{
    ...}
  Class karnivora{
    ...}
}
```

➤ **Static class**

- Hanya kelas bersarang yang bisa statis
(Only nested classes can be static).
- Kelas yang ada kelas bersarang dikenal dengan kelas luar.
- Ada 2 :
 - Static class
 - Non-Static Class

➤ **Static & Non-Static class**

- Static class bersarang tidak memerlukan referensi kelas luar, tapi kelas nested non-static class memerlukan referensi kelas luar.
- Kelas dalam (kelas bersarang non-statis) dapat mengakses anggota kelas Outer statis dan non-statis. Kelas statis tidak dapat mengakses anggota non-statis dari kelas luar. Hanya bisa mengakses anggota statis kelas luar
- Sebuah instance dari inner class tidak dapat dibuat tanpa sebuah instance dari outer class dan inner class dapat merujuk data dan metode yang didefinisikan di outer class dimana ia bersarang, jadi

tidak perlu melewati referensi objek ke konstruktor
dari kelas dalam

Contoh

Class Luar

```
{ void cetakLuar ()  
    { System.out.println ("Cetak Luar .....");  
    // cetakTengah(); → error  
}
```

Class Tengah

```
{      void cetakTengah()  
    { cetakLuar ();  
      System.out.println ("cetak tengah.....");  
      // cetakDalam (); → error  
}
```

Class Dalam

```
{      void cetakDalam ()  
    { cetakLuar ();  
      cetakTengah();  
      System.out,println ( " cetak dalam .....");  
    }  
}  
}  
}
```

// class luar tidak bisa akses dalam / inner

// class dalam / inner bisa akses kelas di luarnya

Main

```
    Public class TestLuar
    {
        Public static void main (String [] a)
        {
            Luar 1=new Luar ();
            Luar.Tengah t=1.new Tengah ();
            Luar.Tengah.Dalam d=t.new Dalam ();
            1.cetakLuar ();
            t.cetakTengah ();
            d.cetakDalam ();
        }
    }
```