

# **3. Percabangan**

# Apa itu percabangan?

Selama ini, program yang kita buat berjalan linear. Tapi, akan ada saat di mana kita ingin menjalankan program kita secara berbeda tergantung kondisi tertentu.

Contoh dalam dunia nyata:

- Jika di luar hujan, maka pakai jas hujan. Jika tidak, maka tidak pakai jas hujan.
- Jika lapar, maka makan

# Apa itu percabangan? - Penjabaran Contoh

Dari kalimat tersebut, kita bisa buat strukturnya

- Jika di luar hujan, maka pakai jas hujan. Jika tidak, maka tidak pakai jas hujan.

Jika <<**suatu kondisi benar**>>, maka <<**lakukan sesuatu**>>. Jika <<**kondisi tidak benar**>>, maka <<**lakukan hal lain**>>

- Jika lapar, maka makan

Jika <<**suatu kondisi benar**>>, maka <<**lakukan sesuatu**>>

# Apa itu percabangan? - Penjabaran contoh

“ Jika <<**suatu kondisi benar**>>, maka <<**lakukan sesuatu**>>. Jika <<**kondisi tidak benar**>>, maka <<**lakukan hal lain**>> ”

Struktur tersebut merupakan struktur dasar dari **percabangan**. Kita akan buat contoh pertama menjadi struktur yang menjadi mirip sebuah kode

```
Jika (cuaca luar sama dengan hujan)
    pakai jas hujan
Jika (cuaca luar tidak sama dengan hujan)
    tidak pakai jas hujan
```

# Apa itu percabangan?

Kita masih bisa ubah lagi apa yang baru kita buat.

cuaca luar tidak sama dengan hujan itu sama saja dengan selain cuaca hujan, atau selain itu, Sehingga kita bisa menuliskannya sebagai:

```
Jika (cuaca luar sama dengan hujan)
    pakai jas hujan
Selain itu
    tidak pakai jas hujan
```

# Apa itu percabangan? - Jika ..., selain itu ...

Sehingga kita sekarang memiliki struktur dasar untuk percabangan dalam pemrograman:

```
Jika (kondisi terpenuhi)
    lakukan sesuatu
Selain itu
    lakukan hal lain
```

# Apa itu percabangan? - Jika ...

Kadang kita hanya ingin melakukan suatu hal kalo suatu kondisi dipenuhi, dan tidak melakukan apa-apa saat kondisinya tidak terpenuhi, seperti contoh kedua

“ Jika **lapar**, maka **makan** ”

Karena tidak ada hal yang butuh dilakukan jika kondisi tidak terpenuhi, maka:

```
Jika (kondisi)  
    lakukan sesuatu
```

# Apa itu percabangan? - Jika ..., selain itu jika ..., selain itu jika ...

Misalkan kita ingin menentukan baju yang kita pakai tiap hari. Jika kita menggunakan `Jika..., selain itu ...`, maka kita hanya bisa menentukan dua tindakan saja.

Oleh karena itu, kita bisa tambahkan kondisi untuk dicek.

“ Jika hari senin, maka baju putih. Selain itu, jika hari selasa, maka baju biru. Selain itu, jika haru rabu, maka baju bebas..... ”



# Apa itu percabangan? - Jika .., selain itu beruntun

Kita bisa ubah pernyataan tadi menjadi

```
Jika (hari sama dengan senin)
    baju putih
Selain itu jika (hari sama dengan selasa)
    baju biru
Selain itu jika (hari sama dengan rabu)
    baju bebas
Selain itu .....
```

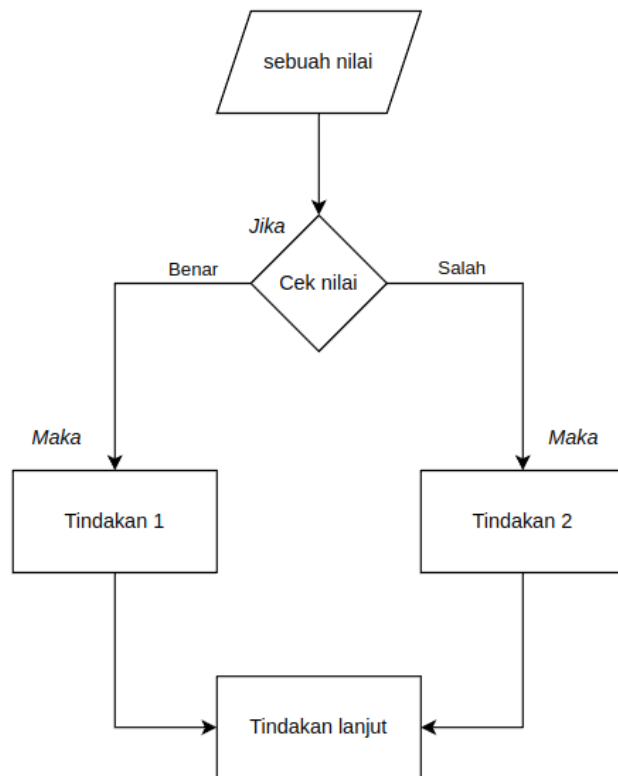
# Apa itu percabangan? - Jika .., selain itu beruntun

Strukturnya menjadi:

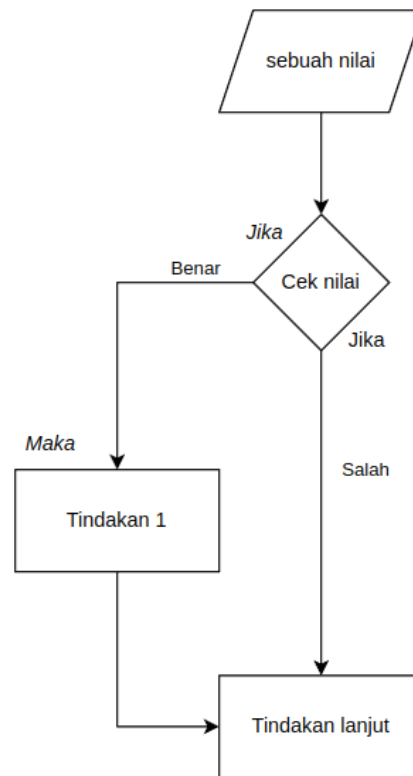
```
Jika (kondisi 1)
    tindakan 1
Selain itu jika (kondisi 2)
    tindakan 2
Selain itu jia (kondisi 3)
    tindakan 3
Selain itu...
```

# Visualisasi Percabangan

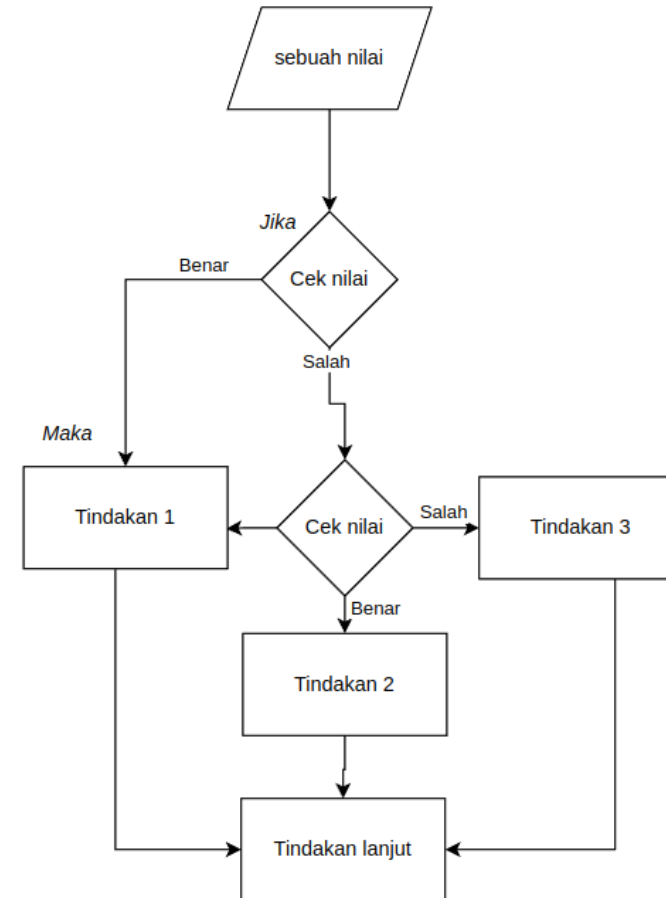
**Jika..., selain itu**



**Jika..., selain itu**



**Jika..., selain itu  
jika ...**



# Latihan

Buatlah teks berikut untuk menjadi pernyataan percabangan:

- Tika diberi tahu oleh orang tuanya untuk pulang ke rumah sebelum jam 6.
- Maria akan makan di luar bersama teman-temannya. Biasanya dia akan memesan makanan yang sama dengan teman dekatnya, Markus. Markus suka makan zupa soup, Maria suka makan iga bakar. Maria masih belum tahu Markus datang atau tidak.

< >

# Percabangan dalam Java

Syntax percabangan Java mirip dengan yang kita sudah buat

```
1 | if (kondisi) { // Jika (kondisi terpenuhi)
2 |     // Lakukan tindakan
3 | } else { // Selain itu
4 |     // Lakukan tindakan lain
5 | }
```

```
1 | if (kondisi) { //Jika (kondisi terpenuhi)
2 |     // Lakukan tindakan
3 | }
```

# Percabangan dalam Java

```
1 | if (kondisi_1) { // Jika (kondisi 1 terpenuhi)  
2 |     // Tindakan 1  
3 | } else if (kondisi3_2) { // Selain itu, jika (kondisi 2 terpenuhi)  
4 |     // Tindakan 2  
5 | } else if (kondisi_3) { // Selain itu, jika (kondisi 3 terpenuhi)  
6 |     // Tindakan 3  
7 | } else ....
```

Sekarang kita sudah tahu syntax dasar percabangan ***if-else*** untuk Java, sekarang kita lihat komponennya

# Kondisi

Saat menentukan keputusan, kita bertanya apakah sesuatu memenuhi kriteria atau tidak. **True** atau **false**

Contoh : Jika **lapar**, maka **makan**

Saat menentukan untuk makan atau tidak, kita bertanya "apakah kita lapar?". Kalau jawabannya **iya (true)**, maka kita akan makan.

Sehingga, dalam percabangan, kita akan bertanya apakah kondisi kita sesuai dengan kondisi untuk melakukan suatu tindakan

# Kondisi - Operasi boolean

Penentuan **benar (true)** atau **salah (false)** dilakukan menggunakan *operasi boolean*

**Operasi boolean** adalah operasi yang membandingkan dua nilai mendapatkan sebuah nilai kebenaran.



## Daftar operator boolean

Operator	Fungsi
==	Cek apakah dua nilai bernilai sama
>	Cek apakah nilai pertama lebih besar dari nilai kedua
>=	Cek apakah nilai pertama lebih besar atau sama dengan nilai kedua
<	Cek apakah nilai pertama lebih kecil daripada nilai kedua
<=	Cek apakah nilai pertama lebih kecil atau sama dengan nilai kedua
!=	Cek apakah nilai pertama tidak sama dengan nilai kedua

# Latihan

Tentukan hasilnya **true** atau **false**

- `1 == 2`
- `4 == 2+2`
- `7 > 8`
- `3 != 3.0`
- `'a' < 'b'`
- `'A' >= 'a'`
- `false == false`

# Operator boolean

Perhatikan: Tanda = beda dengan ==

= berfungsi untuk memberi nilai ke variable (assignment)

== berfungsi untuk membandingkan kesamaan 2 nilai

```
umur = 4 // umur akan bernilai 4
---
umur == 4 // membandingkan nilai dalam variabel umur dengan 4
```

## Daftar operator boolean

Operator	Nama	Fungsi
!	Negasi	Menegasi/membalik nilai boolean
&&	AND	Memberikan nilai benar saat kedua pembandingnya benar
	OR	Memberikan nilai benar saat salah satu pembandingnya benar

Ingat pelajaran Logika Matematika

# Latihan

Tentukan nilai kebenarannya

- $(10 > 50) \parallel (5/2 > 3)$
- `true && false`
- $!(80 < 5 \parallel 5 > 2)$
- `!(false) && !(78 == 5)`

# Kondisi dalam percabangan

Sekarang kita coba masukkan contoh kondisi dalam percabangan

```
1 | int umur = 17;  
2 |  
3 | if (umur > 17) {  
4 |     System.out.println("Sudah dapat memiliki KTP/SIM");  
5 | } else {  
6 |     System.out.println("Belum dapat memiliki KTP/SIM");  
7 | }
```

Apa nilai kebenaran dari kondisinya?

Apa saja tindakannya?

Bagaimana kalau nilai dari umur diganti?

# Latihan

Tentukan apa yang akan dicetak

```
1 | int nilai = 90;
2 |
3 | if (nilai >= 80) {
4 |     System.out.println("A");
5 | } else if (nilai >= 70) {
6 |     System.out.println("B");
7 | } else if (nilai >= 60) {
8 |     System.out.println("C");
9 | } else {
10|     System.out.println("Remidi");
11| }
```

```
1 | int bilangan_1 = 52;
2 | int bilangan_2 = 23;
3 |
4 | if (bilangan_1 % 2 == 0) {
5 |     System.out.println("Genap");
6 | } else {
7 |     System.out.println("Ganjil");
8 | }
9 |
10| if (bilangan_1 > 50) {
11|     System.out.println("Bilangan besar");
12| }
13|
14| if (bilangan_1 > bilangan_2 && bilangan_2 < 1000) {
15|     System.out.println("Bilangan tidak terlalu besar");
16| }
```



# Tindakan

Kita sudah tahu bagaimana cara membuat kondisi dan menentukan kebenarannya. Sekarang kita akan melihat apa yang dilakukan saat kondisi tersebut terpenuhi.

Sekelompok kode yang diawali oleh `{` dan diakhiri `}` disebut sebagai satu blok kode.

```
{ // awal blok kode  
    ... // baris-baris kode  
} // akhir blok kode
```

Baris kode yang ada dalam satu blok akan dianggap dijalankan sepaket.

Dalam pernyataan **if-else**, bisa terdapat beberapa blok kode.

```
if (kondisi) { // awal blok kode 1
    ....
} // akhir blok kode 1
else { //awal blok kode 2
    ...
} // akhir blok kode 2
```

# Code Block

Dalam satu codeblock, bisa terdapat lebih dari satu baris kode.

```
{  
    System.out.println("Baris pertama");  
    System.out.println("Baris kedua");  
}
```

Kedua baris kode tersebut berada dalam satu blok yang sama, sehingga akan dijalankan sepaket.

# Ubah kode kita!

Pekan lalu kita sudah buat data diri kita sebagai variabel. Sekarang, kita akan buat menu untuk menampilkan data diri:

Yang perlu dilakukan:

- Tampilkan menu data diri yang ingin ditampilkan
- Terima input dari user
- Tampilkan data diri berdasarkan nilai yang diinput

```
String nama = "Hersa";
int umur = 26;
String asal = "Temanggung";

Scanner input = new Scanner(System.in);

System.out.println("Data diri:")
System.out.println("1. Nama");
System.out.println("2. Umur");
System.out.println("3. Asal");
System.out.print("Pilih data diri yang ingin ditampilkan: ");

int pilihan = input.nextInt();

if (pilihan == 1) {
    System.out.println("Nama saya " + nama + ".");
} else if (pilihan == 2) {
    System.out.println("Umur saya " + umur + ".");
} else if (pilihan == 3) {
    System.out.println("Asal saya " + asal + ".");
} else {
    System.out.println("Pilihan tidak ada");
}
```