

## Praktikum III : Percabangan I

### Tujuan Praktikum :

1. Praktikan dapat menulis statement menggunakan percabangan I.
2. Praktikan dapat memahami bentuk dasar percabangan I di Java.
3. Praktikan dapat mengimplementasikan percabangan menggunakan if, else-if, if-else if, serta nested if.

### Alur Program

- ❖ Percabangan hanyalah sebuah istilah yang digunakan untuk menyebut alur program yang bercabang. Percabangan juga dikenal dengan “Control Flow”, “Struktur Kondisi”, “Struktur IF”, “Decision”, dsb.



- ❖ Setelah kita menggunakan percabangan, alurnya akan bertambah menjadi seperti ini.



- ❖ Lalu bagaimana cara menulis kode percabangan dalam Java? Caranya: menggunakan kata kunci if, else, switch, dan case.

## Percabangan If

- ❖ Contoh format stuktur IF seperti ini:

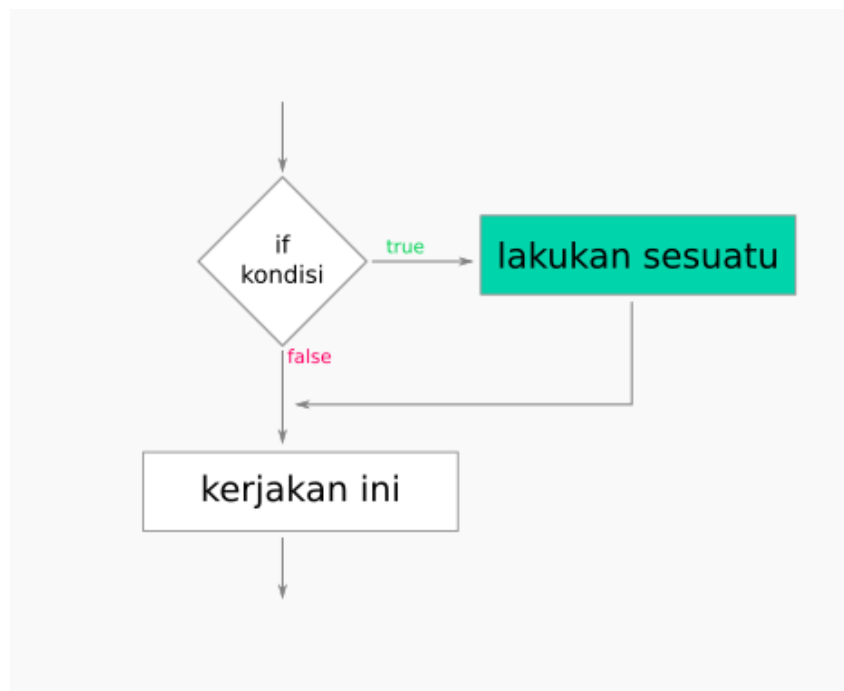
```
if( suatu_kondisi ) {  
    // lakukan sesuatu kalau kondisi benar  
    // Lakukan ini juga  
}
```

- ❖ Suatu\_kondisi hanya bernilai true/false saja. Kita bisa menggunakan operator relasional dan logika pada percabangan.

- ❖ Bentuk deklarasi umum dari percabangan if adalah sebagai berikut:

```
if(suatu_kondisi) {  
    //lakukan sesuatu kalau kondisi benar  
    //lakukan ini juga  
    ...  
}
```

- ❖ Percabangan ini hanya memiliki satu pilihan. Artinya, pilihan di dalam IF hanya akan dikerjakan kalau kondisinya benar.



- ❖ Tapi kalau salah... tidak akan melakukan apa-apa. Alias lanjut eksekusi ke perintah berikutnya.

- ❖ Contoh kasus :

Misalkan ada sebuah toko buku. Mereka memberikan hadiah berupa perlengkapan sekolah kepada pembeli yang belanja di atas Rp100.000

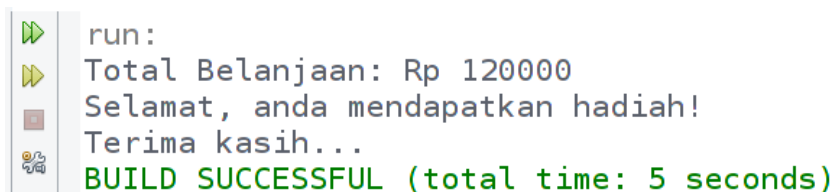
```
import java.util.Scanner;

public class Hadiah {

    public static void main(String[] args) {

        // membuat variabel belanja dan scanner
        int belanja = 0;
        Scanner scan = new Scanner(System.in);
        // mengambil input
        System.out.print("Total Belanjaan: Rp ");
        belanja = scan.nextInt();
        // cek apakah dia belanja di atas 100000
        if ( belanja > 100000 ) {
            System.out.println("Selamat, anda mendapatkan hadiah!");
        }
        System.out.println("Terima kasih...");
    }
}
```

Jalankan programnya dan perhatikanlah hasilnya.



```
run:
Total Belanjaan: Rp 120000
Selamat, anda mendapatkan hadiah!
Terima kasih...
BUILD SUCCESSFUL (total time: 5 seconds)
```

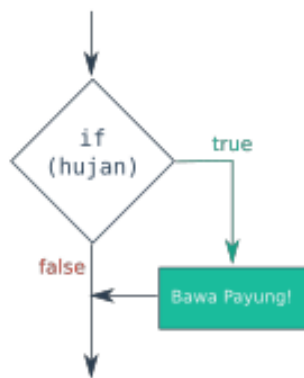
Cobalah untuk memberikan nilai di bawah 100000 dan perhatikan apa akan yang terjadi.

## Percabangan If - Else

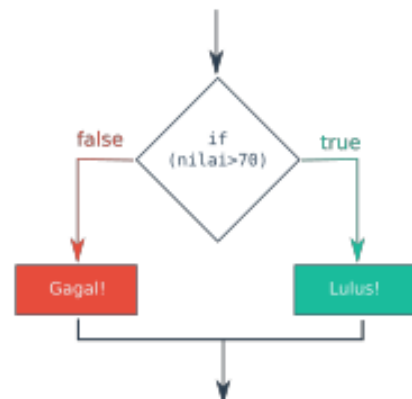
- ❖ Bentuk deklarasi umum dari percabangan if-else adalah sebagai berikut:

```
if(suatu_kondisi) {  
    //lakukan sesuatu kalau kondisi benar  
    //lakukan ini juga  
    ...  
}  
else{  
    //kerjakan ini kalau kondisi pada if salah  
    ...  
}
```

- ❖ Percabangan if-else memiliki pilihan alternatif kalau kondisinya salah.
- ❖ **IF:** “Jika kondisi benar maka kerjakan ini, kalau tidak silahkan lanjut” **IF/ELSE:** “Jika kondisi benar maka kerjakan ini, kalau salah maka kerjakan yang itu, setelah itu lanjut”



```
if (hujan) {  
    System.out.println("Bawa Payung!");  
}
```



```
if (nilai > 70) {  
    System.out.println("Lulus!");  
} else {  
    System.out.println("Gagal!");  
}
```

- ❖ Contoh kasus :

Misalkan, kalau nilai siswa lebih besar dari 70, maka ia dinyatakan lulus. Kalau tidak, maka dia gagal. Programnya bisa kita buat seperti ini:

```

import java.util.Scanner;
public class CekKelulusan {
    public static void main(String[] args) {

        // membuat variabel dan Scanner
        int nilai;
        String nama;
        Scanner scan = new Scanner(System.in);

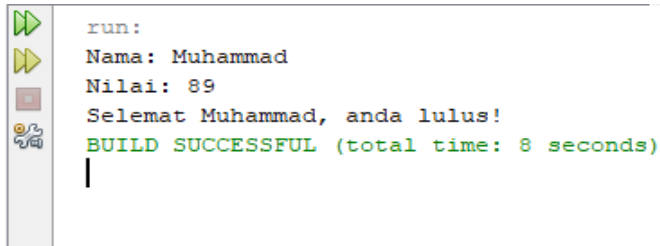
        // mengambil input
        System.out.print("Nama: ");
        nama = scan.nextLine();
        System.out.print("Nilai: ");
        nilai = scan.nextInt();

        // cek apakah dia lulus atau tidak
        if( nilai >= 70 ) {
            System.out.println("Selamat " + nama + ",
anda lulus!");
        }

        else {
            System.out.println("Maaf " + nama + ", anda
gagal");
        }
    }
}

```

Hasil outputnya:



```

run:
Nama: Muhammad
Nilai: 89
Selamat Muhammad, anda lulus!
BUILD SUCCESSFUL (total time: 8 seconds)
|

```

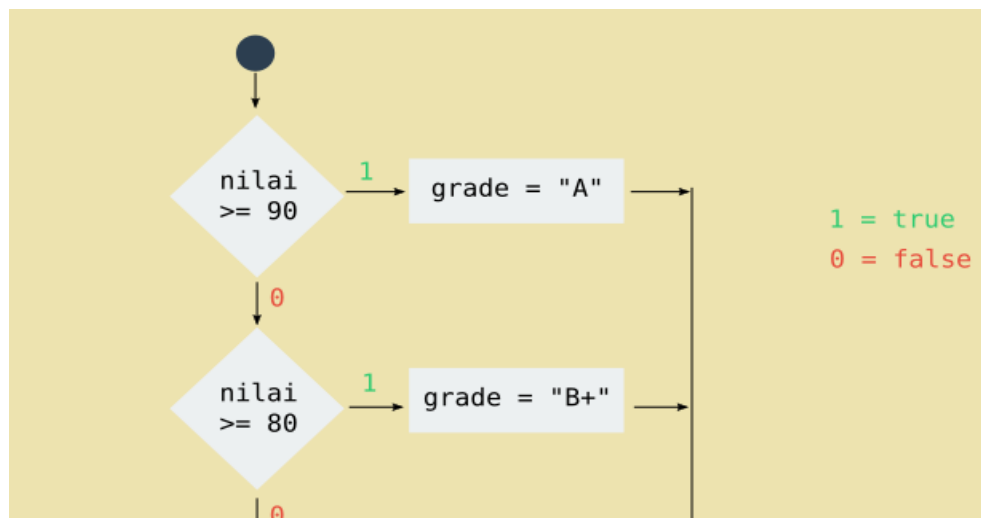
Cobalah untuk merubah nilai yang dimasukkan dan perhatikan apa yang akan terjadi.

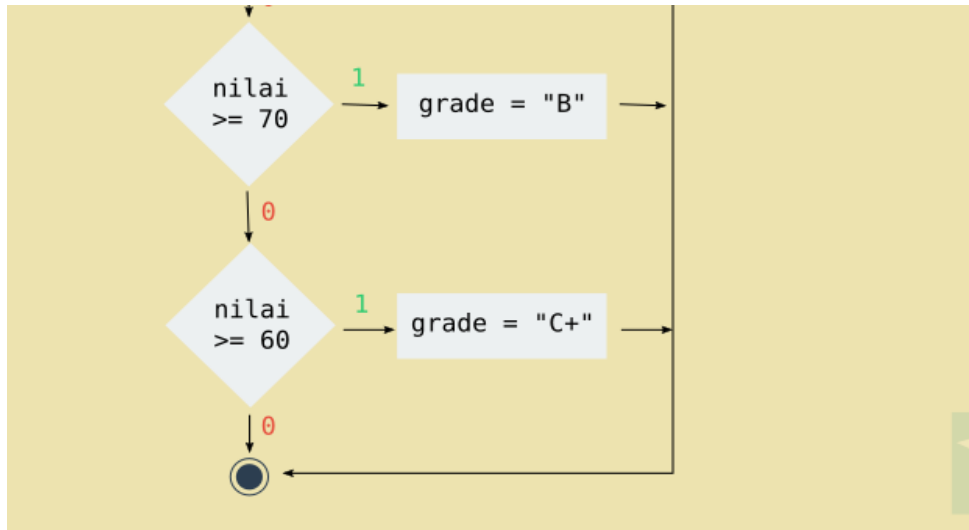
## Percabangan If - Else If

- ❖ Jika percabangan IF/ELSE hanya memiliki dua pilihan saja. Maka percabangan IF/ELSE/IF memiliki lebih dari dua pilihan.
- ❖ Bentuk deklarasi umum dari percabangan if-else if adalah sebagai berikut:

```
if (suatu kondisi) {  
    // maka kerjakan ini  
    // kerjakan perintah ini juga  
    // ...  
}  
else if (kondisi lain) {  
    // kerjakan ini  
    // kerjakan ini juga  
    // ...  
}  
else if (kondisi yang lain lagi) {  
    // kerjakan perintah ini  
    // kerjakan ini juga  
    // ...  
} else {  
    // kerjakan ini kalau  
    // semua kondisi di atas tidak ada yang benar  
    // ...  
}
```

Coba perhatikan contohnya:





Jika nilainya lebih besar dari 90, maka grade-nya "A". Sedangkan kalau lebih besar dari 80, maka "B+". Lebih besar dari 70, maka "B", dan seterusnya.

```
import java.util.Scanner;
public class HitungGrade {
    public static void main(String[] args) {
        // membuat variabel dan scanner
        int nilai;
        String grade;
        Scanner scan = new Scanner(System.in);
        // mengambil input
        System.out.print("Inputkan nilai: ");
        nilai = scan.nextInt();
        // hitung gradenya
        if ( nilai >= 90 ) {
            grade = "A";
        } else if ( nilai >= 80 ) {
            grade = "B+";
        } else if ( nilai >= 70 ) {
            grade = "B";
        } else if ( nilai >= 60 ) {
            grade = "C+";
        } else if ( nilai >= 50 ) {
```

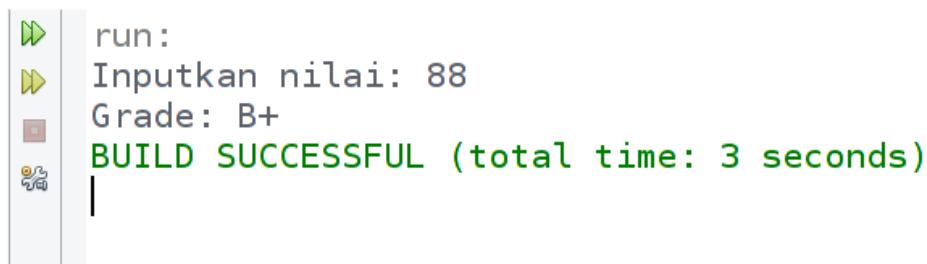
```

        grade = "C";
    } else if ( nilai >= 40 ){
        grade = "D";
    } else {
        grade = "E";
    }

    // cetak hasilnya
    System.out.println("Grade: " + grade);
}
}

```

Hasil Outputnya :



```

run:
Inputkan nilai: 88
Grade: B+
BUILD SUCCESSFUL (total time: 3 seconds)

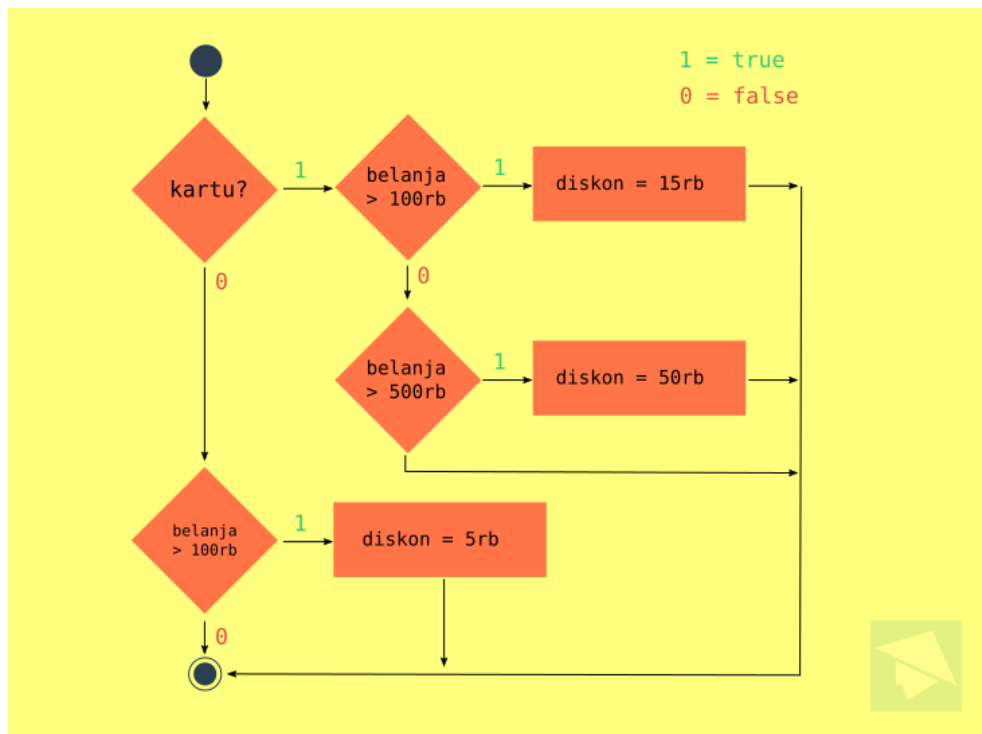
```

## Nested If

- ❖ Kita sudah tahu tiga bentuk dasar percabangan di Java. Selanjutnya, kita coba bahas percabangan yang ada di dalam percabangan (percabangan bersarang).
- ❖ Jadi, percabangan itu bisa dibuat di dalam percabangan. Kadang teknik ini disebut juga NESTED IF.
- ❖ Contoh kasus :
  - ❖ Misalnya ada model bisnis seperti ini di sebuah toko. Ketika orang membayar di kasir, biasanya ditanya ada kartu member untuk mendapatkan diskon dan sebagainya.
  - ❖ Jika pembeli memiliki kartu member maka setiap pembelian dengan total harga lebih dari “Rp500.000” mendapatkan diskon “Rp50.000”, jika pembelian dengan total harga lebih dari “Rp100.000” mendapatkan diskon “Rp15.000”.
  - ❖ Jika pembeli tidak memiliki kartu member maka setiap pembelian dengan total harga lebih dari “Rp100.000” mendapatkan diskon “Rp10.000”



❖ Coba perhatikan flow chart-nya :



Source Code :

```
import java.util.Scanner;
public class Kasir {
    public static void main(String[] args) {
        // deklarasi variabel dan Scanner
        int belanjaan, diskon, bayar;
        String kartu;
        Scanner scan = new Scanner(System.in);

        // mengambil input
        System.out.print("Apakah ada kartu member: ");
        kartu = scan.nextLine();
        System.out.print("Total belanjaan: ");
        belanjaan = scan.nextInt();

        // proses
        if (kartu.equalsIgnoreCase("ya")) {
            if (belanjaan > 500000) {
                diskon = 50000;
            } else if (belanjaan > 100000) {
                diskon = 15000;
            } else {
                diskon = 0;
            }
        } else {
            if (belanjaan > 100000) {
                diskon = 5000;
            }
        }
    }
}
```

```

        } else {
            diskon = 0;
        }
    }
    // total yang harus dibayar
    bayar = belanjaan - diskon;

    // output
    System.out.println("Total Bayar: Rp " + bayar);
}
}

```

Hasil outputnya:

```

run:
Apakah ada kartu member: ya
Total belanjaan: 334000
Total Bayar: Rp 319000
BUILD SUCCESSFUL (total time: 11 seconds)

```

- ❖ Cobalah untuk mengubah nilai yang dimasukkan dan perhatikan hasilnya.  
Mungkin di sana ada yang perlu diperhatikan:
  1. Fungsi `equalsIgnoreCase("ya")` digunakan untuk membandingkan String dengan tidak memperdulikan huruf besar dan kecilnya.
  2. Ada juga Fungsi `equals()`, fungsinya sama. Tapi `equals()` akan memperhatikan CASE hurufnya.
- ❖ Kenapa tidak menggunakan operator `==` atau `!=`?
- ❖ Di Java memang seperti itu. Kalau kita ingin membandingkan nilai String, ya... menggunakan fungsi yang dua tadi. Tapi, kalau membandingkan selain String, maka bisa pakai operator `==` atau `!=`.