



## JOBSHEET 5

### PEMILIHAN 2

#### 1. Tujuan

- Mahasiswa mampu memahami tentang operator logika
- Mahasiswa mampu menyelesaikan permasalahan dengan menggunakan sintaks pemilihan bersarang
- Mahasiswa mampu membuat sebuah program Java yang memanfaatkan sintaks pemilihan bersarang

#### 2. Praktikum

##### 2.1 Percobaan 1

1. Buka text editor. Buat file baru, beri nama **Nested1.java**
2. Tuliskan struktur dasar bahasa pemrograman Java yang terdiri dari fungsi **main()**
3. Tambahkan library Scanner
4. Buat deklarasi **Scanner** dengan nama **sc**
5. Buatlah variabel bertipe **int** dengan nama **nilai**
6. Tambahkan kode berikut ini untuk menerima input dari keyboard

```
System.out.print("Masukkan nilai ujian (0 - 100): ");
nilai = sc.nextInt();
```
7. Buatlah struktur pengecekan kondisi bersarang. Pengecekan pertama digunakan untuk memastikan bahwa nilai yang dimasukkan berada pada rentang 0 – 100. Jika nilai berada pada rentang 0 – 100, maka akan dilakukan pengecekan status kelulusan mahasiswa, yaitu jika nilai di antara 90 – 100 maka nilainya A, jika nilai di antara 80 – 89 maka nilainya B, jika nilai di antara 60 – 79 maka nilainya C, jika nilai di antara 50 – 59 maka nilainya D, dan jika nilai di antara 0 – 49 maka nilainya E. Sedangkan jika nilai berada di luar rentang 0 – 100, maka ditampilkan sebuah informasi yang menyatakan bahwa nilai yang dimasukkan tidak valid.



```

if (nilai >= 0 && nilai <= 100) {
    if (nilai >= 90 && nilai <= 100) {
        System.out.println("Nilai A, EXCELLENT!");
    } else if (nilai >= 80 && nilai <= 89) {
        System.out.println("Nilai B, pertahankan prestasi Anda!");
    } else if (nilai >= 60 && nilai <= 79) {
        System.out.println("Nilai C, tingkatkan prestasi Anda!");
    } else if (nilai >= 50 && nilai <= 59) {
        System.out.println("Nilai D, tingkatkan belajar Anda!");
    } else {
        System.out.println("Nilai E, Anda tidak lulus!");
    }
} else {
    System.out.println("Nilai yang Anda masukkan tidak valid");
}

```

8. Jalankan program tersebut. Amati apa yang terjadi!

### Pertanyaan!

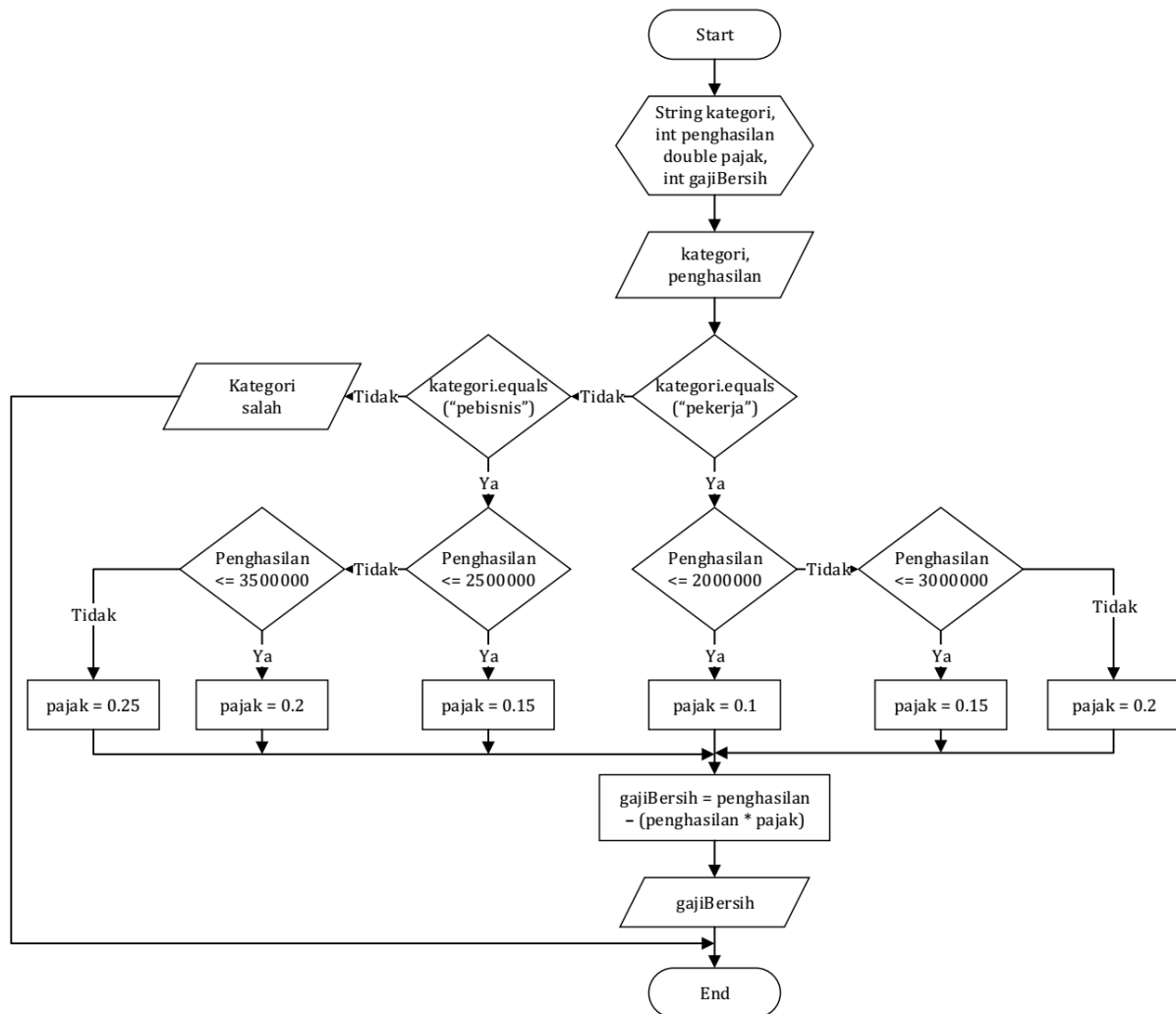
1. Jelaskan fungsi sintaks berikut!

```
if (nilai >= 0 && nilai <= 100)
```

2. Modifikasi kode program pada Percobaan 1 sehingga jika nilai yang dimasukkan kurang dari 0 akan ditampilkan output "Nilai yang Anda masukkan kurang dari 0" dan jika nilai yang dimasukkan lebih dari 100 akan ditampilkan output "Nilai yang Anda masukkan lebih dari 100"!
3. Ubah operator **&&** menjadi **||** pada sintaks `if (nilai >= 0 && nilai <= 100)`. Jalankan program dengan memasukkan nilai = 105 melalui keyboard. Amati apa yang terjadi! Mengapa hasilnya demikian?

### 3.2. Percobaan 2

1. Perhatikan flowchart berikut ini!



Flowchart tersebut digunakan untuk menghitung gaji bersih seseorang setelah dipotong pajak sesuai dengan kategorinya (pekerja dan pebisnis) dan besarnya penghasilan.

2. Buka text editor. Buat file baru, beri nama **Nested2.java**
3. Tuliskan struktur dasar bahasa pemrograman Java yang terdiri dari fungsi **main()**
4. Tambahkan library Scanner
5. Buat deklarasi **Scanner** dengan nama **sc**
6. Deklarasikan variabel **kategori**, **penghasilan**, **gajiBersih**, dan **pajak**

```
String kategori;
int penghasilan, gajiBersih;
double pajak = 0;
```

7. Tambahkan kode berikut ini untuk menerima input dari keyboard



```
System.out.print("Masukkan kategori: ");
kategori = sc.nextLine();
System.out.print("Masukkan besarnya penghasilan: ");
penghasilan = sc.nextInt();
```

8. Buatlah struktur pengecekan kondisi bersarang. Pengecekan pertama digunakan untuk mengecek kategori (pekerja atau pebisnis). Selanjutnya dilakukan pengecekan kedua untuk menentukan besarnya pajak berdasarkan penghasilan yang telah dimasukkan. Kemudian tambahkan kode program untuk menghitung gaji bersih yang diterima setelah dipotong pajak

```
if (kategori.equalsIgnoreCase("pekerja")) {
    if (penghasilan <= 2000000) {
        pajak = 0.1;
    } else if (penghasilan <= 3000000) {
        pajak = 0.15;
    } else {
        pajak = 0.2;
    }
    gajiBersih = (int) (penghasilan - (penghasilan * pajak));
    System.out.println("Gaji bersih yang Anda terima: " + gajiBersih);
} else if (kategori.equalsIgnoreCase("pebisnis")) {
    if (penghasilan <= 2500000) {
        pajak = 0.15;
    } else if (penghasilan <= 3500000) {
        pajak = 0.2;
    } else {
        pajak = 0.25;
    }
    gajiBersih = (int) (penghasilan - (penghasilan * pajak));
    System.out.println("Gaji bersih yang Anda terima: " + gajiBersih);
} else {
    System.out.println("Kategori yang Anda masukkan salah");
}
```

9. Jalankan program tersebut. Amati apa yang terjadi!

### Pertanyaan!

1. Jalankan program dengan memasukkan kategori = pekerja dan penghasilan = 2048485 melalui keyboard. Amati apa yang terjadi! Mengapa angka di belakang koma tidak ditampilkan?
2. Jelaskan fungsi dari `(int)` pada sintaks berikut!

```
gajiBersih = (int) (penghasilan - (penghasilan * pajak));
```

3. Jalankan program dengan memasukkan kategori = PEBISNIS dan penghasilan = 2000000. Amati apa yang terjadi! Apa kegunaan dari **equalsIgnoreCase**?
4. Ubah **equalsIgnoreCase** menjadi **equals**, kemudian jalankan program dengan memasukkan kategori = PEBISNIS dan penghasilan = 2000000. Amati apa yang terjadi! Mengapa hasilnya demikian? Apa kegunaan dari **equals**?

### 3. Tugas

1. Dengan menggunakan tiga nilai yang mewakili panjang tiga sisi sebuah segitiga, tentukan apakah segitiga tersebut **sama sisi** (ketiga sisinya bernilai sama), **sama kaki** (kedua sisinya bernilai sama), atau **sembarang** (tidak ada sisi yang bernilai sama)!
2. Sebuah restoran meminta Anda membuat sebuah program untuk menerima pesanan dari internet. Program yang Anda buat meminta user untuk memasukkan nama makanan dan harga. Setelah itu, user ditawarkan untuk menggunakan pengiriman ekspres. Jika pengguna menolak, maka jenis pengiriman yang digunakan adalah pengiriman reguler. Biaya pengiriman reguler untuk harga makanan kurang dari Rp 100.000 adalah Rp 20.000, sedangkan untuk harga makanan sama dengan atau lebih dari Rp 100.000 biaya pengirimannya adalah Rp 30.000. Untuk jenis pengiriman ekspres, tambahkan biaya tambahan sebesar Rp 25.000 dari standar biaya pengiriman reguler. Tampilkan struk yang berisi nama makanan yang dibeli + harga, biaya pengiriman, dan total yang harus dibayar!

Contoh hasil output program:

```
Masukkan nama makanan: Tuna Salad
Masukkan harga makanan: Rp 115000
Apakah Anda ingin pengiriman ekspres (0 = tidak, 1 = ya)? 0
STRUK PEMBELIAN
Tuna Salad          Rp 115000
Biaya pengiriman    Rp 30000
TOTAL               Rp 145000

Masukkan nama makanan: Beef Bulgogi
Masukkan harga makanan: Rp 78000
Apakah Anda ingin pengiriman ekspres (0 = tidak, 1 = ya)? 1
STRUK PEMBELIAN
Beef Bulgogi        Rp 78000
Biaya pengiriman    Rp 45000
TOTAL               Rp 123000
```