

JOBSHEET 6

Pemilihan 2

1. Tujuan

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

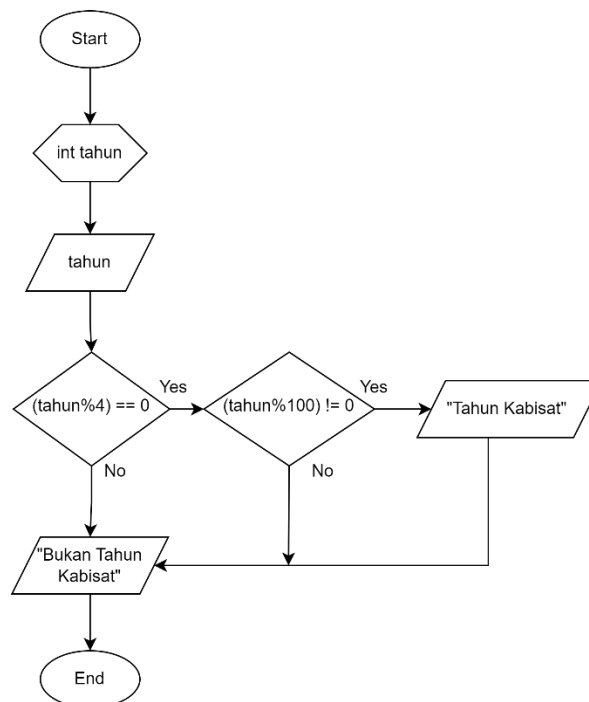
2. Praktikum

2.1 Percobaan 1

Waktu Percobaan 50 Menit

1. Menentukan tahun kabisat atau bukan, dengan ketentuan tahun kabisat adalah tahun kelipatan 4 dan bukan kelipatan 100.

Flowchart berikut adalah algoritma dari penentuan tahun kabisat



2. Selanjutnya kita akan membuat programnya. Buka text editor kemudian simpan dengan nama **Pemilihan2Percobaan1NoAbsen.java**
3. Buatlah struktur dasar java (membuat class dan program main).
4. Tambahkan import library Scanner.
5. Deklarasikan Scanner : beri nama Scanner dengan identitas Absen. Format input **Absen**

6. Tambahkan kode untuk menerima inputan dari keyboard untuk tahun.
7. Buatlah struktur kondisi seperti dibawah ini :

```
if ((tahun % 4) == 0) {  
    if ((tahun % 100) != 0)  
        System.out.println(x:"Tahun Kabisat");  
} else  
    System.out.println(x:"Bukan Tahun Kabisat");
```

8. Jalankan program tersebut, maka outputnya adalah sebagai berikut :

```
Masukkan Tahun : 2004  
Tahun Kabisat
```

9. Push dan commit hasil praktikum anda ke repository

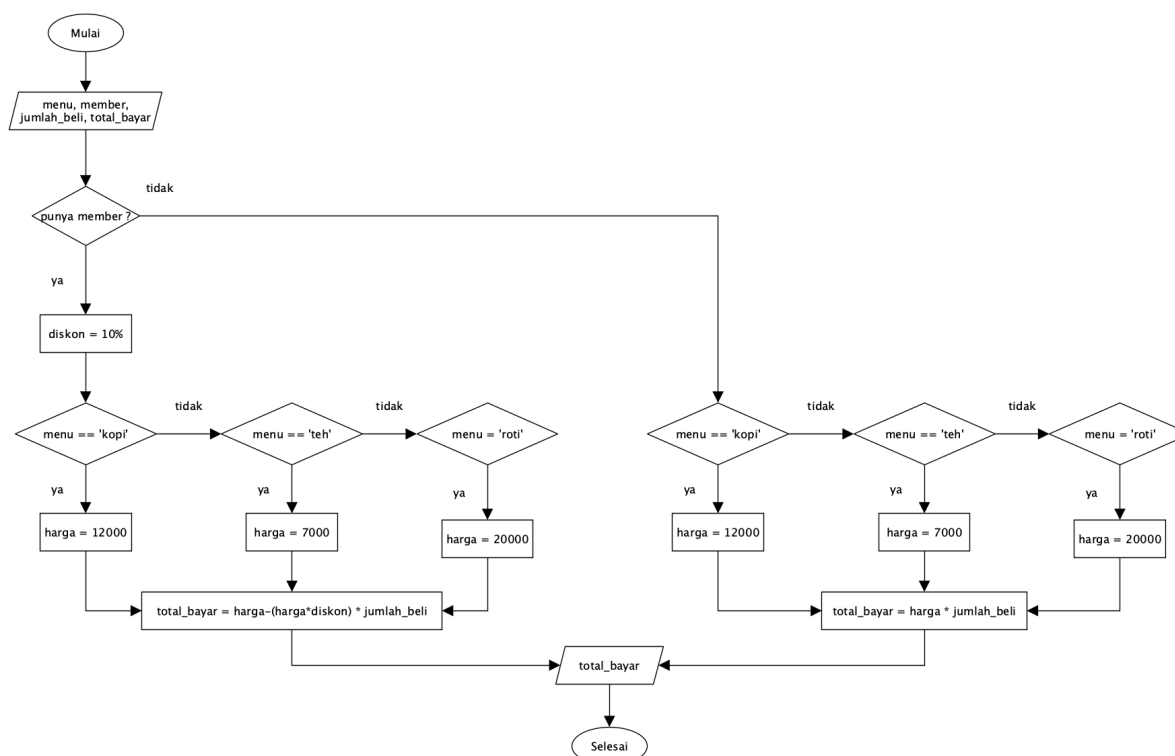
Pertanyaan

1. Bagaimana outputnya ketika diberikan input tahun 2100 ? Jelaskan! Bagaimana agar output sesuai dengan ketentuan (Tahun 2100 bukan tahun kabisat)
2. Modifikasi program sesuai jawaban no 1 !
3. Push dan commit hasil modifikasi anda ke repository !
4. Tahun 2000 adalah kelipatan 4 dan kelipatan 100, tetapi tahun 2000 merupakan tahun kabisat, ketentuan tambahan (pengecualian) adalah ketika tahun kelipatan 100 **dan juga** kelipatan 400 maka tahun tersebut merupakan tahun kabisat. Modifikasi program untuk menyesuaikan ketentuan tersebut ! (selesaikan tanpa menggunakan operator logika)
5. Push dan commit hasil modifikasi anda ke repository !

2.2 Percobaan 2

Waktu Percobaan 50 Menit

1. Menentukan total bayar dari suatu transaksi di kafe dari inputan menu yang dipilih dan memiliki member atau tidak.
2. Perhatikan flowchart berikut ini!



3. Selanjutnya kita akan membuat programnya. Buka text editor kemudian simpan dengan nama **Pemilihan2Percobaan2NoAbsen.java**
4. Buatlah struktur dasar java (membuat class dan program main).
5. Tambahkan import library Scanner.
6. Deklarasikan Scanner : beri nama Scanner dengan identitas Absen. Format input **Absen**
7. Tuliskan perintah untuk memasukkan inputan dari keyboard, untuk pilihan menu dan member.
8. Tambahkan kode program berikut untuk tampilan menu

```
System.out.println(x:"-----");
System.out.println(x:"==== MENU KAFE JTI =====");
System.out.println(x:"-----");
System.out.println(x:"1. Ricebowl");
System.out.println(x:"2. Ice Tea");
System.out.println(x:"3. Paket Bundling (Ricebowl + Ice Tea)");
System.out.println(x:"-----");
System.out.print(s:"masukkan angka dari menu yang dipilih = ");
pilihan_menu = sc.nextInt();
sc.nextLine();
System.out.print(s:"Apakah punya member (y/n) ? = ");
member = sc.nextLine();
System.out.println(x:"-----");
```

9. Buatlah struktur kondisi pertama → jika pembeli memiliki member, seperti dibawah ini :

```

if (member.equalsIgnoreCase(anotherString:"y")) { // Menggunakan equalsIgnoreCase untuk membandingkan string
    diskon = 0.10;
    System.out.println(x:"Besar diskon = 10%");
    if (pilihan_menu == 1) {
        harga = 14000;
        System.out.println("Harga ricebowl = " + harga);
    } else if (pilihan_menu == 2) {
        harga = 3000;
        System.out.println("Harga ice tea = " + harga);
    } else if (pilihan_menu == 3) {
        harga = 15000;
        System.out.println("Harga bundling = " + harga);
    } else {
        System.out.println(x:"Masukkan pilihan menu dengan benar");
        return; // Menghentikan eksekusi lebih lanjut jika pilihan salah
    }

    // Menghitung total bayar setelah diskon
    total_bayar = harga - (harga * diskon);
    System.out.println("Total bayar setelah diskon = " + total_bayar);
}

```

10. Buatlah struktur kondisi kedua → jika pembeli **tidak** memiliki member, seperti dibawah ini :

```

else if (member.equalsIgnoreCase(anotherString:"n")) { // Menggunakan equalsIgnoreCase untuk membandingkan string
    if (pilihan_menu == 1) {
        harga = 14000;
        System.out.println("Harga ricebowl = " + harga);
    } else if (pilihan_menu == 2) {
        harga = 3000;
        System.out.println("Harga ice tea = " + harga);
    } else if (pilihan_menu == 3) {
        harga = 15000;
        System.out.println("Harga bundling = " + harga);
    } else {
        System.out.println(x:"Masukkan pilihan menu dengan benar");
        return; // Menghentikan eksekusi lebih lanjut jika pilihan salah
    }

    // Menghitung total bayar
    System.out.println("Total bayar = " + harga);
} else {
    System.out.println(x:"Member tidak valid");
}
System.out.println(x:"-----");

```

11. Jalankan program tersebut, maka outputnya adalah sebagai berikut :

- Output jika memiliki member :

```

-----
===== MENU KAFE JTI =====
-----
1. Ricebowl
2. Ice Tea
3. Paket Bundling (Ricebowl + Ice Tea)
-----
masukkan angka dari menu yang dipilih = 3
Apakah punya member (y/n) ? = y
-----
Besar diskon = 10%
Harga bundling = 15000.0
Total bayar setelah diskon = 13500.0
-----

```

- Output jika tidak memiliki member :

```

-----
==== MENU KAFE JTI ====
-----
1. Ricebowl
2. Ice Tea
3. Paket Bundling (Ricebowl + Ice Tea)
-----
masukkan angka dari menu yang dipilih = 3
Apakah punya member (y/n) ? = n
-----
Harga bundling = 15000.0
Total bayar = 15000.0
-----
    
```

12. Push dan commit hasil praktikum anda ke repository

Pertanyaan

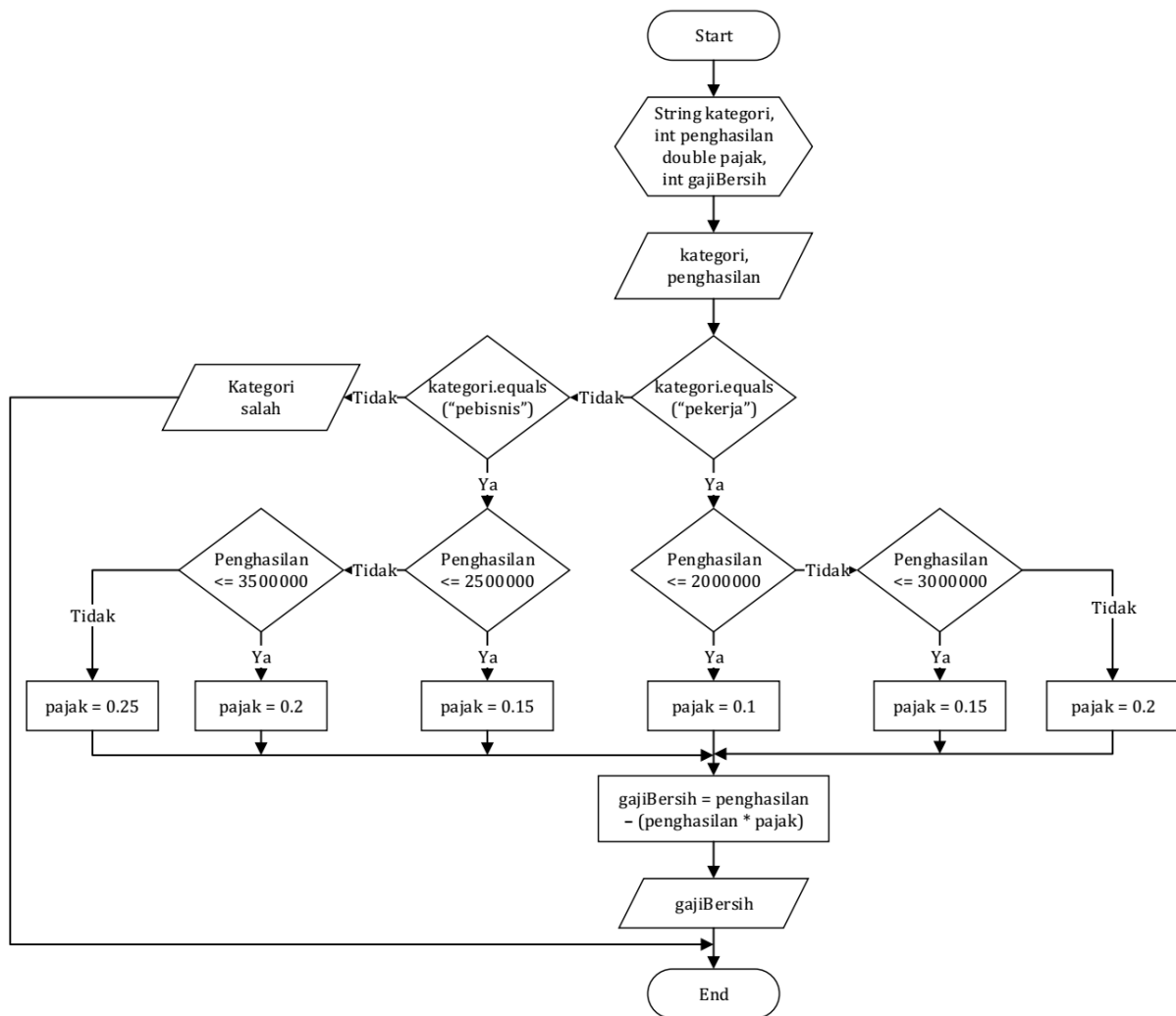
1. Apakah fungsi method `.equalsIgnoreCase()` pada seleksi kondisi member ?
2. Apa perbedaan fungsi method `.equals()` dan `.equalsIgnoreCase()` ?
3. Mengapa terdapat kode program `sc.nextLine();` pada baris setelah `pilihan_menu = sc.nextInt();` ? Jelaskan fungsinya !
4. Bagaimana outputnya ketika diberikan input jenis pembayaran melalui QRIS akan mendapatkan potongan harga Rp.1.000 bagi yang memiliki member maupun yang tidak memiliki member ?
5. Modifikasi program sesuai jawaban no 2 !
6. Push dan commit hasil modifikasi anda ke repository !

2.3 Percobaan 3

Waktu Percobaan 40 Menit

Studi Kasus

1. Perhatikan flowchart di bawah ini, flowchart tersebut digunakan untuk menghitung gaji bersih seseorang setelah dipotong pajak sesuai dengan kategorinya (pekerja dan pebisnis) dan besarnya penghasilan.



- Kita buat program berdasarkan algoritma pada flowchart tersebut. Buka text editor kemudian simpan dengan nama **Pemilihan2Percobaan3NoAbsen.java**
- Buatlah struktur dasar java (membuat class dan program main).
- Tambahkan import library Scanner.
- Deklarasikan Scanner : beri nama Scanner dengan identitas Absen. Format input **Absen**
- Deklarasikan variabel **kategori**, **penghasilan**, **gajiBersih**, dan **pajak**;

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

- Tambahkan kode berikut ini untuk menerima input dari keyboard

```
System.out.print("Masukkan Kategori : ");
kategori = noAbsen.nextLine();
System.out.print("Masukkan Besarnya Penghasilan : ");
penghasilan = noAbsen.nextInt();
```

- 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(anotherString:"pekerja")) {
    if (penghasilan <= 2000000)
        pajak = 0.1;
    else if (penghasilan <= 3000000)
        pajak = 0.15;
    else
        pajak = 0.2;
    gajiBersih = (int) (penghasilan - (pajak * penghasilan));
    System.out.print("Penghasil Bersih : " + gajiBersih);
} else if (kategori.equalsIgnoreCase(anotherString:"pebisnis")) {
    if (penghasilan <= 2500000)
        pajak = 0.15;
    else if (penghasilan <= 3500000)
        pajak = 0.2;
    else
        pajak = 0.25;
    gajiBersih = (int) (penghasilan - (pajak * penghasilan));
    System.out.print("Penghasil Bersih : " + gajiBersih);
} else
    System.out.println(x:"Masukan Kategori Salah ");
```

9. Jalankan program tersebut. Amati apa yang terjadi!

Pertanyaan

1. Jelaskan fungsi dari **(int)** pada sintaks `gajiBersih = (int) (penghasilan - (penghasilan * pajak));`
2. Jalankan program dengan memasukkan kategori = PEBISNIS dan penghasilan = 2000000. Amati apa yang terjadi! Apa kegunaan dari **equalsIgnoreCase**?
3. 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

Waktu Percobaan : 160 Menit

Buatlah kode program berdasarkan flowchart yang telah dibuat pada Tugas pertemuan 6 Matakuliah Dasar Pemrograman ! Push dan commit hasil kode program anda ke repository project Anda!

Catatan : tugas hanya boleh menerapkan materi dari pertemuan 1 hingga pertemuan 6.