

Laporan Hasil Praktikum Dasar Pemrograman

Latihan-6 : Praktikum Dasar Pemrograman



Nama : Ghoffar Abdul Ja'far

NIM : 41720035

Kelas : 1E

JURUSAN TEKNOLOGI INFORMASI

POLITEKNIK NEGERI MALANG

2023/2024

2.1 Percobaan 1

- Menulis Syntax

1. Deklarasi Scanner

```
Scanner input12 = new Scanner(System.in);
```

2. Tambahkan kode untuk menerima inputan dari keyboard untuk tahun.

```
System.out.print("Masukkan Tahun: ");  
tahun = input12.nextInt();
```

3. Buatlah struktur kondisi seperti dibawah ini :

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

- Hasil

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

Pertanyaan!

1. Bagaimana outputnya ketika diberikan input tahun 2100? Jelaskan! Bagaimana agar output sesuai dengan ketentuan (Tahun 2100 bukan tahun kabisat)

```
Masukkan Tahun: 2100
```

Tidak menghasilkan output, dengan menambahkan blok else untuk kondisi ke- setelah if yang pertama

2. Modifikasi program sesuai jawaban no 1 !

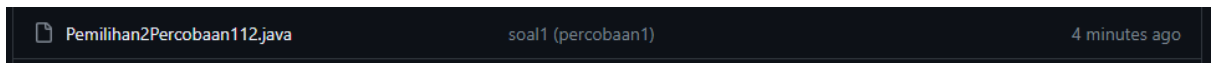
- Perubahan syntax

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

- Hasil

```
Masukkan Tahun: 2100  
Bukan Tahun Kabisat
```

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)

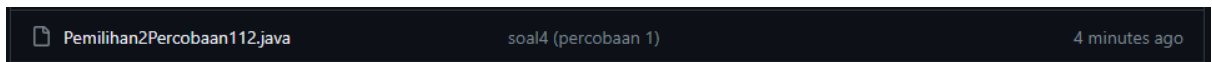
- Perubahan syntax

```
if ((tahun % 4) == 0) {  
    if ((tahun % 100) != 0) {  
        System.out.println("Tahun Kabisat");  
    } else if ((tahun % 400) != 0) {  
        System.out.println("Tahun Kabisat");  
    } else {  
        System.out.println("Bukan Tahun Kabisat");  
    }  
} else {  
    System.out.println("Bukan Tahun Kabisat");  
}
```

- Hasil

```
Masukkan Tahun: 2000  
Bukan Tahun Kabisat
```

5. Push dan commit hasil modifikasi anda ke repository !



2.2 Percobaan 2

- Menuliskan syntax

1. Deklarasi Scanner

```
Scanner input12 = new Scanner(System.in);
```

2. Tuliskan perintah untuk memasukkan inputan dari keyboard, untuk 3 sudut segitiga.

```
System.out.print("Masukkan sudut 1 : ");  
sudut1 = input12.nextInt();  
System.out.print("Masukkan sudut 2 : ");  
sudut2 = input12.nextInt();  
System.out.print("Masukkan sudut 3 : ");  
sudut3 = input12.nextInt();
```

3. Tambahkan kode program berikut untuk menghitung jumlah ketidak sudut segitiga

```
totalSudut = sudut1 + sudut2 + sudut3;
```

4. Buatlah struktur kondisi seperti dibawah ini :

```
if (totalSudut == 180) {  
    if ((sudut1 == 90) || (sudut2 == 90) || (sudut3 == 90)) {  
        System.out.println("Segitiga tersebut adalah segitiga  
siku-siku");  
    } else {  
        System.out.println("Segitiga tersebut adalah bukan  
segitiga siku-siku");  
    }  
} else {  
    System.out.println("Bukan Segitiga");  
}
```

- Hasil

```
Masukkan sudut 1 : 90  
Masukkan sudut 2 : 30  
Masukkan sudut 3 : 60  
Segitiga tersebut adalah segitiga siku-siku
```

Pertanyaan!

1. Jenis bangun datar segitiga selain segitiga siku-siku terdapat segitiga sama sisi, segitiga sama kaki dan segitiga sembarang. Modifikasi program tersebut untuk dapat mengeluarkan output jenis segitiga yang lain (selain segitiga siku-siku). Silahkan menggunakan operator logika untuk menghubungkan multi kondisi.

```
if (totalSudut == 180) {  
    if ((sudut1 == 90) || (sudut2 == 90) || (sudut3 == 90)) {  
        System.out.println("Segitiga tersebut adalah segitiga siku-siku");  
    } else if ((sudut1 == sudut2) && (sudut2 == sudut3)) {  
        System.out.println("Segitiga tersebut adalah segitiga sama sisi");  
    } else if ((sudut1 == sudut2) || (sudut1 == sudut3) || (sudut3 ==  
sudut2)) {  
        System.out.println("Segitiga tersebut adalah bukan segitiga sama  
kaki");  
    } else {  
        System.out.println("Segitiga tersebut adalah bukan segitiga  
sembarang");  
    }  
} else {  
    System.out.println("Bukan Segitiga");  
}
```

2. Push dan commit hasil modifikasi anda ke repository !

2.3 Percobaan 3

- Menuliskan syntax

1. Deklarasi Scanner

```
Scanner input12 = new Scanner(System.in);
```

2. Deklarasikan variabel **kategori**, **penghasilan**, **gajiBersih**, dan **pajak**;

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

3. Tambahkan kode berikut ini untuk menerima input dari keyboard

```
System.out.print("Masukkan Kategori : ");  
kategori = input12.nextLine();  
System.out.print("Masukkan Besarnya penghasilan : ");  
penghasilan = input12.nextInt();
```

- ### 4. 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 - (pajak * penghasilan));  
    System.out.println("Penghasil Bersih : "+ gajiBersih);  
} else if (kategori.equalsIgnoreCase("pebisnis")) {  
    if (penghasilan <= 2000000) {  
        pajak = 0.1;  
    } else if (penghasilan <= 3000000) {  
        pajak = 0.15;  
    } else {  
        pajak = 0.25;  
    }  
    gajiBersih = (int) (penghasilan - (pajak * penghasilan));  
    System.out.println("Penghasil Bersih : "+ gajiBersih);  
} else {  
    System.out.println("Masukan kategori Salah");  
}
```

- Hasil

```
Masukkan Kategori : pebisnis
Masukkan Besarnya penghasilan : 23000000
Penghasil Bersih : 17250000
```

Pertanyaan!

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

```
Masukkan Kategori : PEBISNIS
Masukkan Besarnya penghasilan : 2000000
Penghasil Bersih : 1700000
```

= kegunaan dari equalsIgnoreCase adalah untuk membandingkan dua objek string tanpa memperdulikan perbedaan antara huruf besar dengan huruf kecil

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?

```
Masukkan Kategori : PEBISNIS
Masukkan Besarnya penghasilan : 2000000
Masukan kategori Salah
```

= kegunaan dari equals adalah untuk membandingkan dua objek string yang sama dari segi huruf besar dan kecil, jika ada salah satu ukurannya tidak sama, maka akan menghasilkan false

Tugas

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

```
if ((user.equals(username))) {  
    if ((pw.equals(password))) {  
        System.out.println("Login Berhasil");  
    } else {  
        System.out.println("Password salah");  
    }  
} else {  
    System.out.println("Username salah");  
}
```