LAPORAN PRAKTIKUM BASIS DATA LANJUT

JOB SHEET 5

SINTAKS PEMILIHAN 1

Oleh:

MOCHAMMAD SYAIFUDDIN ZUHRI NIM. 1941720013



PROGRAM STUDI TEKNIK INFORMATIKA

JURUSAN TEKNOLOGI INFORMASI

POLITEKNIK NEGERI MALANG

24 SEPTEMBER 2019

**SINTAKS PEMILIHAN 1**

1. **Tujuan**

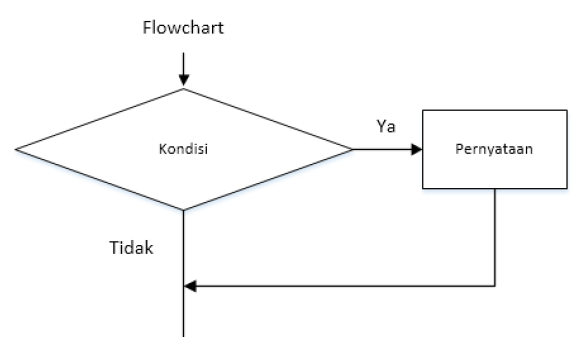
Mahasiswa mampu menyelesaikan permasalahan/studi kasus menggunakan sintaks pemilihan 1 dan mengimplemantasikannya dalam bahasa pemrogaman java.

1. **Alat dan Bahan**
2. PC/Leptop
3. Text Editor (Sublime Text)
4. JDK
5. **Uraian Teori**

Pada kehidupan sehari-hari kita selalu mengambil keputusan dengan mempertimbangkan berbagai hal/kondisi-kondisi. Sintaks pemilihan adalah statement pemilihan yang digunakan untuk mengatur kapan suatu perintah akan dijalankan. Dengan statement ini kita bisa mengatur kapan suatu perintah akan dijalankan, yaitu ketika telah dipenuhinya suatu syarat tertentu. Misalnya:

Jika **nilai lebih dari 70** maka **Diterima**

Pernyataan di atas mengandung sebab akibat. Bila dilihat, keterangan “nilai lebih dari 70” adalah merupakan suatu syarat, sedangakan “DITERIMA” ini akan dilakukan spabila syaratnya terpenuhi atau “nilai lebih dari 70”. Dalam dunia logika, istilah **terpenuhinya syarat** dapat dikatakan **syarat tersebut bernilai benar atau TRUE**. Selanjutnya pernyataan “jika…maka…” dapat diadopsi dalam programming. Untuk menyatakan pernyataan tersebut dalam programming, maka dapat digunakan statement sintaks pemilihan. Pada pembahasan di materi sintaks pemilihan 1 ini akan dipelajari tiga macam sintaks pemilihan yaitu if, if-else, if else if else dan switch..case.

1. Sintaks Pemilihan IF

Bentuk umum:

if (kondisi )

{

pernyataan;

}

* apabila *kondisi* bernilai benar, maka *pernyataan* akan dilaksanakan.
* apabila *kondisi* bernilai salah, maka *pernyataan* tidak akan dilaksanakan.

Pada sintaks pemilihan 1 ini, kita akan menggunakan operator hubungan (*relational operator*). Berikut ini operator hubungan dalam bahasa pemrograman java:

|  |  |
| --- | --- |
| Simbol Operator | Keterangan |
| = = | Sama dengan (*Equal to*) |
| > | Lebih besar dari (*Greater than*) |
| < | Lebih kecil dari (*Less than*) |
| >= | Lebih besar atau sama dengan (*Greater than or Equal to*) |
| <= | Lebih kecil dari atau sama dengan (*Less than or Equal to*) |
| != | Tidak sama dengan (*Not Equal to*) |

Implementasi atau penggunaan operator hubungan pada sintaks pemilihan adalah sebagai berikut:

**if ( )**

**Contoh:**

**if (A>5)**

**if (A==B)**

**if (A!=B)**

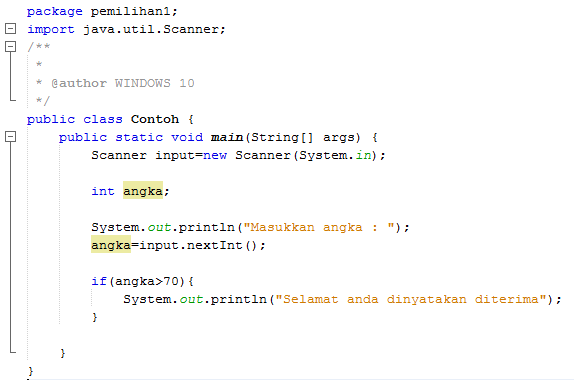
nilai

nilai

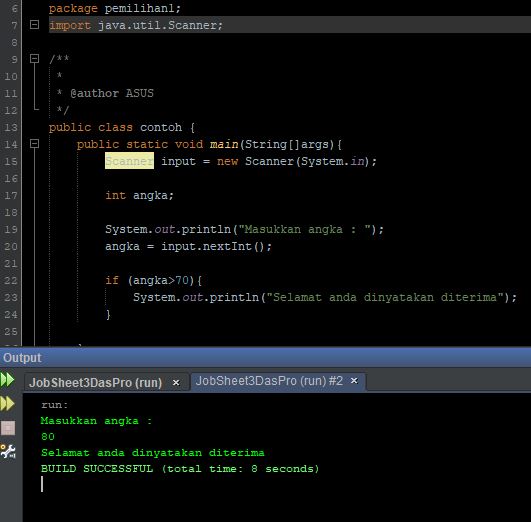
Memeriksa hubungan dua buah nilai

Operator hubungan

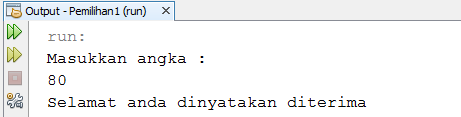
Contoh program:

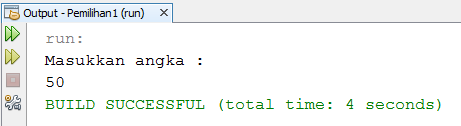


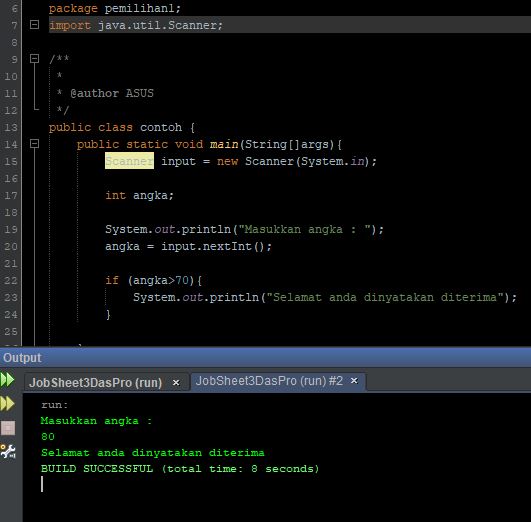
Ketika program diatas di jalankan dan kita masukkan angka 80 maka akan keluar tampilan “selamat anda dinyatakan diterima” hal ini disebabkan karena terdapat sebuah kondisi yang menyatakan bahwa jika nilai lebih dari 70 (>70) maka akan tampil “selamat anda dinyatakan diterima”, sedangkan jika kita masukkan angka 70 atau kurang dari 70 maka tidak akan terdapat tampilan apapun.

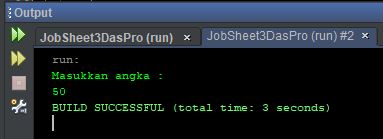


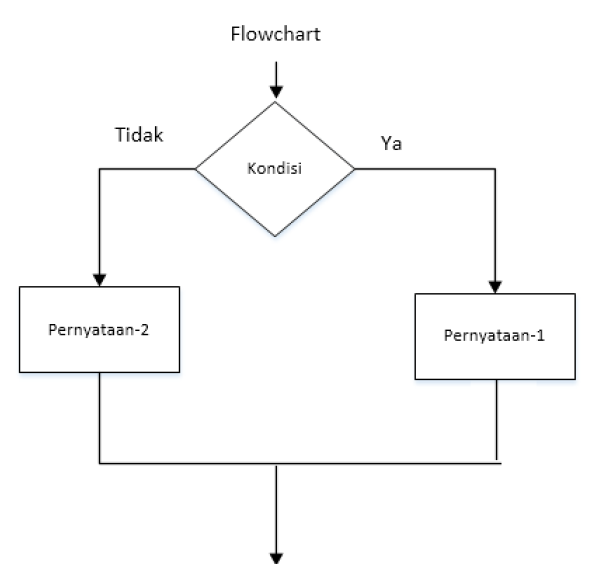
**Output:**











1. Sintaks Pemilihan IF-ELSE

Bentuk umum:

if (kondisi )

{

Pernyataan-1;

}

Else

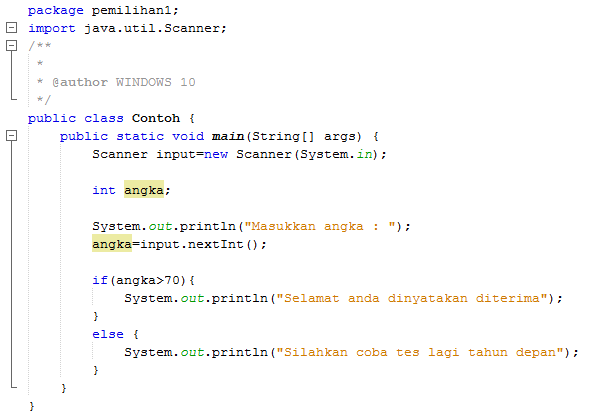
{

Pernyataan-2;

}

Struktur ini minimal memiliki 2 pernyataan. Jika kondisi yang diperiksa bernilai benar atau terpenuhi maka pernyataan pertama yang akan dilaksanakan dan jika kondisi yang diperiksa bernilai salah maka pernyataan yang kedua yang akan dilaksanakan.

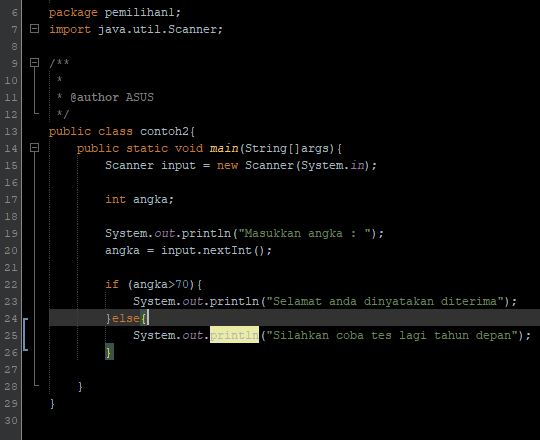
Contoh program:



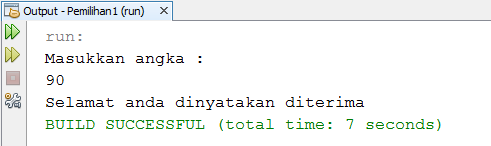
Pada contoh program IF-ELSE diatas tambahkan kode program

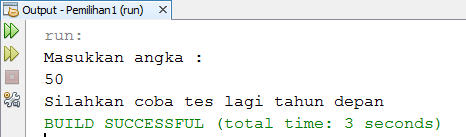


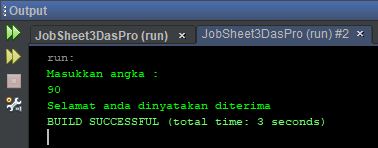
Sehingga ketika angka yang dimasukkan nilainya 70 atau kurang dari 70 maka akan muncul tampilan “Silahkan coba tes lagi tahun depan”

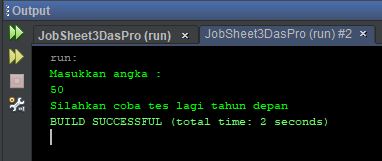


**Output:**



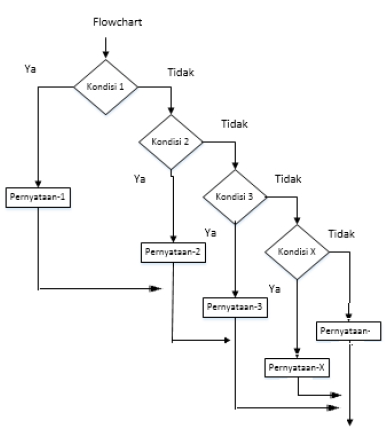
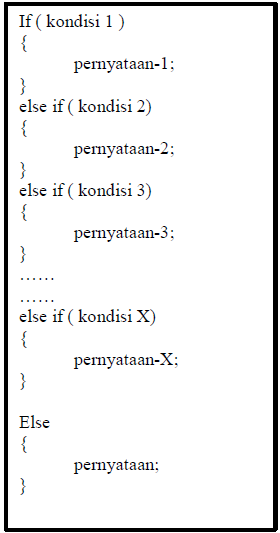






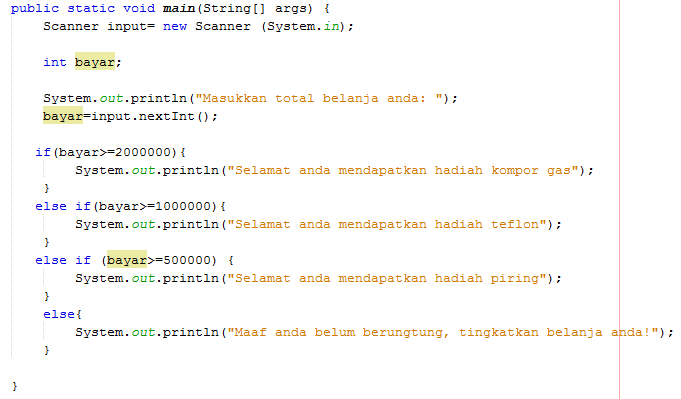
1. Sintaks Pemilihan IF... ELSE IF...ELSE

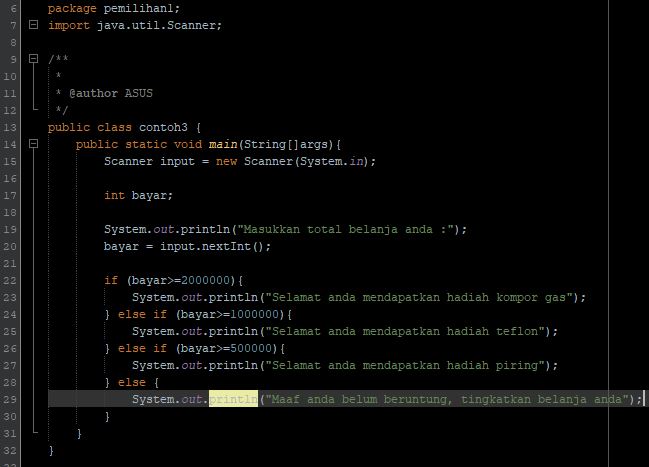
Bentuk umum:



Pada bentuk IF ELSE IF ELSE diatas, pernyataan 1 akan dijalankan apabila “kondisi 1” bernilai BENAR. JIka “kondisi 1” bernilai SALAH, maka akan dicek “kondisi 2”. Jika “kondisi 2” BENAR maka akan dijalankan statement 2, begitu seterusnya. Dan apabila tidak ada satupun syarat yang terpenuhi, barulah statement X akan dikerjakan.

Contoh program:





1. Sintaks Pemilihan SWITCH...CASE

Sintaks pemilihan ini digunakan untuk penyelesaian kondisi dengan kemungkinan yang terjadi cukup banyak. Struktur ini akan melaksanakan salah satu dari beberapa pernyataan “case” tergantung nilai kondisi yang ada di dalam switch. Selanjutnya proses diteruskan hingga ditemukan pernyataan “break”. Jika tidak ada nilai pada case yang sesuai dengan nilai kondisi, maka proses akan diteruskan kepada pernyataan yang ada di bawah “default”.

Bentuk umum:

switch (kondisi)

{

case konstanta -1:

pernyataan -1;

break ;

case konstanta -2:

pernyataan -2;

break;

…

…

case konstanta –x:

pernyataan –x;

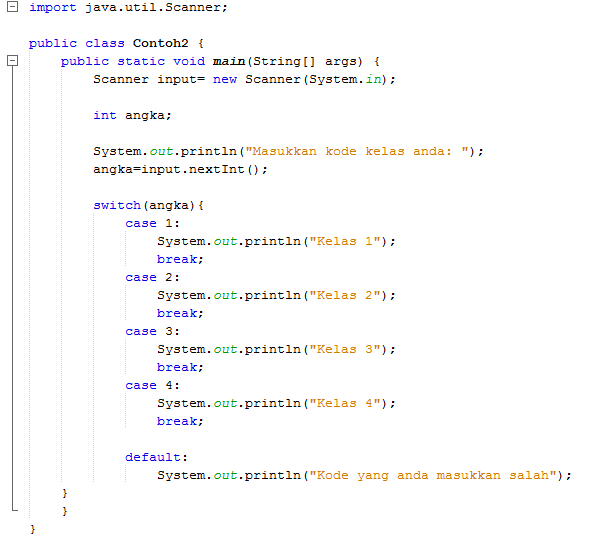
break –x;

default:

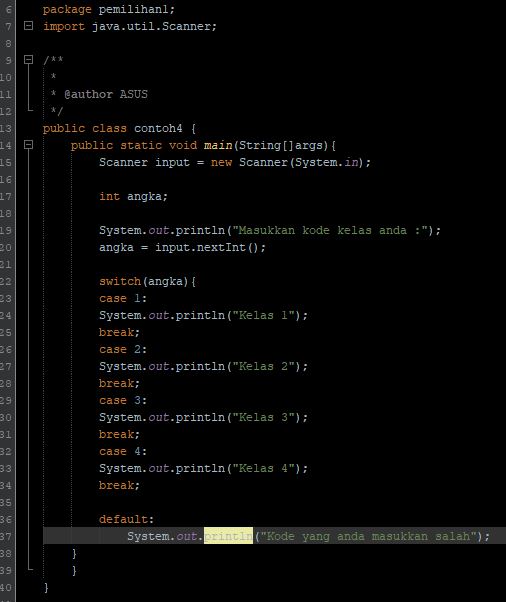
pernyataan;

}

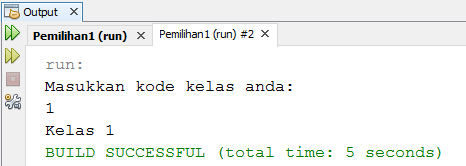
Contoh program:

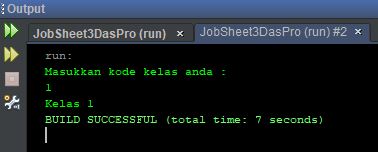


Pada contoh program switch case diatas jika dimasukkan angka 1 maka outputnya “kelas 1” dan seterusnya sampai angka 4. Jika inputan selain angka 1 s/d 4 maka outputnya adalah “kode yang anda masukkan salah”



**Output:**





1. **Langkah Praktikum**

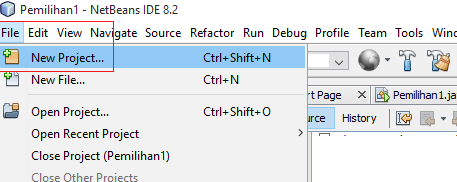
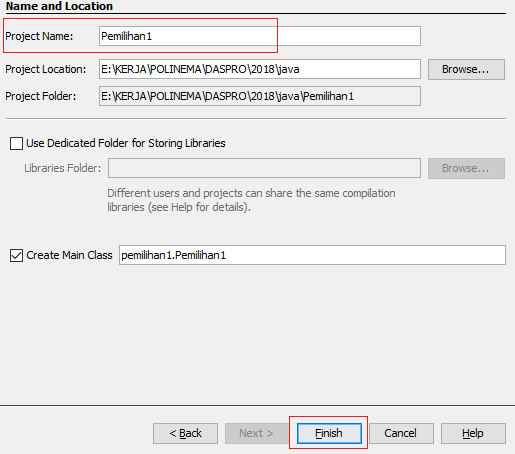
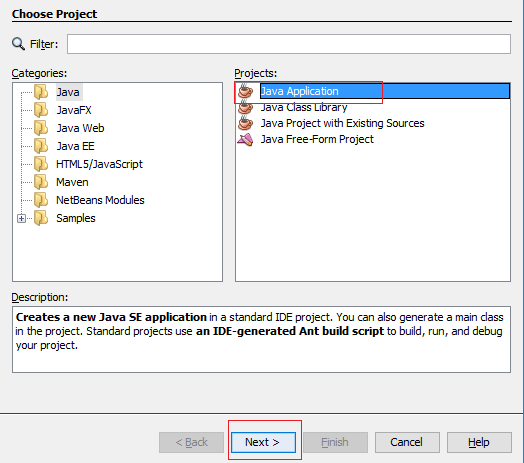
**Percobaan 1**

1. Perhatikan flowchart dibawah ini!

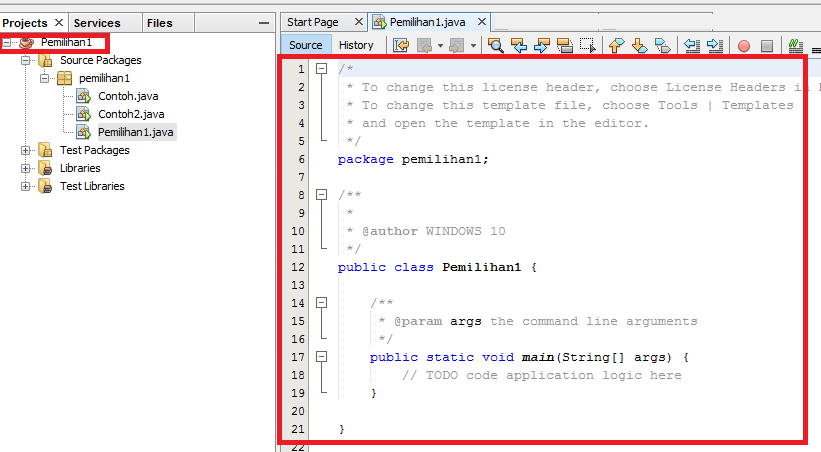
E:\KERJA\POLINEMA\DASPRO\ganjilgenap.tif

Flowchart diatas digunakan untuk menentukan bilangan ganjil/genap, selanjutnya kita akan membuat programnya berdasarkan flowchart di atas!

1. Buka netbeans yang sudah anda instal
2. Buat project baru dengan nama “Pemilihan1” dengan cara file -> New Project -> Java Application ->Next ->Pemilihan1 ->Finish

1. Maka akan terdapat 1 buat project dengan nama Pemilihan 1 dan didalamnya terdapat 1 file dengan nama Pemilihan1.java



1. Tambahkan import library Scanner.
2. Deklarasikan Scanner:



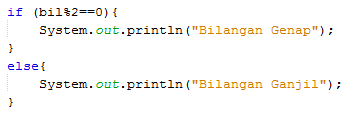
1. Buatlah variabel bertipe **int** dengan nama **bil**



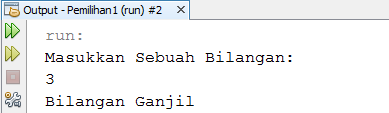
1. Tambahkan kode berikut ini untuk menerima inputan dari keyboard:



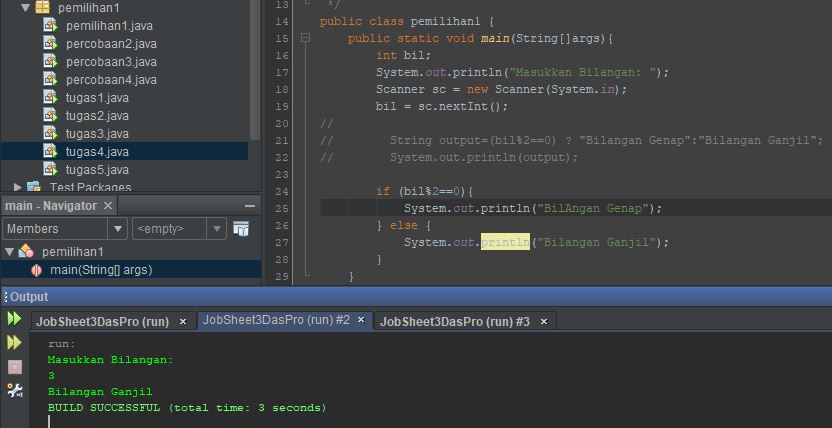
1. Buatlah struktur kondisi untuk mengecek apakah bilangan tersebut merupakan bilangan genap atau ganjil



1. Jalankan program, maka outputnya adalah sebagai berikut:

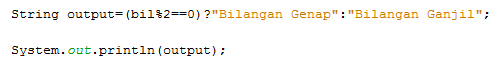
****

**Jawab :**

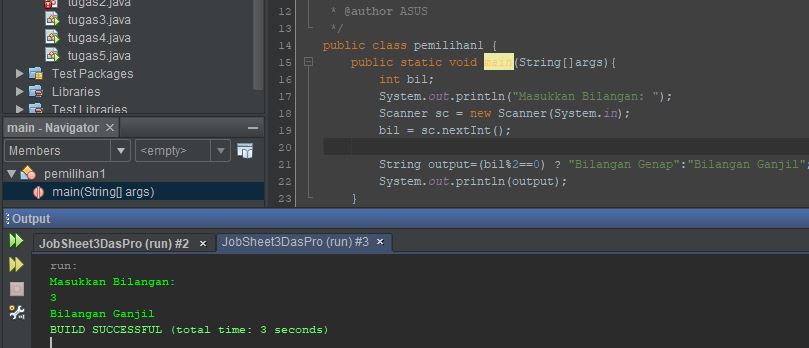


**Pertanyaan!**

1. Modifikasi program diatas dibagian struktur pemilihannya sehingga menjadi sebagai berikut:



1. Jalankan dan amatilah hasilnya!

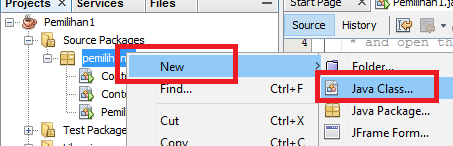
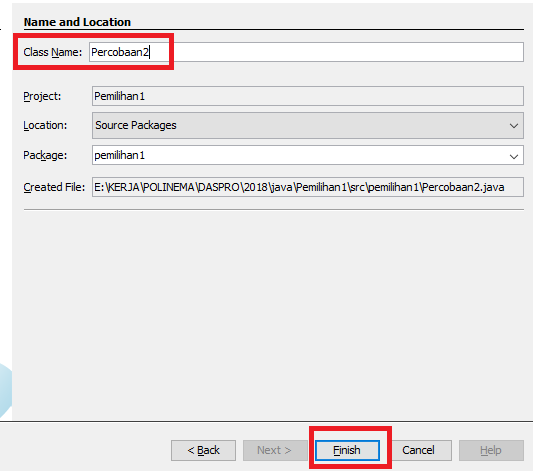


1. Jelaskan mengapa output program yang dimodifikasi sama dengan output program sebelum dimodifikasi!

**Jawab :** Karena dalam modifikasi menggunakan operator ternary. Operator ternary tersebut fungsinya sebagai pemilihan semisal true or false.

**Percobaan 2**

1. Buat file baru beri nama “Percobaan2.java” di project “Pemilihan1”.

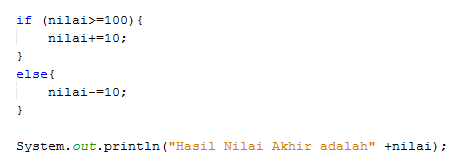
1. Tambahakan library Scanner.
2. Buatlah deklarasi Scanner.
3. Buat variabel nilai bertipe int.



1. Tuliskan perintah untuk memasukkan inputan.

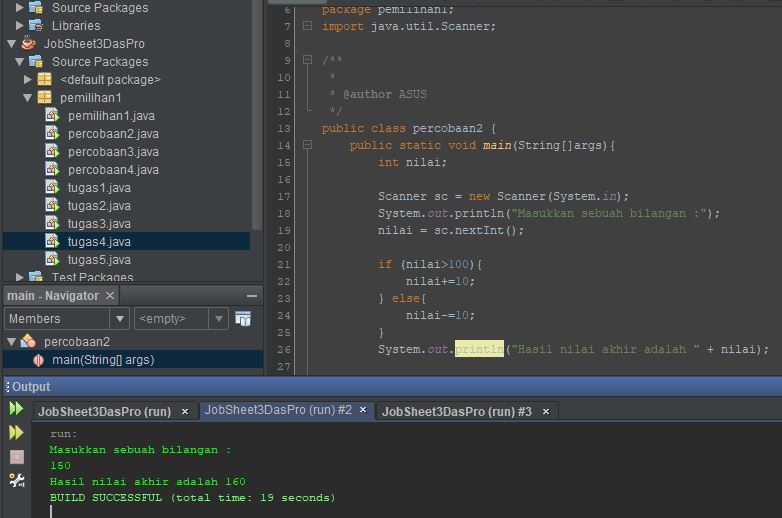


1. Tambahkan kode program kondisi dibawah ini



1. Jalankan program. Amati apa yang terjadi!

**Jawab :**



**Pertanyaan!**

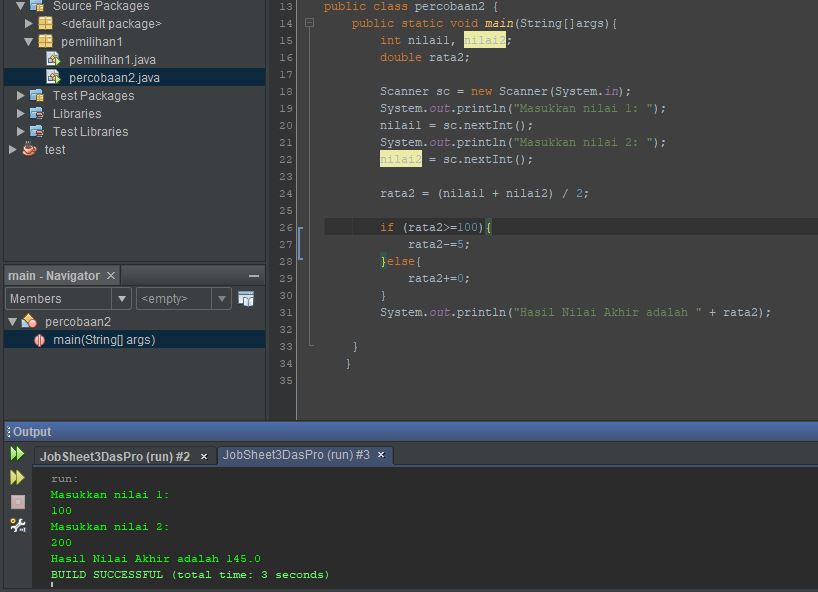
1. Jelaskan fungsi kode program berikut:

|  |
| --- |
| *nilai+=10;*  *nilai-=10;* |

**Jawab :** Fungsi kode program nilai+=10 adalah untuk menambah 10 pada hasil output. Sedangkan niali-=10 untuk mengurangi 10 pada hasil output.

1. Modifikasilah program diatas dimana inputannya yang awalnya hanya satu kemudian diganti 2 inputan (misal : nilai1 dan nilai2), lakukan perhitungan rata-rata kedua nilai tersebut jika nilainya lebih dari sama dengan 100 maka dikurangi 5, sedangkan jika nilai rata-rata tersebut kurang dari 100 maka akan langsung dicetak!

**Jawab :**



**Percobaan 3**

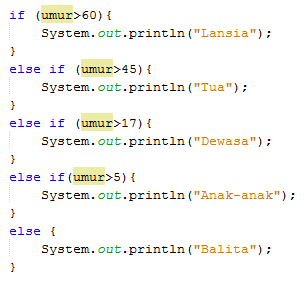
1. Buat file baru beri nama “Percobaan3.java” di project “Pemilihan1”.
2. Tambahakan library Scanner.
3. Buatlah deklarasi Scanner.
4. Buat variabel umur bertipe int.



1. Tuliskan perintah untuk memasukkan inputan.

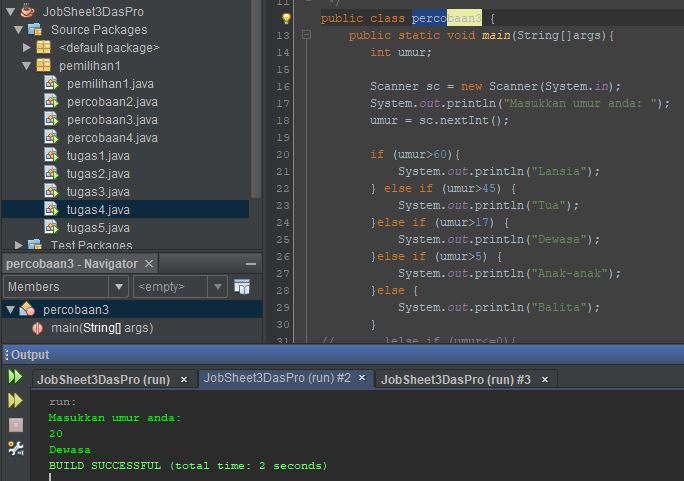


1. Tambahkan kode program kondisi dibawah ini



1. Jalankan program. Amati apa yang terjadi!

**Jawab :**



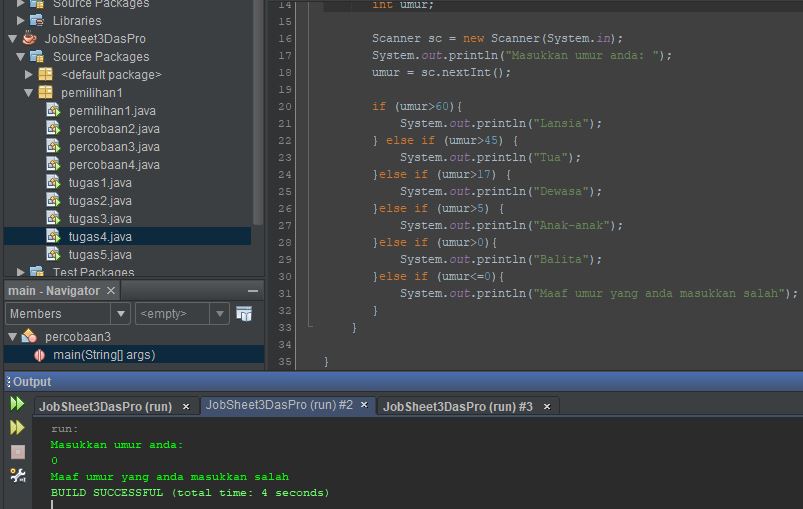
**Pertanyaan!**

1. Berapa jumlah kondisi yang ada pada program di percobaan 3? Jelaskan!

**Jawab :** sebelum modifikasi ada 4 kondisi, dan setelah modifikasi dari nomor 2 ada 5 kondisi.

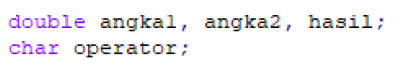
1. Modifikasi program diatas sehingga jika umur yang dimasukkan 0 tahun atau kurang dari 0 akan tampil output “Maaf umur yang anda masukkan salah”

**Jawab :**

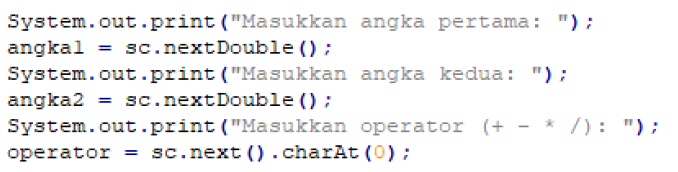


**Percobaan 4**

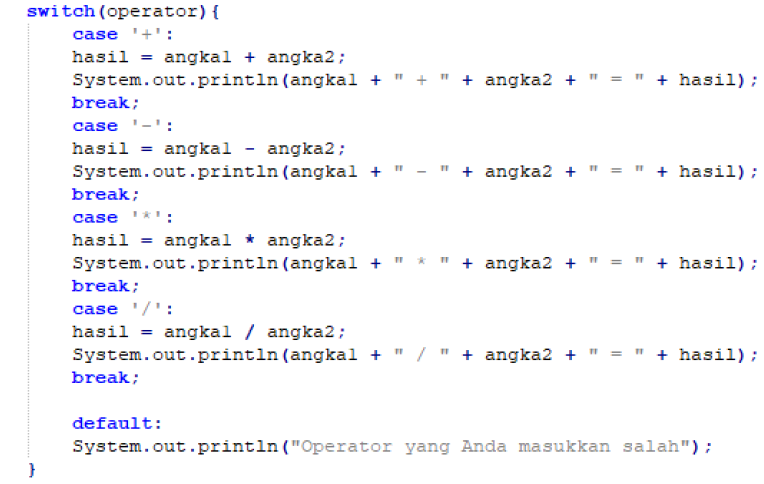
1. Buat file baru beri nama “Percobaan4.java” di project “Pemilihan1”.
2. Tambahakan library Scanner.
3. Buatlah deklarasi Scanner.
4. Buat variabel-variabel berikut:



1. Tuliskan perintah untuk memasukkan inputan.

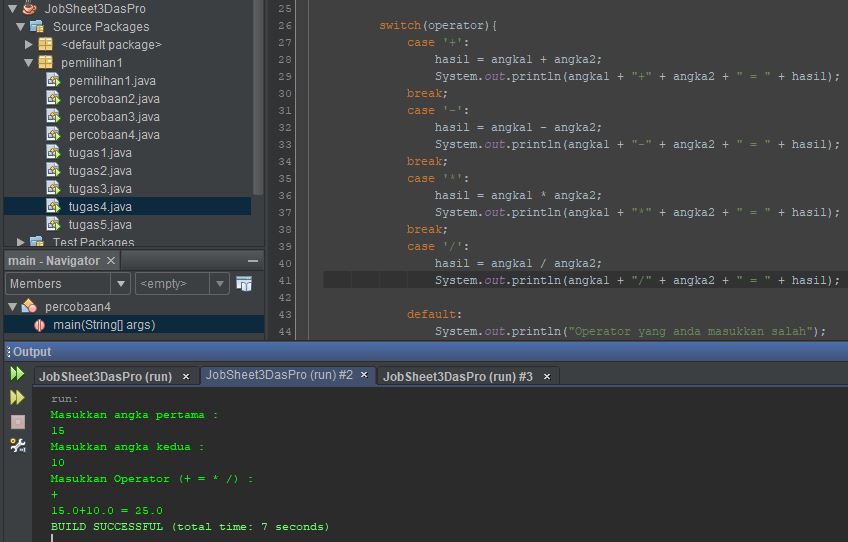


1. Tambahkan kode program kondisi dibawah ini



1. Jalankan program. Amati apa yang terjadi!

**Jawab :**



**Pertanyaan!**

1. Jelaskan fungsi dari *break* dan *default* pada percobaan 4 diatas!

**Jawab :** Fungsi break adalah untuk menghentikan proses running dalam setiap kondisi. Jika tidak memasukkan break, maka running akan terus lanjut sampai akhir. Sedangkan fungsi default adalah untuk menampilkan hasil terakhir jika tidak ada kondisi yang sesuai.

1. Jelaskan fungsi perintah kode program dibawah ini pada percobaan 4!

|  |
| --- |
| operator = sc.next().charAt(0); |

**Jawab :** Fungsi kode program tersebut adalah untuk memanggil input dari keyboard. Yang nantinya akan muncul variable operator yang sudah dimasukkan.