

**PRAKTIKUM  
PEMROGRAMAN BERORIENTASI OBJEK**

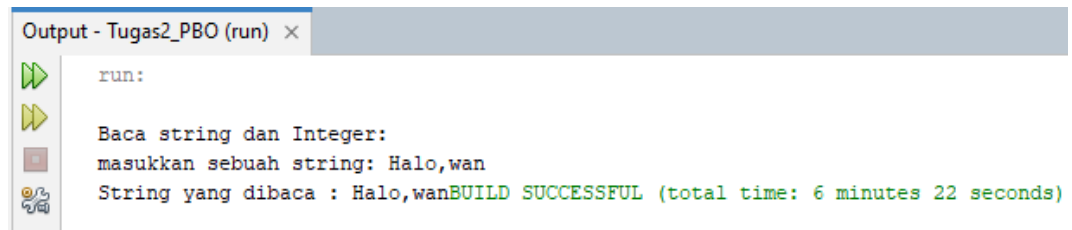


Nama : Wa Ode Wan Afifah Ahmad  
Stambuk : 13020230180  
Frekuensi : TI\_PBO-13 (B3)  
Dosen : Mardiyah Hasnawi, S.Kom., M.T., MTA

**PROGRAM STUDI TEKNIK INFORMATIKA  
FAKULTAS ILMU KOMPUTER  
UNIVERSITAS MUSLIM INDONESIA  
MAKASSAR  
2025**

## HASIL OUTPUT :

### 1. Program 1

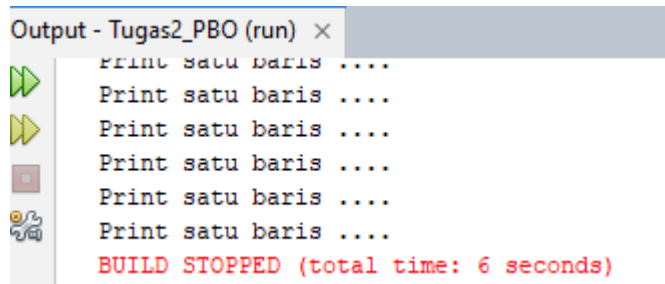


```
Output - Tugas2_PBO (run) x
run:
Baca string dan Integer:
masukkan sebuah string: Halo,wan
String yang dibaca : Halo,wan
BUILD SUCCESSFUL (total time: 6 minutes 22 seconds)
```

#### Penjelasan:

Pada program ini digunakan untuk membaca dan menampilkan sebuah string. Inputan akan dibaca oleh BufferedReader. Program tersebut mula-mul akan meminta pengguna untuk memasukkan sebuah string, inputan yang telah dimasukkan pengguna akan dibaca dan ditampilkan. Program ini hanya membaca satu baris saja.

### 2. Program 2

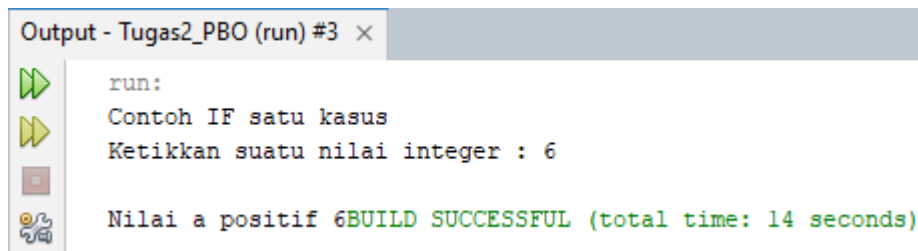


```
Output - Tugas2_PBO (run) x
Print satu baris ....
Print satu baris ....
Print satu baris ....
Print satu baris ....
Print satu baris ....
Print satu baris ....
BUILD STOPPED (total time: 6 seconds)
```

#### Penjelasan:

Pada program ini menggunakan while loop, tetapi program tersebut ada perulangan yang tidak berujung atau pesan yang dicetak tidak akan berhenti. Dengan kondisi while(true), menyatakan bahwa jika kondisinya benar maka program akan terus dieksekusi dan tidak berhenti. Tetapi jika ingin dihentikan menggunakan ctrl + c.

### 3. Program 3

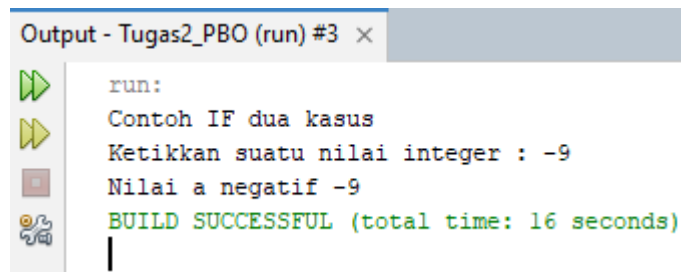


```
Output - Tugas2_PBO (run) #3 x
run:
Contoh IF satu kasus
Ketikkan suatu nilai integer : 6
Nilai a positif 6
BUILD SUCCESSFUL (total time: 14 seconds)
```

### Penjelasan:

Program tersebut menggunakan pengkondisian if, dimana setelah pengguna memasukkan suatu nilai integer dan menyimpannya ke variabel a yaitu `a = masukan.nextInt();` Setelah itu program akan memeriksa dengan if (`a >= 0`), nilai yang telah diinputkan sebelumnya diperiksa apakah nilai tersebut positif dan diatas atau sama dengan 0, jika iya nilai yang diinputkan akan dicetak.

### 4. Program 4



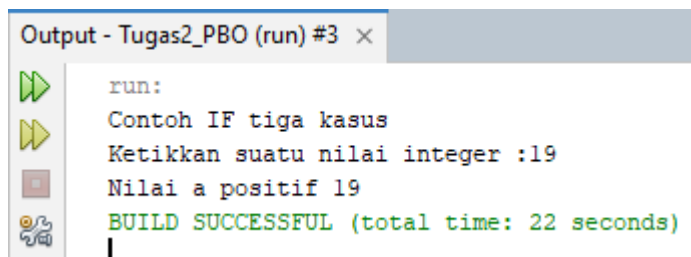
```
Output - Tugas2_PBO (run) #3 x
run:
Contoh IF dua kasus
Ketikkan suatu nilai integer : -9
Nilai a negatif -9
BUILD SUCCESSFUL (total time: 16 seconds)
```

### Penjelasan:

Pada program ini menggunakan pengkondisin if else. Dimana pengguna diminta untuk memasukkan sebuah nilai dengan menggunakan `a=masukan.nextInt();` yang variabel a nya telah di deklarasikan int a yaitu memiliki tipe data integer. Nilai yang telah diinputkan akan diperiksa menggunakan pengkondisian if else, yaitu:

- if (`a >= 0`) { , nilai diperiksa apakah nilai yang diinputkan positif dan lebih atau sama dengan 0, jika iya akan dicetak.
- }else /\* `a < 0` \*/  
    {  
        System.out.println ("Nilai a negatif "+ a);  
Tetapi jika nilai yang dimasukkan adalah negative maka perintah yang dijalankan adalah else dan nilainya akan dieksekusi

### 5. Program 5



```
Output - Tugas2_PBO (run) #3 x
run:
Contoh IF tiga kasus
Ketikkan suatu nilai integer :19
Nilai a positif 19
BUILD SUCCESSFUL (total time: 22 seconds)
```

### Penjelasan:

Program tersebut menggunakan pengkondisian percabangan if else if else, artinya di dalam sebuah kondisi if akan terdapat kondisi selanjutnya yaitu else if .

Pengkondisian ini digunakan jika kondisinya lebihh dari dua. Pada program tersebut setelah pengguna telah menginputkan nilai,maka akan di cek dengan pengkondisian yaitu:

- `if (a > 0){`  
`System.out.println ("Nilai a positif "+ a);`

Kondisi pertama, jika nilai yang diinputkan lebih besar dari 0 maka kondisi pertama yang akan dicetak.

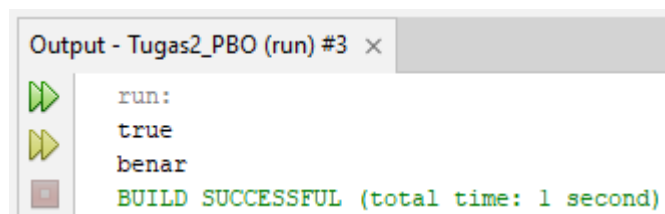
- `else if (a == 0){`  
`System.out.println ("Nilai Nol "+ a);`

Kondisi kedua, nilai yang diinputkan akandibandingkan apakah nilainya sama dengan 0 atau bukan, jika iya maka kondisi kedua yang dicetak.

- `else /* a > 0 */ {`  
`System.out.println ("Nilai a negatif "+ a);`

Kondisi ketiga akan dicetak jika nilai yang diinputkanbukan merupakan positif ataupun 0. Artinya jika nilainya negatifmaka kondisi ketiga yang akan dicetak

## 6.Program 6



### Penjelasan:

Pada program ini menggunakan if else untuk memeriksa setiap kondisi, dimana sebuah variabel yang memiliki nama bool dengan tipe data boolean. Tipe data Boolean hanya memiliki dua nilai yaitu true atau false. Pada program variabel bool diinisialisasi dengan nilai true, dan pada bagian pengkondisiannya yaitu :

- `if(bool) {`  
`System.out.print("true\n");`  
`} else`

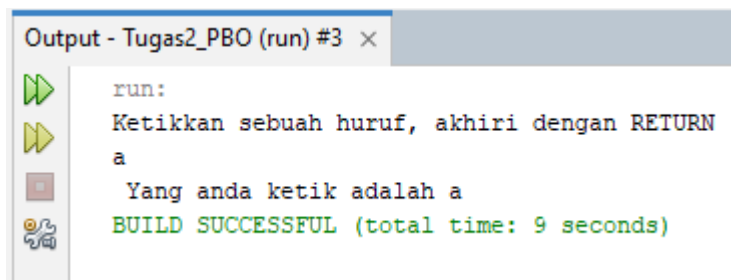
```
System.out.print("false\n");
```

Jika bool sama dengan true, maka akan mencetak if, tetapi jika nilainya bernilai salah maka program akan mengeksekusi elsenya.

- ```
if(!bool) {  
    System.out.print("salah\n");  
} else  
    System.out.print("benar\n");
```

Pada program ini menggunakan operator negasi. Pada operator ini mengembalikan nilai boolean, karena bool sama dengan true sedangkan jika menggunakan negasi yaitu !bool maka sama dengan false.

## 7. Program 7

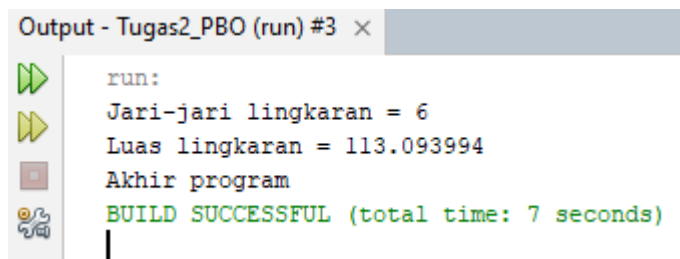


```
Output - Tugas2_PBO (run) #3 x  
run:  
Ketikkan sebuah huruf, akhiri dengan RETURN  
a  
Yang anda ketik adalah a  
BUILD SUCCESSFUL (total time: 9 seconds)
```

### Penjelasan:

Pada program variable cc dideklarasikan dengan menggunakan tipe data char, yang akan menyimpan karakter yang diinputkan oleh pengguna. Program tersebut menggunakan switch case. Jika pengguna memasukkan salah satu dari huruf vokal yaitu a, i, u, e, o, maka program akan mencetak nilai yang sesuai. Contohnya jika pengguna menginputkan u maka output akan menghasilkan u yang di proses adalah case u. Tetapi jika menginputkan selain dari huruf vokal, maka program akan memproses default dengan output yaitu “yang anda ketik adalah huruf mati.

## 8. Program 8

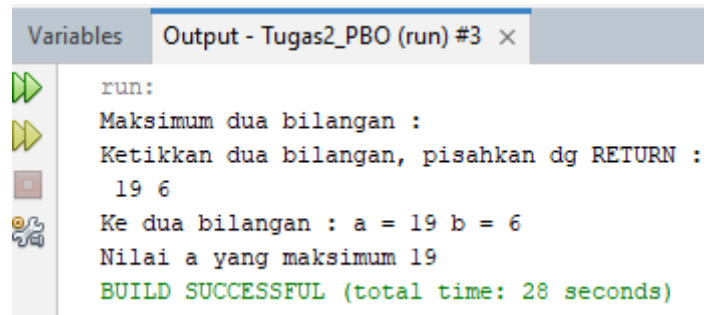


```
Output - Tugas2_PBO (run) #3 x  
run:  
Jari-jari lingkaran = 6  
Luas lingkaran = 113.093994  
Akhir program  
BUILD SUCCESSFUL (total time: 7 seconds)  
|
```

### Penjelasan:

Pada program tersebut mendeklarasikan nilai phi dengan nilai 3.145f dengan tipe data float. Pada deklarasi ini menggunakan final, yang artinya nilai phi tersebut tidak dapat diubah, karena nilai phi tersebut sudah mutlak. Selanjutnya pengguna diminta untuk menginputkan nilai jari jari pada deklarasi variabel r. Nilai jari-jari yang telah diinputkan akan di eksekusi untuk mencari luas lingkaran dengan rumus yaitu  $\phi * r * r$ .

## 9. Program 9



```

run:
Maksimum dua bilangan :
Ketikkan dua bilangan, pisahkan dg RETURN :
19 6
Ke dua bilangan : a = 19 b = 6
Nilai a yang maksimum 19
BUILD SUCCESSFUL (total time: 28 seconds)

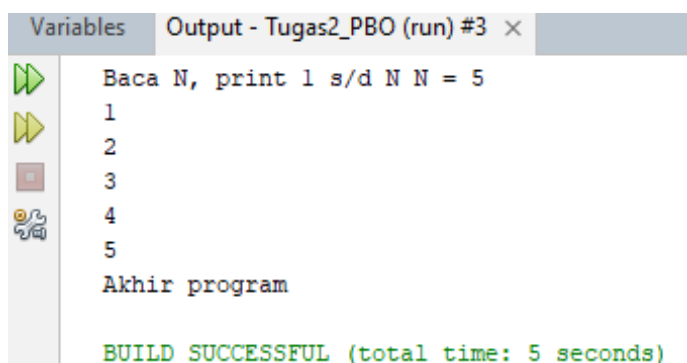
```

### Penjelasan:

Pada program tersebut meminta pengguna untuk memasukkan nilai variabel a dan b dengan tipe data integer, menggunakan pengondisian if else untuk menentukan inputan dari dua bilangan. Dengan kondisi yaitu:

- if (a >= b){  
System.out.println ("Nilai a yang maksimum "+ a);  
Jika nilai yang telah diinputkan pada a lebih besar sama dengan b maka outputya adalah nilai yang maksimum adalah a, begitupun sebaliknya.
- }else /\* a > b \*/{  
System.out.println ("Nilai b yang maksimum: "+ b);  
Jika nilai yang telah diinputkan pada b lebih besar sama dengan a maka outputya adalah nilai yang maksimum adalah b.

## 10. Program 10



```

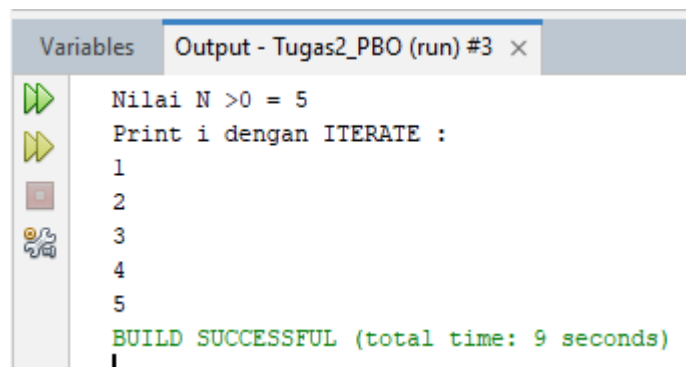
Variables Output - Tugas2_PBO (run) #3 x
Baca N, print 1 s/d N N = 5
1
2
3
4
5
Akhir program
BUILD SUCCESSFUL (total time: 5 seconds)

```

### Penjelasan:

Program tersebut memiliki dua variabel *i* dan *N* yang dideklarasikan dengan tipe data *int* atau bilangan bulat. Dimana *N* akan menyimpan bilangan yang dimasukkan oleh pengguna yang menentukan batas dari perulangan *for*. Variabel *i* digunakan untuk variabel penghitung pada perulangan. Setelah menginputkan nilai *N*, program akan melakukan perulangan untuk mencetak dari 1 hingga ke *N*. *i* diinisialisasi dengan 1 dan perulangan akan terus berjalan selama *i* kurang dari atau sama dengan *N*.

### 11. Program 11



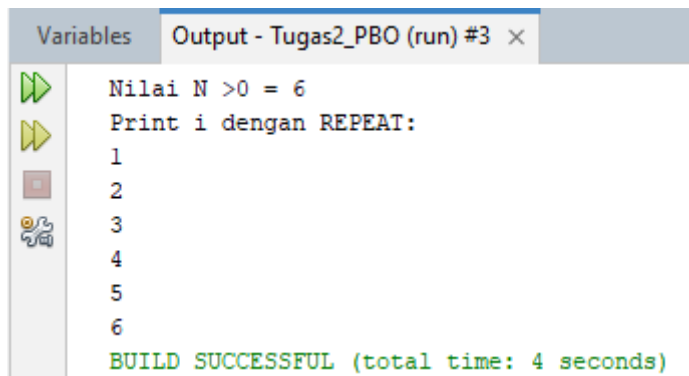
```
Variables | Output - Tugas2_PBO (run) #3 x
[Run] [Stop] [Debug] [Run and Debug]
Nilai N >0 = 5
Print i dengan ITERATE :
1
2
3
4
5
BUILD SUCCESSFUL (total time: 9 seconds)
```

### Penjelasan:

Program tersebut menggunakan perulangan *for ( ; ; )* untuk membuat perulangan tanpa kondisi eklipsit. Maksudnya adalah perulangan ini tidak memiliki kondisi awal, kondisi terminasi, ataupun increment. Oleh Karen itu perulangan ini akan berjalan tanpa henti kecuali ada perintah *break* yang dieksekusi. Program ini memilikipengkondisian yaitu:

- `if (i == N)`  
    `break;`  
Hal ini memeriksa apakah nilai *i* sama dengan *N*, jika iya perintah *break* akan dieksekusi dan mengakhiri proses perulangan.
- `else {`  
    `i++;`  
Tetapi jika *i* tidak sama dengan *N*, maka *else* yang akan dieksekusi dan menambah nilai *i* dengan 1 dan perulangan akan mencapai kondisi *i*==*N* dan perulangan tersebut berhenti.

## 12. Program 12

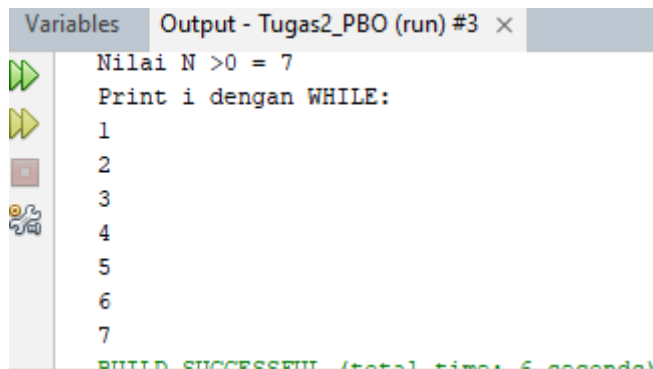


The screenshot shows a window titled "Output - Tugas2\_PBO (run) #3". It contains the following text: "Nilai N >0 = 6", "Print i dengan REPEAT:", followed by the numbers 1 through 6 on separate lines. At the bottom, it says "BUILD SUCCESSFUL (total time: 4 seconds)". On the left side of the window, there are icons for running, stepping through, and debugging the code.

### Penjelasan:

Pada program ini menggunakan do while, yang artinya minimal dilakukan sekali setelah itu proses perulangannya. Variabel i dan N dideklarasikan dengan tipe data integer. dan variabel i diinisialisasi dengan nilai 1. Program akan mencetak nilai i dan ditambah dengan 1. Perulangan akan terus dilakukan hingga i lebih besar dari N.

## 13. Program 13



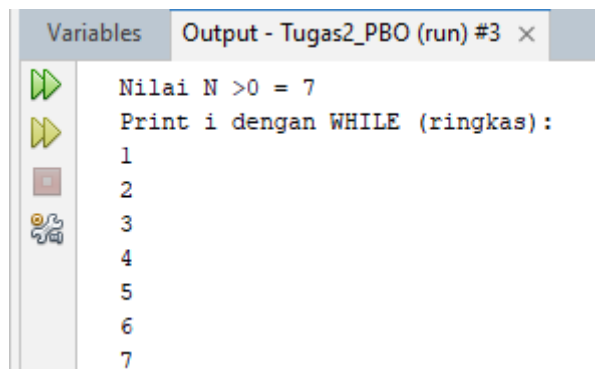
The screenshot shows a window titled "Output - Tugas2\_PBO (run) #3". It contains the following text: "Nilai N >0 = 7", "Print i dengan WHILE:", followed by the numbers 1 through 7 on separate lines. At the bottom, it says "BUILD SUCCESSFUL (total time: 6 seconds)". On the left side of the window, there are icons for running, stepping through, and debugging the code.

### Penjelasan:

Pada program menggunakan perulangan while untuk mencetak angka 1 yang telah dideklarasikan dengan variabel I, dan perulangan terus terjadi hingga N. Dengan perulangan whilenya yaitu memeriksa apakah I kurang dari sama dengan N, jika iya akan dieksekusi nilai i dan melakukan increment. Proses ini akan terus dilakukan hingga kondisi  $i \leq N$  terpenuhi.

## 14. Program 14

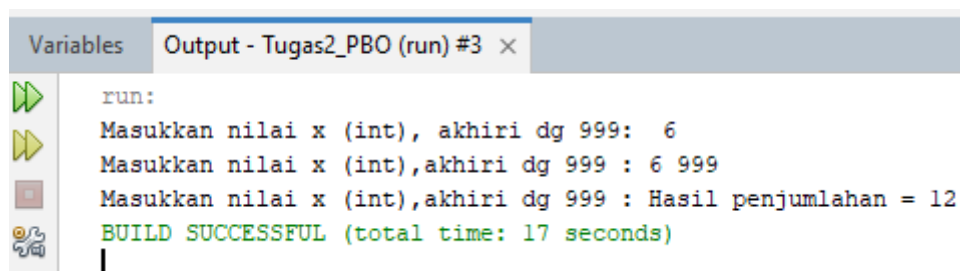




### Penjelasan:

Variabel *i* dan *N* dideklarasikan dengan tipe data integer. dan variabel *i* diinisialisasi dengan nilai 1 dan *N* diminta untuk memasukkan nilai dan menyimpannya. Selanjutnya program akan melakukan perulangan while untuk mencetak nilai *I* selama kurang dari atau sama dengan *N*, dan setelah nilai *i* dicetak selanjutnya akan ditambah 1.

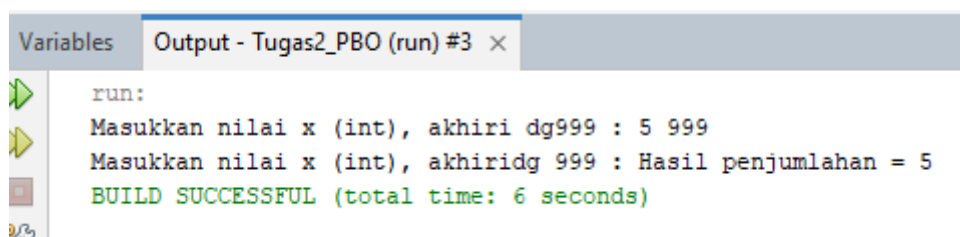
### 15. Program 15



### Penjelasan:

Program diminta untuk memasukkan nilai integer, dengan *sum* yang telah diinisialisasi ke 0 untuk menyimpan total penjumlahan. Jika input pertama adalah 999, program mencetak "Kasus kosong" dan berakhir. Program terus meminta input nilai integer. Jika input adalah 999, perulangan berhenti, tetapi jika tidak, nilai input ditambahkan ke *sum*.

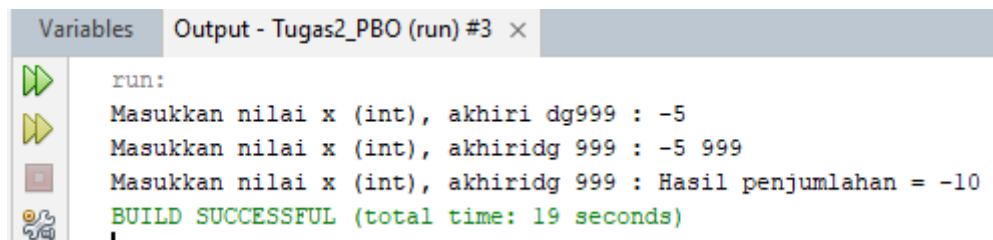
### 16. Program 16



### Penjelasan:

Program tersebut menggunakan perulangan do while, dimana program meminta pengguna untuk memasukkan nilai integer x. Jika nilai tersebut adalah 999, maka program akan mencetak “kasus kosong” dan program tersebut berhenti. Tetapi jika nilai x bukan 999, maka program menginisialisasi sum dengan 0 dan melakukan perulangan do while, nilai x ditambahkan ke sum dan menggunakan memasukkan nilai x berikutnya hingga menginputkan nilai 999.

### 17. Program 17

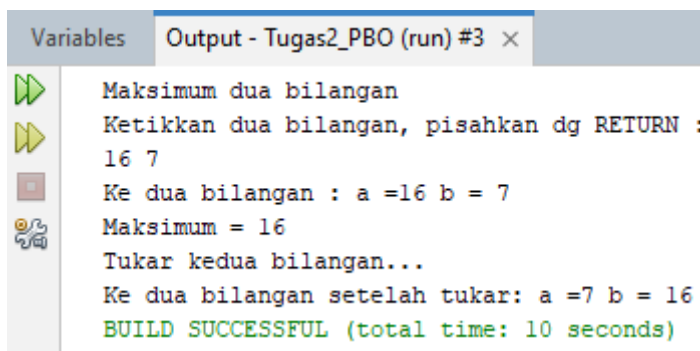


```
run:
Masukkan nilai x (int), akhiri dg 999 : -5
Masukkan nilai x (int), akhiridg 999 : -5 999
Masukkan nilai x (int), akhiridg 999 : Hasil penjumlahan = -10
BUILD SUCCESSFUL (total time: 19 seconds)
```

### Penjelasan:

Program meminta pengguna untuk memasukkan nilai integer x, dengan sum yang diinisialisasi dengan 0. Perulangan while digunakan untuk terus meminta inputan selama x tidak sama dengan 999. Dalam setiap iterasi, nilai x ditambahkan ke sum, tetapi setelah memasukkan 999 perulangannya akan berhenti dan program akan mencetak total penjumlahan dari nilai-nilai yang dimasukkan.

### 18. Program 18



```
Maksimum dua bilangan
Ketikkan dua bilangan, pisahkan dg RETURN :
16 7
Ke dua bilangan : a =16 b = 7
Maksimum = 16
Tukar kedua bilangan...
Ke dua bilangan setelah tukar: a =7 b = 16
BUILD SUCCESSFUL (total time: 10 seconds)
```

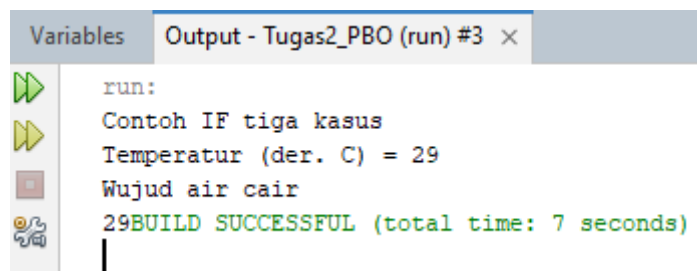
### Penjelasan:

Pada program ini menggunakan prosedur dan fungsi, dengan dua parameter integer a dan b, dengan menggunakan operator ternary akan mengembalikan nilai

maksimum di antara kedua bilangan. Selanjutnya nilai a dan b akan ditukar dengan menggunakan variabel sementara yaitu temp. maka:

- Menggunakan objek scanner untuk membaca dua bilangan bulat dari pengguna.
- Mencetak kedua bilangan yang dibaca.
- Memanggil fungsi untuk mencetak nilai maksimum, dengan fungsi maxab
- Memanggil prosedur untuk menukar kedua bilangan dan mencetak hasilnya, dengan prosedur tukar

## 19. Program 19



The screenshot shows a Java IDE's output window titled "Output - Tugas2\_PBO (run) #3". The output text is as follows:

```
run:
Contoh IF tiga kasus
Temperatur (der. C) = 29
Wujud air cair
29BUILD SUCCESSFUL (total time: 7 seconds)
```

### Penjelasan:

Pada program tersebut menentukan wujud air berdasarkan temperatur yang dimasukkan oleh pengguna. Program meminta pengguna untuk memasukkan temperatur dalam derajat celcius yang sudah dideklarasikan di variabel T. Pada program ini memiliki pengkondisian if else if else untuk menentukan wujud air berdasarkan temperature yaitu:

- Jika  $T < 0$  maka akan mencetak wujud air beku
- Jika  $0 \leq T$  maka akan mencetak wujud air cair
- Jika  $T > 100$  maka akan mencetak wujud air uap/gas