

**LAPORAN
PEMROGRAMAN BERORIENTASI OBJEK**



Nama : Nazwa Syalaisa Haq
Stambuk : 13020230063
Kelas : B1
Dosen : Mardiyah Hasnawi, S.Kom., M.T

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

Output:

Penjelasan:

Saat program dijalankan, pengguna diminta memasukkan sebuah string. Setelah mengetikkan teks dan menekan **Enter**, input tersebut disimpan dalam variabel str. Program kemudian menampilkan kembali teks yang telah dimasukkan menggunakan `System.out.println`. Jika pengguna langsung menekan **Enter** tanpa mengetikkan apapun, program akan menampilkan string kosong. Program ini juga memperlihatkan bagaimana membaca input dari pengguna menggunakan `BufferedReader` serta menampilkan kembali hasil input dalam Java.

Output:

[illegible]

Penjelasan:

Program ini menjalankan perulangan tanpa henti menggunakan while(true), yang berarti program akan terus berjalan tanpa batas. Saat dijalankan, program pertama-tama mencetak pesan "Program akan looping, akhiri dengan ^c" sebagai peringatan bahwa program harus dihentikan secara manual. Setelah itu, perulangan akan terus mencetak "Print satu baris" berulang kali di layar tanpa berhenti, kecuali pengguna menekan Ctrl + C atau menutup program secara paksa.

3. Class If1**Output:**

```
D:\Nazwa Syalaisa Haq_0063_B1\Tugas 2\Source Code>javac If1.java
D:\Nazwa Syalaisa Haq_0063_B1\Tugas 2\Source Code>java If1
Contoh IF satu kasus
Ketikkan suatu nilai integer : 3

Nilai a positif 3
D:\Nazwa Syalaisa Haq_0063_B1\Tugas 2\Source Code>
```

Penjelasan:

Program ini membaca sebuah bilangan integer dari pengguna dan mengecek apakah nilainya positif atau nol. Pertama, program menggunakan Scanner untuk menerima input dari keyboard. Setelah pengguna memasukkan angka, nilai tersebut disimpan dalam variabel a. Kemudian, program memeriksa apakah a lebih besar atau sama dengan nol menggunakan pernyataan if. Jika kondisi ini terpenuhi, program mencetak pesan bahwa nilai a adalah positif beserta angkanya. Jika angka yang dimasukkan negatif, program tidak menampilkan apa pun karena tidak ada blok else.

4. Class If2**Output:**

```
D:\Nazwa Syalaisa Haq_0063_B1\Tugas 2\Source Code>java If2
Contoh IF dua kasus
Ketikkan suatu nilai integer :4
Nilai a positif 4

D:\Nazwa Syalaisa Haq_0063_B1\Tugas 2\Source Code>javac If2.java
D:\Nazwa Syalaisa Haq_0063_B1\Tugas 2\Source Code>java If2
Contoh IF dua kasus
Ketikkan suatu nilai integer :-5
Nilai a negatif -5
D:\Nazwa Syalaisa Haq_0063_B1\Tugas 2\Source Code>
```

Penjelasan:

Program ini membaca sebuah bilangan integer dari pengguna dan menentukan apakah bilangan tersebut positif atau negatif. Pertama, program menggunakan Scanner untuk menerima input dari keyboard. Setelah pengguna memasukkan angka, nilainya disimpan dalam variabel a. Kemudian, program mengecek kondisi dengan if-else: jika a lebih besar atau

sama dengan nol, program mencetak bahwa angka tersebut positif; jika tidak, program mencetak bahwa angka tersebut negatif. Dengan adanya blok else, program akan selalu memberikan output, baik untuk angka positif maupun negatif.

5. Class If3

Output:

```
D:\Nazwa Syalaisa Haq_0063_B1\Tugas 2\Source Code>java If3
Contoh IF tiga kasus
Ketikkan suatu nilai integer :6
Nilai a positif 6

D:\Nazwa Syalaisa Haq_0063_B1\Tugas 2\Source Code>javac If3.java

D:\Nazwa Syalaisa Haq_0063_B1\Tugas 2\Source Code>java If3
Contoh IF tiga kasus
Ketikkan suatu nilai integer :0
Nilai Nol 0

D:\Nazwa Syalaisa Haq_0063_B1\Tugas 2\Source Code>javac If3.java

D:\Nazwa Syalaisa Haq_0063_B1\Tugas 2\Source Code>java If3
Contoh IF tiga kasus
Ketikkan suatu nilai integer :-9
Nilai a negatif -9

D:\Nazwa Syalaisa Haq_0063_B1\Tugas 2\Source Code>
```

Penjelasan:

Program ini membaca sebuah bilangan integer dari pengguna dan menentukan apakah bilangan tersebut positif, nol, atau negatif. Pertama, program menggunakan Scanner untuk menerima input dari keyboard dan menyimpannya dalam variabel a. Kemudian, program mengecek kondisi dengan if-else if-else: jika a lebih dari nol, program mencetak bahwa angka tersebut positif; jika a sama dengan nol, program mencetak bahwa angka tersebut adalah nol; dan jika tidak memenuhi kedua kondisi tersebut, program mencetak bahwa angka tersebut negatif. Dengan tiga kasus ini, program dapat menangani semua kemungkinan nilai yang dimasukkan pengguna.

6. Class KasusBoolean

Output:

```
D:\Nazwa Syalaisa Haq_0063_B1\Tugas 2\Source Code>javac KasusBoolean.java

D:\Nazwa Syalaisa Haq_0063_B1\Tugas 2\Source Code>java KasusBoolean
true
benar

D:\Nazwa Syalaisa Haq_0063_B1\Tugas 2\Source Code>
```

Penjelasan:

Program ini menunjukkan penggunaan tipe data boolean dalam Java. Pertama, variabel bool bertipe boolean dideklarasikan dan diberi nilai true. Kemudian, program mengecek nilainya dengan if: jika bool bernilai true, program mencetak "true", jika tidak, mencetak "false". Setelah itu, program melakukan pengecekan dengan negasi (!bool), yang membalik nilai true

menjadi false. Karena bool awalnya true, negasinya menjadi false, sehingga program melewati cetakan "salah" dan justru mencetak "benar".

7. Class KasusSwitch

Output:

```
D:\Nazwa Syalaisa Haq_0063_B1\Tugas 2\Source Code>javac KasusSwitch.java

D:\Nazwa Syalaisa Haq_0063_B1\Tugas 2\Source Code>java KasusSwitch
Ketikkan sebuah huruf, akhiri dengan RETURN
a
Yang anda ketik adalah a

D:\Nazwa Syalaisa Haq_0063_B1\Tugas 2\Source Code>javac KasusSwitch.java

D:\Nazwa Syalaisa Haq_0063_B1\Tugas 2\Source Code>java KasusSwitch
Ketikkan sebuah huruf, akhiri dengan RETURN
n
Yang anda ketik adalah huruf mati

D:\Nazwa Syalaisa Haq_0063_B1\Tugas 2\Source Code>
```

Penjelasan:

Program ini menggunakan struktur switch untuk mengecek apakah huruf yang dimasukkan oleh pengguna merupakan huruf vokal atau bukan. Pertama, program meminta pengguna mengetikkan sebuah huruf menggunakan Scanner, lalu mengambil karakter pertama dari input dengan `charAt(0)`. Kemudian, program membandingkan karakter tersebut dengan huruf vokal 'a', 'u', 'e', 'i', dan 'o'. Jika karakter yang dimasukkan sesuai dengan salah satu huruf vokal, program mencetak pesan yang sesuai. Jika bukan huruf vokal, program masuk ke blok default dan mencetak bahwa huruf yang diketik adalah huruf mati (konsonan).

8. Class Konstant

Output:

```
D:\Nazwa Syalaisa Haq_0063_B1\Tugas 2\Source Code>javac Konstant.java

D:\Nazwa Syalaisa Haq_0063_B1\Tugas 2\Source Code>java Konstant
Jari-jari lingkaran =16
Luas lingkaran = 804.224
Akhir program

D:\Nazwa Syalaisa Haq_0063_B1\Tugas 2\Source Code>
```

Penjelasan:

Program ini menghitung luas lingkaran dengan menggunakan konstanta PHI. Pertama, variabel PHI dideklarasikan sebagai final float dengan nilai 3.1415, yang berarti nilainya tidak bisa diubah. Program kemudian meminta pengguna memasukkan jari-jari lingkaran (r) menggunakan Scanner. Setelah itu, program menghitung luas lingkaran dengan rumus $\pi \times r \times r$ dan mencetak hasilnya. Terakhir, program menampilkan pesan "Akhir program" untuk menandakan bahwa eksekusi telah selesai.

9. Class Max2

Output:

```
D:\Nazwa Syalaisa Haq_0063_B1\Tugas 2\Source Code>javac Max2.java

D:\Nazwa Syalaisa Haq_0063_B1\Tugas 2\Source Code>java Max2
Maksimum dua bilangan :
Ketikkan dua bilangan, pisahkan dg RETURN :
16
10
Ke dua bilangan : a = 16 b = 10
Nilai a yang maksimum 16

D:\Nazwa Syalaisa Haq_0063_B1\Tugas 2\Source Code>javac Max2.java

D:\Nazwa Syalaisa Haq_0063_B1\Tugas 2\Source Code>java Max2
Maksimum dua bilangan :
Ketikkan dua bilangan, pisahkan dg RETURN :
10
16
Ke dua bilangan : a = 10 b = 16
Nilai b yang maksimum: 16

D:\Nazwa Syalaisa Haq_0063_B1\Tugas 2\Source Code>
```

Penjelasan:

Program ini membaca dua bilangan yang dimasukkan oleh pengguna, kemudian menentukan dan menampilkan bilangan yang lebih besar. Pertama, program meminta pengguna mengetikkan dua bilangan dengan memisahkannya menggunakan tombol RETURN. Setelah itu, bilangan tersebut disimpan dalam variabel a dan b, lalu ditampilkan di layar. Program kemudian menggunakan struktur if-else untuk membandingkan kedua bilangan: jika a lebih besar atau sama dengan b, maka a adalah nilai maksimum; jika tidak, maka b yang lebih besar. Program ini menunjukkan cara membaca input pengguna dengan Scanner serta menentukan nilai maksimum dalam Java.

10. Class PriFor

Output:

```
D:\Nazwa Syalaisa Haq_0063_B1\Tugas 2\Source Code>javac PriFor.java

D:\Nazwa Syalaisa Haq_0063_B1\Tugas 2\Source Code>java PriFor
Baca N, print 1 s/d N
N = 3
1
2
3
Akhir program

D:\Nazwa Syalaisa Haq_0063_B1\Tugas 2\Source Code>
```

Penjelasan:

Program ini membaca input dari pengguna menggunakan Scanner untuk menentukan batas angka yang akan dicetak. Setelah pengguna memasukkan nilai N, program menggunakan perulangan for untuk mencetak angka dari 1 hingga N secara berurutan. Setiap iterasi mencetak satu angka ke layar, lalu setelah perulangan selesai, program menampilkan pesan "Akhir program" sebagai tanda bahwa eksekusi telah selesai. Program ini juga menunjukkan bagaimana cara membaca input pengguna dan menggunakan perulangan untuk menampilkan sejumlah angka dalam Java.

11. Class PrintIterasi

Output:

```
D:\Nazwa Syalaisa Haq_0063_B1\Tugas 2\Source Code>javac PrintIterasi.java

D:\Nazwa Syalaisa Haq_0063_B1\Tugas 2\Source Code>java PrintIterasi
Nilai N >0 = 5
Print i dengan ITERATE :
1
2
3
4
5

D:\Nazwa Syalaisa Haq_0063_B1\Tugas 2\Source Code>
```

Penjelasan:

Program ini meminta pengguna memasukkan nilai N yang lebih besar dari 0, lalu mencetak angka dari 1 hingga N menggunakan perulangan for tanpa kondisi. Variabel i dimulai dari 1 dan akan terus bertambah setiap iterasi. Jika nilai i sudah sama dengan N, perulangan dihentikan dengan break. Program ini menunjukkan penggunaan perulangan for tanpa ekspresi kondisi dalam Java untuk mengontrol iterasi dengan logika di dalam bloknya.

12. Class PrintRepeat

Output:

```
D:\Nazwa Syalaisa Haq_0063_B1\Tugas 2\Source Code>java PrintRepeat
Nilai N >0 = 8
Print i dengan REPEAT:
1
2
3
4
5
6
7
8

D:\Nazwa Syalaisa Haq_0063_B1\Tugas 2\Source Code>
```

Penjelasan:

Program ini meminta pengguna memasukkan nilai N yang lebih besar dari 0, lalu mencetak angka dari 1 hingga N menggunakan perulangan do-while. Perulangan ini dijalankan setidaknya sekali, karena pengecekan kondisi dilakukan setelah pernyataan dalam blok perulangan dieksekusi. Variabel i dimulai dari 1 dan terus bertambah setiap iterasi hingga mencapai N, kemudian perulangan berhenti. Program ini menunjukkan penggunaan perulangan do-while dalam Java, di mana minimal satu iterasi selalu dijalankan.

13. Class PrintWhile

Output:

```
D:\Nazwa Syalaisa Haq_0063_B1\Tugas 2\Source Code>javac PrintWhile.java

D:\Nazwa Syalaisa Haq_0063_B1\Tugas 2\Source Code>java PrintWhile
Nilai N >0 = 4
Print i dengan WHILE:
1
2
3
4

D:\Nazwa Syalaisa Haq_0063_B1\Tugas 2\Source Code>
```

Penjelasan:

Program ini meminta pengguna memasukkan nilai N yang lebih besar dari 0, lalu mencetak angka dari 1 hingga N menggunakan perulangan while. Variabel i dimulai dari 1 dan selama nilainya kurang dari atau sama dengan N, perulangan terus berjalan, mencetak nilai i setiap iterasi, lalu menambahkannya satu per satu. Program ini menunjukkan penggunaan perulangan while dalam Java, yang berguna ketika jumlah iterasi belum pasti dan bergantung pada kondisi tertentu.

14. Class PrintWhile1

Output:

```
D:\Nazwa Syalaisa Haq_0063_B1\Tugas 2\Source Code>javac PrintWhile1.java

D:\Nazwa Syalaisa Haq_0063_B1\Tugas 2\Source Code>java PrintWhile1
Nilai N >0 = 7
Print i dengan WHILE (ringkas):
1
2
3
4
5
6
7

D:\Nazwa Syalaisa Haq_0063_B1\Tugas 2\Source Code>
```

Penjelasan:

Program ini meminta pengguna memasukkan nilai N yang lebih besar dari 0, lalu mencetak angka dari 1 hingga N menggunakan perulangan while dalam bentuk yang lebih ringkas. Variabel i diinisialisasi dengan nilai 1, kemudian dalam perulangan while, nilai i langsung dicetak dan ditingkatkan dengan i++ dalam satu baris. Program ini menunjukkan cara menulis perulangan while secara lebih efisien dalam Java.

15. Class PrintXinterasi

Output:

```
D:\Nazwa Syalaisa Haq_0063_B1\Tugas 2\Source Code>javac PrintXinterasi.java

D:\Nazwa Syalaisa Haq_0063_B1\Tugas 2\Source Code>java PrintXinterasi
Masukkan nilai x (int), akhiri dg 999: 999
Kasus kosong
Hasil penjumlahan = 0

D:\Nazwa Syalaisa Haq_0063_B1\Tugas 2\Source Code>javac PrintXinterasi.java

D:\Nazwa Syalaisa Haq_0063_B1\Tugas 2\Source Code>java PrintXinterasi
Masukkan nilai x (int), akhiri dg 999: 10
Masukkan nilai x (int),akhiri dg 999 : 50
Masukkan nilai x (int),akhiri dg 999 : 999
Hasil penjumlahan = 60

D:\Nazwa Syalaisa Haq_0063_B1\Tugas 2\Source Code>
```

Penjelasan:

Program ini membaca serangkaian angka dari pengguna dan menjumlahkannya hingga pengguna memasukkan 999 sebagai tanda berhenti. Jika angka pertama yang dimasukkan adalah 999, program langsung mencetak "Kasus kosong", karena tidak ada angka yang diproses. Jika tidak, angka pertama disimpan ke dalam Sum, lalu program masuk ke perulangan for(;;) yang terus meminta input sampai pengguna memasukkan

999. Setiap angka yang dimasukkan ditambahkan ke Sum, dan setelah perulangan berakhir, hasil penjumlahan ditampilkan. Program ini menunjukkan cara membaca input pengguna dengan Scanner serta menghitung jumlah angka secara iteratif dalam Java.

16. Class PrintXRepeat

Output:

```
D:\Nazwa Syalaisa Haq_0063_B1\Tugas 2\Source Code>javac PrintXRepeat.java

D:\Nazwa Syalaisa Haq_0063_B1\Tugas 2\Source Code>java PrintXRepeat
Masukkan nilai x (int), akhiri dg 999 : 999
Kasus kosong

D:\Nazwa Syalaisa Haq_0063_B1\Tugas 2\Source Code>javac PrintXRepeat.java

D:\Nazwa Syalaisa Haq_0063_B1\Tugas 2\Source Code>java PrintXRepeat
Masukkan nilai x (int), akhiri dg 999 : 50
Masukkan nilai x (int), akhiri dg 999 : 50
Masukkan nilai x (int), akhiri dg 999 : 999
Hasil penjumlahan = 100

D:\Nazwa Syalaisa Haq_0063_B1\Tugas 2\Source Code>
```

Penjelasan:

Program ini membaca serangkaian angka dari pengguna dan menjumlahkannya hingga pengguna memasukkan 999 sebagai tanda berhenti. Jika angka pertama yang dimasukkan adalah 999, program akan mencetak "Kasus kosong" karena tidak ada angka yang dijumlahkan. Jika tidak, program menggunakan perulangan do-while untuk terus meminta angka dari pengguna dan menambahkannya ke variabel Sum. Perulangan ini terus berjalan sampai pengguna memasukkan 999, lalu program menampilkan hasil penjumlahan dari semua angka yang telah dimasukkan. Program ini menunjukkan cara membaca input pengguna dengan Scanner serta melakukan iterasi dengan do-while dalam Java.

17. Class PrintXWhile

Output:

```
D:\Nazwa Syalaisa Haq_0063_B1\Tugas 2\Source Code>javac PrintXWhile.java

D:\Nazwa Syalaisa Haq_0063_B1\Tugas 2\Source Code>java PrintXWhile
Masukkan nilai x (int), akhiri dg 999 : 999
Hasil penjumlahan = 0

D:\Nazwa Syalaisa Haq_0063_B1\Tugas 2\Source Code>javac PrintXWhile.java

D:\Nazwa Syalaisa Haq_0063_B1\Tugas 2\Source Code>java PrintXWhile
Masukkan nilai x (int), akhiri dg 999 : 10
Masukkan nilai x (int), akhiri dg 999 : 16
Masukkan nilai x (int), akhiri dg 999 : 999
Hasil penjumlahan = 26

D:\Nazwa Syalaisa Haq_0063_B1\Tugas 2\Source Code>
```

Penjelasan:

Program ini membaca sejumlah angka dari pengguna dan menjumlahkannya hingga pengguna memasukkan 999 sebagai tanda berhenti. Perulangan while digunakan untuk terus meminta input angka selama angka yang dimasukkan bukan 999. Setiap angka yang dimasukkan akan ditambahkan ke variabel Sum. Setelah pengguna mengetikkan 999,

program keluar dari perulangan dan menampilkan hasil penjumlahan dari semua angka yang telah dimasukkan. Program ini menunjukkan cara membaca input pengguna dengan Scanner serta menggunakan while-loop dalam Java untuk melakukan iterasi.

18. Class SubProgram

Output:

```
D:\Nazwa Syalaisa Haq_0063_B1\Tugas 2\Source Code>javac SubProgram.java

D:\Nazwa Syalaisa Haq_0063_B1\Tugas 2\Source Code>java SubProgram
Maksimum dua bilangan
Ketikkan dua bilangan, pisahkan dg RETURN :
14
31
Ke dua bilangan : a = 14 b = 31
Maksimum = 31
Tukar kedua bilangan...

Ke dua bilangan setelah tukar: a = 31 b = 14

D:\Nazwa Syalaisa Haq_0063_B1\Tugas 2\Source Code>
```

Penjelasan:

Program ini menggunakan dua fungsi untuk menentukan bilangan terbesar dan menukar dua bilangan. Fungsi `maxab()` digunakan untuk mencari nilai maksimum antara dua angka dengan membandingkan keduanya. Jika `a` lebih besar atau sama dengan `b`, maka `a` dikembalikan sebagai nilai terbesar, jika tidak, maka `b` yang dikembalikan. Sementara itu, fungsi `tukar()` bertujuan untuk menukar nilai dua bilangan menggunakan variabel sementara `temp`. Namun, karena Java menggunakan `pass-by-value`, perubahan yang terjadi dalam fungsi `tukar()` tidak mempengaruhi nilai asli variabel di dalam `main()`.

Saat program dijalankan, pengguna diminta memasukkan dua bilangan. Program kemudian mencetak bilangan terbesar menggunakan fungsi `maxab()`. Setelah itu, program mencoba menukar nilai kedua bilangan dengan memanggil fungsi `tukar()`, tetapi karena Java hanya mengirim salinan nilai variabel, perubahan yang terjadi di dalam fungsi tidak berpengaruh pada variabel asli. Akhirnya, program menunjukkan hasil tanpa adanya perubahan pada bilangan setelah pemanggilan fungsi `tukar()`.

19. Class Tempair

Output:

```
D:\Nazwa Syalaisa Haq_0063_B1\Tugas 2\Source Code>javac Tempair.java

D:\Nazwa Syalaisa Haq_0063_B1\Tugas 2\Source Code>java Tempair
Contoh IF tiga kasus
Temperatur (der. C) = -5
Wujud air beku
-5
D:\Nazwa Syalaisa Haq_0063_B1\Tugas 2\Source Code>javac Tempair.java

D:\Nazwa Syalaisa Haq_0063_B1\Tugas 2\Source Code>java Tempair
Contoh IF tiga kasus
Temperatur (der. C) = 31
Wujud air cair
31
D:\Nazwa Syalaisa Haq_0063_B1\Tugas 2\Source Code>javac Tempair.java

D:\Nazwa Syalaisa Haq_0063_B1\Tugas 2\Source Code>java Tempair
Contoh IF tiga kasus
Temperatur (der. C) = 130
Wujud air uap/gas
130
D:\Nazwa Syalaisa Haq_0063_B1\Tugas 2\Source Code>
```

Penjelasan:

Program ini menentukan wujud air berdasarkan suhu yang dimasukkan oleh pengguna. Pengguna diminta memasukkan temperatur dalam derajat Celsius, lalu program menggunakan struktur IF tiga kasus untuk menentukan apakah air dalam keadaan beku, cair, atau uap. Jika suhu kurang dari 0°C, program mencetak bahwa air dalam keadaan beku. Jika suhu berada dalam rentang 0°C hingga 100°C, program menyatakan bahwa air dalam bentuk cair. Sedangkan jika suhu lebih dari 100°C, program menampilkan bahwa air berada dalam bentuk uap/gas. Program ini sederhana tetapi efektif dalam menunjukkan penggunaan percabangan IF-ELSE IF untuk mengatasi berbagai kondisi.