

Tugas II

Makassar, 06 Maret 2025

PRAKTIKUM
PEMROGRAMAN BERORIENTASI OBJEK



Nama : Ghadiyahtul Fadiyah Ahmad
Nim : 13020230085
Dosen : Mardiyah Hasnawi, S.Kom., M.T.

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

Tugas Program PBO

Program 1

output :

```
D:\PBO\Tugas2\Source Code>javac BacaString.java
D:\PBO\Tugas2\Source Code>java BacaString.java

Baca string dan Integer:
masukkan sebuah string: fadiyah
String yang dibaca : fadiyah
D:\PBO\Tugas2\Source Code>\_
```

keterangan :

Program Java ini simpel banget dan fungsinya buat baca input dari pengguna terus nampilin lagi hasilnya. Jadi, pertama programnya dikompilasi dulu pakai `javac BacaString.java`, abis itu dijalankan dengan `java BacaString`. Pas dijalankan, programnya minta kita masukin sebuah string, misalnya "fadiyah", terus dia langsung nampilin lagi teks yang kita masukin tadi. Intinya, program ini nunjukin cara dasar buat ambil input dan cetak output di Java, yang bisa jadi dasar buat bikin program yang lebih kompleks nantinya, kayak baca angka atau ngolah input sebelum ditampilin

Program 2

output :

[illegible]

keterangan :

Program **ForEver.java** ini bikin perulangan yang jalan terus tanpa henti (infinite loop), terus-menerus nge-print "**Print satu baris**" sampai kita stop secara manual pakai **Ctrl + C (^C)**. Pas dijalankan, program ngasih tahu kalau bakal looping terus dan harus dihentikan paksa. Kemungkinan besar, program ini pakai `while(true)`, jadi nggak ada batasannya buat berhenti sendiri. Biar nggak kejadian kayak gini, lebih

baik ditambahin cara buat berhenti, misalnya baca input dari user atau pakai hitungan tertentu supaya loop-nya bisa selesai sendiri.

Program 3

output :

```
D:\PBO\Tugas2\Source Code>javac If1.java

D:\PBO\Tugas2\Source Code>java If1.java
Contoh IF satu kasus
Ketikkan suatu nilai integer : 3

Nilai a positif 3
D:\PBO\Tugas2\Source Code>
```

keterangan :

Program **If1.java** ini pakai percabangan **IF** buat ngecek apakah angka yang dimasukkan pengguna itu positif. Pas dijalankan, program minta input angka integer, lalu pengguna masukin angka **3**. Karena **3 adalah angka positif**, program langsung nge-print "**Nilai a positif 3**". Kalau angkanya negatif atau nol, kemungkinan program nggak bakal nge-print apa-apa, tergantung gimana logikanya dibuat. Ini contoh sederhana dari percabangan IF yang cuma jalan kalau syaratnya terpenuhi.

Program 4

output :

```
D:\PBO\Tugas2\Source Code>javac If2.java

D:\PBO\Tugas2\Source Code>java If2.java
Contoh IF dua kasus
Ketikkan suatu nilai integer :3
Nilai a positif 3

D:\PBO\Tugas2\Source Code>_
```

keterangan :

Program Java ini meminta pengguna memasukkan sebuah angka bulat, lalu menggunakan struktur **if** untuk mengecek apakah angka tersebut positif. Jika angka lebih dari nol, program akan menampilkan pesan bahwa angka tersebut positif, seperti yang terlihat pada output "Nilai a positif 3" ketika pengguna memasukkan angka 3. Jika angka nol atau negatif, program kemungkinan memiliki kondisi lain untuk menampilkan pesan berbeda (meskipun tidak terlihat di gambar). Program ini adalah contoh sederhana penggunaan **if** dalam Java untuk menangani dua kemungkinan kasus.

Program 5

output :

```
D:\PBO\Tugas2\Source Code>javac If3.java

D:\PBO\Tugas2\Source Code>java If3.java
Contoh IF tiga kasus
Ketikkan suatu nilai integer :2
Nilai a positif 2

D:\PBO\Tugas2\Source Code>_
```

keterangan :

Program Java ini meminta pengguna memasukkan sebuah angka bulat, lalu menggunakan struktur **if-else if-else** untuk menangani tiga kemungkinan kasus: jika angka positif, nol, atau negatif. Pada contoh ini, pengguna memasukkan angka **2**, sehingga program menampilkan "Nilai a positif 2" karena angka tersebut lebih dari nol. Jika pengguna memasukkan **0**, program kemungkinan akan menampilkan pesan bahwa angka tersebut nol, dan jika memasukkan angka negatif, program akan memberi tahu bahwa angkanya negatif. Program ini adalah contoh penggunaan **percabangan IF dengan tiga kondisi** dalam Java.

Program 6

output :

```
D:\PBO\Tugas2\Source Code>javac KasusBoolean.java

D:\PBO\Tugas2\Source Code>java KasusBoolean.java
true
benar

D:\PBO\Tugas2\Source Code>_
```

keterangan :

Program Java ini menggunakan tipe data **boolean** untuk menentukan kondisi **true** (**benar**) atau **false** (**salah**). Saat dijalankan, program mencetak "true" dan "benar", yang menunjukkan bahwa variabel boolean yang digunakan dalam program memiliki nilai **true**. Kemungkinan besar, program ini memiliki kondisi seperti boolean status = true; dan kemudian mencetak nilai dari variabel tersebut, diikuti dengan pernyataan tambahan seperti if (status) { System.out.println("benar"); }. Program ini adalah contoh sederhana penggunaan **tipe data boolean dalam Java**.

Program 7

output :

```
D:\PBO\Tugas2\Source Code>javac KasusSwitch.java

D:\PBO\Tugas2\Source Code>java KasusSwitch.java
Ketikkan sebuah huruf, akhiri dengan RETURN
a
    Yang anda ketik adalah a

D:\PBO\Tugas2\Source Code>
```

keterangan :

Program Java ini meminta pengguna untuk mengetikkan sebuah huruf, lalu menggunakan struktur **switch-case** untuk mencocokkan input yang diberikan. Ketika pengguna mengetik huruf 'a' dan menekan ENTER, program mengenali input tersebut dan menampilkan output "**Yang anda ketik adalah a**". Kemungkinan besar, program memiliki kode seperti switch(huruf), dengan beberapa **case** yang menangani berbagai kemungkinan input. Jika huruf yang diketik cocok dengan salah satu **case**, program akan menampilkan pesan sesuai dengan input tersebut. Ini adalah contoh penggunaan **switch-case dalam Java** untuk menangani berbagai pilihan berdasarkan nilai yang dimasukkan.

Program 8

output :

```
D:\PBO\Tugas2\Source Code>javac Konstant.java

D:\PBO\Tugas2\Source Code>java Konstant.java
Jari-jari lingkaran =12
Luas lingkaran = 452.37598
Akhir program

D:\PBO\Tugas2\Source Code>_
```

keterangan :

Program Java ini menghitung luas lingkaran dengan menggunakan **konstanta** untuk nilai π (pi). Dari output yang terlihat, program menetapkan **jari-jari = 12** dan menggunakan rumus luas lingkaran $\pi \times r^2$ untuk mendapatkan hasil **452.37598**. Kemungkinan besar, program menggunakan deklarasi konstanta seperti final double $PI = 3.14159$; agar nilai π tetap dan tidak berubah. Setelah menghitung luas lingkaran, program menampilkan hasilnya dan mengakhiri eksekusi dengan menampilkan pesan "Akhir program". Ini adalah contoh penggunaan **konstanta dalam Java** untuk memastikan nilai tetap dalam perhitungan matematika.

Program 9

ouput :

```
D:\PBO\Tugas2\Source Code>javac Max2.java
D:\PBO\Tugas2\Source Code>java Max2.java
Maksimum dua bilangan :
Ketikkan dua bilangan, pisahkan dg RETURN :
4
1
Ke dua bilangan : a = 4 b = 1
Nilai a yang maksimum 4
D:\PBO\Tugas2\Source Code>
```

keterangan :

Program Java ini meminta pengguna memasukkan dua bilangan, lalu menentukan bilangan mana yang lebih besar menggunakan struktur **if-else**. Dari output, pengguna memasukkan angka **4** dan **1**, kemudian program mencetak bahwa **a = 4** dan **b = 1**. Setelah itu, program mengevaluasi nilai terbesar dan menampilkan "**Nilai a yang maksimum 4**", karena **4 lebih besar dari 1**. Jika angka yang dimasukkan berbeda, program akan menyesuaikan outputnya. Ini adalah contoh sederhana dari **percabangan if-else dalam Java** untuk membandingkan dua angka dan menentukan nilai maksimum.

Program 10

output :

```
D:\PBO\Tugas2\Source Code>javac PriFor.java
D:\PBO\Tugas2\Source Code>java PriFor.java
Baca N, print 1 s/d N N = 5
1
2
3
4
5
Akhir program
D:\PBO\Tugas2\Source Code>_
```

keterangan :

Program Java ini membaca input **N** dari pengguna, lalu mencetak angka dari **1 hingga N** menggunakan perulangan **for**. Dari output, pengguna memasukkan **N = 5**, sehingga program mencetak angka **1, 2, 3, 4, 5** secara berurutan. Kemungkinan besar, program menggunakan struktur `for (int i = 1; i <= N; i++) { System.out.println(i); }` untuk mencetak angka dari 1 sampai N. Setelah mencetak semua angka, program menampilkan "**Akhir program**" dan berhenti. Ini adalah contoh sederhana dari penggunaan **perulangan for dalam Java** untuk mencetak deret angka.

Program 11

output :

```
D:\PBO\Tugas2\Source Code>javac PrintIterasi.java

D:\PBO\Tugas2\Source Code>java PrintIterasi.java
Nilai N >0 = 1
Print i dengan ITERATE :
1

D:\PBO\Tugas2\Source Code>_
```

keterangan :

Program Java ini menampilkan angka secara iteratif berdasarkan nilai **N** yang diberikan oleh pengguna. Dari output, terlihat bahwa nilai **N = 1**, sehingga program hanya mencetak angka **1**. Kemungkinan besar, program menggunakan perulangan seperti **for** atau **while** untuk mencetak angka dari 1 hingga N. Jika N lebih besar, program akan mencetak lebih banyak angka sesuai dengan iterasi yang dilakukan. Program ini merupakan contoh penggunaan **perulangan dalam Java** untuk melakukan iterasi berdasarkan kondisi yang diberikan.

Program 12

output :

```
D:\PBO\Tugas2\Source Code>javac PrintRepeat.java

D:\PBO\Tugas2\Source Code>java PrintRepeat.java
Nilai N >0 = 3
Print i dengan REPEAT:
1
2
3

D:\PBO\Tugas2\Source Code>
```

keterangan :

Program Java ini membaca nilai **N** dari pengguna dan mencetak angka dari **1 hingga N** menggunakan perulangan, kemungkinan dengan **do-while** atau **while** loop. Dari output, terlihat bahwa pengguna memasukkan **N = 3**, sehingga program mencetak angka **1, 2, 3** secara berurutan. Kata "REPEAT" dalam output menunjukkan bahwa program mungkin menggunakan perulangan **do-while**, yang mengeksekusi blok kode setidaknya sekali sebelum mengevaluasi kondisi. Jika N lebih besar, program akan mencetak lebih banyak angka sesuai dengan jumlah iterasi yang dilakukan.

Program 13

output :

```
D:\PBO\Tugas2\Source Code>javac PrintWhile.java

D:\PBO\Tugas2\Source Code>java PrintWhile.java
Nilai N >0 = 4
Print i dengan WHILE:
1
2
3
4

D:\PBO\Tugas2\Source Code>
```

keterangan :

Program Java ini membaca nilai **N** dari pengguna dan mencetak angka dari **1 hingga N** menggunakan perulangan **while**. Pada output ini, pengguna memasukkan **N = 4**, sehingga program mencetak angka **1, 2, 3, 4** secara berurutan. Perulangan **while** akan terus mengeksekusi blok kode selama kondisi (misalnya $i \leq N$) masih bernilai **true**. Jika **N** lebih besar, program akan mencetak lebih banyak angka sesuai dengan jumlah iterasi.

Program 14

output :

```
D:\PBO\Tugas2\Source Code>javac PrintWhile1.java

D:\PBO\Tugas2\Source Code>java PrintWhile1.java
Nilai N >0 = 2
Print i dengan WHILE (ringkas):
1
2

D:\PBO\Tugas2\Source Code>
```

keterangan :

Program **PrintWhile1.java** membaca input **N** dari pengguna dan mencetak angka dari **1 hingga N** menggunakan perulangan **while** secara lebih ringkas dibanding versi sebelumnya (**PrintWhile.java**), seperti terlihat pada hasil saat pengguna memasukkan **N = 2**, sehingga hanya mencetak **1 dan 2**.

Program 15

output :

```
D:\PBO\Tugas2\Source Code>javac PrintXinterasi.java

D:\PBO\Tugas2\Source Code>java PrintXinterasi.java
Masukkan nilai x (int), akhiri dg 999 : 59
Masukkan nilai x (int), akhiri dg 999 : 12
Masukkan nilai x (int), akhiri dg 999 : 999
Hasil penjumlahan = 71

D:\PBO\Tugas2\Source Code>_
```

keterangan :

Program **PrintXinterasi.java** menerima sejumlah bilangan bulat dari pengguna dan menjumlahkannya hingga pengguna memasukkan nilai **999** sebagai tanda berhenti. Pada contoh ini, pengguna memasukkan **59** dan **12**, kemudian memasukkan **999** untuk mengakhiri input, sehingga program menampilkan hasil penjumlahan **71**.

Program 16

output :

```
D:\PBO\Tugas2\Source Code>javac PrintXRepeat.java

D:\PBO\Tugas2\Source Code>java PrintXRepeat.java
Masukkan nilai x (int), akhiri dg 999 : 70
Masukkan nilai x (int), akhiri dg 999 : 23
Masukkan nilai x (int), akhiri dg 999 : 999
Hasil penjumlahan = 93

D:\PBO\Tugas2\Source Code>
```

keterangan :

Program **PrintXRepeat.java** menerima input bilangan bulat dari pengguna dan menjumlahkannya hingga pengguna memasukkan nilai **999** sebagai tanda berhenti. Pada contoh ini, pengguna memasukkan **70** dan **23**, lalu memasukkan **999** untuk mengakhiri input, sehingga program menampilkan hasil penjumlahan **93**.

Program 17

output :

```
D:\PBO\Tugas2\Source Code>javac PrintXWhile.java

D:\PBO\Tugas2\Source Code>java PrintXWhile.java
Masukkan nilai x (int), akhiri dg 999 : 33
Masukkan nilai x (int), akhiri dg 999 : 57
Masukkan nilai x (int), akhiri dg 999 : 999
Hasil penjumlahan = 90

D:\PBO\Tugas2\Source Code>
```

keterangan :

Program **PrintXWhile.java** menggunakan perulangan **while** untuk menerima input bilangan bulat dari pengguna dan menjumlahkannya hingga pengguna memasukkan nilai **999** sebagai tanda berhenti. Dalam contoh ini, pengguna memasukkan **33** dan **57**, kemudian memasukkan **999** untuk mengakhiri input, sehingga program menampilkan hasil penjumlahan **90**.

Program 18

output :

```
D:\PBO\Tugas2\Source Code>javac SubProgram.java

D:\PBO\Tugas2\Source Code>java SubProgram.java
Maksimum dua bilangan
Ketikkan dua bilangan, pisahkan dg RETURN :
2
6
Ke dua bilangan : a = 2 b = 6
Maksimum = 6
Tukar kedua bilangan...
Ke dua bilangan setelah tukar: a = 6 b = 2

D:\PBO\Tugas2\Source Code>
```

keterangan :

Program **SubProgram.java** menerima dua bilangan dari pengguna dan menampilkan bilangan terbesar di antara keduanya. Dalam contoh ini, pengguna memasukkan angka **2** dan **6**, lalu program menentukan bahwa angka maksimum adalah **6**. Selanjutnya, program menukar nilai kedua bilangan sehingga nilai **a** menjadi **6** dan nilai **b** menjadi **2**, kemudian menampilkan hasil setelah pertukaran.

Program 19

output :

```
D:\PBO\Tugas2\Source Code>javac Tempair.java

D:\PBO\Tugas2\Source Code>java Tempair.java
Contoh IF tiga kasus
Temperatur (der. C) = -3
Wujud air beku
-3
D:\PBO\Tugas2\Source Code>java Tempair.java
Contoh IF tiga kasus
Temperatur (der. C) = 40
Wujud air cair
40
D:\PBO\Tugas2\Source Code>java Tempair.java
Contoh IF tiga kasus
Temperatur (der. C) = 150
Wujud air uap/gas
150
D:\PBO\Tugas2\Source Code>
```

keterangan :

Program **Tempair.java** menentukan wujud air berdasarkan suhu yang dimasukkan pengguna. Jika suhu di bawah **0°C**, air berwujud **beku/es**; jika suhu antara **0°C hingga kurang dari 100°C**, air berwujud **cair**; dan jika suhu **100°C atau lebih**, air berwujud **uap/gas**. Dalam hasil eksekusi, program diuji dengan tiga nilai suhu: **-3°C** (hasilnya air beku), **40°C** (hasilnya air cair), dan **150°C** (hasilnya air uap/gas).