

TUGAS
PEMROGRAMAN BERORIENTASI OBJEK



Nama : Nayla Ananda
Stambuk : 13020230112
Kelas : B2
Mata Kuliah : Pemrograman Berorientasi objek
Dosen : Mardiyah Hasnawi, S.Kom., M.T.,MTA

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

Gambar di bawah merupakan langkah-langkah untuk masuk dan mengakses folder yang ingin kita akses yakni folder tugas PBO lalu folder Tugas 2 lalu masuk ke folder Kode Program dengan menggunakan perintah CD. Setelah itu mengetikkan perintah DIR untuk melihat file apa saja yang ada di dalam folder Tugas 1 seperti pada gambar tersebut.

```
C:\>cd Tugas PBO
```

```
C:\Tugas PBO> CD Tugas 2
```

```
C:\Tugas PBO\Tugas 2>CD Kode Program
```

```
C:\Tugas PBO\Tugas 2\Kode Program>
```

1. Program pertama (BacaString)

Output Program pertama :

```
C:\Tugas PBO\Tugas 2\Kode Program>javac BacaString.java
```

```
C:\Tugas PBO\Tugas 2\Kode Program>java BacaString
```

```
Baca string dan Integer:
masukkan sebuah string: Nayla
String yang dibaca : Nayla
```

Program pertama ini meminta kita untuk memasukkan sebuah string, lalu mencetak kembali string tersebut. Ketika program dijalankan, akan muncul pesan "**masukkan sebuah string:**", dan kita diminta mengetik sebuah teks. Setelah kita menekan enter, program akan membaca string menggunakan `BufferedReader`, kemudian mencetak hasilnya dengan format "**String yang dibaca: <teks_input>**". Misalnya, saya mengetik "Nayla", outputnya adalah "**String yang dibaca : Nayla**". Program ini hanya menangani satu input string tanpa validasi lebih lanjut.

2. Program kedua (ForEver)

Output Program Kedua :

```
C:\Tugas PBO\Tugas 2\Kode Program>javac ForEver.java
```

```
C:\Tugas PBO\Tugas 2\Kode Program>java ForEver  
Program akan looping, akhiri dengan ^c  
Print satu baris ....
```

Program Kedua Program ini menggunakan perulangan while (true), yang berarti program akan terus berjalan tanpa henti kecuali dihentikan secara paksa. Saat dijalankan, program pertama-tama mencetak pesan "**Program akan looping, akhiri dengan ^c**". Setelah itu, program akan terus mencetak "Print satu baris" secara berulang-ulang. Karena tidak ada kondisi berhenti dalam perulangan ini, program akan terus mencetak output hingga dihentikan secara manual. Program ini bisa digunakan untuk simulasi perulangan tanpa batas atau menunggu suatu kondisi eksternal sebelum berhenti.

3. Program Ketiga (If1)

Output Kode Ketiga :

```
C:\Tugas PBO\Tugas 2\Kode Program>javac If1.java
```

```
C:\Tugas PBO\Tugas 2\Kode Program>java If1  
Contoh IF satu kasus  
Ketikkan suatu nilai integer : 7
```

```
Nilai a positif 7
```

Program Ketiga Program ini membaca sebuah bilangan bulat dari pengguna dan hanya mencetaknya jika nilainya nol atau lebih besar (positif). Saat dijalankan, program meminta input dengan teks "**Ketikkan suatu nilai integer:**". Jika pengguna memasukkan angka positif, misalnya saya memasukkan nilai 7, program akan mencetak "**Nilai a positif 7**". Namun, jika saya mengetikkan angka negatif seperti -3, program tidak akan mencetak apa pun karena tidak ada kondisi untuk menangani angka negatif. Oleh karena itu, program ini hanya memberikan output untuk angka nol dan positif.

4. Program Keempat (If2)

Output Program Keempat :

```
C:\Tugas PBO\Tugas 2\Kode Program>javac If2.java
```

```
C:\Tugas PBO\Tugas 2\Kode Program>java If2
```

Contoh IF dua kasus

Ketika suatu nilai integer :3

Nilai a positif 3

Program Keempat Program ini merupakan pengembangan dari **If1** yang menangani dua kondisi berbeda: angka positif dan negatif. Saat pengguna memasukkan angka nol atau lebih besar, program mencetak "**Nilai a positif <angka>**", dan jika angka negatif, program mencetak "**Nilai a negatif <angka>**". Misalnya, saya mengetikkan angka 3, outputnya "**Nilai a positif 3**", dan jika inputnya -4, outputnya "**Nilai a negatif -4**". Dengan adanya kondisi else, program ini lebih lengkap dibandingkan **If1**.

5. Program Kelima (If3)

Output Program Kelima :

```
C:\Tugas PBO\Tugas 2\Kode Program>javac If3.java
```

```
C:\Tugas PBO\Tugas 2\Kode Program>java If3
```

Contoh IF tiga kasus

Ketikkan suatu nilai integer :4

Nilai a positif 4

Pada **Program Kelima**, Program ini lebih lengkap karena mencakup tiga kemungkinan: angka positif, nol, dan negatif. Jika saya memasukkan angka lebih besar dari nol, program mencetak "**Nilai a positif <angka>**", jika angka nol, outputnya "**Nilai Nol 0**", dan jika angka negatif, outputnya "**Nilai a negatif <angka>**". Misalnya, disini saya mengetikkan angka 4, maka outputnya adalah "**Nilai a positif 4**", sedangkan jika inputnya 0, outputnya "**Nilai Nol 0**".

6. Program Keenam (KasusBoolean)

Output Program Keenam :

```
C:\Tugas PBO\Tugas 2\Kode Program>javac KasusBoolean.java
```

```
C:\Tugas PBO\Tugas 2\Kode Program>java KasusBoolean
```

true

benar

Program Keenam Program ini mendemonstrasikan penggunaan variabel boolean. Pada awalnya, bool diatur menjadi true, sehingga program langsung mencetak "**true**". Kemudian, program memeriksa !bool (negasi dari true yang berarti false), sehingga bagian else akan dieksekusi, mencetak "**benar**". Dengan kata lain, program ini hanya mencetak "true" diikuti dengan "benar" setiap kali dijalankan.

7. Program Ketujuh (KasusSwitch)

Output Program Ketujuh :

```
C:\Tugas PBO\Tugas 2\Kode Program>javac KasusSwitch.java
```

```
C:\Tugas PBO\Tugas 2\Kode Program>java KasusSwitch
```

```
Ketikkan sebuah huruf, akhiri dengan RETURN
```

```
e
```

```
Yang anda ketik adalah e
```

Program Ketujuh, Program ini meminta kita memasukkan satu karakter. Jika karakter tersebut adalah huruf vokal (a, u, e, i, o), maka program mencetak "**Yang anda ketik adalah <huruf>**". Disini saya mengetik "e", outputnya adalah "**Yang anda ketik adalah e**". Jika pengguna memasukkan huruf lain seperti "z", program mencetak "**Yang anda ketik adalah huruf mati**".

8. Program Kedelapan (Konstant)

Output Program Kedelapan :

```
C:\Tugas PBO\Tugas 2\Kode Program>javac Konstant.java
```

```
C:\Tugas PBO\Tugas 2\Kode Program>java Konstant
```

```
Jari-jari lingkaran = 10
```

```
Luas lingkaran = 314.15002
```

```
Akhir program
```

Program Kedelapan Program **Konstant** menghitung luas lingkaran berdasarkan jari-jari yang dimasukkan pengguna. Nilai π (**pi**) diwakili oleh konstanta PHI = 3.1415. Setelah pengguna memasukkan jari-jari, misalnya 5, program menghitung luas dengan rumus $\text{PHI} * r * r$, menghasilkan **78.5375**. Hasil ini ditampilkan dalam format "**Luas lingkaran = 78.5375**", lalu program mencetak "**Akhir program**" sebagai tanda eksekusi selesai. Disini saya sebagai pengguna memasukkan 10, luas yang ditampilkan adalah **314.15002**. Program ini membantu menghitung luas lingkaran dengan cepat tanpa harus memasukkan nilai π secara manual.

9. Program Kesembilan (Max2)

Output Program Kesembilan :

```
C:\Tugas PBO\Tugas 2\Kode Program>javac Max2.java
```

```
C:\Tugas PBO\Tugas 2\Kode Program>java Max2
```

```
Maksimum dua bilangan:
```

```
Ketikkan dua bilangan, pisahkan dg RETURN:
```

```
9 1
```

```
Ke dua bilangan: a = 9 b = 1
```

```
Nilai a yang maksimum: 9
```

Program Kesembilan Program **Max2** menentukan bilangan terbesar dari dua angka yang dimasukkan pengguna. Setelah pengguna mengetikkan dua bilangan, misalnya **8** dan **3**, program mencetak "**Ke dua bilangan : a = 8 b = 3**", lalu membandingkan keduanya menggunakan if. Karena **8 lebih besar dari 3**, program mencetak "**Nilai a yang maksimum 8**". Jika pengguna memasukkan **4 dan 10**, program mencetak "**Nilai b yang maksimum: 10**". Jika kedua angka sama, misalnya **6 dan 6**, program tetap mencetak "**Nilai a yang maksimum 6**". Disini saya sebagai pengguna memasukkan nilai "**a = 9 dan b = 1**", maka program akan membandingkan nilai mana yang maksimal, disini nilai a menjadi nilai maksimum maka dari itu program akan menampilkan "**nilai a yang maksimum : 9**" Program ini berguna untuk menemukan nilai terbesar dari dua angka tanpa menggunakan fungsi Math.max().

10. Program Kesepuluh (PriFor)

Output Program Kesepuluh :

```
C:\Tugas PBO\Tugas 2\Kode Program>javac PriFor.java
```

```
C:\Tugas PBO\Tugas 2\Kode Program>java PriFor
```

```
Baca N, print 1 s/d N N = 7
```

```
1
```

```
2
```

```
3
```

```
4
```

```
5
```

```
6
```

```
7
```

```
Akhir program
```

Program Kesepuluh, Program ini membaca nilai N dari pengguna dan mencetak angka dari 1 hingga N menggunakan perulangan for. Ketika program dijalankan, pengguna diminta memasukkan angka N dengan pesan "**Baca N, print 1 s/d N**", lalu mengetikkan angka. Disini saya memasukkan angka 7, maka program akan menjalankan perulangan for ($i = 1; i \leq N; i++$), yang berarti mencetak angka **1, 2, 3, 4, 5, 6, 7** secara berurutan. Setelah perulangan selesai, program mencetak "**Akhir program**" untuk menandakan bahwa eksekusi telah selesai. Program ini berguna untuk menampilkan daftar angka dari 1 hingga N dengan metode iterasi for, yang mudah dipahami dan efisien dalam pengulangan yang memiliki batas yang jelas.

11. Program Kesebelas (PrintIterasi)

Output Program Kesebelas :

```
C:\Tugas PB0\Tugas 2\Kode Program>javac PrintIterasi.java

C:\Tugas PB0\Tugas 2\Kode Program>java PrintIterasi
Nilai N >0 = 5
Print i dengan ITERATE :
1
2
3
4
5
```

Program Kesebelas, Program ini juga mencetak angka dari 1 hingga N, tetapi menggunakan perulangan for(;;), yang merupakan bentuk perulangan tanpa batas (infinite loop). Namun, program ini menggunakan pernyataan break untuk keluar dari perulangan ketika $i == N$. Pada awalnya, program meminta pengguna memasukkan nilai N, lalu mencetak angka dari 1 hingga N satu per satu. Disini saya memasukkan angka 5, maka program akan mencetak angka **1, 2, 3, 4, 5**, lalu berhenti setelah mencapai nilai N. Program ini menunjukkan bagaimana perulangan tak terbatas dapat dikontrol menggunakan kondisi break.

12. Program Keduabelas (PrintRepeat)

Output Program Keduabelas :

```
C:\Tugas PBO\Tugas 2\Kode Program>javac PrintRepeat.java

C:\Tugas PBO\Tugas 2\Kode Program>java PrintRepeat
Nilai N >0 = 9
Print i dengan REPEAT:
1
2
3
4
5
6
7
8
9
```

Program Keduabelas Program ini serupa dengan **PrintIterasi**, tetapi menggunakan perulangan do-while, yang menjamin bahwa perintah dalam blok do akan dieksekusi setidaknya satu kali. Program ini akan terus mencetak angka dari 1 hingga N, kemudian berhenti ketika nilai i melebihi N. Jika saya memasukkan angka 9, maka program akan mencetak **1, 2, 3, 4, 5, 6, 7, 8, 9** dalam satu baris. Keunggulan dari metode do-while adalah eksekusi pertama pasti terjadi, meskipun N bernilai negatif atau nol.

13. Program Ketigabelas (PrintWhile)

Output Program Ketigabelas :

```
C:\Tugas PBO\Tugas 2\Kode Program>javac PrintWhile.java

C:\Tugas PBO\Tugas 2\Kode Program>java PrintWhile
Nilai N >0 = 3
Print i dengan WHILE:
1
2
3
```

Program Ketigabelas, Program ini mencetak angka dari 1 hingga N menggunakan perulangan while, di mana kondisi while ($i \leq N$) akan dievaluasi sebelum menjalankan perintah di dalamnya. Program meminta pengguna memasukkan N, lalu mencetak angka satu per satu hingga mencapai batas yang telah ditentukan. Jika pengguna memasukkan 3, program akan mencetak angka **1, 2, 3** dan kemudian berhenti. Program ini mirip dengan **PriFor**, tetapi menggunakan while, yang lebih fleksibel untuk kasus ketika jumlah iterasi tidak diketahui sebelumnya.

14. Program Keempatbelas (PrintWhile1)

Output Program Keempatbelas :

```
C:\Tugas PBO\Tugas 2\Kode Program>javac PrintWhile1.java

C:\Tugas PBO\Tugas 2\Kode Program>java PrintWhile1
Nilai N >0 = 2
Print i dengan WHILE (ringkas):
1
2
```

Program Keempatbelas Program ini adalah versi lebih ringkas dari **PrintWhile**. Perbedaannya adalah pada cara mencetak nilai i, yaitu dengan langsung menulis `System.out.println(i++)`, yang secara otomatis meningkatkan nilai i setelah mencetak angka. Fungsionalitasnya tetap sama, yaitu mencetak angka dari 1 hingga N menggunakan perulangan while. Saya selaku pengguna disini memasukkan angka 2 output yang dihasilkan adalah **1, 2** mirip dengan program sebelumnya, tetapi ditulis dengan lebih ringkas.

15. Program Kelimabelas (PrintXIterasi)

Output Program kelimabelas :

```
C:\Tugas PBO\Tugas 2\Kode Program>javac PrintXinterasi.java

C:\Tugas PBO\Tugas 2\Kode Program>java PrintXinterasi
Masukkan nilai x (int), akhiri dg 999: 8
Masukkan nilai x (int), akhiri dg 999: 7
Masukkan nilai x (int), akhiri dg 999: 999
Hasil penjumlahan = 15
```

Program Kelimabelas, Program ini membaca serangkaian angka dari pengguna dan menjumlahkannya hingga pengguna memasukkan 999, yang berfungsi sebagai tanda berhenti (sentinel). Saya di sini memasukkan angka 8 dan 7, maka program akan menjumlahkan angka-angka tersebut dan menampilkan hasilnya. Jika 999 dimasukkan sebagai angka pertama, program mencetak "**Kasus kosong**", yang berarti tidak ada angka yang dihitung. Jika pengguna memasukkan 8, 7, 999, maka outputnya adalah "**Hasil penjumlahan = 15**".

16.Program Keenambelas (PrintXRepeat)

Output Program Keenambelas :

```
C:\Tugas PBO\Tugas 2\Kode Program>javac PrintXRepeat.java
```

```
C:\Tugas PBO\Tugas 2\Kode Program>java PrintXRepeat
Masukkan nilai x (int), akhiri dg 999 : 12
Masukkan nilai x (int), akhiri dg 999 : 89
Masukkan nilai x (int), akhiri dg 999 : 999
Hasil penjumlahan = 101
```

Program Keenambelas Program ini sama dengan **PrintXIterasi**, tetapi menggunakan do-while. Ini berarti program akan selalu meminta input setidaknya sekali sebelum memeriksa apakah angka yang dimasukkan adalah 999. Disini saya memasukkan angka seperti **12, 89**, hasil akhirnya adalah "**Hasil penjumlahan = 101**", sama seperti sebelumnya.

17.Program Ketujuhbelas (PrintXWhile)

Output Program Ketujuhbelas :

```
C:\Tugas PBO\Tugas 2\Kode Program>javac PrintXWhile.java
```

```
C:\Tugas PBO\Tugas 2\Kode Program>java PrintXWhile
Masukkan nilai x (int), akhiri dg 999 : 33
Masukkan nilai x (int), akhiri dg 999 : 77
Masukkan nilai x (int), akhiri dg 999 : 999
Hasil penjumlahan = 110
```

Program Ketujuhbelas Program ini menggunakan perulangan while, tetapi fungsinya tetap sama seperti **PrintXIterasi** dan **PrintXRepeat**. Pengguna diminta memasukkan angka satu per satu hingga memasukkan 999, lalu program menjumlahkan semua angka yang diberikan sebelum 999. Disini saya memasukkan **33, 77, 999**, outputnya adalah "**Hasil penjumlahan = 110**".

18. Program Kedelapanbelas (SubProgram)

Output Program Kedelapanbelas :

```
C:\Tugas PBO\Tugas 2\Kode Program>javac SubProgram.java

C:\Tugas PBO\Tugas 2\Kode Program>java SubProgram
Maksimum dua bilangan
Ketikkan dua bilangan, pisahkan dg RETURN:
7 10
Ke dua bilangan : a = 7 b = 10
Maksimum = 10
Tukar kedua bilangan...
Ke dua bilangan setelah tukar: a = 10 b = 7
```

Program Kedelapanbelas Program ini memiliki dua fungsi: **maxab** untuk mencari bilangan terbesar dari dua angka, dan **tukar** untuk menukar nilai dua angka. Ketika program dijalankan, saya memasukkan dua angka, lalu program mencetak angka terbesar menggunakan maxab. Setelah itu, program memanggil fungsi tukar, yang mencoba menukar dua bilangan, tetapi karena Java menggunakan **passing by value**, nilai asli di dalam main() tidak berubah. disini saya memasukkan angka 7 dan 10 maka program akan mencetak **a= 7, b=10**. Lalu program akan menentukan nilai maksimumnya itu yang mana yakni nilai 10. Lalu setelah itu menjalankan perintah tukar kedua bilangan yang akhirnya menghasilkan **a= 10 dan b=7**.

19. Program Kesembilanbelas (Tempair)

Output Program Kesembilanbelas :

```
C:\Tugas PBO\Tugas 2\Kode Program>javac Tempair.java

C:\Tugas PBO\Tugas 2\Kode Program>java Tempair
Contoh IF tiga kasus
Temperatur (der. C) = 70
Wujud air cair pada suhu 70°C
```

Program Kesembilanbelas Program ini membaca suhu dalam derajat Celcius (T) dan menentukan wujud air. Jika $T < 0$, program mencetak "**Wujud air beku**", jika $0 \leq T \leq 100$, program mencetak "**Wujud air cair**", dan jika $T > 100$, program mencetak "**Wujud air uap/gas**". Disini saya memasukkan angka 70 sebagai temperatur, maka output yang di hasilan adalah program akan mencetak "**wujud air cair pada suhu 70 derajat C**".