

TUGAS 1

PEMROGRAMAN BERORIENTASI OBJEK

Sri Ulfiani Rifai

13020230350

B1/Teknik Informatika

```
Command Prompt
Microsoft Windows [Version 10.0.18363.1139]
(c) 2019 Microsoft Corporation. All rights reserved.

C:\Users\Fadli>java -version
java version "23.0.2" 2025-01-21
Java(TM) SE Runtime Environment (build 23.0.2+7-58)
Java HotSpot(TM) 64-Bit Server VM (build 23.0.2+7-58, mixed mode, sharing)

C:\Users\Fadli>javac -version
javac 23.0.2

C:\Users\Fadli>D:

D:\>cd PBO

D:\PBO>cd Tugas1
```

Pertama masuk ke Command Prompt, lalu masuk ke **java -version** untuk menampilkan versi java yang terinstall di sistem. Lalu, ditampilkan outputnya menunjukkan laptop saya menggunakan **Java 23.0.2** yang rilis pada 21 Januari 2025. Selanjutnya, masuk ke **javac -version** untuk menampilkan versi dari **Java Compiler (javac)** yang digunakan untuk mengompilasi kode Java. Masuk ke **D:**, artinya berpindah ke drive **D:** dari drive **C:** jika folder yang kamu buat tersimpan di drive **D:**. Kemudian masuk ke **cd PBO**, untuk masuk ke folder **PBO** di drive **D:**. Terakhir, masuk ke **cd Tugas1** untuk masuk ke folder tersebut yang berada dalam folder **PBO**.

1. Class Asgdll

```
Command Prompt

D:\PBO\Tugas1>dir
Volume in drive D is NAVAL09
Volume Serial Number is 928E-D3C9

Directory of D:\PBO\Tugas1

26/02/2025  08:57    <DIR>          .
26/02/2025  08:57    <DIR>          ..
26/02/2025  08:57                252 Asgdll.java
               1 File(s)                252 bytes
               2 Dir(s)  14.085.668.864 bytes free

D:\PBO\Tugas1>javac Asgdll.java

D:\PBO\Tugas1>java Asgdll.java
f : 20.0
f11: 10.0

D:\PBO\Tugas1>dir
Volume in drive D is NAVAL09
Volume Serial Number is 928E-D3C9

Directory of D:\PBO\Tugas1

26/02/2025  08:58    <DIR>          .
26/02/2025  08:58    <DIR>          ..
26/02/2025  08:58                876 Asgdll.class
26/02/2025  08:57                252 Asgdll.java
               2 File(s)                1.128 bytes
               2 Dir(s)  14.085.664.768 bytes free
```

Perintah **dir** pertama menunjukkan bahwa folder hanya berisi file **Asgdll.java**. Setelah dikompilasi dengan **javac Asgdll.java**, terbentuk file **Asgdll.class**. Program kemudian dijalankan dengan **java Asgdll.java**, menghasilkan output tertentu. Perintah **dir** terakhir menunjukkan tambahan file **Asgdll.class**, menandakan bahwa kompilasi berhasil.

Dari output ini, terdapat dua nilai yang dicetak ke layar. Nilai **f : 20.0** dan **f11: 10.0** yang berasal dari dua variabel atau metode dalam kode program yang berisi perhitungan atau inisialisasi nilai tertentu. Output menunjukkan bahwa program telah berjalan sesuai dengan logika yang dituliskan dalam **Asgdll.java**.

2. Class Assign

```
Command Prompt

D:\PBO\Tugas1>dir
Volume in drive D is NAVAL09
Volume Serial Number is 928E-D3C9

Directory of D:\PBO\Tugas1

26/02/2025  08:59    <DIR>        .
26/02/2025  08:59    <DIR>        ..
26/02/2025  08:58             876 Asgdll.class
26/02/2025  08:57             252 Asgdll.java
26/02/2025  08:59             247 Assign.java
                3 File(s)          1.375 bytes
                2 Dir(s)      14.085.537.792 bytes free

D:\PBO\Tugas1>javac Assign.java

D:\PBO\Tugas1>java Assign.java
hello
Ini nilai i :5

D:\PBO\Tugas1>dir
Volume in drive D is NAVAL09
Volume Serial Number is 928E-D3C9

Directory of D:\PBO\Tugas1

26/02/2025  08:59    <DIR>        .
26/02/2025  08:59    <DIR>        ..
26/02/2025  08:58             876 Asgdll.class
26/02/2025  08:57             252 Asgdll.java
26/02/2025  08:59             889 Assign.class
26/02/2025  08:59             247 Assign.java
                4 File(s)          2.264 bytes
                2 Dir(s)      14.085.533.696 bytes free
```

Perintah `dir` pertama menampilkan isi folder `D:\PBO\Tugas1`, yang awalnya berisi tiga file: `Asgdll.class`, `Asgdll.java`, dan `Assign.java`. Selanjutnya, perintah **`javac Assign.java`** digunakan untuk mengompilasi file **`Assign.java`**, menghasilkan file **`Assign.class`**. Program kemudian dijalankan dengan perintah **`java Assign.java`**, yang mengeksekusi kode di dalamnya. Setelah eksekusi selesai, perintah `dir` terakhir menunjukkan tambahan file **`Assign.class`**.

Output menunjukkan bahwa program menampilkan kata "hello", diikuti dengan teks yang mencetak nilai variabel `i`, yang bernilai 5. Hal ini menunjukkan bahwa program memiliki perintah untuk mencetak string dan variabel integer ke layar.

3. Class ASIGNi

```
Command Prompt

D:\PBO\Tugas1>javac ASIGNi.java

D:\PBO\Tugas1>java ASIGNi.java
Karakter = A
Karakter = Z
Karakter = A
Karakter = Z
Bilangan integer (short) = 1
          (int) = 1
          (long)= 10000
Bilangan Real x = 50.20000076293945
Bilangan Real y = 50.2

D:\PBO\Tugas1>dir
Volume in drive D is NAVAL09
Volume Serial Number is 928E-D3C9

Directory of D:\PBO\Tugas1

26/02/2025  09:00    <DIR>        .
26/02/2025  09:00    <DIR>        ..
26/02/2025  08:58             876 Asgdll.class
26/02/2025  08:57             252 Asgdll.java
26/02/2025  08:59             889 Assign.class
26/02/2025  08:59             247 Assign.java
26/02/2025  09:00             1.371 ASIGNi.class
26/02/2025  09:00             1.021 ASIGNi.java
                6 File(s)          4.656 bytes
                2 Dir(s)      14.085.459.968 bytes free
```

Perintah **`javac ASIGNi.java`** digunakan untuk mengompilasi file **`ASIGNi.java`**, kemudiang menghasilkan file **`ASIGNi.class`**. Setelah itu, program dijalankan dengan perintah **`java ASIGNi.java`**, yang mengeksekusi kode dalam file tersebut. Setelah

eksekusi selesai, perintah `dir` dijalankan kembali untuk melihat perubahan dalam folder.

Pada output ini menunjukkan bahwa program mencetak beberapa nilai karakter (A dan Z) serta berbagai tipe bilangan integer (short, int, dan long) dengan nilai tertentu. Selain itu, terdapat dua bilangan real (x dan y) dengan nilai yang hampir sama, tetapi x memiliki lebih banyak digit desimal karena kemungkinan menggunakan tipe data float atau double. Hal ini menunjukkan bahwa program bertujuan untuk mendemonstrasikan berbagai tipe data di Java.

4. Class BacaData

```
D:\PBO\Tugas1>javac BacaData.java
D:\PBO\Tugas1>java BacaData.java
Contoh membaca dan menulis, ketik nilai integer:
5
Nilai yang dibaca : 5
D:\PBO\Tugas1>dir
Volume in drive D is NAVAL09
Volume Serial Number is 928E-D3C9

Directory of D:\PBO\Tugas1

26/02/2025  09:04    <DIR>          .
26/02/2025  09:04    <DIR>          ..
26/02/2025  08:58             876 Asgdl1.class
26/02/2025  08:57             252 Asgdl1.java
26/02/2025  08:59             889 Asign.class
26/02/2025  08:59             247 Asign.java
26/02/2025  09:00             1.371 ASIGNi.class
26/02/2025  09:00             1.021 ASIGNi.java
26/02/2025  09:04             1.073 BacaData.class
26/02/2025  09:04             504 BacaData.java
                8 File(s)              6,233 bytes
                2 Dir(s)          14,085,324,800 bytes free
```

Perintah **javac BacaData.java** digunakan untuk mengompilasi file **BacaData.java**, menghasilkan file **BacaData.class**. Setelah itu, program dijalankan dengan perintah **java BacaData.java**, yang memungkinkan pengguna untuk memasukkan nilai saat eksekusi. Setelah program selesai dijalankan, perintah `dir` menunjukkan bahwa file **BacaData.class** telah ditambahkan.

Saat program **BacaData.java** dijalankan, program pertama-tama menampilkan pesan yang meminta pengguna untuk memasukkan sebuah bilangan integer. Setelah pengguna mengetik angka dan menekan *Enter*, program akan membaca nilai yang dimasukkan dan menampilkannya kembali di layar. Pada gambar, pengguna memasukkan angka 5, sehingga program mencetak "Nilai yang dibaca : 5". Hal ini menunjukkan bahwa program menggunakan mekanisme input-output di Java, di mana input dari pengguna diambil dan langsung ditampilkan kembali sebagai output.

5. Class Bacakar

```
Command Prompt
D:\PBO\Tugas1>javac Bacakar.java
D:\PBO\Tugas1>java Bacakar.java
hello
baca 1 karakter : S
baca 1 bilangan : 4
5
4
bye
```

Saat program **Bacakar.java** dijalankan, pertama-tama program mencetak kata "hello" sebagai sapaan awal. Kemudian, program meminta pengguna untuk memasukkan satu karakter dengan menampilkan pesan "baca 1 karakter :". Pengguna memasukkan karakter S, yang kemudian dibaca oleh program. Selanjutnya, program meminta

pengguna untuk memasukkan sebuah bilangan dengan pesan "baca 1 bilangan :", lalu pengguna memasukkan angka 4. Setelah itu, program mencetak kembali karakter yang telah dibaca (S) dan angka yang dimasukkan (4). Terakhir, program menampilkan pesan "bye" sebagai tanda bahwa eksekusi telah selesai. Output ini menunjukkan bahwa program menggunakan mekanisme input-output untuk membaca dan menampilkan data yang dimasukkan oleh pengguna.

6. Class Casting1

```
D:\PBO\Tugas1>javac Casting1.java
D:\PBO\Tugas1>java Casting1.java
5.0
6.0
2
3.200000047683716
53
53.0
53.0
3
3.14
```

Saat **Casting1.java** dijalankan, output menunjukkan hasil konversi tipe data di Java. Nilai 5.0 dan 6.0 berasal dari konversi integer ke double. Angka 2 dan 3.200000047683716 menunjukkan perubahan tipe dari double atau float. Nilai 53 kemungkinan berasal dari karakter ASCII. Angka 53.0 menunjukkan konversi integer ke double. Nilai 3 berasal dari pembulatan ke integer, dan 3.14 tetap dalam format desimal. Output ini menggambarkan bagaimana Java menangani *type casting* atau konversi tipe data secara otomatis dengan perintah eksplisit dalam program.

7. Class Casting2

```
D:\PBO\Tugas1>javac Casting2.java
D:\PBO\Tugas1>java Casting2.java
a : 67
k : 45.0
d : 100.0
n : 9
m : 5
l : 3.2
k : 67.0
c : 9.0
l : 3.2
```

Saat **Casting2.java** dijalankan, output menunjukkan hasil konversi berbagai tipe data di Java. Nilai a : 67 kemungkinan berasal dari karakter ASCII yang dikonversi ke integer. Nilai k : 45.0 dan d : 100.0 menunjukkan konversi angka ke tipe double. Nilai n : 9 dan m : 5 berasal dari pembulatan atau pemotongan bilangan saat dikonversi ke integer. Nilai l : 3.2 tetap dalam format desimal, menunjukkan tipe float atau double. Nilai k : 67.0 kemungkinan hasil konversi integer ke double, sedangkan c : 9.0 menunjukkan angka yang dikonversi ke double. Secara keseluruhan, output menunjukkan bagaimana Java menangani *type casting* antara berbagai tipe data.

8. Class Ekspresi

```
D:\PBO\Tugas1>javac Ekspresi.java
D:\PBO\Tugas1>java Ekspresi.java
x = 1
y = 2
hasil ekspresi = (x*y):x:y = 1
```

Saat **Ekspresi.java** dijalankan, program menampilkan nilai $x = 1$ dan $y = 2$, yang menunjukkan bahwa variabel x dan y telah diinisialisasi dengan nilai tersebut. Kemudian, ekspresi $(x < y) ? x : y$ dievaluasi menggunakan operator ternary. Karena $x < y$ bernilai true (1 kurang dari 2), maka ekspresi menghasilkan nilai x , yaitu 1. Oleh karena itu, output akhirnya adalah hasil ekspresi $= (x < y) ? x : y = 1$.

9. Class Ekspresi1

```
D:\PBO\Tugas1>javac Ekspresi1.java

D:\PBO\Tugas1>java Ekspresi1.java
x/y (format integer) = 0
x/y (format float) = 0
x/y (format integer) = 0.5
x/y (format float) = 0.5
float(x)/float(y) (format integer) = 0.5
float(x)/float(y) (format float) = 0.5
x/y (format integer) = 3
x/y (format float) = 3
```

Saat **Ekspresi1.java** dijalankan, program menunjukkan hasil pembagian x dan y dalam berbagai format tipe data. Pada awalnya, **x/y (format integer) = 0** menunjukkan bahwa pembagian dua bilangan bulat menghasilkan hasil integer, sehingga desimal dibulatkan ke bawah. **x/y (format float) = 0** kemungkinan karena kedua nilai masih dianggap sebagai integer. Selanjutnya, **x/y (format integer) = 0.5** dan **x/y (format float) = 0.5** menunjukkan pembagian dengan hasil desimal, kemungkinan karena setidaknya salah satu variabel dikonversi ke float. Kemudian, **$\text{float}(x)/\text{float}(y)$ (format integer) = 0.5** dan **$\text{float}(x)/\text{float}(y)$ (format float) = 0.5** menunjukkan bahwa pembagian dilakukan sepenuhnya dalam format float. Terakhir, **x/y (format integer) = 3** dan **x/y (format float) = 3** menunjukkan bahwa operasi pembagian menghasilkan hasil yang tepat tanpa desimal, kemungkinan karena nilai x dan y berbeda dari kasus sebelumnya. Output memperlihatkan perbedaan antara operasi pembagian dalam tipe data int dan float.

10. Class Hello

```
D:\PBO\Tugas1>javac Hello.java

D:\PBO\Tugas1>java Hello.java
Hello
Hello World
Welcome
```

Saat **Hello.java** dijalankan, program mencetak tiga baris output. Baris pertama menampilkan "Hello", yang berasal dari perintah `System.out.println("Hello");`. Baris kedua mencetak "Hello World", menunjukkan bahwa program memiliki perintah tambahan seperti `System.out.println("Hello World");`. Baris terakhir menampilkan "Welcome", yang berarti ada perintah serupa untuk mencetak kata tersebut.

Secara keseluruhan, program ini adalah contoh sederhana dalam Java untuk mencetak teks ke layar menggunakan `System.out.println()`.

11. Class Incr

```
D:\PBO\Tugas1>javac Incr.java
D:\PBO\Tugas1>java Incr.java
Nilai i : 5
Nilai j : 3
```

Output program menunjukkan dua nilai yang dihasilkan setelah operasi tertentu pada variabel *i* dan *j*. Baris pertama mencetak "Nilai *i* : 5", menunjukkan bahwa variabel *i* memiliki nilai 5 setelah proses tertentu. Baris kedua mencetak "Nilai *j* : 3", yang berarti variabel *j* bernilai 3 setelah dieksekusi. Program ini menggunakan operasi *increment* atau *decrement* untuk mengubah nilai variabel sebelum dicetak.

12. Class Oper1

```
Command Prompt
D:\PBO\Tugas1>javac Oper1.java
D:\PBO\Tugas1>java Oper1.java
n = 10
x = 1
y = 2
n & 8 = 8
x & ~ 8 = 1
y << 2 = 8
y >> 3 = 0
```

Output program menunjukkan hasil dari berbagai operasi bitwise (*bitwise AND*, *NOT*, *left shift*, dan *right shift*) pada variabel *n*, *x*, dan *y*.

- *n* = 10, *x* = 1, dan *y* = 2 menunjukkan nilai awal variabel.
- *n* & 8 = 8 berarti operasi *bitwise AND* antara 10 (1010 dalam biner) dan 8 (1000 dalam biner) menghasilkan 1000 atau 8.
- *x* & ~ 8 = 1 menunjukkan operasi *bitwise AND* antara 1 (0001 dalam biner) dan komplement dari 8 (~1000 = 0111), hasilnya tetap 1.
- *y* << 2 = 8 berarti *y* (2 atau 0010 dalam biner) digeser ke kiri 2 bit menjadi 1000 atau 8.
- *y* >> 3 = 0 berarti *y* (2 atau 0010) digeser ke kanan 3 bit menjadi 0000, hasilnya 0.

Program mendemonstrasikan bagaimana operasi bitwise bekerja pada bilangan biner di Java.

13. Class Oper2

```
Command Prompt
D:\PBO\Tugas1>javac Oper2.java
D:\PBO\Tugas1>java Oper2.java
i = 3
j = 4
i & j = 0
i | j = 7
i ^ j = 7
81.0
~i = -4
```

Output program ini menunjukkan hasil dari berbagai operasi bitwise (*AND*, *OR*, *XOR*, *NOT*) serta perhitungan pada bilangan desimal.

- *i* = 3 dan *j* = 4 menunjukkan nilai awal variabel.
- *i* & *j* = 0 adalah hasil *bitwise AND* antara 3 (0011 dalam biner) dan 4 (0100 dalam biner), yang menghasilkan 0000 atau 0.

- $i | j = 7$ adalah hasil *bitwise OR*, yang menggabungkan bit dari i dan j ($0011 | 0100 = 0111$ atau 7).
- $i \wedge j = 7$ adalah hasil *bitwise XOR*, yang menghasilkan 1 hanya jika bit berbeda ($0011 \wedge 0100 = 0111$ atau 7).
- 81.0 kemungkinan merupakan hasil dari operasi aritmetika seperti perpangkatan atau pembagian.
- $\sim i = -4$ adalah hasil dari operasi *bitwise NOT* pada 3 (~ 0011 menjadi 1100 dalam bentuk bilangan negatif, yaitu -4 dalam representasi bilangan bertanda).

Program mendemonstrasikan penggunaan operasi bitwise dan aritmetika dalam Java.

14. Class Oper3

```

Command Prompt

D:\PBO\Tugas1>javac Oper3.java

D:\PBO\Tugas1>java Oper3.java
true
false
true
true
true

```

Output program ini menunjukkan hasil dari operasi logika (*boolean*), yang biasanya melibatkan operator seperti $\&\&$ (AND), $\|$ (OR), dan $!$ (NOT).

- `true` kemungkinan berasal dari ekspresi yang menghasilkan nilai benar, seperti perbandingan dua nilai yang sama atau penggunaan operator OR ($\|$).
- `false` muncul ketika suatu kondisi tidak terpenuhi, misalnya pada operasi AND ($\&\&$) jika salah satu operand bernilai *false*.
- Urutan output menunjukkan berbagai kombinasi operasi logika yang menghasilkan nilai akhir *true* atau *false*.

15. Class Oper4

```

Command Prompt

D:\PBO\Tugas1>javac Oper4.java

D:\PBO\Tugas1>java Oper4.java
Nilai e = 10
Nilai k = 0
Nilai k = 4

```

Output program ini menunjukkan hasil operasi pada variabel e dan k .

- Nilai $e = 10$ berarti variabel e diberi nilai awal 10.
- Nilai $k = 0$ kemungkinan menunjukkan bahwa k diinisialisasi dengan nilai 0 sebelum dilakukan operasi lebih lanjut.
- Nilai $k = 4$ menandakan bahwa ada operasi yang mengubah nilai k , seperti operasi aritmatika atau *bitwise*.

16. Class Oprator

```
Command Prompt
D:\PBO\Tugas1>javac Oprator.java

D:\PBO\Tugas1>java Oprator.java
Hasil dari berbagai operasi:

Boolean AND      : false
Boolean OR       : true
Boolean NOT      : false
Boolean XOR      : true

Operasi Numerik (Integer):
Penjumlahan      : 7
Pengurangan      : 3
Perkalian         : 10
Pembagian bulat  : 2
Modulo           : 1

Operasi Numerik (Float):
Penjumlahan      : 10.0
Pengurangan      : 0.0
Perkalian         : 25.0
Pembagian        : 1.0

Operasi Relasional (Integer):
i == j           : false
i != j           : true
i < j            : false
i > j            : true
i <= j           : false
i >= j           : true

Operasi Relasional (Float):
x != y           : false
x < y            : false
x > y            : false
x <= y           : true
x >= y           : true
```

Output program ini menunjukkan hasil dari berbagai operasi logika, numerik, dan relasional dalam Java.

- **Operasi Boolean** mencakup AND (&&), OR (||), NOT (!), dan XOR (^), yang bekerja berdasarkan nilai kebenaran dari variabel.
- **Operasi Numerik** terbagi menjadi integer dan float, mencakup penjumlahan, pengurangan, perkalian, pembagian, dan modulo untuk integer, serta operasi serupa untuk float tanpa modulo.
- **Operasi Relasional** membandingkan dua angka, baik dalam format integer maupun float, menggunakan operator seperti ==, !=, <, >, <=, dan >=.