

LAPORAN
PEMROGRAMAN BERBASIS OBJEK



Nama : Rizqi Ananda Jalil
Stambuk : 13020230244
Kelas : TI_PBO-14 (B4)
Dosen : Mardiyah Hasnawi, S.Kom., M.T., MTA

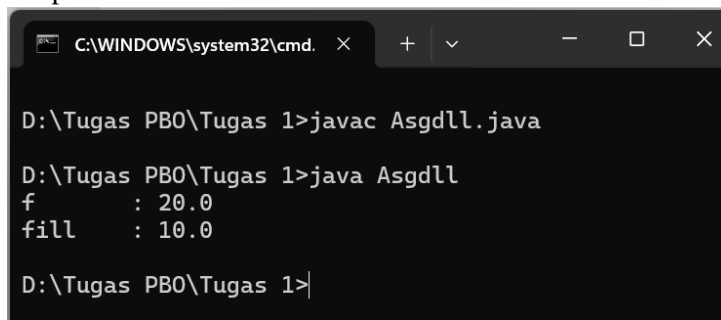
PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS ILMU KOMPUTER
UNIVERSITAS MUSLIM INDONESIA
MAKASSAR
2025

Tugas 1 – Pemrograman Berbasis Objek

1. Penjelasan Source Code :

Pada program dengan deklarasi bernama class Asgdl1, terdapat dua variabel yaitu f dengan tipe data float yang bernilai 20.0f dan fill dengan tipe data double namun belum memiliki nilai. Kemudian dideklarasikan lagi variabel fill yang diberi nilai 10.0f. Pada dasarnya variabel fill yang diberikan nilai 10.0f tersebut bertipe data float tapi karena double yang mendeklarasikan fill tadi memiliki presisi lebih tinggi, maka nilai tersebut otomatis dikonversi tanpa kita perlu melakukan perubahan secara manual. Setelah programnya dijalankan, program tersebut menampilkan output f : 20.0 dan fill : 10.0 menggunakan sintaks System.out.println() yang menunjukkan bahwa nilai f tetap dalam bentuk float, sedangkan fill dalam bentuk double setelah dideklarasikan sebelumnya namun tanpa desimal tambahan karena nilai nya dari 10.0f.

Output:



```
C:\WINDOWS\system32\cmd. x + - □ x

D:\Tugas PBO\Tugas 1>javac Asgdl1.java

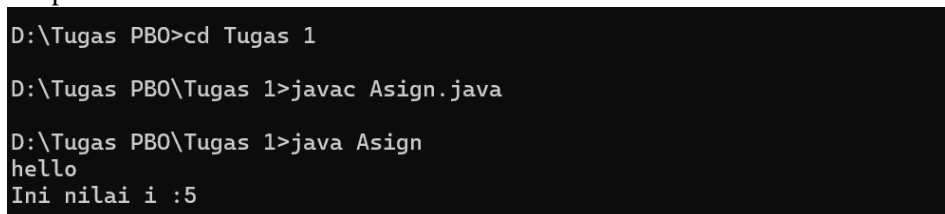
D:\Tugas PBO\Tugas 1>java Asgdl1
f      : 20.0
fill   : 10.0

D:\Tugas PBO\Tugas 1>
```

2. Penjelasan Source Code :

Pada program dengan class Assign, sebenarnya yang harus dicetak lebih awal adalah "hello" namun karena dengan sintaks System.out.println java menggunakan sistem penyimpanan sementara sehingga output nya tertahan. Maka yang tampil hanya hasil dari algoritma System.out. println("Ini nilai i : " + 1); yaitu 1 yang sebelumnya dideklarasikan variabel i dengan nilai 5. Namun dalam kode untuk menampilkannya diberikan secara langsung nilai 1 sehingga yang tampil bukan nilai dari variabel i yaitu 5.

Output:



```
D:\Tugas PBO>cd Tugas 1

D:\Tugas PBO\Tugas 1>javac Assign.java

D:\Tugas PBO\Tugas 1>java Assign
hello
Ini nilai i :5
```

3. Penjelasan Source Code:

Pada program dengan class ASIGNi terdapat didalamnya beberapa variabel dengan tipe data berbeda yaitu short ks dengan menyimpan nilai 1, int ki dengan menyimpan nilai 1, long kl dengan menyimpan nilai 10.000, char c dengan menyimpan nilai 65 yang dimana dalam code ASCII angka 65 mewakili karakter A, kemudian ada variabel c1 dengan tipe data char yang menyimpan karakter 'z',

variabel x dengan tipe data double yang menyimpan nilai 50.2f namun karena 50.2f adalah float sehingga dikonversi menjadi angka yang lebih panjang yaitu 50.20000076293945 karena double memiliki presisi yang lebih panjang. sedangkan variabel y yang pada dasarnya bertipe data float dengan menyimpan nilai 50.2f maka tetap menampilkan 50.2

Output:

```
D:\Tugas PBO\Tugas 1>javac ASIGNi.java

D:\Tugas PBO\Tugas 1>java ASIGNi
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:\Tugas PBO\Tugas 1>
```

4. Penjelasan Source Code:

Program dengan nama class BacaData membaca input angka dari yang diinputkan melalui class Scanner dari package java.util. Kemudian dideklarasikan variabel a dengan tipe data integer dan variabel masukan. Kemudian dibuatkan objek Scanner untuk membaca inputan yang disimpan dalam variabel a dengan masukan.nextInt();. Dalam hal ini program meminta pengguna untuk menginputkan angka, saya sendiri memasukkan angka 17, maka yang tampil adalah "Nilai yang dibaca : 17".

Output:

```
02/27/2025 05:41 PM 319 BacaData.java
7 File(s) 3,990 bytes
2 Dir(s) 52,154,191,872 bytes free

D:\Tugas PBO\Tugas 1>javac BacaData.java

D:\Tugas PBO\Tugas 1>java BacaData
Contoh membaca dan menulis, ketik nilai integer:
17
Nilai yang dibaca : 17
D:\Tugas PBO\Tugas 1>
```

5. Penjelasan Source Code:

Pada program dengan nama class Bacakar menggunakan 3 import library yaitu BufferedReader, IOException, dan InputStreamReader. Dalam program ini terdapat 2 variabel yaitu cc dengan tipe data char dan bil dengan tipe data integer. Kemudian menampilkan hello dengan sintaks perintah System.out.print("hello\n");. Lalu menampilkan teks "baca 1 karakter" dengan sintaks System.out.print("baca 1 karakter : "); menggunakan variabel cc = dataIn.readLine().charAt(0) untuk membaca seluruh baris input sebagai string dan mengambil karakter pertama dari inputan. Lalu menampilkan teks "baca 1 bilangan: " dengan sintaks System.out.print("baca 1 bilangan : "); menggunakan variabel bil = Integer.parseInt(dataIn.readLine()); untuk membaca input sebagai string dan mengubah string menjadi bilangan integer. Nah, pada yang diinputkan tadi saya memasukkan karakter r dan bilangan 7 maka pada

System.out.print(cc + "\n" + bil + "\n"); yang tampil adalah r 7. Lalu menampilkan teks bye dari sintaks System.out.print ("bye \n");

Output:

```
02/27/2025 03:41 PM 519 Bacadata.java
02/27/2025 11:37 PM 694 Bacakar.java
          9 File(s)      5,756 bytes
          2 Dir(s)  52,153,278,464 bytes free

D:\Tugas PBO\Tugas 1>javac Bacakar.java

D:\Tugas PBO\Tugas 1>java Bacakar
hello
baca 1 karakter : a
baca 1 bilangan : 7
a
7
bye

D:\Tugas PBO\Tugas 1>
```

6. Penjelasan Source Code:

Pada program dengan nama class Casting1, didalamnya terdapat beberapa variabel diantaranya yaitu int a = 5, b = 6; float d = 2.f, e = 3.2f; char g = '5'; double k = 3.14; kemudian dilakukan konversi pada sintaks perintah untuk menampilkannya diantaranya yaitu :

- Variabel a menyimpan nilai 5 yang bertipe data integer dikonversi ke float, sehingga hasilnya menjadi 5.0
- Variabel b menyimpan nilai 6 yang bertipe data integer dikonversi ke double, sehingga hasilnya menjadi 6.0
- Variabel d menyimpan nilai 2.f yang bertipe data float dikonversi ke integer, sehingga hasilnya menjadi 2
- Variabel e menyimpan nilai 3.2f yang bertipe data float dikonversi ke double, sehingga hasilnya tetap 3.2 karena double bisa merepresentasikan nilai float tanpa perubahan.
- Variabel g menyimpan karakter '5' yang bertipe data char dikonversi ke integer, sehingga hasilnya menjadi 53 karena berdasarkan kode ASCII karakter '5' memiliki nilai ASCII 53.
- Variabel k menyimpan nilai 3.14 yang bertipe data double dikonversi ke integer, sehingga hasilnya menjadi 3 karena integer merupakan bilangan bulat sehingga 3.14 dipotong menjadi 3.
- Variabel k menyimpan nilai 3.14 yang bertipe data double dikonversi ke float, sehingga tetap 3.14 karena double bisa merepresentasikan nilai float tanpa perubahan meskipun dengan presisi lebih rendah.

Output:

```
02/27/2025 11:37 PM 694 Casting1.java
02/27/2025 11:43 PM 459 Casting1.java
          11 File(s)     7,627 bytes
          2 Dir(s)  52,153,274,368 bytes free

D:\Tugas PBO\Tugas 1>javac Casting1.java

D:\Tugas PBO\Tugas 1>java Casting1
5.0
6.0
2
3.2000000047683716
53
53.0
53.0
3
3.14

D:\Tugas PBO\Tugas 1>
```

7. Penjelasan Source Code:

Pada program dengan class Casting2 juga menggunakan metode konversi tipe data sama seperti sebelumnya seperti mengubah teks (String) menjadi angka, angka menjadi teks, dan mengonversi antara jenis angka yang berbeda. Awalnya, program mendeklarasikan beberapa variabel dengan tipe int, float, double, char, dan String.

- Integer.parseInt(n) = Mengubah "67" menjadi int 67, lalu menyimpannya ke variabel a.
- Double.parseDouble(m) = Mengubah "45" menjadi double 45.0, lalu menyimpannya ke k.
- Float.parseFloat(l) = Mengubah "100" menjadi float 100.0, lalu menyimpannya ke d.
- String.valueOf(b) = Mengubah int 9 menjadi "9", lalu menyimpannya ke n.
- String.valueOf(g) = Mengubah char '5' menjadi "5", lalu menyimpannya ke m.
- String.valueOf(e) = Mengubah float 3.2f menjadi "3.2", lalu menyimpannya ke l.
- Double.valueOf(a).intValue(). Variabel a = 67 (int) dikonversi ke Double lalu dikonversi kembali ke int, tetap menghasilkan 67 kemudian variabel k tetap bernilai 67.0 karena bertipe double.
- Integer.valueOf(b).doubleValue(), Variabel b = 9 (int) dikonversi ke Integer, lalu diubah ke double, menghasilkan 9.0 sehingga c bernilai 9.0.
- Variabel l awalnya bertipe String dengan nilai "100". Kemudian, nilai `l` diubah menggunakan String.valueOf(e), di mana e adalah variabel bertipe float dengan nilai 3.2f. Proses ini mengubah nilai l dari "100" menjadi "3.2" dalam bentuk String, sehingga ketika dicetak, yang ditampilkan adalah "3.2".

Output:

```
02/27/2025 11:43 PM          459 Casting1.java
02/27/2025 11:47 PM          648 Casting2.java
          13 File(s)        8,839 bytes
          2 Dir(s)  52,153,274,368 bytes free

D:\Tugas PB0\Tugas 1>javac Casting2.java

D:\Tugas PB0\Tugas 1>java Casting2
a : 67
k : 45.0
d : 100.0
n : 9
m : 5
l : 3.2
k : 67.0
c : 9.0
l : 3.2

D:\Tugas PB0\Tugas 1>
```

Penjelasan:

8. Penjelasan Source Code:

Program dengan class Ekspresi mendeklarasikan dua variabel integer yaitu x dengan nilai 1 dan y dengan nilai 2. Program kemudian mencetak nilai x dan y ke konsol menggunakan System.out.print(), dengan setiap nilai diikuti dengan baris baru. Selanjutnya program menggunakan operator ternary (x < y) ? x : y untuk mengevaluasi ekspresi sebagai pernyataan kondisi sederhana yang mencetak apakah x lebih kecil dari y. Karena 1 < 2 bernilai true, maka ekspresi menghasilkan nilai x yaitu 1. Hasil ini kemudian ditampilkan dengan teks "hasil ekspresi = (x < y) ? x : y = 1" di konsol.

Output:

```

02/28/2025 03:58 AM 267 Ekspresi.java
15 File(s) 10,850 bytes
2 Dir(s) 52,153,270,272 bytes free

D:\Tugas PBO\Tugas 1>javac Ekspresi.java

D:\Tugas PBO\Tugas 1>java Ekspresi
x = 1
y = 2
hasil ekspresi = (x<y) ? x:y = 1
D:\Tugas PBO\Tugas 1>

```

9. Penjelasan Source Code:

Program dengan class Ekspresi1 menerapkan bagaimana tipe data integer dan float beroperasi pada pembagian di Java.

- Pertama, variabel x dan y dideklarasikan sebagai integer dengan nilai 1 dan 2, sedangkan fx dan fy dideklarasikan sebagai float. Saat mencetak hasil x/y, karena kedua variabel bertipe integer maka hasilnya adalah pembagian integer yang dipotong menjadi 0. Setelah fx dan fy diisi dengan x dan y, hasil pembagian fx/fy berubah menjadi float yaitu 0.5.
- Kedua, ekspresi (float)x dan (float)y menunjukkan bahwa jika x dan y dikonversi ke float sebelum pembagian, hasilnya tetap 0.5. Setelah x dan y diubah menjadi 10 dan 3, hasil pembagian integer x/y menjadi 3, karena bagian desimal dibuang. Namun, jika dikonversi ke float terlebih dahulu hasilnya tetap 3.33333333.

Output:

```

02/28/2025 03:58 AM 267 Ekspresi.java
02/28/2025 04:05 AM 702 Ekspresi1.java
17 File(s) 12,600 bytes
2 Dir(s) 52,153,266,176 bytes free

D:\Tugas PBO\Tugas 1>javac Ekspresi1.java

D:\Tugas PBO\Tugas 1>java Ekspresi1
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
D:\Tugas PBO\Tugas 1>

```

10. Penjelasan Source Code:

Program dengan class Hello ini menerapkan perbedaan antara print() dan println() dalam mencetak teks ke konsol.

- Pertama, System.out.print("Hello"); mencetak "Hello" tanpa berpindah ke baris baru.
 - Kedua, System.out.print("\nHello "); mencetak karakter newline (\n), sehingga teks "Hello " muncul di baris berikutnya
 - Ketiga, System.out.println("World"); mencetak "World" dan langsung berpindah ke baris baru karena menggunakan println()
 - Keempat, System.out.println("Welcome"); mencetak "Welcome" di baris baru.
- Output:

```

02/28/2025 04:09 AM 215 Hello.java
19 File(s) 14,115 bytes
2 Dir(s) 52,153,262,080 bytes free

D:\Tugas PBO\Tugas 1>javac Hello.java

D:\Tugas PBO\Tugas 1>java Hello
Hello
Hello World
Welcome

D:\Tugas PBO\Tugas 1>

```

11. Penjelasan Source Code:

Program dengan nama class Incr menerapkan penggunaan post-increment (`i++`) dan pre-increment (`++i`).

- Pertama, variabel `i` dan `j` dideklarasikan sebagai integer. Variabel `i` diinisialisasi dengan nilai 3.
- Kedua, `j = i++` menggunakan post-increment, yang berarti nilai `i` (3) diberikan ke `j` terlebih dahulu, lalu `i` bertambah menjadi 4.

Dalam perintah `System.out.println`, ekspresi `++i` adalah pre-increment, yang meningkatkan `i` dari 4 ke 5 sebelum mencetaknya. Sementara itu, nilai `j` tetap 3 karena tidak mengalami perubahan setelah assignment. Maka hasil outputnya adalah "Nilai `i` : 5" dan "Nilai `j` : 3"

Output:

```

02/28/2025 04:12 AM 178 Incr.java
21 File(s) 14,785 bytes
2 Dir(s) 52,153,262,080 bytes free

D:\Tugas PBO\Tugas 1>javac Incr.java

D:\Tugas PBO\Tugas 1>java Incr
Nilai i : 5
Nilai j : 3

D:\Tugas PBO\Tugas 1>

```

12. Penjelasan Source Code:

Program dengan class `Oper1` menerapkan penggunaan operasi bitwise di Java, termasuk AND (`&`), NOT (`~`), left shift (`<<`), dan right shift (`>>`).

- Variabel `n`, `x`, dan `y` dideklarasikan dengan nilai 10 (1010), 1 (0001), dan 2 (0010) dalam bentuk biner. Diantara operasi yang dilakukan yaitu :
 - `n & 8` menghasilkan 8 karena 1010 & 1000 menghasilkan 1000 (8 dalam desimal).
 - `x & ~8` menghasilkan 1 karena menggunakan komplemen bit dari 8 (1000 -> 0111 dalam 4-bit), sehingga 0001 & 0111 tetap 1.
 - `y << 2` menggeser bit `y` (2 -> 0010) ke kiri dua kali, menghasilkan 1000 (8 dalam desimal).
 - `y >> 3` menggeser bit `y` ke kanan tiga kali, sehingga semua bit hilang dan hasilnya menjadi 0.

Output:

```
02/28/2025 04:15 AM 426 Oper1.java
23 File(s) 16,083 bytes
2 Dir(s) 52,153,257,984 bytes free

D:\Tugas PBO\Tugas 1>javac Oper1.java

D:\Tugas PBO\Tugas 1>java Oper1
n = 10
x = 1
y = 2
n & 8 = 8
x & ~ 8 = 1
y << 2 = 8
y >> 3 = 0

D:\Tugas PBO\Tugas 1>
```

13. Penjelasan Source Code:

Program dengan class Oper2 menerapkan operasi bitwise dan matematika pada variabel bertipe char. Variabel i dan j diinisialisasi dengan nilai 3 dan 4, yang dalam representasi ASCII merepresentasikan karakter dengan nilai numerik yang sama.

Hasil operasi:

- (int) i dan (int) j menghasilkan 3 dan 4 karena dilakukan konversi eksplisit ke integer.
- i & j (bitwise AND) pada 00000011 (3) & 00000100 (4) menghasilkan 00000000 (0).
- i | j (bitwise OR) menghasilkan 00000111 (7).
- i ^ j (bitwise XOR) menghasilkan 00000111 (7).
- Math.pow(i, j) menghasilkan 81.0.
- ~i (bitwise NOT) menghasilkan $3^4 = 81.0$. Operasi ~i (bitwise NOT) membalik semua bit dari 3 (00000011) menjadi -4 dalam representasi two's complement.

Output:

```
02/28/2025 04:19 AM 413 Oper2.java
25 File(s) 17,597 bytes
2 Dir(s) 52,153,253,888 bytes free

D:\Tugas PBO\Tugas 1>javac Oper2.java

D:\Tugas PBO\Tugas 1>java Oper2
i = 3
j = 4
i & j = 0
i | j = 7
i ^ j = 7
81.0
~i = -4

D:\Tugas PBO\Tugas 1>
```

14. Penjelasan Source Code:

Program dengan nama class Oper3 mendemonstrasikan penggunaan operator logika (&&, ||) dan bitwise (&, |) dalam pernyataan if.

1. if (true && true) mengevaluasi true, sehingga System.out.println(true && true); mencetak true.
2. if (true & true) juga mengevaluasi true, tetapi System.out.println(true & false); mencetak false, karena true & false pada bitwise AND adalah false.
3. if (true) langsung bernilai true, sehingga System.out.println(true); mencetak true.
4. if (true || true) mengevaluasi true, sehingga System.out.println(true); mencetak true.
5. if (true | false) menggunakan OR bitwise, yang menghasilkan true, sehingga System.out.println(true | false); mencetak true.

Output:

```
02/28/2025 04:23 AM 344 Oper3.java
27 File(s) 19,074 bytes
2 Dir(s) 52,153,249,792 bytes free

D:\Tugas PB0\Tugas 1>javac Oper3.java

D:\Tugas PB0\Tugas 1>java Oper3
true
false
true
true
true
D:\Tugas PB0\Tugas 1>
```

15. Penjelasan Source Code:

Program Oper4 menggunakan operator ternary (? :) untuk menentukan nilai variabel berdasarkan kondisi tertentu. Variabel i dan j diinisialisasi dengan nilai 0, sedangkan c dan d bertipe char memiliki nilai 8 dan 10. Variabel e dan k ditentukan dengan operator ternary berdasarkan perbandingan (int)c > (int)d dan i > j, menghasilkan nilai e = 10 dan k = 0. Setelah nilai i dan j diperbarui menjadi 2 dan 3, nilai k diperbarui menjadi 3 berdasarkan kondisi i++ > j++. Output akhir program ini adalah "Nilai e = 10", "Nilai k = 0", dan "Nilai k = 3".

Output:

```
02/28/2025 04:25 AM 381 Oper4.java
29 File(s) 19,875 bytes
2 Dir(s) 52,153,249,792 bytes free

D:\Tugas PB0\Tugas 1>javac Oper4.java

D:\Tugas PB0\Tugas 1>java Oper4
Nilai e = 10
Nilai k = 0
Nilai k = 4
D:\Tugas PB0\Tugas 1>
```

16. Penjelasan Source Code:

Program Oprator mendemonstrasikan berbagai operasi logika, aritmatika, dan relasional pada tipe data boolean, int, dan float. Variabel Bool1 dan Bool2 diinisialisasi dengan true dan false, kemudian diterapkan operasi logika seperti AND, OR, NOT, dan XOR yang menghasilkan output false, true, false, dan true. Operasi aritmatika dilakukan pada variabel int i = 5 dan j = 2, termasuk

penjumlahan, pengurangan, perkalian, pembagian, dan modulus. Variabel float $x = 5.0$ dan $y = 5.0$ juga diterapkan operasi serupa. Operasi relasional menunjukkan hasil evaluasi perbandingan antara nilai int dan float, seperti $i == j$, $i != j$, $i > j$, dan $i < j$ untuk int, serta $x != y$, $x < y$, $x > y$, $x <= y$, dan $x >= y$ untuk float.

Output:

```
D:\Tugas PBO\Tugas 1>javac Oprator.java

D:\Tugas PBO\Tugas 1>java Oprator
Silahkan baca teksnya dan tambahkan perintah untuk menampilkan output
Bool1 AND Bool2: false
Bool1 OR Bool2: true
NOT Bool1: false
Bool1 XOR Bool2: true
i + j: 7
i - j: 3
i * j: 10
i / j: 2
i % j: 1
x + y: 10.0
x - y: 0.0
x / y: 1.0
x * y: 25.0
i == j: false
i != j: true
i < j: false
i > j: true
i <= j: false
i >= j: true
x != y: false
x < y: false
x > y: false
x <= y: true
x >= y: true

D:\Tugas PBO\Tugas 1>
```