

**PEMROGRAMAN BERBASIS OBJEK
(PBO)**



Nama : Nayla Dwi Rianti Putri

Kelas : B2

Nim : 13020230154

Dosen : Mardiyah Hasnawati, S.Kom.,M.T.

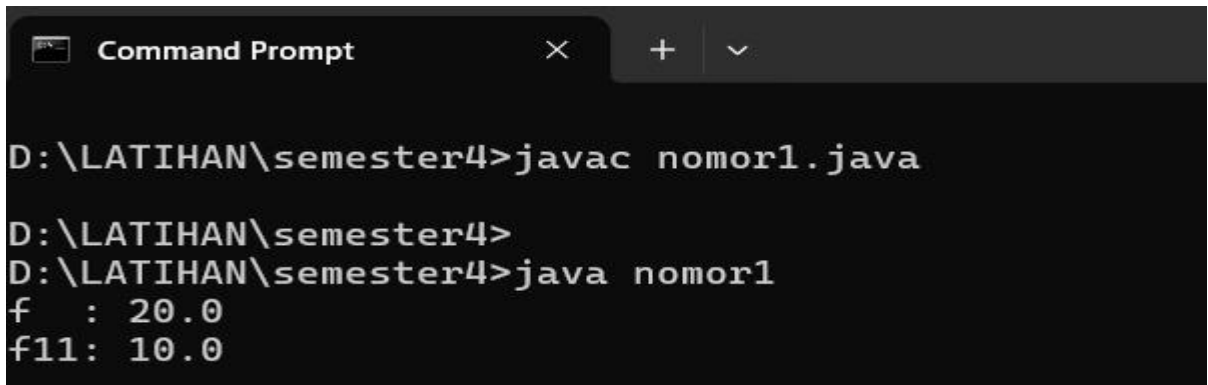
**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS ILMU KOMPUTER
UNIVERSITAS MUSLIM INDONESIA
MAKASSAR**

2024

1. Penjelasan

- `Float f = 20.0f`: Mendeklarasikan variabel `f` bertipe `float` dan menginisialisasinya dengan nilai 20.0.
- `Double f11`: Mendeklarasikan variabel `f11` bertipe `double` tanpa inisialisasi
- `f11 = 10.0f`: Menginisialisasi variabel `f11` dengan nilai 10.0. Meskipun `f11` bertipe `double`, nilai `10.0f` (yang merupakan `float`) dapat diassign ke `double` karena `float` dapat dipromosikan ke `double`.
- Secara keseluruhan, program ini mendeklarasikan dua variabel numerik, menginisialisasi mereka, dan mencetak nilainya ke konsol.

Output:p



```
Command Prompt

D:\LATIHAN\semester4>javac nomor1.java

D:\LATIHAN\semester4>
D:\LATIHAN\semester4>java nomor1
f : 20.0
f11: 10.0
```

2. Enjelasan

- `int i`: Mendeklarasikan variabel `i` bertipe `int` (bilangan bulat) tanpa inisialisasi.
- `System.out.print("hello\n");`: Mencetak string "hello" di konsol diikuti dengan karakter newline (`\n`), yang berarti baris berikutnya akan dimulai di bawahnya.
- `i = 5`: Menginisialisasi variabel `i` dengan nilai 5.
- `System.out.println("Ini nilai i : " + i);`: Mencetak string "Ini nilai i :" diikuti dengan nilai dari variabel `i` (yaitu 5) ke konsol. `println` secara otomatis menambahkan newline setelah output.
- Secara keseluruhan, program ini mendeklarasikan sebuah variabel integer, menginisialisasinya, dan mencetak pesan ke konsol yang menunjukkan nilai dari variabel tersebut.

Output

```
D:\LATIHAN\semester4>java nomor2
hello
Ini nilai i :5

D:\LATIHAN\semester4>
```

3. Penjelasan

- Program mencetak karakter `c` dan `c1` ke konsol. Karakter `c` akan dicetak sebagai 'A' (karena 65 dalam ASCII adalah 'A'), dan `c1` akan dicetak sebagai 'Z'.
- Program juga mencetak nilai dari variabel `ks`, `ki`, `kl`, `x`, dan `y` ke konsol.
- Secara keseluruhan, program ini mendeklarasikan beberapa variabel dengan berbagai tipe data, menginisialisasi mereka, dan mencetak nilai serta karakter yang diwakili oleh variabel tersebut ke konsol.

Output

```
Command Prompt
D:\LATIHAN\semester4>javac nomor3.java
D:\LATIHAN\semester4>java nomor3
Karakter = A
Karakter = Z
Karakter = A
Karakter = Z
Bilangan integer (short) = 1
(int) = 1
(long)= 10000
Bilangan Real x = 50.200000076293945
Bilangan Real y = 50.2

D:\LATIHAN\semester4>
```

4. Penjelasan

- program ini menunjukkan cara dasar untuk membaca input integer dari pengguna dan menampilkannya kembali ke konsol.

Output

```
D:\LATIHAN\semester4>javac nomor4.java

D:\LATIHAN\semester4>java nomor4
Contoh membaca dan menulis, ketik nilai integer:
|
```

5. Penjelasan

- `InputStreamReader isr = new InputStreamReader(System.in);`: Membuat objek `InputStreamReader` untuk membaca input dari konsol.
- `BufferedReader dataIn = new BufferedReader(isr);`: Membuat objek `BufferedReader` untuk membaca data dengan lebih efisien dari `InputStreamReader`.
- `BufferedReader dataIn = new BufferedReader(new InputStreamReader(System.in));`: Alternatif cara untuk membuat objek `BufferedReader` yang sama.
- Secara keseluruhan, program ini menunjukkan cara membaca karakter dan bilangan bulat dari input pengguna menggunakan `BufferedReader` dan mencetak hasilnya ke konsol.

Output

```
Command Prompt
D:\LATIHAN\semester4>javac nomor5.java
D:\LATIHAN\semester4>java nomor5
hello
baca 1 karakter : N
baca 1 bilangan : 10
N
10
bye
```

6. Penjelasan

- , program ini menunjukkan cara melakukan konversi tipe data di Java, termasuk konversi dari `int` ke `float`, `double`, dan sebaliknya, serta konversi dari `char` ke `int`, `float`, dan `double`. Program ini juga memberikan pemahaman tentang bagaimana nilai ASCII dari karakter dapat digunakan dalam konversi.

Output:

```
D:\LATIHAN\semester4>javac nomor6.java
D:\LATIHAN\semester4>java nomor6
5.0
6.0
2
3.2000000047683716
53
53.0
53.0
3
3.14
```

7. Penjelasan

- program ini menunjukkan cara melakukan konversi antara tipe data yang berbeda di Java, termasuk konversi dari `String` ke `int`, `double`, dan `float`, serta konversi dari `int`, `char`, dan `float` kembali ke `String`. Program ini juga memberikan pemahaman tentang penggunaan kelas pembungkus (wrapper classes) seperti `Integer` dan `Double` untuk konversi tipe data.

Output:

```

Command Prompt
D:\LATIHAN\semester4>javac nomor7.java
D:\LATIHAN\semester4>java nomor7
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:\LATIHAN\semester4>

```

8. penjelasan

- Program menggunakan operator ternary untuk mengevaluasi ekspresi $(x < y)$. Jika kondisi tersebut benar (true), maka nilai `x` akan dicetak; jika salah (false), maka nilai `y` yang akan dicetak.
- program ini menunjukkan penggunaan variabel, operator ternary, dan cara mencetak output ke konsol dalam Java. Ini adalah contoh sederhana dari pengkondisian dalam pemrograman.

Output:

```

D:\LATIHAN\semester4>javac nomor8.java
D:\LATIHAN\semester4>java nomor8
x = 1
y = 2
hasil ekspresi = (x<y)?x:y = 1
D:\LATIHAN\semester4>

```

9. penjelasan

- Program mencetak hasil pembagian `x` dan `y` dalam format integer dan float. Namun, karena `x` dan `y` adalah integer, hasil pembagian `x/y` dalam format integer akan dibulatkan ke bawah (floor division).
- Untuk mendapatkan hasil yang benar dalam format float, `fx` dan `fy` diinisialisasi dengan nilai `x` dan `y`:

- program ini menunjukkan bagaimana pembagian antara tipe data integer dan float bekerja di Java, serta pentingnya casting untuk mendapatkan hasil yang diinginkan. Program ini juga memberikan pemahaman tentang bagaimana hasil pembagian dapat berbeda tergantung pada tipe data yang digunakan.

Output:

```
D:\LATIHAN\semester4>java nomor9
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
```

10. Penjelasan

- `System.out.print("Hello");`: Mencetak string "Hello" ke konsol tanpa menambahkan newline setelahnya. Ini berarti kursor tetap berada di akhir string yang dicetak.
- `System.out.print("\nHello ");`: Mencetak newline (`\n`) terlebih dahulu, kemudian mencetak "Hello ". Ini akan memindahkan kursor ke baris baru sebelum mencetak "Hello ".
- `System.out.println("World");`: Mencetak string "World" ke konsol dan secara otomatis menambahkan newline setelahnya. Ini berarti kursor akan pindah ke baris berikutnya setelah mencetak "World".
- `System.out.println("Welcome");`: Mencetak string "Welcome" ke konsol dan juga menambahkan newline setelahnya.
- program ini mendemonstrasikan cara dasar untuk menampilkan teks di konsol menggunakan Java, serta perbedaan antara metode `print` dan `println`

Output:

```
D:\LATIHAN\semester4>javac nomor10.java

D:\LATIHAN\semester4>java nomor10
Hello
Hello World
Welcome
```

11. Penjelasan

- untuk memahami perbedaan antara operator post-increment (`i++`) dan pre-increment (`++i`):
- `i++` mengembalikan nilai `i` sebelum increment, sedangkan `++i` mengembalikan nilai `i` setelah increment.

Output:

```
D:\LATIHAN\semester4>javac nomor11.java

D:\LATIHAN\semester4>java nomor11
Nilai i : 5
Nilai j : 3

D:\LATIHAN\semester4>
```

12. penjelsan

- Kode ini menunjukkan penggunaan operator bitwise dalam Java, termasuk AND (`&`), negasi (`~`), shift left (`<<`), dan shift right (`>>`).

Output:

```
D:\LATIHAN\semester4>javac nomor12.java

D:\LATIHAN\semester4>java nomor12
n = 10
x = 1
y = 2
n & 8 = 8
x & ~ 8 = 1
y << 2 = 8
y >> 3 = 0

D:\LATIHAN\semester4>
```

13. Penjelasan

- Kode ini menunjukkan penggunaan operator bitwise dalam Java, termasuk AND (`&`), OR (`|`), XOR (`^`), dan negasi (`~`).
- Penting untuk diingat bahwa operator `^` dalam Java bukan untuk pangkat, tetapi untuk operasi XOR.
- Metode `Math.pow()` digunakan untuk melakukan operasi pemangkatan, yang berbeda dari operasi bitwise.

Output:

```
Command Prompt
D:\LATIHAN\semester4>javac nomor13.java

D:\LATIHAN\semester4>java nomor13
i = 3
j =
i & j = 0
i | j = 7
i ^ j = 7
81.0
~i = -4
```

14. Penjelsan

- Kode ini menunjukkan penggunaan operator logika (`&&`, `||`) dan operator bitwise (`&`, `|`) dalam Java.
- Operator logika (`&&`, `||`) digunakan untuk evaluasi kondisi boolean, sedangkan operator bitwise (`&`, `|`) digunakan untuk operasi bitwise pada bilangan bulat, tetapi dapat juga digunakan dalam konteks boolean.
- operator `&&` dan `||` adalah operator pendek (short-circuit), yang berarti jika operand pertama sudah cukup untuk menentukan hasil, operand kedua tidak akan dievaluasi. Sebaliknya, operator `&` dan `|` akan selalu mengevaluasi kedua operand.

Output:

```
D:\LATIHAN\semester4>javac nomor14.java

D:\LATIHAN\semester4>java nomor14
true
false
true
true
true
```

15. penjelasan

- ntuk memahami bagaimana operator ternary bekerja dan bagaimana post-increment (`i++`) mempengaruhi nilai yang dibandingkan.
- Kode ini juga menunjukkan bagaimana karakter dapat dibandingkan setelah dikonversi ke tipe integer.

Output:


```
D:\LATIHAN\semester4>javac nomor15.java

D:\LATIHAN\semester4>java nomor15
Nilai e = 10
Nilai k = 0
Nilai k = 4
D:\LATIHAN\semester4>
```

16. Penjelasan

- Operasi bilangan
`&&` (AND), `||` (OR), `!` (NOT), dan `^` (XOR) digunakan untuk operasi logika.
- operasi Relasional
 Menggunakan operator seperti `==`, `!=`, `<`, `>`, `<=`, dan `>=` untuk membandingkan nilai dan menghasilkan boolean.
- operasi numerik
 Operasi dasar seperti penjumlahan (`+`), pengurangan (`-`), pembagian (`/`), perkalian (`*`), dan modulo (`%`) digunakan untuk tipe data `int` dan `float`.

Output:

```
Command Prompt
D:\LATIHAN\semester4>javac nomor16.java

D:\LATIHAN\semester4>java nomor16
Silahkan baca teksnya dan tambahkan perintah untuk menampilkan output

D:\LATIHAN\semester4>
```