

Tugas 1

Makassar, 21 Februari 2025

**PRAKTIKUM
PEMROGRAMAN BERBASIS OBJEK**



Nama : Rofifah Gina Tamala
Stambuk : 13020230091
Dosen : Mardiyah Hasnawi, S.Kom.,MT.

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

Tugas 1 – Pemrograman Berorientasi Objek

1. Class Asgdll

Output:

```
D:\Tugas PBO\Tugas 1>java Asgdll
f : 20.0
f11: 10.0

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

Keterangan:

Class dengan nama Asgdll untuk mencetaak nilai dari dua variable dengan tipe data float dan double. Deklarasi variable f= 20.0 tipe data float dan variable fl1 dengan tipe data double tanpa nilai awal. Kemudian fl1 diberi nilai 10.0 maka output yang keluar System.out.println nilai dari f dan fl1.

2. Class Asign

Output:

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

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

Keterangan:

Class dengan nama Asign untuk mencetak teks ke layer dengan deklarasi variable I tipe data int tanpa nilai awal. Kemudian output System.out.print mencetak "hello" lalu pindah baris. i=5; memberikan nilai 5 ke variable i. System.out.println akan mencetak "ini nilai i : 5".

3. Class ASIGNi

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 = 50.200000076293945
Bilangan Real y = 50.2
```

Keterangan:

Class dengan nama ASIGNi untuk mendeklarasikan beberapa variable dengan berbagai tipe data short, int, long, char, double, float. short ks=1;, int ki= 1; , long kl=10000; char c=65; karakter yang diinisialisasi dengan nilai 'A', char c1=Z karakter diinisialisasi langsung "Z". double x= 50.2f, float y=50.2f.

4. Class BacaData

Output:

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

D:\Tugas PBO\Tugas 1>java BacaData
Contoh membaca dan menulis, ketik nilai integer:
2
Nilai yang dibaca : 2
```

Keterangan:

Class dengan nama BacaData dengan deklarasi variable a tipe data integer untuk menyimpan inputan dan variable Scanner masukkan untuk objek Scanner. Kemudian menampilkan output System.out.print ("Nilai yang dibaca:"+a); yang menampilkan angka yang dimasukkan. Jika ada dua tanda titik koma (masukkan.nextInt(); ;), akan terjadi ERROR sintaks.

5. Class Bacakar

Output:

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

D:\Tugas PBO\Tugas 1>java Bacakar
hello
baca 1 karakter : e
baca 1 bilangan : 2
e
2
bye
```

Keterangan:

Class Bacakar menggunakan `BufferedReader` untuk membaca inputan. Dengan `BufferedReader` yang membaca satu karakter dan menyimpannya dalam variable `cc`, objek `dataIn` membaca bilangan integer dan menyimpannya dalam variable `bil`. Output akan membaca karakter dan bilangan yang dimasukkan sebelum menampilkan "bye". Jika inputan tidak sesuai tipe data akan ERROR.

6. Class Casting1

Output:

```
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
```

Keterangan:

Class Casting1 dengan deklarasi variable `a` dan `b` bertipe integer dikonversi ke float dan double, sedangkan `d` dan `e` bertipe float dikonversi

ke integer dan double. Kemudian g dikonversi ke int,float dan double. Yang menghasilkan ASCII dari '5' yaitu 53. Variable k bertipe double dikonversi ke int untuk menghapus desimalnya dan float mempertahankan desimalnya.

7. Class Casting2

Output:

```
D:\Tugas PBO\Tugas 1>javac Casting2.java

D:\Tugas PBO\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
```

Keterangan:

Class Casting2 dengan deklarasi variable n,m dan l bertipe string dikonversi ke int,double,float menggunakan Integer.parseInt(), Double.parseDouble() dan Float.parseFloat() lalu ditampilkan. Kemudian, variable b,g dan e dikonversi ke string menggunakan String.valueOf(). Kemudian, a dikonversikan ke Double lalu dikembalikan ke int menggunakan Double.valueOf(a).intValue(), sedangkan b dikonversi ke double menggunakan Integer.valueOf(b).doubleValue().

8. Class Ekspresi

Output:

```
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
```

Keterangan:

Class Ekspresi dengan deklarasi variable $x=1$ dan $y=2$ kemudian mencetak nilainya. Ekspresi $((x < y) ? x : y)$ digunakan untuk membaca apakah x lebih kecil dari y . jika benar ($x < y$) maka nilai x dikembalikan. Jika salah maka nilai y dikembalikan karena $1 < 2$ bernilai true. Menghasilkan nilai 1 dalam output.

9. Class Ekspresi1

Output:

```
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
```

Keterangan:

Class Ekspresi1 dengan deklarasi variable $x=1$ dan $y=2$, serta fx dan fy bertipe float. saat membagi x/y dalam format integer, hasilnya 0 karena pembagian antara dua bilangan bulat akan dibulatkan kebawah. Setelah x dan y dikonversi ke float ($fx = x$; $fy = y$). Ketika x dan y diubah menjadi 10 dan 3 hasil pembagian int tetap dibulatkan ($10/3=3$), sedangkan dalam float hasilnya lebih akurat menjadi 3.3333.

10. Class Hello

Output:

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

D:\Tugas PBO\Tugas 1>java

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

Keterangan:

Class Hello untuk menampilkan teks ke layar menggunakan System.out.print() dan System.out.println(). Baris pertama cetak "Hello", selanjutnya "World" otomatis pindah kebaris berikutnya dan diikuti oleh "Welcome" baris berikutnya.

11. Class Incr

Output:

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

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

Keterangan:

Class Incr dengan deklarasi variable i inisialilasi dengan 3, j=i++ menyimpan nilai i ke j sebelum i bertambah 1. Sehingga j tetap 3 dan i menjadi 4. Kemudian akan mencetak ++i, artinya, i dinaikkan terlebih dahulu menjadi 5 sebelum dicetak, sementara j tetap 3. Perbedaan antara **post-increment (i++)** dan **pre-increment (++i)** adalah bahwa post-increment mengembalikan nilai sebelum peningkatan, sedangkan pre-increment mengembalikan nilai setelah peningkatan.

12. Class Oprator

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 && Bool2: false
Bool1 || Bool2: true
!Bool1: false
Bool1 ^ 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: 25.0
x / y: 1.0
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
```

Keterangan:

Class Oper1 dengan variable n=10 biner 1010, x=1 biner 0001 dan y=2 biner 0010. n&8 melakukan operasi AND antara 1010 dan 1000 hasilnya 8. x&~8 melakukan operasi AND dengan 8 hasilnya 1. Operasi (y<<2) left shift menggeser bit y dua posisi ke kiri hasilnya 8. Sedangkan (y>>3) right shift menggeser bit y tiga posisi ke kanan hasilnya 0. Operator bitwise sering digunakan dalam manipulasi bit-level.

13. Class Oper2

Output:

```
D:\Tugas PB0\Tugas 1>javac Oper2.java

D:\Tugas PB0\Tugas 1>java Oper2
i = 3
j =
i & j = 0
i | j = 7
i ^ j = 7
81.0
~i = -4
```

Keterangan:

Class Oper2 dengan operasi bitwise dalam java dengan variable bertipe data char yang dikonverikan ke integer. Variable i =3 dan j=4. i&j AND hasilnya 0. i|j OR hasilnya 7. Dan i^j XOR hasilnya 7.

14. Class Oper3

Output:

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

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

Keterangan:

Class Oper3 menggunakan operator logika dan bitwise logika dalam java. Pernyataan pertama mencetak true karena true&true. Kedua menggunakan & namun memiliki kesalahan karena System.out.println(true&false); hasilnya false. Pernyataan ketiga hasilnya true karena kondisi if(true). Pernyataan keempat menggunakan logika ||

artinya mencetak setelah menemukan true hasilnya true. Pernyataan terakhir | selalu mengevaluasi kedua operand dan true|false tetap true. Perbedaan utama antara &&/|| dan &/| adalah bahwa && dan || memiliki short-circuit evaluation, sedangkan & dan | tidak.

15. Class Oper4

Output:

```
D:\Tugas PBO\Tugas 1>javac Oper4.java

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

Keterangan:

Class Oper4 menggunakan operator ternary(?:) dalam java untuk menentukan nilai berdasarkan kondisi tertentu. Deklarasi variable c=8 dan d=10 dikonversi ke int. lalu ekspresi ((int)c > (int)d) ? c : d mengevaluasi nilai terbesar, sehingga e = 10. Selanjutnya, k = ((i > j) ? i : j) memilih nilai terbesar antara i = 0 dan j = 0, sehingga k = 0. Setelah i diubah menjadi 2 dan j menjadi 3, ekspresi ((i++ > j++) ? i : j) mengevaluasi 2 > 3 (false), sehingga nilai j yang telah bertambah menjadi 4 dikembalikan ke k.

16. Class Oprator

Output:

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

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

Keterangan:

Class Operator menggunakan operator logika, aritmatika dan relasional dalam java. Operator logika untuk variable Boolean && (AND), || (OR), ! (NOT), dan ^ (XOR). Operator aritmetika untuk variabel integer i,j dan float x,y penjumlahan, pengurangan, perkalian, pembagian, dan modulus untuk mendapatkan sisa baginya. Operator relasional untuk membandingkan nilai numerik seperti ==, !=, <, >, <= dan >=. Namun, program ini tidak mencetak output karena tidak ada perintah `System.out.println()` untuk menampilkan hasil perhitungan.