

# Laporan Tugas 1 (Tugas Praktek)

Nama : Nur Ikhwan Alfiansyah

Nim : 13020220131

Kelas : A4 – TI

## 1. Program Asgdll

```
C:\Users\ASUS\Documents\UMI\SEMESTER 4\PBO\Tugas 1\Tugas1_SourceCode_Praktek>javac Asgdll.java  
C:\Users\ASUS\Documents\UMI\SEMESTER 4\PBO\Tugas 1\Tugas1_SourceCode_Praktek>java Asgdll  
f : 20.0  
f11: 10.0
```

Penjelasan:

Program Java ini bertujuan untuk menunjukkan perbedaan antara tipe data float(f) dan double(fil) dalam Java

- **float f = 20.0f;** Mendeklarasikan variabel **f** dengan tipe data float dan memberinya nilai 20.0.
- **double fil;** Mendeklarasikan variabel **fil** dengan tipe data double.
- **fil = 10.0f;** Menginisialisasi variabel **fil** dengan nilai 10.0 (secara implisit dikonversi menjadi double).
- **System.out.println("f : " + f + "\nf11: " + fil);** Mencetak nilai variabel **f** dan **fil**.

## 2. Program Assign

```
C:\Users\ASUS\Documents\UMI\SEMESTER 4\PBO\Tugas 1\Tugas1_SourceCode_Praktek>javac Assign.java  
C:\Users\ASUS\Documents\UMI\SEMESTER 4\PBO\Tugas 1\Tugas1_SourceCode_Praktek>java Assign  
hello  
Ini nilai i : 5
```

Penjelasan:

Program Java ini bertujuan untuk menunjukkan penggunaan variabel dan mencetak nilai variabel ke konsol.

- **public class Assign:** Mendefinisikan kelas Java dengan nama “Assign”.
- **public static void main(String[] args):** Metode utama yang dieksekusi oleh JVM (Java Virtual Machine) saat program dijalankan.

- **int i;** Mendeklarasikan variabel **i** dengan tipe data **int**.
- **System.out.print("hello\n");** Mencetak "hello" ke konsol tanpa mengganti baris.
- **i = 5;** Memberikan nilai 5 ke variabel **i**.
- **System.out.println("Ini nilai i : " + i);** Mencetak nilai variabel **i** dengan pesan "Ini nilai i : " ke konsol, diikuti oleh nilai variabel **i**. **println** digunakan untuk mencetak teks dan nilai variabel dalam baris baru.

### 3. Program ASIGNi

```
C:\Users\ASUS\Documents\UMI\SEMESTER 4\PBO\Tugas 1\Tugas1_SourceCode_Praktek>javac ASIGNi.java

C:\Users\ASUS\Documents\UMI\SEMESTER 4\PBO\Tugas 1\Tugas1_SourceCode_Praktek>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
```

Penjelasan:

Program Java ini bertujuan untuk menunjukkan penggunaan berbagai tipe data dan mencetak nilainya ke konsol.

- **public class ASIGNi:** Mendefinisikan kelas Java dengan nama "ASIGNi".
- **public static void main(String[] args):** Metode utama yang dieksekusi oleh JVM saat program dijalankan.
- **short ks = 1;, int ki = 1;, long kl = 10000;:** Mendeklarasikan variabel **ks** dengan tipe data **short**, **ki** dengan tipe data **int**, dan **kl** dengan tipe data **long**, dan memberi nilai awal ke masing-masing variabel.
- **char c = 65;, char c1 = 'Z';** Mendeklarasikan variabel **c** dan **c1** dengan tipe data **char**, dan memberi nilai awal 'A' dan 'Z'.
- **double x = 50.2f;, float y = 50.2f;:** Mendeklarasikan variabel **x** dengan tipe data **double** dan **y** dengan tipe data **float**, dan memberi nilai awal 50.2f.
- **System.out.println("Karakter = " + c);** Mencetak nilai variabel **c** ke konsol dengan pesan "Karakter = ".

- **System.out.println ("Bilangan integer (short) = "+ ks);** Mencetak nilai variabel **ks** ke konsol dengan pesan "Bilangan integer (short) = ".
- **System.out.println("\t(int) = "+ ki);** Mencetak nilai variabel **ki** ke konsol dengan pesan "(int) = ", menggunakan tabulasi untuk indentasi.
- **System.out.println("Bilangan Real x = "+ x);** Mencetak nilai variabel **x** ke konsol dengan pesan "Bilangan Real x = ".
- **System.out.println("Bilangan Real y = "+ y);** Mencetak nilai variabel **y** ke konsol dengan pesan "Bilangan Real y = ".

#### 4. Program BacaData

```
C:\Users\ASUS\Documents\UMI\SEMESTER 4\PBO\Tugas 1\Tugas1_SourceCode_Praktek>javac BacaData.java

C:\Users\ASUS\Documents\UMI\SEMESTER 4\PBO\Tugas 1\Tugas1_SourceCode_Praktek>java BacaData
Contoh membaca dan menulis, ketik nilai integer:
131
Nilai yang dibaca : 131
```

Penjelasan:

Program Java ini bertujuan untuk membaca input integer dari pengguna menggunakan kelas Scanner dan mencetaknya ke konsol.

- **import java.util.Scanner;** Mengimpor kelas Scanner untuk membaca input.
- **public class BacaData:** Mendefinisikan kelas Java dengan nama "BacaData".
- **public static void main(String[] args):** Metode utama.
- **int a;** Deklarasi variabel **a** untuk menyimpan nilai integer.
- **Scanner masukan;** Deklarasi variabel **masukan** untuk membaca input.
- **masukan = new Scanner(System.in);** Inisialisasi objek Scanner untuk membaca input dari konsol.
- **a = masukan.nextInt();** Mengambil input integer dari pengguna dan menyimpannya dalam variabel **a**.
- **System.out.print("Nilai yang dibaca : " + a);** Mencetak nilai yang dibaca dari pengguna ke konsol.

## 5. Program Bacakar

```
C:\Users\ASUS\Documents\UMI\SEMESTER 4\PBO\Tugas 1\Tugas1_SourceCode_Praktek>javac Bacakar.java

C:\Users\ASUS\Documents\UMI\SEMESTER 4\PBO\Tugas 1\Tugas1_SourceCode_Praktek>java Bacakar
hello
baca 1 karakter : f
baca 1 bilangan : 3
f
3
bye
```

Penjelasan:

Program Java ini bertujuan untuk membaca input karakter dan bilangan dari pengguna menggunakan kelas `BufferedReader` dan `InputStreamReader`, kemudian mencetaknya ke konsol.

- **Import `java.io.BufferedReader;`, `import java.io.IOException;`, `import java.io.InputStreamReader;`**: Mendeklarasikan impor kelas-kelas yang dibutuhkan untuk membaca input dari pengguna.
- **`public class Bacakar`**: Mendefinisikan kelas Java dengan nama “Bacakar”.
- **`public static void main(String[] args) throws IOException`**: Metode utama yang dapat melemparkan `IOException`. Ini berarti program dapat menghasilkan pengecualian `IOException` yang tidak ditangkap.
- **`char cc; int bil;`**: Mendeklarasikan variabel `cc` untuk menyimpan karakter dan `bil` untuk menyimpan bilangan bulat.
- **`InputStreamReader isr = new InputStreamReader(System.in);`**: Membuat objek `InputStreamReader` yang membaca input dari `System.in` (konsol).
- **`BufferedReader dataIn = new BufferedReader(isr);`**: Membuat objek `BufferedReader` untuk membaca input karakter menggunakan `InputStreamReader`.
- **`BufferedReader datAIn = new BufferedReader(new InputStreamReader(System.in));`**: Membuat objek `BufferedReader` lainnya untuk membaca input bilangan bulat secara langsung dari `System.in`.
- **`cc = dataIn.readLine().charAt(0);`**: Membaca satu baris input dari pengguna, kemudian mengambil karakter pertama dari baris tersebut.

- **bil = Integer.parseInt(datAIn.readLine());**: Membaca satu baris input dari pengguna sebagai string, kemudian mengonversinya menjadi bilangan bulat dengan menggunakan metode **parseInt** dari kelas Integer.
- **System.out.print(cc + "\n" + bil + "\n");**: Mencetak karakter dan bilangan yang dibaca dari pengguna ke konsol.
- **System.out.print("bye \n");**: Mencetak pesan "bye" ke konsol.

## 6. Program Casting1

```
C:\Users\ASUS\Documents\UMI\SEMESTER 4\PBO\Tugas 1\Tugas1_SourceCode_Praktek>javac Casting1.java
C:\Users\ASUS\Documents\UMI\SEMESTER 4\PBO\Tugas 1\Tugas1_SourceCode_Praktek>java Casting1
5.0
6.0
2
3.200000047683716
53
53.0
53.0
3
3.14
```

Penjelasan:

Program Java ini bertujuan untuk Menunjukkan penggunaan penukaran tipe data (casting) dalam Java.

- **public class Casting1**: Mendefinisikan kelas Java dengan nama “Casting1”.
- **public static void main(String[] args)**: Metode utama yang dieksekusi ketika program dimulai.
- **int a=5, b=6;**: Mendeklarasikan dua variabel bertipe data integer (**int**) dengan nama **a** dan **b** dan memberikan nilai awal **5** dan **6**.
- **float d=2.f, e=3.2f;**: Mendeklarasikan dua variabel bertipe data float (**float**) dengan nama **d** dan **e** dan memberikan nilai awal **2.f** dan **3.2f**.
- **char g='5';**: Mendeklarasikan variabel bertipe data karakter (**char**) dengan nama **g** dan memberikan nilai awal **'5'**.
- **double k=3.14;**: Mendeklarasikan variabel bertipe data double (**double**) dengan nama **k** dan memberikan nilai awal **3.14**.
- **System.out.println()**: Mencetak output ke konsol.

- **(float)a, (double)b, (int)d, (double)e, (int)g, (float)g, (double)g, (int)k, (float)k:**  
Contoh penukaran tipe data dengan melemparkan (casting) nilai dari satu tipe data ke tipe data lainnya. Misalnya, **(float)a** mengubah nilai variabel **a** yang bertipe data integer menjadi float.

## 7. Program Casting2

```
C:\Users\ASUS\Documents\UMI\SEMESTER 4\PBO\Tugas 1\Tugas1_SourceCode_Praktek>javac Casting2.java

C:\Users\ASUS\Documents\UMI\SEMESTER 4\PBO\Tugas 1\Tugas1_SourceCode_Praktek>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
```

Penjelasan:

Program Java ini bertujuan untuk menunjukkan penggunaan konsep *casting* antara tipe data primitif dan objek, serta penggunaan metode parsing untuk mengubah string menjadi tipe data primitif.

- **public class Casting2:** Mendefinisikan kelas Java dengan nama “Casting2”.
- **public static void main(String[] args):** Metode utama yang dieksekusi ketika program dimulai.
- Deklarasi dan inisialisasi variabel dengan berbagai tipe data, termasuk int, float, char, double, dan String.
- Menggunakan metode parsing untuk mengubah string menjadi tipe data primitif: **Integer.parseInt(n)**, **Double.parseDouble(m)**, **Float.parseFloat(l)**.
- Menggunakan metode **valueOf()** untuk mengubah nilai variabel menjadi string: **String.valueOf(b)**, **String.valueOf(g)**, **String.valueOf(e)**.

- Konversi tipe data antara tipe data primitif dan objek menggunakan metode **valueOf()** dan **intValue()** atau **doubleValue()**: **Double.valueOf(a).intValue()**, **Integer.valueOf(b).doubleValue()**.
- **System.out.println()**: Mengeluarkan output ke konsol, menunjukkan nilai-nilai variabel setelah dilakukan operasi *casting* dan konversi.

## 8. Program Ekspresi

```
C:\Users\ASUS\Documents\UMI\SEMESTER 4\PBO\Tugas 1\Tugas1_SourceCode_Praktek>javac Ekspresi.java

C:\Users\ASUS\Documents\UMI\SEMESTER 4\PBO\Tugas 1\Tugas1_SourceCode_Praktek>java Ekspresi
x = 1
y = 2
hasil ekspresi = (x<y)?x:y = 1
```

Penjelasan:

Program Java ini bertujuan untuk menunjukkan penggunaan operator ternary dalam ekspresi untuk membandingkan dua nilai dan mengembalikan nilai yang lebih kecil di antara keduanya.

- **public class Ekspresi**: Mendefinisikan kelas Java dengan nama "Ekspresi".
- **public static void main(String[] args)**: Ini adalah metode utama di mana eksekusi program dimulai. Parameter **args** adalah array string yang berisi argumen yang diberikan saat menjalankan program.
- **int x = 1; int y = 2;**: Mendeklarasikan dua variabel bertipe data integer, **x** dan **y**, dengan nilai masing-masing 1 dan 2.
- **System.out.print("x = " + x + "\n");**: Mencetak nilai dari variabel **x** ke konsol dengan format "x = nilai\_x".
- **System.out.print("y = " + y + "\n");**: Mencetak nilai dari variabel **y** ke konsol dengan format "y = nilai\_y".
- **System.out.print("hasil ekspresi = (x<y)?x:y = " + ((x < y) ? x : y));**: Ini adalah bagian penting dari program. Ini mencetak hasil dari ekspresi ternary **(x < y) ? x : y**. Ekspresi ini membandingkan nilai **x** dan **y**. Jika **x** kurang dari **y**, maka hasilnya adalah

nilai **x**, jika tidak, maka hasilnya adalah nilai **y**. Hasil dari ekspresi ini kemudian dicetak ke konsol.

## 9. Program Ekspresi1

```
C:\Users\ASUS\Documents\UMI\SEMESTER 4\PBO\Tugas 1\Tugas1_SourceCode_Praktek>javac Ekspresi1.java
C:\Users\ASUS\Documents\UMI\SEMESTER 4\PBO\Tugas 1\Tugas1_SourceCode_Praktek>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
```

Penjelasan:

Program Java ini bertujuan untuk menunjukkan perbedaan hasil operasi pembagian antara tipe data integer dan float dalam bahasa pemrograman Java.

- **public class Ekspresi1:** Mendefinisikan kelas Java dengan nama "Ekspresi1".
- **public static void main(String[] args):** Ini adalah metode utama di mana eksekusi program dimulai. Parameter **args** adalah array string yang berisi argumen yang diberikan saat menjalankan program.
- **int x = 1; int y = 2; float fx; float fy;** Mendeklarasikan empat variabel, **x** dan **y** sebagai integer, dan **fx** serta **fy** sebagai float.
- **System.out.print ("x/y (format integer) = "+ x/y);** Mencetak hasil dari operasi pembagian **x** oleh **y** dengan format integer. Perhatikan bahwa operasi ini dilakukan menggunakan tipe data integer, sehingga nanti hasilnya juga integer.
- **System.out.print ("\nx/y (format float) = "+ x/y);** Mencetak hasil dari operasi pembagian **x** oleh **y** dengan format float. Namun, karena **x** dan **y** adalah integer, hasilnya akan tetap integer.
- **fx=x; fy=y;** Mengkonversi nilai **x** dan **y** ke tipe data float dan disimpan dalam variabel **fx** dan **fy**.
- **System.out.print ("\nx/y (format integer) = "+ fx/fy);** Mencetak hasil dari operasi pembagian **fx** oleh **fy** dengan format integer. Karena **fx** dan **fy** adalah float, hasilnya akan sesuai dengan format float.



- **System.out.print ("\nx/y (format float) = "+ fx/fy);** Mencetak hasil dari operasi pembagian **fx** oleh **fy** dengan format float. Hasilnya juga float.
- **System.out.print("\nfloat(x)/float(y) (format integer)= "+ (float)x/(float)y);** Mencetak hasil dari operasi pembagian nilai float **x** oleh nilai float **y** dengan format integer.
- **System.out.print("\nfloat(x)/float(y) (format float) = "+ (float)x/(float)y);** Mencetak hasil dari operasi pembagian nilai float **x** oleh nilai float **y** dengan format float.
- **x = 10; y = 3;** Mengubah nilai **x** menjadi 10 dan **y** menjadi 3.
- **System.out.print ("\nx/y (format integer) = "+ x/y);** Mencetak hasil dari operasi pembagian **x** oleh **y** dengan format integer setelah perubahan nilai.
- **System.out.print ("\nx/y (format float = "+ x/y);** Mencetak hasil dari operasi pembagian **x** oleh **y** dengan format float setelah perubahan nilai.

## 10. Program PrintHello

```
C:\Users\ASUS\Documents\UMI\SEMESTER 4\PBO\Tugas 1\Tugas1_SourceCode_Praktek>javac PrintHello.java
C:\Users\ASUS\Documents\UMI\SEMESTER 4\PBO\Tugas 1\Tugas1_SourceCode_Praktek>java PrintHello
Hello
Hello World
Welcome
```

Penjelasan:

Program Java ini bertujuan untuk mencetak beberapa pesan ke konsol.

- **public class PrintHello:** Mendefinisikan kelas Java dengan nama "PrintHello".
- **public static void main(String[] args):** Metode utama dari program. Ini adalah tempat eksekusi program dimulai. Parameter **args** adalah argumen baris perintah yang dapat diberikan saat menjalankan program.
- **System.out.print("Hello");** Mencetak teks "Hello" ke konsol tanpa melakukan perubahan baris baru.

- **System.out.print("\nHello ");** Mencetak teks "Hello" ke konsol dengan melakukan perubahan baris baru sebelumnya. Penambahan spasi setelah "Hello" akan menghasilkan output dengan spasi setelah baris baru.
- **System.out.println("World");** Mencetak teks "World" ke konsol dengan melakukan perubahan baris baru setelahnya.
- **System.out.println("Welcome");** Mencetak teks "Welcome" ke konsol dengan melakukan perubahan baris baru setelahnya.

## 11. Program Incr

```
C:\Users\ASUS\Documents\UMI\SEMESTER 4\PBO\Tugas 1\Tugas1_SourceCode_Praktek>javac Incr.java

C:\Users\ASUS\Documents\UMI\SEMESTER 4\PBO\Tugas 1\Tugas1_SourceCode_Praktek>java Incr
Nilai i : 5
Nilai j : 3
```

Penjelasan:

Program Java ini bertujuan untuk menunjukkan perbedaan antara operator penambahan (++i) dan operator penambahan dengan penugasan (i++).

- **public class Incr:** Mendefinisikan kelas Java dengan nama "Incr".
- **public static void main(String[] args):** Metode utama dari program. Ini adalah tempat eksekusi program dimulai. Parameter **args** adalah argumen baris perintah yang dapat diberikan saat menjalankan program.
- **int i, j;** Mendeklarasikan dua variabel integer **i** dan **j**.
- **i = 3;** Menginisialisasi variabel **i** dengan nilai 3.
- **j = i++;** Menugaskan nilai dari **i** ke **j**, kemudian meningkatkan nilai **i** dengan 1. Karena operator penugasan (++ setelah variabel), nilai **j** akan memiliki nilai **i** sebelum penambahan.
- **System.out.println ("Nilai i : " + (++i) + "\nNilai j : " + j);** Mencetak nilai variabel **i** setelah peningkatan (++i) dan nilai variabel **j**. Perhatikan bahwa ++i menambahkan nilai **i** sebelum digunakan dalam pernyataan cetak, sedangkan **j** memegang nilai **i** sebelum peningkatan.

## 12. Program Oper1

```
C:\Users\ASUS\Documents\UMI\SEMESTER 4\PBO\Tugas 1\Tugas1_SourceCode_Praktek>javac Oper1.java

C:\Users\ASUS\Documents\UMI\SEMESTER 4\PBO\Tugas 1\Tugas1_SourceCode_Praktek>java Oper1
n = 10
x = 1
y = 2
n & 8 = 8
x & ~ 8 = 1
y << 2 = 8
y >> 3 = 0
```

Penjelasan:

Program Java ini bertujuan untuk mendemonstrasikan penggunaan operator bitwise pada bilangan bulat. Program ini mengilustrasikan penggunaan operator bitwise seperti AND (&), NOT (~), dan shift kiri (<<) serta shift kanan (>>), untuk memanipulasi nilai variabel-variabel integer.

- **public class Oper1:** Mendefinisikan kelas Java dengan nama "Oper1".
- **public static void main(String[] args):** Metode utama dari program. Ini adalah tempat eksekusi program dimulai. Parameter **args** adalah argumen baris perintah yang dapat diberikan saat menjalankan program.
- **int n = 10; int x = 1; int y = 2;** Mendeklarasikan tiga variabel bertipe data integer, yaitu **n**, **x**, dan **y**, dengan nilai masing-masing 10, 1, dan 2.
- **System.out.println ("n = "+ n);** Mencetak nilai dari variabel **n**.
- **System.out.println ("x = "+ x);** Mencetak nilai dari variabel **x**.
- **System.out.println ("y = "+ y);** Mencetak nilai dari variabel **y**.
- **System.out.println("n & 8 = "+ (n & 8));** Mencetak hasil operasi bitwise AND antara nilai **n** (1010) dan 8 (1000).
- **System.out.println ("x & ~ 8 = "+ (x & ~8));** Mencetak hasil operasi bitwise AND antara nilai **x** (1) dan komplement dari 8 (0111).
- **System.out.println ("y << 2 = "+ (y << 2));** Mencetak hasil operasi bitwise shift kiri pada nilai **y** (10) sebanyak 2 bit.
- **System.out.println ("y >> 3 = "+ (y >> 3));** Mencetak hasil operasi bitwise shift kanan pada nilai **y** (10) sebanyak 3 bit.

### 13. Program Oper2

```
C:\Users\ASUS\Documents\UMI\SEMESTER 4\PBO\Tugas 1\Tugas1_SourceCode_Praktek>javac Oper2.java  
C:\Users\ASUS\Documents\UMI\SEMESTER 4\PBO\Tugas 1\Tugas1_SourceCode_Praktek>java Oper2  
i = 3  
j =  
i & j = 0  
i | j = 7  
i ^ j = 7  
81.0  
~i = -4
```

Penjelasan:

Program Java ini bertujuan untuk mendemonstrasikan penggunaan operator bitwise dan beberapa operasi matematika dasar dalam bahasa Java. Program ini mengilustrasikan penggunaan operator bitwise seperti AND (&), OR (|), XOR (^), dan NOT (~), serta menggunakan fungsi matematika dasar seperti pangkat dengan menggunakan kelas **Math**.

- **public class Oper2:** Mendefinisikan kelas Java dengan nama "Oper2".
- **public static void main(String[] args):** Metode utama dari program. Ini adalah tempat eksekusi program dimulai. Parameter **args** adalah argumen baris perintah yang dapat diberikan saat menjalankan program.
- **char i, j;** Mendeklarasikan dua variabel bertipe data char, yaitu **i** dan **j**.
- **i = 3; j = 4;** Menginisialisasi nilai **i** dengan 3 dan nilai **j** dengan 4.
- **System.out.println("i = " + (int) i);** Mencetak nilai dari variabel **i**. **(int)** digunakan untuk mengonversi nilai char ke dalam bentuk integer sebelum mencetak.
- **System.out.println("j = " + j);** Mencetak nilai dari variabel **j**.
- **System.out.println("i & j = " + (i & j));** Mencetak hasil operasi bitwise AND antara nilai **i** dan **j**.
- **System.out.println("i | j = " + (i | j));** Mencetak hasil operasi bitwise OR antara nilai **i** dan **j**.
- **System.out.println("i ^ j = " + (i ^ j));** Mencetak hasil operasi bitwise XOR antara nilai **i** dan **j**.
- **System.out.println(Math.pow(i, j));** Menggunakan fungsi **Math.pow()** untuk menghitung nilai dari **i** pangkat **j**.

- **System.out.println("~i = "+ ~i);**: Mencetak hasil operasi bitwise NOT dari nilai **i**.

#### 14. Program Oper3

```
C:\Users\ASUS\Documents\UMI\SEMESTER 4\PBO\Tugas 1\Tugas1_SourceCode_Praktek>javac Oper3.java
C:\Users\ASUS\Documents\UMI\SEMESTER 4\PBO\Tugas 1\Tugas1_SourceCode_Praktek>java Oper3
true
false
true
true
true
```

Penjelasan:

Program Java ini bertujuan untuk menunjukkan perbedaan antara operator logika "&&" dan "&", serta "||" dan "|".

- **public class Oper3**: Mendefinisikan kelas Java dengan nama "Oper3".
- **public static void main(String[] args)**: Metode utama dari program. Ini adalah tempat eksekusi program dimulai. Parameter **args** adalah argumen baris perintah yang dapat diberikan saat menjalankan program.
- **if (true && true)**: Menggunakan operator logika "&&" untuk mengevaluasi hasil dari ekspresi "true && true". Karena kedua operan benar (true), hasilnya adalah true, dan kondisi if menjadi benar, sehingga kode di dalamnya dieksekusi. Mencetak hasil "true".
- **if (true & true)**: Menggunakan operator bitwise AND "&" untuk mengevaluasi hasil dari ekspresi "true & true". Meskipun kedua operan benar (true), hasilnya adalah true, namun ini adalah operasi bitwise, sehingga setiap bit dari nilai true akan diperhitungkan. Mencetak hasil "true".
- **if (true)**: Mencetak "true", karena kondisi **if** selalu dieksekusi jika ekspresi yang diberikan kepadanya bernilai true.
- **if (true || true)**: Menggunakan operator logika "||" untuk mengevaluasi hasil dari ekspresi "true || true". Karena setidaknya satu operan adalah benar (true), hasilnya adalah true, dan kondisi if menjadi benar, sehingga kode di dalamnya dieksekusi. Mencetak hasil "true".
- **if (true|false)**: Menggunakan operator bitwise OR "|" untuk mengevaluasi hasil dari ekspresi "true | false". Karena setidaknya satu operan adalah benar (true), hasilnya

adalah true, namun ini adalah operasi bitwise, sehingga setiap bit dari nilai true dan false akan diperhitungkan. Mencetak hasil "true".

## 15. Program Oper4

```
C:\Users\ASUS\Documents\UMI\SEMESTER 4\PBO\Tugas 1\Tugas1_SourceCode_Praktek>javac Oper4.java
C:\Users\ASUS\Documents\UMI\SEMESTER 4\PBO\Tugas 1\Tugas1_SourceCode_Praktek>java Oper4
Nilai e = 10
Nilai k = 0
Nilai k = 4
```

Penjelasan:

Program Java ini bertujuan untuk mendemonstrasikan penggunaan operator ternary dalam pemrograman. Program ini mengilustrasikan penggunaan operator ternary untuk membuat keputusan berdasarkan hasil evaluasi ekspresi kondisional.

- **public class Oper4:** Mendefinisikan kelas Java dengan nama "Oper4".
- **public static void main(String[] args):** Metode utama dari program. Tempat eksekusi program dimulai. Parameter **args** adalah argumen baris perintah yang dapat diberikan saat menjalankan program.
- **int i = 0; int j = 0; char c = 8; char d = 10;** Mendeklarasikan beberapa variabel dengan tipe data yang berbeda. **i** dan **j** adalah integer, sedangkan **c** dan **d** adalah char.
- **int e = (((int)c > (int)d) ? c : d);** Menggunakan operator ternary untuk membandingkan dua nilai char setelah dikonversi ke integer. Jika nilai dari **c** lebih besar dari **d**, maka nilai **e** akan menjadi **c**, jika tidak, nilai **e** akan menjadi **d**.
- **int k = ((i > j) ? i : j);** Menggunakan operator ternary untuk membandingkan dua nilai integer **i** dan **j**. Jika nilai **i** lebih besar dari **j**, maka nilai **k** akan menjadi **i**, jika tidak, nilai **k** akan menjadi **j**.
- **System.out.print ("Nilai e = "+ e);** Mencetak nilai dari variabel **e**.
- **System.out.print ("\nNilai k = "+ k);** Mencetak nilai dari variabel **k**.
- **i = 2; j = 3; k = ((i++>j++) ? i : j) ;** Mengubah nilai dari variabel **i** dan **j**, kemudian menggunakan operator ternary untuk membandingkan nilai **i** dan **j**. Pemakaian operator **++** menyebabkan peningkatan nilai variabel setelah operasi perbandingan dilakukan.

- **System.out.print ("\nNilai k = "+ k);**: Mencetak nilai dari variabel **k** setelah perubahan nilai **i** dan **j**.

## 16. Program Oprator

```
C:\Users\ASUS\Documents\UMI\SEMESTER 4\PBO\Tugas 1\Tugas1_SourceCode_Praktek>javac Oprator.java
C:\Users\ASUS\Documents\UMI\SEMESTER 4\PBO\Tugas 1\Tugas1_SourceCode_Praktek>java Oprator
Silahkan baca teksnya dan tambahkan perintah program di bawah ini untuk menampilkan output program
Hasil Boolean AND: false
Hasil Boolean OR: true
Hasil Boolean NOT: false
Hasil Boolean XOR: true
Hasil Penjumlahan: 7
Hasil Pengurangan: 3
Hasil Pembagian: 2
Hasil Perkalian: 10
Hasil Pembagian Bulat: 2
Hasil Sisa Modulo: 1
Hasil Penjumlahan Float: 10.0
Hasil Pengurangan Float: 0.0
Hasil Pembagian Float: 1.0
Hasil Perkalian Float: 25.0
Hasil Relasional ==: false
Hasil Relasional !=: true
Hasil Relasional <: false
Hasil Relasional >: true
Hasil Relasional <=: false
Hasil Relasional >=: true
Hasil Relasional Float !=: false
Hasil Relasional Float <: false
Hasil Relasional Float >: false
Hasil Relasional Float <=: true
Hasil Relasional Float >=: true
```

Penjelasan:

Program Java ini bertujuan untuk menunjukkan berbagai operasi logika, numerik, dan relasional dalam bahasa pemrograman Java. Program ini bertujuan untuk mendemonstrasikan berbagai operasi dasar dalam pemrograman Java serta cara penggunaannya.

- **public class Oprator:** Mendefinisikan kelas Java dengan nama "Oprator".
- **public static void main(String[] args):** Metode utama dari program. Tempat eksekusi program dimulai. Parameter **args** adalah argumen baris perintah yang dapat diberikan saat menjalankan program.
- **Deklarasi Variabel:** Berbagai variabel seperti **Bool1**, **Bool2**, **TF**, **i**, **j**, **hsl**, **x**, **y**, dan **res** dideklarasikan dengan tipe data yang sesuai.
- **Operasi Logika:** Dilakukan operasi logika seperti AND (**&&**), OR (**||**), NOT (**!**), dan XOR (**^**) antara variabel-variabel boolean, dengan hasilnya dicetak.
- **Operasi Numerik:** Dilakukan operasi matematika seperti penjumlahan, pengurangan, perkalian, dan pembagian antara variabel-variabel integer dan float, dengan hasilnya dicetak.
- **Operasi Relasional Numerik:** Dilakukan operasi relasional seperti kesetaraan (**==**), ketidaksetaraan (**!=**), kurang dari (**<**), lebih besar dari (**>**), kurang dari atau sama dengan

( $\leq$ ), dan lebih besar dari atau sama dengan ( $\geq$ ) antara variabel-variabel integer dan float, dengan hasilnya dicetak.