

Laporan Tugas Praktek

Nama : Sahrul Adiyanto

Nim : 13020220143

Kelas : A4

- Program Asigdll

```
C:\Tugas 1\Tugas1_SourceCode_Praktek>javac Asgdll.java
C:\Tugas 1\Tugas1_SourceCode_Praktek>java Asgdll
f : 20.0
f11: 10.0
```

Penjelasan : Program ini merupakan contoh sederhana dalam bahasa pemrograman Java yang menunjukkan penggunaan tipe data float dan double. Tujuan dari program ini adalah untuk menunjukkan konsep penggunaan tipe data floating point dalam Java.

public class Asgdll: Mendefinisikan kelas publik dengan nama Asgdll.

public static void main(String[] args): Metode utama yang dieksekusi saat program dijalankan.

float f= 20.0f;: Deklarasi dan inisialisasi variabel f dengan tipe data float.

double f11;: Deklarasi variabel f11 dengan tipe data double.

f11=10.0f;: Menginisialisasi variabel f11 dengan nilai float. Konversi tipe data secara otomatis terjadi karena float lebih kecil dari double.

System.out.println("f : "+f+ "\nf11: "+f11);: Mencetak nilai variabel f dan f11 ke konsol.

- Program Assign

```
C:\Tugas 1\Tugas1_SourceCode_Praktek>javac Assign.java
C:\Tugas 1\Tugas1_SourceCode_Praktek>java Assign
hello
Ini nilai i : 5
```

Penjelasan : Program ini juga merupakan contoh sederhana dalam bahasa pemrograman Java. Tujuannya adalah untuk menunjukkan penggunaan pernyataan dalam satu baris dan beberapa pernyataan pada baris yang sama.

public class Assign: Mendefinisikan kelas publik dengan nama Assign.

public static void main(String[] args): Metode utama yang dieksekusi saat program dijalankan.

int i;: Deklarasi variabel i dengan tipe data int.

System.out.print("hello\n"); i = 5;; Mencetak "hello" ke konsol tanpa newline, kemudian menginisialisasi variabel i dengan nilai 5 pada baris yang sama.

System.out.println("Ini nilai i : " + i);; Mencetak nilai variabel i ke konsol dengan pesan tambahan.

- Program ASIGNi

```
C:\Tugas 1\Tugas1_SourceCode_Praktek>javac ASIGNi.java

C:\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 ini bertujuan untuk menunjukkan penggunaan beberapa tipe data dan mencetak nilai-nilai variabel ke konsol.

public class ASIGNi: Mendefinisikan kelas publik dengan nama ASIGNi.

public static void main(String[] args): Metode utama yang dieksekusi saat program dijalankan.

short ks = 1; int ki = 1; long kl = 10000; char c = 65; char c1 = 'Z'; double x = 50.2f; float y = 50.2f;; Pendeklarasian dan inisialisasi beberapa variabel dengan tipe data yang berbeda.

System.out.println("Karakter = "+ c);; Mencetak nilai variabel c ke konsol sebagai karakter.

System.out.println ("Bilangan integer (short) = "+ ks);; Mencetak nilai variabel ks ke konsol sebagai bilangan integer short.

System.out.println("Bilangan Real x = "+ x);; Mencetak nilai variabel x ke konsol sebagai bilangan real.

System.out.println("\t(int) = "+ ki);; Mencetak nilai variabel ki ke konsol sebagai bilangan integer int dengan indentasi.

- Program BacaData

```
C:\Tugas 1\Tugas1_SourceCode_Praktek>javac BacaData.java

C:\Tugas 1\Tugas1_SourceCode_Praktek>java BacaData
Contoh membaca dan menulis, ketik nilai integer:
4
Nilai yang dibaca : 4
```

Penjelasan : Program ini menggunakan kelas Scanner untuk membaca input dari pengguna dan mencetaknya ke konsol.

`import java.util.Scanner;;` Mendeklarasikan impor dari pustaka utilitas Java yang mengandung kelas Scanner, yang akan digunakan untuk membaca input dari pengguna.

`public class BacaData:` Mendefinisikan kelas publik dengan nama BacaData.

`public static void main(String[] args):` Metode utama yang dieksekusi saat program dijalankan.

`int a; Scanner masukan;;` Mendeklarasikan variabel a sebagai integer dan variabel masukan sebagai objek Scanner.

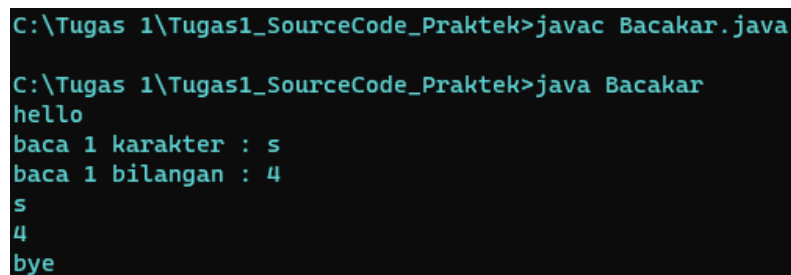
`System.out.print("Contoh membaca dan menulis, ketik nilai integer: \n");` Mencetak pesan instruksi ke konsol, meminta pengguna untuk memasukkan sebuah nilai integer.

`masukan = new Scanner(System.in);` Membuat objek Scanner baru dengan menggunakan input stream dari System.in, yang akan digunakan untuk membaca input dari pengguna melalui keyboard.

`a = masukan.nextInt();` Menggunakan objek Scanner untuk membaca nilai integer yang dimasukkan oleh pengguna dan menyimpannya dalam variabel a.

`System.out.print("Nilai yang dibaca : "+ a);` Mencetak nilai yang dibaca dari input pengguna ke konsol.

- Program Bacakar



```
C:\Tugas 1\Tugas1_SourceCode_Praktek>javac Bacakar.java
C:\Tugas 1\Tugas1_SourceCode_Praktek>java Bacakar
hello
baca 1 karakter : s
baca 1 bilangan : 4
s
4
bye
```

Penjelasan : Program ini membaca input dari pengguna menggunakan kelas `BufferedReader` dan `InputStreamReader`, kemudian mencetaknya ke konsol.

`import java.io.BufferedReader;` dan `import java.io.InputStreamReader;;` Mendeklarasikan impor dari pustaka Java yang mengandung kelas `BufferedReader` dan `InputStreamReader`, yang akan digunakan untuk membaca input dari pengguna melalui keyboard.

`public class Bacakar:` Mendefinisikan kelas publik dengan nama Bacakar.

`public static void main(String[] args) throws IOException:` Metode utama yang dieksekusi saat program dijalankan, dan menunjukkan bahwa program ini dapat melemparkan `IOException` yang mungkin terjadi saat membaca input.

`char cc; int bil;` Mendeklarasikan variabel cc sebagai char dan variabel bil sebagai int.

`InputStreamReader isr = new InputStreamReader(System.in); BufferedReader dataIn = new BufferedReader(isr);` Membuat objek `InputStreamReader` dan `BufferedReader` untuk membaca input dari pengguna.

`BufferedReader dataIn = new BufferedReader(new InputStreamReader(System.in));` Membuat objek `BufferedReader` untuk membaca input dari pengguna dengan lebih singkat.

`System.out.print("baca 1 karakter : ");` Mencetak pesan instruksi ke konsol, meminta pengguna untuk memasukkan sebuah karakter.

`cc = dataIn.readLine().charAt(0);` Membaca input karakter yang dimasukkan oleh pengguna dan menyimpannya dalam variabel `cc`.

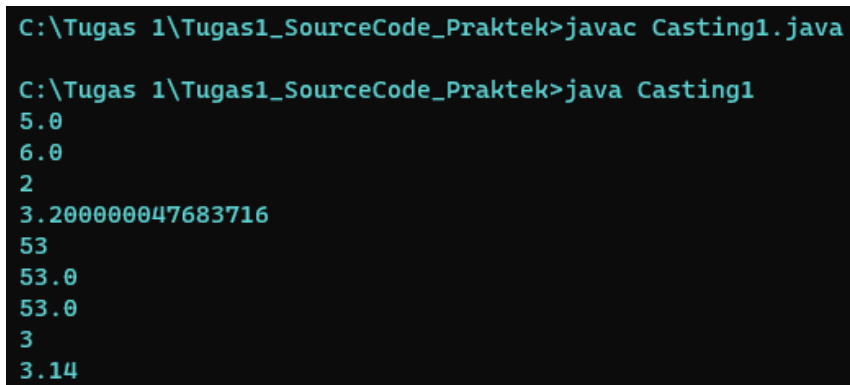
`System.out.print("baca 1 bilangan : ");` Mencetak pesan instruksi ke konsol, meminta pengguna untuk memasukkan sebuah bilangan.

`bil = Integer.parseInt(dataIn.readLine());` Membaca input bilangan yang dimasukkan oleh pengguna dan menyimpannya dalam variabel `bil` setelah mengonversi dari string menjadi integer.

`System.out.print(cc + "\n" + bil + "\n");` Mencetak karakter dan bilangan yang dibaca dari input pengguna ke konsol.

`System.out.print("bye \n");` Mencetak pesan "bye" ke konsol.

- Program Casting1



```
C:\Tugas 1\Tugas1_SourceCode_Praktek>javac Casting1.java
C:\Tugas 1\Tugas1_SourceCode_Praktek>java Casting1
5.0
6.0
2
3.2000000047683716
53
53.0
53.0
3
3.14
```

Penjelasan : Program ini menunjukkan penggunaan operasi casting untuk mengubah tipe data dari satu jenis ke jenis lainnya.

`public class Casting1:` Mendefinisikan kelas publik dengan nama `Casting1`.

`public static void main(String[] args):` Metode utama yang dieksekusi saat program dijalankan.

`int a=5, b=6; float d=2.f, e=3.2f; char g='5'; double k=3.14;` Deklarasi dan inisialisasi beberapa variabel dengan tipe data yang berbeda.

System.out.println((float)a);: Mencetak nilai variabel a yang telah di-cast ke tipe data float.
System.out.println((double)b);: Mencetak nilai variabel b yang telah di-cast ke tipe data double.

System.out.println((int)d);: Mencetak nilai variabel d yang telah di-cast ke tipe data int.
System.out.println((double)e);: Mencetak nilai variabel e yang telah di-cast ke tipe data double.

System.out.println((int)g);: Mencetak nilai variabel g yang merupakan karakter '5' yang telah di-cast ke tipe data int (mengambil nilai ASCII dari karakter).

System.out.println((float)g);: Mencetak nilai variabel g yang merupakan karakter '5' yang telah di-cast ke tipe data float (mengambil nilai ASCII dari karakter).

System.out.println((double)g);: Mencetak nilai variabel g yang merupakan karakter '5' yang telah di-cast ke tipe data double (mengambil nilai ASCII dari karakter).

System.out.println((int)k);: Mencetak nilai variabel k yang merupakan bilangan pecahan 3.14 yang telah di-cast ke tipe data int (pembulatan ke bawah).

System.out.println((float)k);: Mencetak nilai variabel k yang merupakan bilangan pecahan 3.14 yang telah di-cast ke tipe data float.

- Program Casting2

```
C:\Tugas 1\Tugas1_SourceCode_Praktek>javac Casting2.java

C:\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 : menunjukkan penggunaan operasi casting antara tipe data primitif dan tipe data objek.

public class Casting2: Mendefinisikan kelas publik dengan nama Casting2.

public static void main(String[] args): Metode utama yang dieksekusi saat program dijalankan.

int a=8,b=9; float d=2.f,e=3.2f; char g='5'; double k=3.14; String n="67",m="45",l="100";: Deklarasi dan inisialisasi beberapa variabel dengan tipe data yang berbeda.

`a = Integer.parseInt(n);`: Mengubah nilai variabel `n` yang bertipe `String` menjadi tipe data `int` menggunakan metode `parseInt` dari kelas `Integer`, kemudian menyimpannya dalam variabel `a`.

`k = Double.parseDouble(m);`: Mengubah nilai variabel `m` yang bertipe `String` menjadi tipe data `double` menggunakan metode `parseDouble` dari kelas `Double`, kemudian menyimpannya dalam variabel `k`.

`d = Float.parseFloat(l);`: Mengubah nilai variabel `l` yang bertipe `String` menjadi tipe data `float` menggunakan metode `parseFloat` dari kelas `Float`, kemudian menyimpannya dalam variabel `d`.

`n = String.valueOf(b);`: Mengubah nilai variabel `b` yang bertipe `int` menjadi tipe data `String` menggunakan metode `valueOf` dari kelas `String`, kemudian menyimpannya dalam variabel `n`.

`m = String.valueOf(g);`: Mengubah nilai variabel `g` yang bertipe `char` menjadi tipe data `String` menggunakan metode `valueOf` dari kelas `String`, kemudian menyimpannya dalam variabel `m`.

`l = String.valueOf(e);`: Mengubah nilai variabel `e` yang bertipe `float` menjadi tipe data `String` menggunakan metode `valueOf` dari kelas `String`, kemudian menyimpannya dalam variabel `l`.

`k = Double.valueOf(a).intValue();`: Mengubah nilai variabel `a` yang bertipe `int` menjadi tipe data `Double` menggunakan metode `valueOf` dari kelas `Double`, kemudian mengonversikannya ke tipe data `int` menggunakan metode `intValue()`, dan kemudian menyimpannya dalam variabel `k`.

`double c = Integer.valueOf(b).doubleValue();`: Mengubah nilai variabel `b` yang bertipe `int` menjadi tipe data `Integer` menggunakan metode `valueOf` dari kelas `Integer`, kemudian mengonversikannya ke tipe data `double` menggunakan metode `doubleValue()`, dan kemudian menyimpannya dalam variabel `c`.

`System.out.println("a : "+a+"\nk : "+k+"\nd : "+d);`: Mencetak nilai variabel `a`, `k`, dan `d` ke konsol.

`System.out.println("n : "+n+"\nm : "+m+"\nl : "+l);`: Mencetak nilai variabel `n`, `m`, dan `l` ke konsol.

`System.out.println("k : "+k+"\nc : "+c+"\nl : "+l);`: Mencetak nilai variabel `k`, `c`, dan `l` ke konsol.

- Program Ekespresi

```
C:\Tugas 1\Tugas1_SourceCode_Praktek>javac Eksespresi.java

C:\Tugas 1\Tugas1_SourceCode_Praktek>java Eksespresi
x = 1
y = 2
hasil eksespresi = (x<y)?x:y = 1
```

Penjelasan : Program ini menunjukkan penggunaan operator ternary (?:) dalam eksespresi.

public class Eksespresi: Mendefinisikan kelas publik dengan nama Eksespresi.

public static void main(String[] args): Metode utama yang dieksekusi saat program dijalankan.

int x = 1; int y = 2;; Mendeklarasikan dan menginisialisasi variabel x dan y dengan nilai 1 dan 2.

System.out.print("x = "+ x + "\n");: Mencetak nilai variabel x ke konsol.

System.out.print("y = "+ y + "\n");: Mencetak nilai variabel y ke konsol.

System.out.print("hasil eksespresi = (x<y)?x:y = " + ((x < y) ? x : y));: Mencetak hasil dari eksespresi ternary (x<y)?x:y, yang akan mengembalikan nilai x jika x kurang dari y, dan mengembalikan nilai y jika tidak, ke konsol.

- Program Eksespresi1

```
C:\Tugas 1\Tugas1_SourceCode_Praktek>javac Eksespresi1.java

C:\Tugas 1\Tugas1_SourceCode_Praktek>java Eksespresi1
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 ini menunjukkan penggunaan operator pemangkatan dan pembagian dalam eksespresi, serta konversi tipe data.

public class Eksespresi1: Mendefinisikan kelas publik dengan nama Eksespresi1.

public static void main(String[] args): Metode utama yang dieksekusi saat program dijalankan.

int x = 1; int y = 2; float fx; float fy;; Mendeklarasikan variabel x dan y dengan tipe data int, dan variabel fx dan fy dengan tipe data float.

System.out.print("x/y (format integer) = "+ x/y);: Mencetak hasil pembagian x oleh y dalam format integer ke konsol. Perhatikan bahwa operasi ini menggunakan pembagian integer sehingga hasilnya juga integer.

`System.out.print("\nx/y (format float) = "+ x/y);` Mencetak hasil pembagian x oleh y dalam format float ke konsol. Perhatikan bahwa operasi ini juga menggunakan pembagian integer sehingga hasilnya masih integer.

`fx=x; fy=y;` Menginisialisasi variabel fx dan fy dengan nilai x dan y.

`System.out.print("\nx/y (format integer) = "+ fx/fy);` Mencetak hasil pembagian fx oleh fy dalam format integer ke konsol.

`System.out.print("\nx/y (format float) = "+ fx/fy);` Mencetak hasil pembagian fx oleh fy dalam format float ke konsol.

`System.out.print("\nfloat(x)/float(y) (format integer)= "+ (float)x/(float)y);` Mencetak hasil pembagian nilai float dari x oleh nilai float dari y dalam format integer ke konsol.

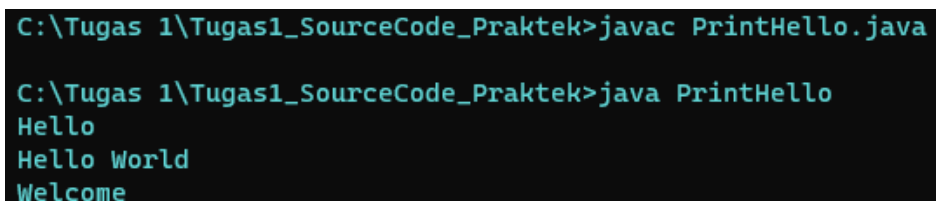
`System.out.print("\nfloat(x)/float(y) (format float) = "+ (float)x/(float)y);` Mencetak hasil pembagian nilai float dari x oleh nilai float dari y dalam format float ke konsol.

`x = 10; y = 3;` Mengubah nilai variabel x dan y.

`System.out.print("\nx/y (format integer) = "+ x/y);` Mencetak hasil pembagian x oleh y dalam format integer ke konsol.

`System.out.print("\nx/y (format float) = "+ x/y);` Mencetak hasil pembagian x oleh y dalam format float ke konsol.

- Program PrintHello



```
C:\Tugas 1\Tugas1_SourceCode_Praktek>javac PrintHello.java
C:\Tugas 1\Tugas1_SourceCode_Praktek>java PrintHello
Hello
Hello World
Welcome
```

Penjelasan : Program ini mencetak beberapa baris teks ke konsol.

`public class PrintHello:` Mendefinisikan kelas publik dengan nama PrintHello.

`public static void main(String[] args):` Metode utama yang dieksekusi saat program dijalankan.

`System.out.print("Hello");` Mencetak teks "Hello" ke konsol tanpa membuat baris baru.

`System.out.print("\nHello ");` Mencetak teks "Hello" ke konsol dengan membuat baris baru sebelumnya.

`System.out.println("World");` Mencetak teks "World" ke konsol diikuti oleh baris baru.

`System.out.println("Welcome");` Mencetak teks "Welcome" ke konsol diikuti oleh baris baru.

- Program Incr

```
C:\Tugas 1\Tugas1_SourceCode_Praktek>javac Incr.java  
  
C:\Tugas 1\Tugas1_SourceCode_Praktek>java Incr  
Nilai i : 5  
Nilai j : 3
```

Penjelasan : Program ini menunjukkan penggunaan operator penambahan (++), khususnya dalam konteks postfix dan prefix.

public class Incr: Mendefinisikan kelas publik dengan nama Incr.

public static void main(String[] args): Metode utama yang dieksekusi saat program dijalankan.

int i, j;: Mendeklarasikan dua variabel, i dan j, yang memiliki tipe data int.

i = 3;: Menginisialisasi variabel i dengan nilai 3.

j = i++;: Memberikan nilai variabel i ke variabel j, kemudian meningkatkan nilai variabel i dengan 1. Karena operator ++ berada setelah variabel (postfix), nilai j akan sama dengan nilai i sebelum penambahan.

System.out.println("Nilai i : " + (++i) + "\nNilai j : " + j);: Mencetak nilai variabel i yang sudah ditambah 1 (dengan operator ++ prefix) dan nilai variabel j ke konsol. Perhatikan bahwa nilai i di-preincrement sebelum digunakan dalam operasi pencetakan, sehingga nilai yang dicetak adalah 5 (nilai i yang sudah di-increment) sedangkan nilai j tetap 3 (nilai i sebelum di-increment).

- Program Oper1

```
C:\Tugas 1\Tugas1_SourceCode_Praktek>javac Oper1.java  
  
C:\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 ini menunjukkan penggunaan beberapa operator bitwise, seperti AND (&), NOT (~), dan shift left (<<), shift right (>>).

public class Oper1: Mendefinisikan kelas publik dengan nama Oper1.

public static void main(String[] args): Metode utama yang dieksekusi saat program dijalankan.

int n = 10; /* 1010 */: Mendeklarasikan variabel n dan menginisiasinya dengan nilai 10 dalam format biner (1010).

int x = 1; /* 1 */: Mendeklarasikan variabel x dan menginisiasinya dengan nilai 1.

`int y = 2; /* 10 */`: Mendeklarasikan variabel `y` dan menginisialisasinya dengan nilai 2 dalam format biner (10).

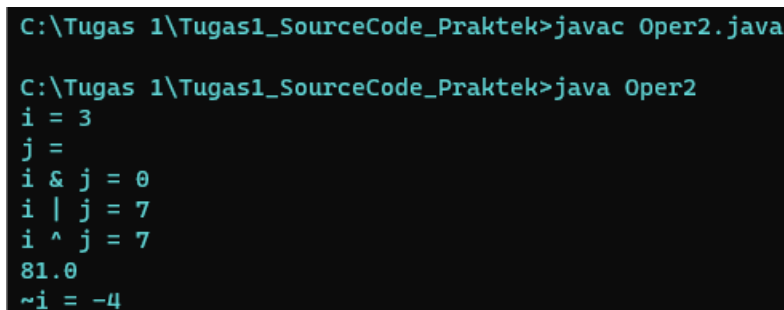
`System.out.println("n & 8 = "+ (n & 8)); /* 1010 AND 1000 */`: Melakukan operasi bitwise AND antara `n` (1010) dan 8 (1000), kemudian mencetak hasilnya ke konsol.

`System.out.println("x & ~8 = "+ (x & ~8)); /* 1 AND 0111 */`: Melakukan operasi bitwise AND antara `x` (1) dan komplement dari 8 (0111), kemudian mencetak hasilnya ke konsol.

`System.out.println("y << 2 = "+ (y << 2)); /* 10==> 1000 = 8 */`: Melakukan operasi shift left pada `y` (10) sebanyak 2 bit, kemudian mencetak hasilnya ke konsol.

`System.out.println("y >> 3 = "+ (y >> 3)); /* 10==>0000 = 0 */`: Melakukan operasi shift right pada `y` (10) sebanyak 3 bit, kemudian mencetak hasilnya ke konsol.

- Program Oper2



```
C:\Tugas 1\Tugas1_SourceCode_Praktek>javac Oper2.java

C:\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 ini menunjukkan perbedaan antara operator logika `&&` dan `&`, serta antara operator logika `||` dan `|`.

`public class Oper3`: Mendefinisikan kelas publik dengan nama `Oper3`.

`public static void main(String[] args)`: Metode utama yang dieksekusi saat program dijalankan.

`if (true && true) { System.out.println(true && true); }`: Melakukan pengecekan apakah kedua operand pada operator logika `&&` adalah `true`. Jika benar, maka hasilnya akan `true`. Pernyataan ini hanya mencetak `true` ke konsol jika kondisinya benar.

`if (true & true) { System.out.println(true & false); }`: Melakukan operasi bitwise AND antara `true` dan `true`. Hasilnya `false`, karena nilai `false` adalah 0 dalam representasi biner. Pernyataan ini mencetak `false` ke konsol.

`if (true) { System.out.println(true); }`: Mencetak `true` ke konsol, karena kondisinya selalu benar.

`if (true || true) { System.out.println(true); }`: Melakukan pengecekan apakah salah satu dari kedua operand pada operator logika `||` adalah `true`. Jika salah satunya `true`, maka hasilnya akan `true`. Pernyataan ini mencetak `true` ke konsol jika kondisinya benar.

`if (true|false) { System.out.println(true|false); }`: Melakukan operasi bitwise OR antara true dan false. Hasilnya true, karena nilai true adalah 1 dalam representasi biner. Pernyataan ini mencetak true ke konsol.

- Program Oper3

```
C:\Tugas 1\Tugas1_SourceCode_Praktek>javac Oper3.java  
  
C:\Tugas 1\Tugas1_SourceCode_Praktek>java Oper3  
true  
false  
true  
true  
true
```

Penjelasan : Program ini menunjukkan perbedaan antara operator logika `&&` dan `&`, serta antara operator logika `||` dan `|`.

`public class Oper3`: Mendefinisikan kelas publik dengan nama Oper3.

`public static void main(String[] args)`: Metode utama yang dieksekusi saat program dijalankan.

`if (true && true) { System.out.println(true && true); }`: Melakukan pengecekan apakah kedua operand pada operator logika `&&` adalah true. Jika benar, maka hasilnya akan true. Pernyataan ini hanya mencetak true ke konsol jika kondisinya benar.

`if (true & true) { System.out.println(true & false); }`: Melakukan operasi bitwise AND antara true dan true. Hasilnya true, karena nilai true adalah 1 dalam representasi biner. Pernyataan ini mencetak false ke konsol.

`if (true) { System.out.println(true); }`: Mencetak true ke konsol, karena kondisinya selalu benar.

`if (true || true) { System.out.println(true); }`: Melakukan pengecekan apakah salah satu dari kedua operand pada operator logika `||` adalah true. Jika salah satunya true, maka hasilnya akan true. Pernyataan ini mencetak true ke konsol jika kondisinya benar.

`if (true|false) { System.out.println(true|false); }`: Melakukan operasi bitwise OR antara true dan false. Hasilnya true, karena nilai true adalah 1 dalam representasi biner. Pernyataan ini mencetak true ke konsol.

- Program Oper4

```
C:\Tugas 1\Tugas1_SourceCode_Praktek>javac Oper4.java  
  
C:\Tugas 1\Tugas1_SourceCode_Praktek>java Oper4  
Nilai e = 10  
Nilai k = 0  
Nilai k = 4
```

Penjelasan : Program ini menunjukkan penggunaan operator ternary (?:) dan operator penambahan (++).

public class Oper4: Mendefinisikan kelas publik dengan nama Oper4.

public static void main(String[] args): Metode utama yang dieksekusi saat program dijalankan.

int i = 0; int j = 0;: Mendeklarasikan dua variabel i dan j dengan nilai awal 0.

char c = 8; char d = 10;: Mendeklarasikan dua variabel c dan d dengan nilai karakter 8 dan 10.

int e = (((int)c > (int)d) ? c: d);: Menggunakan operator ternary untuk membandingkan nilai dari c dan d. Jika nilai dari c lebih besar dari d (dalam representasi ASCII), maka nilai e akan sama dengan c, jika tidak, maka nilai e akan sama dengan d.

int k = ((i>j) ? i: j);: Menggunakan operator ternary untuk membandingkan nilai dari i dan j. Jika nilai dari i lebih besar dari j, maka nilai k akan sama dengan i, jika tidak, maka nilai k akan sama dengan j.

System.out.print("Nilai e = "+ e);: Mencetak nilai e ke konsol.

System.out.print("\nNilai k = "+ k);: Mencetak nilai k ke konsol.

i = 2; j = 3;: Mengubah nilai dari variabel i dan j menjadi 2 dan 3.

k = ((i++>j++) ? i: j);: Menggunakan operator ternary untuk membandingkan nilai dari i dan j setelah melakukan penambahan dengan postfix increment. Jika nilai dari i setelah penambahan lebih besar dari nilai j setelah penambahan, maka nilai k akan sama dengan i sebelum penambahan, jika tidak, maka nilai k akan sama dengan j sebelum penambahan.

System.out.print("\nNilai k = "+ k);: Mencetak nilai k setelah operasi increment ke konsol.

- Program Oprator

```
C:\Tugas 1\Tugas1_SourceCode_Praktek>javac Oprator.java
C:\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 : menunjukkan penggunaan berbagai operator, termasuk operator boolean dan operator numerik.

public class Operator: Mendefinisikan kelas publik dengan nama Operator.

public static void main(String[] args): Metode utama yang dieksekusi saat program dijalankan.

Mendeklarasikan beberapa variabel boolean (Bool1, Bool2, TF), variabel integer (i, j, hsl), dan variabel float (x, y, res).

Melakukan operasi boolean seperti AND (&&), OR (||), NOT (!), dan XOR (^) antara variabel boolean Bool1 dan Bool2, kemudian mencetak hasilnya ke konsol.

Melakukan operasi numerik seperti penjumlahan (+), pengurangan (-), pembagian (/), perkalian (*), pembagian bulat, dan sisa modulo (%) antara variabel integer i dan j, kemudian mencetak hasilnya ke konsol.

Melakukan operasi numerik seperti penjumlahan (+), pengurangan (-), pembagian (/), dan perkalian (*) antara variabel float x dan y, kemudian mencetak hasilnya ke konsol.

Melakukan operasi relasional numerik seperti ==, !=, <, >, <=, dan >= antara variabel integer i dan j, serta antara variabel float x dan y, kemudian mencetak hasilnya ke konsol.