

Nama & NPM	Topik:	Tanggal:
Dewi Margiani_G1F022037	Operator	08 September 2022

[Nomor 1] Identifikasi Masalah Operator Aritmatika

```

1 package operator;
2
3 public class latihan1 {
4     public static void main(String[] args) {
5         // deklarasi nilai
6         int a = 20, b = 3;
7         //operator aritmatika
8         System.out.println("a: " + a);
9         System.out.println("b: " + b);
10        System.out.println("a + b = " + (a + b));
11    }
12 }
13

```

```

<terminated> latihan1 [1] [Java Application] C:\Users\Lenovo\p2\pool\plugins\org.eclipse.justi.openjdk.hotspot.jre.full.win
a: 20
b: 3
a + b = 23

```

- 1.1. Tambahkan baris `System.out.println("a + b = " + (a + b));` Ubahlah operator (+) dengan tanda (-, *, / , %)

```

1 package operator;
2
3 public class lat1 {
4     public static void main(String[] args) {
5         // deklarasi nilai
6         int a = 20, b = 3;
7         //operator aritmatika
8         System.out.println("a: " + a);
9         System.out.println("b: " + b);
10        System.out.println("a + b = " + (a + b));
11        System.out.println("a - b = " + (a - b));
12        System.out.println("a * b = " + (a * b));
13        System.out.println("a / b = " + (a / b));
14        System.out.println("a % b = " + (a % b));
15    }
16 }
17

```

```

<terminated> lat1 [Java Application] C:\Users\aby\p2\pool\plugins\org.eclipse.justi.openjdk.hotspot.jre.full.win32.x86_64.17.0.4
a: 20
b: 3
a + b = 23
a - b = 17
a * b = 60
a / b = 6
a % b = 2

```

- 1.2. Analisa perhitungan matematika yang terjadi!

Pada operasi kali ini menggunakan Operator Binary dimana tanda (+) mengartikan operasi penjumlahan, tanda (-) mengartikan operasi pengurangan, tanda (*) mengartikan operasi perkalian, tanda (/) mengartikan operasi pembagian, tanda (%) mengartikan operasi sisa pembagian.

Operasi diatas dengan nilai a = 20 dan nilai b = 3,

a+b (20+3) artinya nilai a ditambah dengan nilai b dan mendapatkan hasil 23.

a-b (20-3) artinya nilai a dikurang dengan nilai b dan hasilnya adalah 17.

a*b (20x3) artinya nilai a dikalikan dengan nilai b dan mendapatkan hasil 60.

a/b (20/3) artinya nilai a dibagi dengan nilai b hasilnya adalah 6

dan yang terakhir adalah a%b artinya nilai a dibagi dengan nilai b namun yang dimaksud tanda % adalah sisa bagi, maka luaran nya adalah 2 yakni sisa dari pembagian 20 dibagi 3.

[Nomor 1] Penyusunan Algoritma

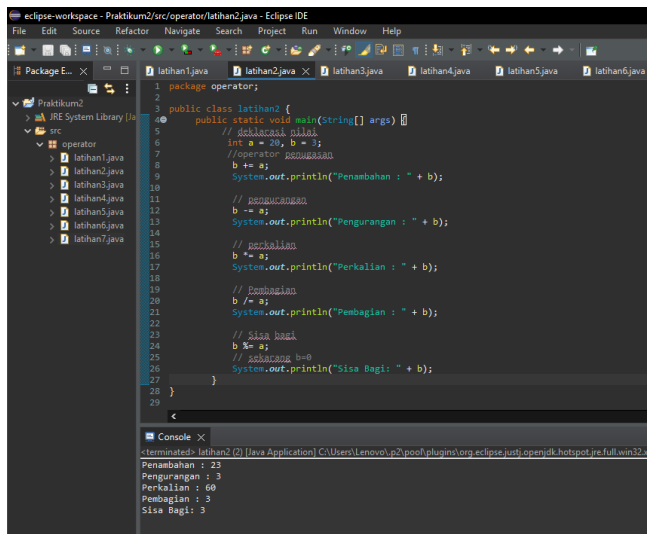
- a) Mulai
- b) Masukkan nilai a dan b
- c) Masukkan operasi seperti $a+b$
- d) Run, dan lihat hasilnya
- e) Selesai.

[Nomor 1] Kesimpulan

- 1) Analisa
Dari program diatas dapat disimpulkan bahwa operator binary terdiri dari $+$ (penjumlahan), $-$ (pengurangan), $*$ (perkalian) , $/$ (pembagian), dan $\%$ (sisa bagi).
- 2) Evaluasi
Program diatas, saya menjadi mengerti penggunaan operator binary pada suatu program.
- 3) Kreasi
Hal baru yang saya temukan adalah operator binary $\%$ (sisa bagi) yakni misal $7\%2$ maka sisa 1, penjelasannya adalah 2 dikali 3 hasilnya 6 dan itu mendekati angka 7 namun tersisa 1 maka angka 1 inilah yang akan menjadi luaran $7\%2$.

[Nomor 2] Identifikasi Masalah Operator Penugasan

2.1. Bandingkan hasil Contoh 1 dengan Contoh 2!



```
1 package operator;
2
3 public class latihan2 {
4     public static void main(String[] args) {
5         // deklarasi nilai
6         int a = 20, b = 3;
7         //operator penjumlahan
8         b += a;
9         System.out.println("Penambahan : " + b);
10
11         // pengurangan
12         b -= a;
13         System.out.println("Pengurangan : " + b);
14
15         // perkalian
16         b *= a;
17         System.out.println("Perkalian : " + b);
18
19         // pembagian
20         b /= a;
21         System.out.println("Pembagian : " + b);
22
23         // sisa bagi
24         b %= a;
25         System.out.println("Sisa Bagi: " + b);
26     }
27 }
28
29
```

Console Output:

```
<terminated> latihan2 (2) [Java Application] C:\Users\Lenovo\p2\pool\plugins\org.eclipse.justi.openjdk.hotspot.jre.full.win32.x86_64.jre\bin\java.exe
Penambahan : 23
Pengurangan : 3
Perkalian : 60
Pembagian : 3
Sisa Bagi: 3
```

Perbandingan dari Latihan 1 adalah pada

Latihan 2 ini tidak terdapat int hasil dan Latihan 1 menggunakan operand aritmatika sedangkan Latihan 2 menggunakan operand penugasan :

int a = 20, b = 3;

- operasi penjumlahan **b += a** atau **b = b+a** jadi **b=20+3** dan hasilnya **23**,
- nah pada operasi pengurangan **b -=a** atau **b=b-a** akan tetapi b nya bukan nilai **int b=3** akan tetapi nilai b dari hasil penjumlahan **b=20+3** tadi, jadi **b=23-20** sehingga memperoleh hasil **3**.
- Pada operasi perkalian **b *=a** atau **b=b*a** (nilai b berasal dari hasil operasi pengurangan), jadi **b=3*20** hasilnya **60**
- Pada operasi pembagian **b /=a** atau **b =b/a** (nilai b berasal dari hasil operasi perkalian), jadi **b=60/20** hasilnya **30**
- Pada operasi sisa bagi, **b %=a** atau **b=b%a** (nilai b berasal dari hasil operasi perkalian), jadi **b=30%20** hasilnya **3**.

[Nomor 2] Penyusunan Algoritma

- 1) Mulai
- 2) Masukkan nilai int a=20 , b=3
- 3) Masukkan operasi binary
- 4) Run
- 5) Lihat hasil luaran
- 6) Selesai

[Nomor 2] Kesimpulan

- 1) Analisa
Dari program diatas dapat disimpulkan bahwa operator binary terdiri dari +(penjumlahan), -(pengurangan), *(perkalian) ,/(pembagian), dan %(sisa bagi).
- 2) Evaluasi
Program diatas, saya menjadi mengerti perbedaan menuliskan **int hasil** dan yang tidak dituliskan **int hasil**.
- 3) Kreasi
Hal baru yang saya temukan adalah operasi binary tanpa memasukkan variable hasil. Contohnya int a=20, b=3 : melakukan operasi penjumlahan **b+=a** atau **b=3+20** maka Ketika variable b dipanggil hasilnya 23. Kemudian operasi pengurangan **b-=a** tetapi input

b bukan b=3 tapi input nilai b dari hasil operasi penjumlahan yakni b=23. Jadi b=a sama dengan b=23-20 maka hasilnya 3.

[Nomor 3] Identifikasi Masalah

Operator Relasional: <, >, <=, >=, =, ==, !=

```

package operator;

public class latihan3 {
    public static void main(String[] args) {
        int nilaiA = 12;
        int nilaiB = 4;
        boolean hasil;

        System.out.println("A = " + nilaiA + "\n B = " + nilaiB);
        // apakah A lebih besar dari B?
        hasil = nilaiA > nilaiB;
        System.out.println("Hasil A > B = " + hasil);

        // apakah A lebih kecil dari B?
        hasil = nilaiA < nilaiB;
        System.out.println("Hasil A < B = " + hasil);

        // apakah A lebih besar samadengan B?
        hasil = nilaiA >= nilaiB;
        System.out.println("Hasil A >= B = " + hasil);

        // apakah A lebih kecil samadengan B?
        hasil = nilaiA <= nilaiB;
        System.out.println("Hasil A <= B = " + hasil);

        // apakah nilai A sama dengan B?
        hasil = nilaiA == nilaiB;
        System.out.println("Hasil A == B = " + hasil);

        // apakah nilai A tidak samadengan B?
        hasil = nilaiA != nilaiB;
        System.out.println("Hasil A != B = " + hasil);
    }
}

```

```

package operator;

public class lat3 {
    public static void main(String[] args) {
        int nilaiA = 4;
        int nilaiB = 4;
        boolean hasil;

        System.out.println("A = " + nilaiA + "\n B = " + nilaiB);
        // apakah A lebih besar dari B?
        hasil = nilaiA > nilaiB;
        System.out.println("Hasil A > B = " + hasil);

        // apakah A lebih kecil dari B?
        hasil = nilaiA < nilaiB;
        System.out.println("Hasil A < B = " + hasil);

        // apakah A lebih besar samadengan B?
        hasil = nilaiA >= nilaiB;
        System.out.println("Hasil A >= B = " + hasil);

        // apakah A lebih kecil samadengan B?
        hasil = nilaiA <= nilaiB;
        System.out.println("Hasil A <= B = " + hasil);

        // apakah nilai A sama dengan B?
        hasil = nilaiA == nilaiB;
        System.out.println("Hasil A == B = " + hasil);

        // apakah nilai A tidak samadengan B?
        hasil = nilaiA != nilaiB;
        System.out.println("Hasil A != B = " + hasil);
    }
}

```

3.1. Ubahlah nilai A = 4 dan B = 4. Analisa perubahan yang terjadi!

```

package operator;

public class latihan3 {
    public static void main(String[] args) {
        int nilaiA = 4;
        int nilaiB = 4;
        boolean hasil;

        System.out.println("A = " + nilaiA + "\n B = " + nilaiB);
        // apakah A lebih besar dari B?
        hasil = nilaiA > nilaiB;
        System.out.println("Hasil A > B = " + hasil);

        // apakah A lebih kecil dari B?
        hasil = nilaiA < nilaiB;
        System.out.println("Hasil A < B = " + hasil);

        // apakah A lebih besar samadengan B?
        hasil = nilaiA >= nilaiB;
        System.out.println("Hasil A >= B = " + hasil);

        // apakah A lebih kecil samadengan B?
        hasil = nilaiA <= nilaiB;
        System.out.println("Hasil A <= B = " + hasil);

        // apakah nilai A sama dengan B?
        hasil = nilaiA == nilaiB;
        System.out.println("Hasil A == B = " + hasil);

        // apakah nilai A tidak samadengan B?
        hasil = nilaiA != nilaiB;
        System.out.println("Hasil A != B = " + hasil);
    }
}

```

3.2 Bandingkan bagaimana perbedaan nilai A dan B mempengaruhi nilai luaran!

Perubahan yang terjadi adalah pada hasil luaran :

- Hasil A > B = true berubah menjadi false (sebab nilai A tidak lebih besar dari nilai B)
- Hasil A < B = false berubah menjadi false (sebab nilai A tidak lebih kecil dari nilai B)
- Hasil A >= B = true berubah menjadi true (sebab nilai A tidak lebih besar sama dengan nilai dari B)
- Hasil A <= B = false berubah menjadi true (sebab nilai A tidak lebih kecil sama dengan nilai dari B)
- Hasil A == B = false berubah menjadi true (sebab nilai A sama besarnya dengan nilai B)
- Hasil A != B = true berubah menjadi false (sebab nilai A tidak lebih besar dari nilai B)

[Nomor 3] Penyusunan Algoritma

- 1) Mulai

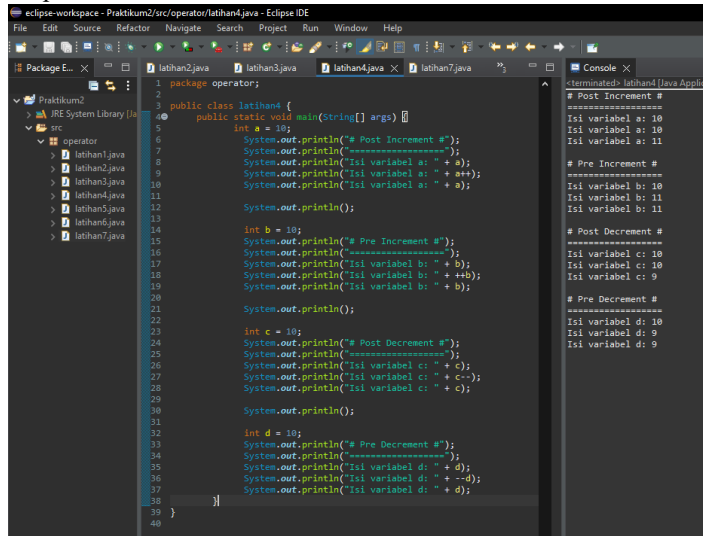
- 2) Masukkan int nilaiA dan nilaiB
- 3) Masukkan operasi penugasan
- 4) Lakukan Run, Lihat Luaran
- 5) Selesai

[Nomor 3] Kesimpulan

- 1) Analisa
Dari program diatas tipe data Boolean digunakan untuk menentukan true atau false.
- 2) Evaluasi
Untuk menghasilkan luaran true atau false membutuhkan suatu program misal dimana nilai $a = 4$, $b = 4$. Maka membutuhkan operasi relasional berikut **$a > b = + \text{hasil}$** ; luaran yang tampil adalah false karena nilai a tidak lebih besar dari nilai b.
- 3) Kesimpulan
Saya dapat memahami penggunaan Operator relasional ($<$, $>$, $<=$, $>=$, $=$, $==$, $!=$)

[Nomor 4] Identifikasi Masalah Operator Increment dan Decrement: ++, --

- Operator Increment dan Decrement digunakan pada operand bilangan bulat.
- Operator increment untuk menaikkan nilai variabel sebesar satu,
- Operator decrement untuk menurunkan nilai variabel sebesar satu



```
1 package operator;
2
3 public class latihan4 {
4     public static void main(String[] args) {
5         int a = 10;
6         System.out.println("# Post Increment #");
7         System.out.println("=====");
8         System.out.println("Isi variabel a: " + a);
9         System.out.println("Isi variabel a: " + ++a);
10        System.out.println("Isi variabel a: " + a);
11        System.out.println();
12
13        int b = 10;
14        System.out.println("# Pre Increment #");
15        System.out.println("=====");
16        System.out.println("Isi variabel b: " + b);
17        System.out.println("Isi variabel b: " + ++b);
18        System.out.println("Isi variabel b: " + b);
19        System.out.println("Isi variabel b: " + b);
20
21        System.out.println();
22
23        int c = 10;
24        System.out.println("# Post Decrement #");
25        System.out.println("=====");
26        System.out.println("Isi variabel c: " + c);
27        System.out.println("Isi variabel c: " + c--);
28        System.out.println("Isi variabel c: " + c);
29
30        System.out.println();
31
32        int d = 10;
33        System.out.println("# Pre Decrement #");
34        System.out.println("=====");
35        System.out.println("Isi variabel d: " + d);
36        System.out.println("Isi variabel d: " + --d);
37        System.out.println("Isi variabel d: " + d);
38    }
39 }
40 }
```

4.1. Berdasarkan luaran program Contoh 4, bandingkan hasil Post dan Pre untuk Increment dan Decrement!

Ciri yang paling mudah diingat adalah bila post tandanya selalu dibelakang variable sedangkan pre tanda selalu didepan variable.

- Post -Increment yakni `a++` artinya menampilkan nilai `int a=10` akan tetap tampil 10, namun setelah increment baru ditambah sebanyak 1 angka sehingga menjadi 11
- Pre – Increment berbeda dengan post , artinya nilai `int a= 10` langsung ditambah 1 sehingga menjadi 11 dan setelah di pre juga akan tetap sama tampil 11.
- Post – Decrement yakni `a--` artinya `int a= 10` akan tetap ditampilkan 10 baru setelahnya dikurangi 1 angkatan , sehingga luaran nya menjadi 9
- Pre – Decrement yakni `--a` artinya `int a=10` dikurangi 1 angka terlebih dahulu sehingga hasilnya 9 dan setelahnya pun tetap 9 yang akan di tampilkan

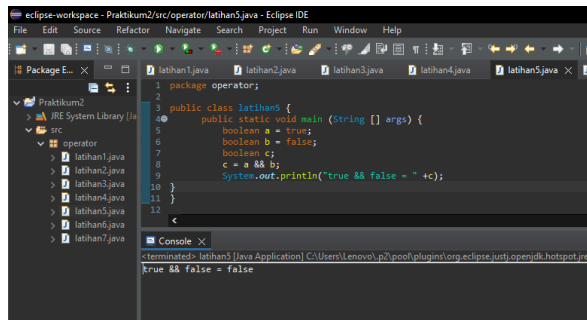
[Nomor 4] Penyusunan Algoritma

- 6) Mulai
- 7) Masukkan nilai `int a = 10`
- 8) Lakukan operasi post dan pre
- 9) Run
- 10) Selesai

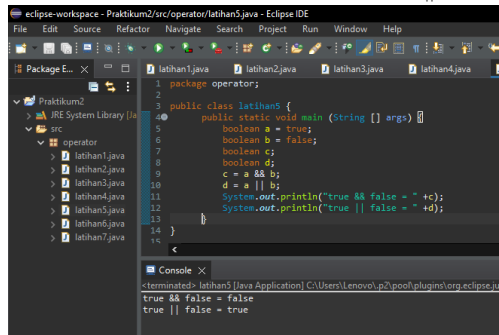
[Nomor 4] Kesimpulan

- 1) Analisa
Increment adalah penambahan . Decrement adalah pengurangan. Lalu Pre adalah yang tanda `(++,--)` di awal variable contohnya `++a` dan `--a` . Dan Post adalah yang tanda `(++,--)` diakhir variable contohnya `a++` dan `a--`.
- 2) Kesimpulan
Kesimpulannya adalah saya dapat memahami apa itu Pre dan Post increment, serta Pre dan Post decrement.

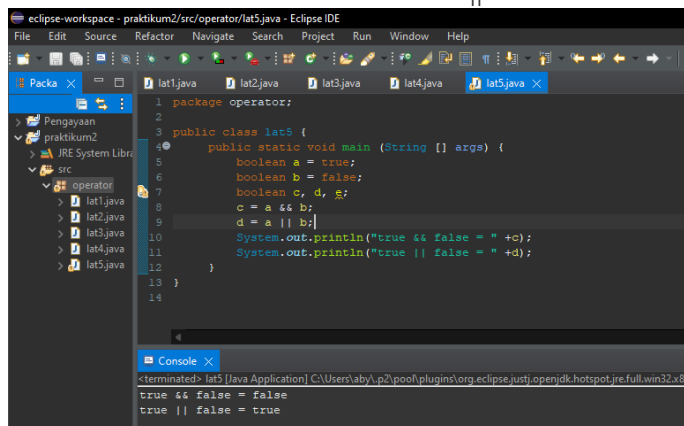
[Nomor 5] Identifikasi Masalah Operator Logika: &&, ||, !



5.1. Tambahkan baris kode untuk memeriksa a || b.

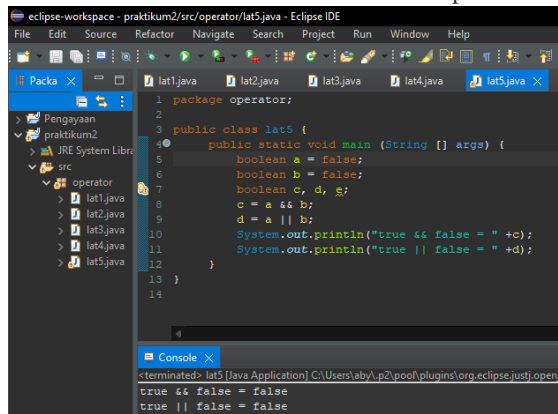


5.1. Tambahkan baris kode untuk memeriksa a || b.



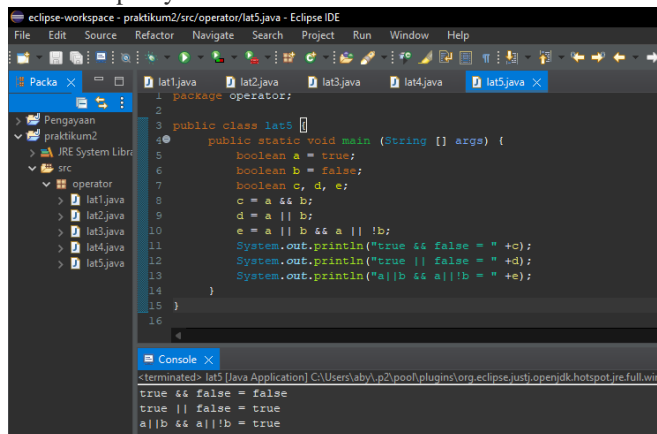
Pada AND (&&) akan menilai true apabila kedua operand bernilai true. Dan OR (||) akan menilai true apabila salah satu boolean bernilai true

5.2. Ubahlah nilai a = false dan b = false. Analisa perubahan dan perbedaan boolean yang terjadi!



Perubahan yang terjadi adalah hasil dari kedua operator AND dan OR adalah false. Sebab operator AND akan menghasilkan nilai true apabila keduanya bernilai true dan OR akan menghasilkan true bila salah satu atau kedua operand bernilai true.

5.3. Apabila diketahui pernyataan `a || b && a || !b`. Uraikan urutan logika yang akan dikerjakan! Analisa luaran true atau false dari pernyataan tersebut!



The screenshot shows the Eclipse IDE interface. The main editor displays a Java file named `lat5.java` with the following code:

```
1 package operator;
2
3 public class lat5 {
4     public static void main (String [] args) {
5         boolean a = true;
6         boolean b = false;
7         boolean c, d, e;
8         c = a && b;
9         d = a || b;
10        e = a || b && a || !b;
11        System.out.println("true && false = " + c);
12        System.out.println("true || false = " + d);
13        System.out.println("a || b && a || !b = " + e);
14    }
15 }
16
```

The console output at the bottom shows the results of the program execution:

```
<terminated> lat5 [Java Application] C:\Users\aby\p2\p00f\plugins\org.eclipse.justi.openjdk.hotspot.jre.full.win32
true && false = false
true || false = true
a || b && a || !b = true
```

Urutan operator logika Boolean diatas adalah :

(`a || b`) menghasilkan true. Sedangkan (`a || !b`) menghasilkan true. karena `a=true` , `b= false` tetapi disini dituliskan `!b` (NOT `b`) yang artinya kebalikan dari input atau kebalikan dari `b=false` yakni `b=true`. Jadi (`a || b && a || !b`) menghasilkan nilai true.

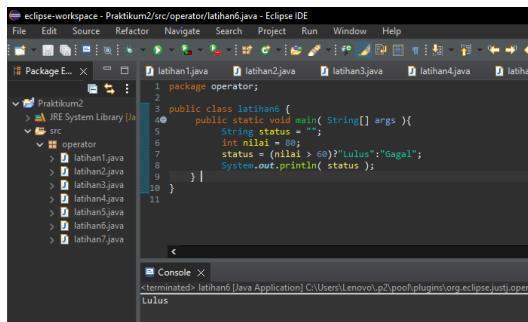
[Nomor 5] Penyusunan Algoritma

- 1) Mulai
- 2) Masukkan Boolean true dan false
- 3) Msukkan operasi operator AND, OR, XOR , dll.
- 4) Run
- 5) Selesai

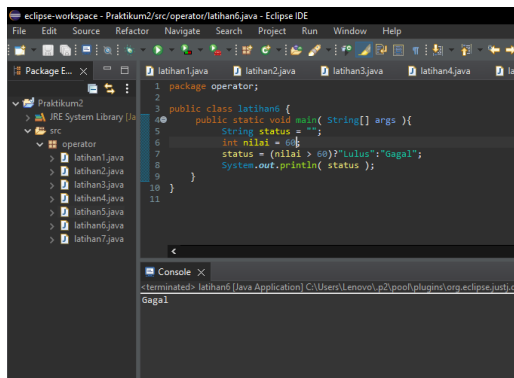
[Nomor 5] Kesimpulan

- 1) Analisa
- 2) Evaluasi

[Nomor 6] Identifikasi Masalah Operator Kondisional (Ternary): ?:



6.1. Berdasarkan Contoh 6, ubahlah nilai = 60. Analisis hasil dan proses yang terjadi!



Mengapa luarannya Gagal, sebab nilai yang diinput tidak lebih besar (>) daripada 60. Tetapi jika tanda > diubah menjadi >= maka akan menghasilkan luaran Lulus.

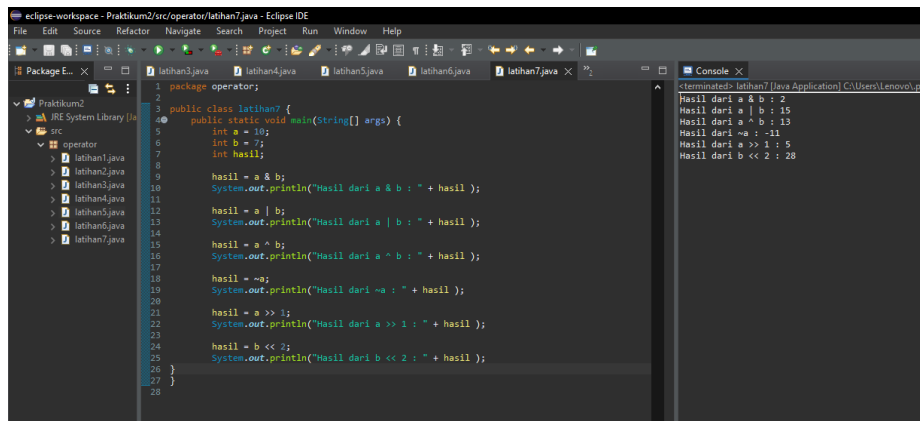
[Nomor 6] Penyusunan Algoritma

- 1 Mulai
- 2 Masukkan String status = "";
- 3 Masukkan int nilai = 60;
- 4 Panggil variable status
- 5 Run
- 6 Selesai

[Nomor 6] Kesimpulan

- 1 Analisa
Dari program diatas, penggunaan operator kondisional dapat menghasilkan luaran Lulus adalah apabila input nilai lebih besar dari 60.
- 2 Kreasi
Hal yang saya kembangkan adalah saya berhasil mencoba semua operator kondisional, salah satunya adalah tanda >=. Misalnya input int nilai = 60; status = (60 >= 60) ? "Lulus" : "Gagal"; System.out.println(status); Maka luaran yang ditampilkan adalah Lulus.

[Nomor 7] Identifikasi Masalah Operator Bitwise: &, |, ^, ~, <<, >>, >>>



```
package operator;

public class latihan7 {
    public static void main(String[] args) {
        int a = 10;
        int b = 7;
        int hasil;

        hasil = a & b;
        System.out.println("Hasil dari a & b : " + hasil);

        hasil = a | b;
        System.out.println("Hasil dari a | b : " + hasil);

        hasil = a ^ b;
        System.out.println("Hasil dari a ^ b : " + hasil);

        hasil = ~a;
        System.out.println("Hasil dari ~a : " + hasil);

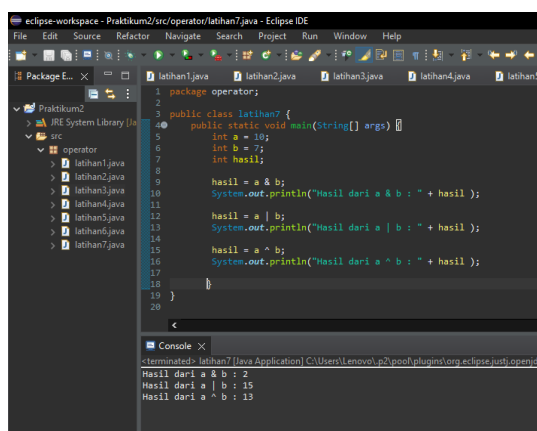
        hasil = a >> 1;
        System.out.println("Hasil dari a >> 1 : " + hasil);

        hasil = b << 2;
        System.out.println("Hasil dari b << 2 : " + hasil);
    }
}
```

Console Output:

```
Hasil dari a & b : 2
Hasil dari a | b : 15
Hasil dari a ^ b : 13
Hasil dari ~a : -11
Hasil dari a >> 1 : 5
Hasil dari b << 2 : 28
```

Pilihlah 3 perhitungan Contoh 7, kemudian uraikan perhitungan biner! Simpulkan hasilnya!



```
package operator;

public class latihan7 {
    public static void main(String[] args) {
        int a = 10;
        int b = 7;
        int hasil;

        hasil = a & b;
        System.out.println("Hasil dari a & b : " + hasil);

        hasil = a | b;
        System.out.println("Hasil dari a | b : " + hasil);

        hasil = a ^ b;
        System.out.println("Hasil dari a ^ b : " + hasil);
    }
}
```

Console Output:

```
Hasil dari a & b : 2
Hasil dari a | b : 15
Hasil dari a ^ b : 13
```

Pada perhitungan biner saya mencoba menguraikannya sebagai berikut :

Nilai a = 10 (desimal) bentuk biner nya 1010

b = 7 (desimal) bentuk biner nya 111

- 1) Operasi yang pertama **a & b = 10 & 7**
(dalam bentuk biner 1010 & 111) hasilnya 2 (biner 10). Operasi & (AND) terhadap kedua variable. Operasi bitwise “AND” ini akan memproses bit per bit dari kedua variable, jika kedua bit sama sama 1, maka hasilnya juga 1, selain kondisi tersebut nilai akhirnya 0.
- 2) Operasi kedua **a | b = 10 | 7** hasilnya 15
(binernya 1010 | 111) hasilnya 15 (biner 1111), operasi |(OR) hasilnya akan bernilai 0 jika kedua bit bernilai 0, selain itu nilai bit akan di set menjadi 1.
- 3) Operasi yang ke tiga **a ^ b = 10 ^ 7**
(binernya 1010 ^ 111) hasilnya 13 (binernya 1101), operasi ^ (XOR) hasilnya akan bernilai 1 apabila salah satu dari kedua variable bernilai 1 (namun tidak keduanya). Atau dengan kata lain jika kedua bit berlainan, hasilnya 1 tapi jika sama-sama 0 atau sama-sama 1 hasilnya 0.

[Nomor 7] Penyusunan Algoritma

- 1) Mulai
- 2) Masukkan int a, b, dan hasil
- 3) Masukkan operasi operator bitwise
- 4) Run
- 5) Selesai

[Nomor 7] Kesimpulan

1) Analisa

Menurut analisa saya bahwasahannya di program ini menggunakan operasi bitwise. Saya dapat mengerti perhitungan biner. Pada biner 0 artinya variable bernilai false dan 1 artinya variable bernilai true.

Refleksi

Tuliskan singkat tentang pengalaman belajar, pemaknaan pengetahuan yang baru, tantangan yang dihadapi pada minggu tersebut.

- Pada minggu ini saya memperelajari tentang operator dan pada operator ini terdapat banyak jenis jenis operator diantaranya adalah operator aritmatika, operator penugasan, operator relasioanal, operator increment dan decrement, operator logika , operator kondisional, serta operator bitwise