

Pemrograman berorientasi objek

Nama : salman alfariji

Kelas : ti20 a

Nim : 20200040033

Penjelasan tugas PBO point 2

1.

```
14      int a = 100;
15      int b = 20;
16      int c = 528;
17      int d = 70;
18      int e = 256;
19      int f = 3;
20      int g = 128;
21      int h = 5;
22      int i = 600;
23      int j = 4;
24      int k = 423;
25      int l = 2;
26      int hasil;
27
28      hasil = a & b;
29      System.out.println("Hasil perhitungan dari a & b = " + hasil );
30      System.out.println("-----");
31
32      hasil = c & d;
```

latihan2 > main >

Output - pembelajaran2 (run) x

run:
Hasil perhitungan dari a & b = 4

Operasi bitwise “and” ini akan memproses bit per bit dari kedua variabel, jika kedua bit sama-sama 1, maka hasilnya juga 1, selain kondisi tersebut, nilai akhirnya adalah 0.

2.

```
      hasil = c | d;
      System.out.println("Hasil perhitungan dari c | d = " + hasil );
      System.out.println("-----");
```

latihan2 > main >

ut - pembelajaran2 (run) x

Hasil perhitungan dari c | d = 598

Operasi | (OR), hasilnya akan bernilai 0 jika kedua bit bernilai 0, selain itu nilai bit akan di set menjadi 1.

3. 4. 5. 6

```
15
16     hasil = e >> f;
17     System.out.println("Hasil perhitungan dari a >> 1 = " + hasil );
18     System.out.println("-----");
19
20     hasil = g << h;
21     System.out.println("Hasil perhitungan dari b << 2 = " + hasil );
22     System.out.println("-----");
23
24     hasil = i >> j;
25     System.out.println("Hasil perhitungan dari a >> 1 = " + hasil );
26     System.out.println("-----");
27
28     hasil = k << l;
29     System.out.println("Hasil perhitungan dari b << 2 = " + hasil );
30     System.out.println("-----");
31 }
32
33
```

latihan2 > main >

Output - pembelajaran2 (run) x

```
> Hasil perhitungan dari a >> 1 = 32
> -----
> Hasil perhitungan dari b << 2 = 4096
> -----
> Hasil perhitungan dari a >> 1 = 37
> -----
> Hasil perhitungan dari b << 2 = 1692
> -----
BUILD SUCCESSFUL (total time: 0 seconds)
```

Operator **shift right** ">>" dimana bahasa Java akan menggeser posisi bit dalam variabel a ke kanan sebanyak 1 tempat. Selain itu Operator shift right menggeser nilai biner variabel a ke arah kanan, sehingga digit paling kanan akan dihapus. Operator shift right ini akan menghasilkan nilai asal / 2.

Operator **shift left** "<<" dimana nilai variabel b akan digeser sebanyak 2 digit ke kiri. Dan selain itu Ketika hasil pergeseran ke kiri, digit paling kanan akan diisi angka 0. Setiap penggeseran 1 tempat ke kiri akan mengkali 2 nilai asal.