Laporan Praktikum ke 3

Variable, Tipe data, Operator dan metode input di JAVA



Oleh :

Nur Afifah (E31201690) Semester 1

Program Studi Manejemen Informatika Jurusan Teknologi Informasi Politeknik Negeri Jember

2020

1. **Dasar Teori**
2. Pada pertemuan Praktek hari ini kita belajar Variabel pada Java Variaber singkatnya adalah sebagai tempat menyimpan nilai sementara.Kemudian, kita juga belajar Tipe data. Tipe data adalah jenis data yang tersimpan dalam variabel.

Macam-macam Tipe Data

Berikut ini macam-macam tipe data pada Java:

1. char: Tipe data karakter, contoh Z
2. int: angka atau bilangan bulat, contoh 29
3. float: bilangan desimal, contoh 2.1
4. double: bilangan desimal juga, tapi lebih besar kapasistanya, contoh 2.1
5. String: kumpulan dari karakter yang membentuk teks, contoh Hello World!
6. boolean: tipe data yang hanya bernilai true dan false.
7. Imputan di Java menggunakan Buffered Reader dan Scanner

Buffered Reader merupakan salah satu Class Library yang di sediakan oleh Java OOP untuk melakukan proses Input Class. Class tsb dapat dipanggil dengan melakukan import library; seperti :

**Import Java.io.BufferedReader   
Import Java.io.InputStreamReader  
Import Java.io.IOException**

Yang ke dua kita mempelajari inputan di Java menggunakan Scanner.

Scanner merupakan class yang menyediakan fungsi-fungsi untuk mengambil input dari keyboard.

1. Operator di Java

Operator dalam pemrograman digunakan untuk melakukan operasi tertentu.

Misalkan kita ingin menjumlahkan nilai dari variabel x dan y, maka kita bisa menggunakan operator penjumlahan (+).

Yang akan di praktekkan yaitu =

1. Melakukan increment dan decrement nilai

Operator **increment** dan **decrement** adalah sebutan untuk operasi seperti a++, dan a- Ini sebenarnya penulisan singkat dari operasi **a = a + 1** serta **a = a – 1**.

**Increment** digunakan untuk menambah variabel sebanyak 1 angka, sedangkan **decrement**digunakan untuk mengurangi variabel sebanyak 1 angka.

Penulisannya menggunakan tanda tambah 2 kali untuk **increment**, dan tanda kurang 2 kali untuk **decrement**. Penempatan tanda tambah atau kurang ini boleh di awal seperti ++a dan --a, atau di akhir variabel seperti a++ dan a--.

1. Melakukan operasi bit

**Bitwise** atau biasa di sebut Bit adalah operator khusus untuk menangani operasi logika bilangan biner dalam bentuk bit.

**Bilangan biner** sendiri merupakan jenis bilangan yang hanya terdiri dari 2 jenis angka, yakni 0 dan 1. Jika nilai asal yang dipakai bukan bilangan biner, akan dikonversi secara otomatis oleh compiler Java menjadi bilangan biner. Misalnya 7 desimal = 0111 dalam bilangan biner.

Dalam penerapannya, operator bitwise tidak terlalu sering dipakai, kecuali anda sedang membuat program yang harus memproses bit-bit komputer. Selain itu operator ini cukup rumit dan harus memiliki pemahaman tentang sistem bilangan biner. Dalam bahasan kali ini saya menganggap sudah paham beda antara bilangan biner (basis 2) dan bilangan desimal (basis 10).

1. Melakukan operasi komplemen

Operator pelengkap bitwise (~), yang melakukan negasi bitwise dari nilai integer. Negasi bitwise berarti setiap bit dalam nomor tersebut toggled. Dengan kata lain, semua biner 0 menjadi 1 dan semua biner 1 menjadi 0.

x = 8;

Y = ~ x;

Bilangan bulat disimpan dalam memori sebagai rangkaian biner yang dapat berupa 0 atau 1. Bilangan dianggap negatif jika bit orde tertinggi dalam bilangan tersebut diatur ke 1. Karena pelengkap bitwise mengubah semua bit dalam sebuah bilangan termasuk bit pesanan tinggi (tanda bit). Karena angkanya menjadi negatif, ini ada dalam bentuk pujian 2. Untuk mengetahui nilai desimalnya, pertama-tama kita akan mengurangi 1 dari angka tersebut dan kemudian melakukan pujian l padanya (yaitu, ubah 1 menjadi 0 dan 0 menjadi 1).

1. Melakukan operasi shift

Operator bit shift merupakan jenis operator yang tidak berbeda dengan operator [sebelumnya](https://bmadi.wordpress.com/2014/12/22/operator-bitwise-di-java/). Akan tetapi, operator ini hanya menggeser posisi bit saja. Java, memiliki tiga macam operasi untuk operator ini, yaitu

|  |
| --- |
| 1. >>  geser bit ke kanan secara aritmatik.  2. <<< geser bit ke kiri secara aritmatik.  3  >>  geser bit ke ke kanan secara logika. |

1. Menggunakan logical operator

Operator logika memiliki satu atau lebih operand boolean yang menghasilkan nilai boolean. Pada Operetor logika Terdapat enam jenis operator yaitu: && (logika AND), & (boolean logika AND), || (logika OR), | (boolean logika inclusive OR), ^ (boolean logika exclusive OR), dan ! (logika NOT).

1. Menggunakan operator boolean and

Hasil Operand Logika && (AND) akan bernilai **true** jika kedua operand bernilai **true** dan akan bernilai **false** jika ada salah satu operand yang bernilai **false**. Untuk lebih jelasnya coba perhatikan tabel kebenaran berikut ini.

|  |  |  |
| --- | --- | --- |
| **---- OPERAND1 ----** | **---- OPERAND2 ----** | **---- HASIL ----** |
| **true** | **true** | **true** |
| **true** | **false** | **false** |
| **false** | **true** | **false** |
| **false** | **false** | **false** |

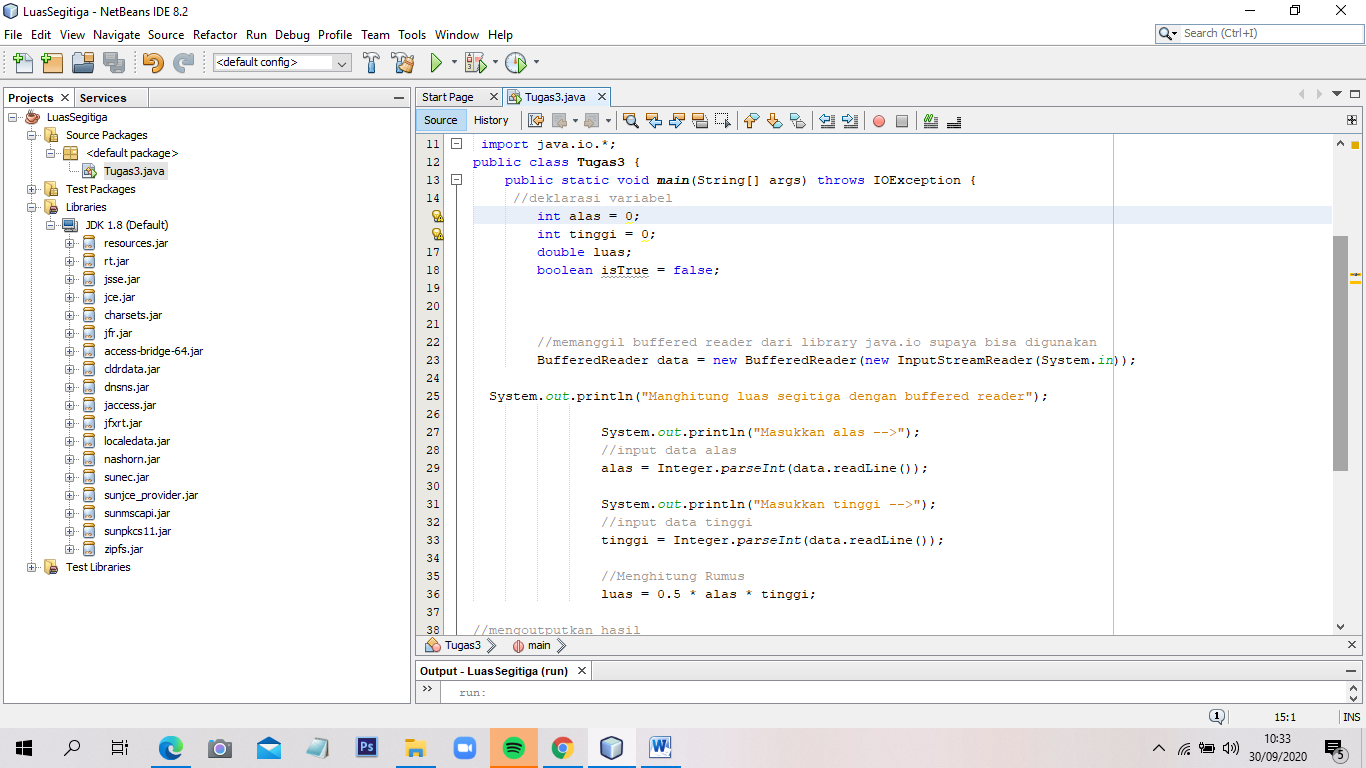
1. Menggunakan operator boolean and short-circuit

Short Circuit ini ada dalam tipe data Boolean. Dalam tipe data boolean kita mengetahui logika operator seperti '&, '&&', '|', dan '| |'.

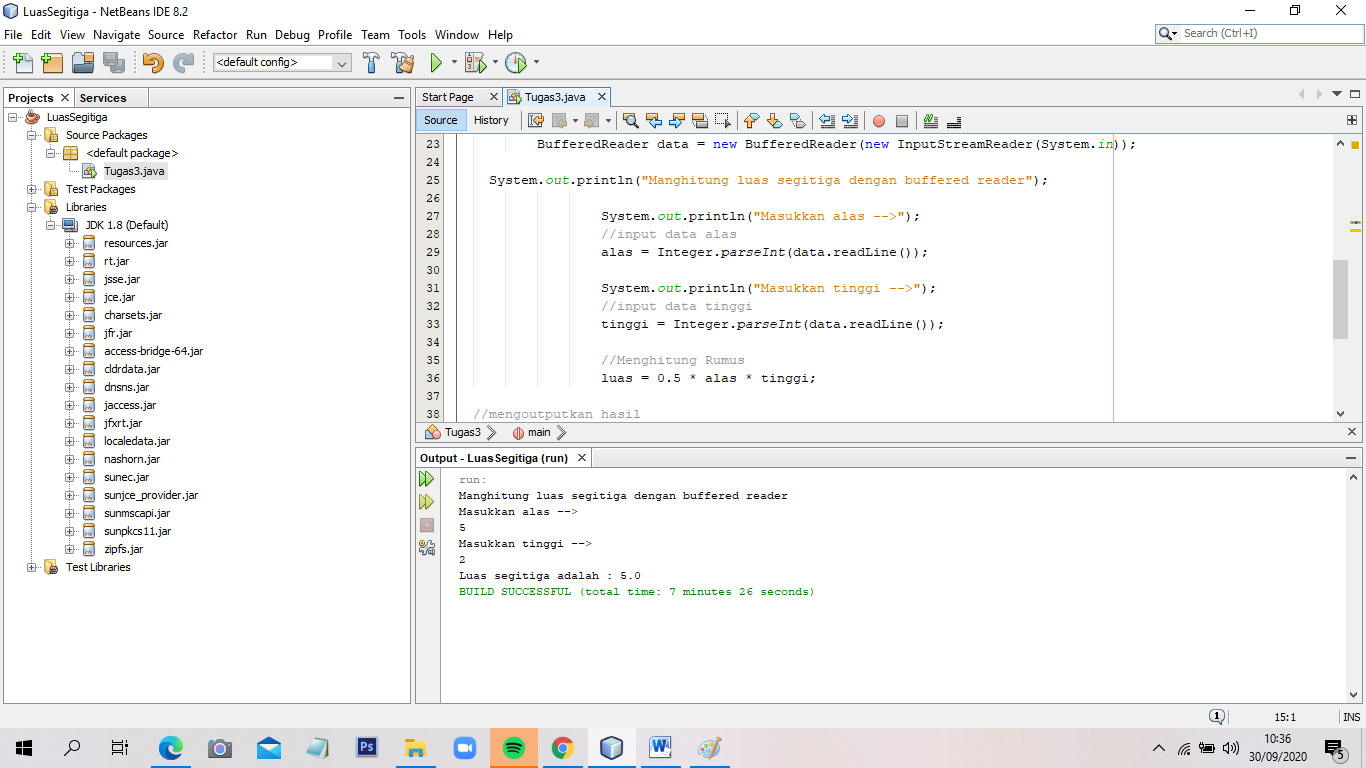
1. Menggunakan boolean or

Perbedaan dasar antara logika | | (OR) dan | (inclisive OR) adalah bahwa logika  | | (OR) mendukung **short-circuit** evaluations (proses evaluasi sebagian), sementara | tidak

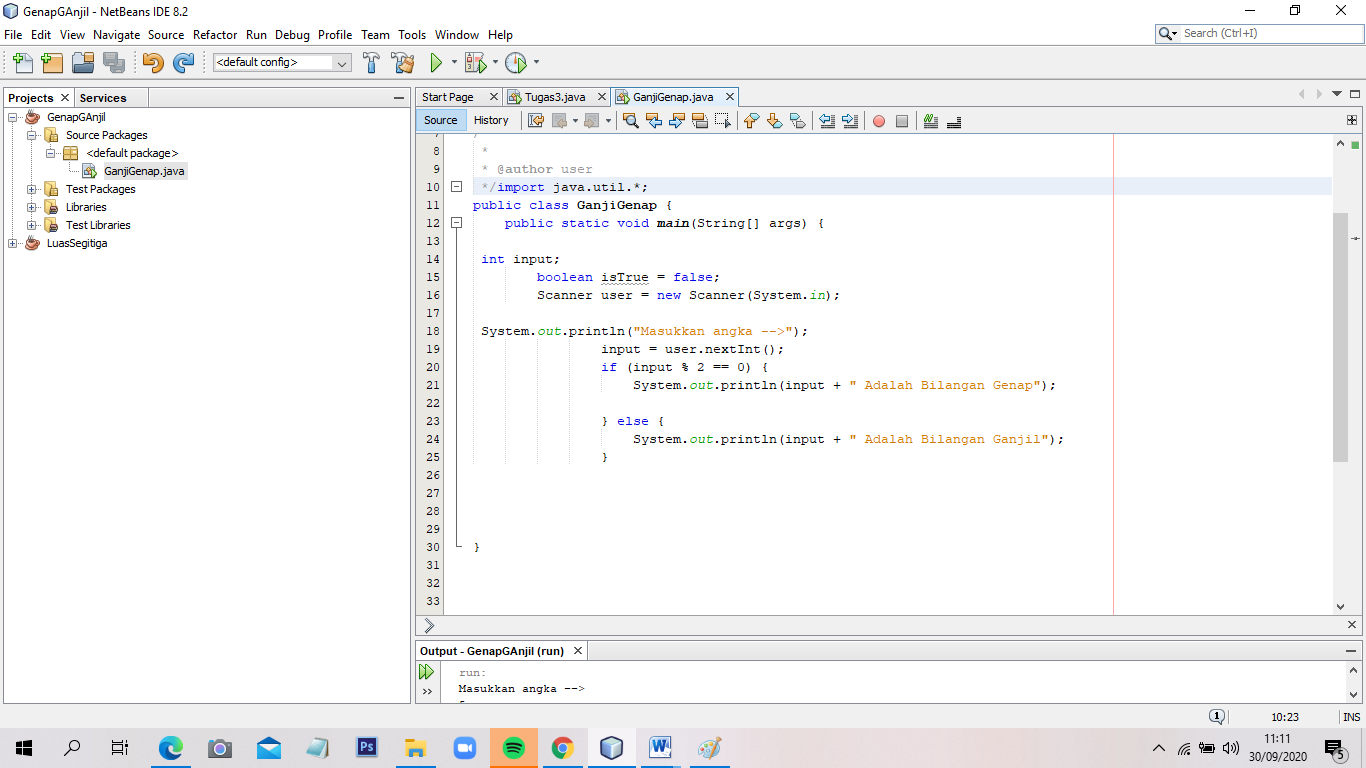
1. **Hasil Uji Coba Praktikum.**
2. Buffered Reader.Membuat sebuah program untuk menghitung luas segitiga.



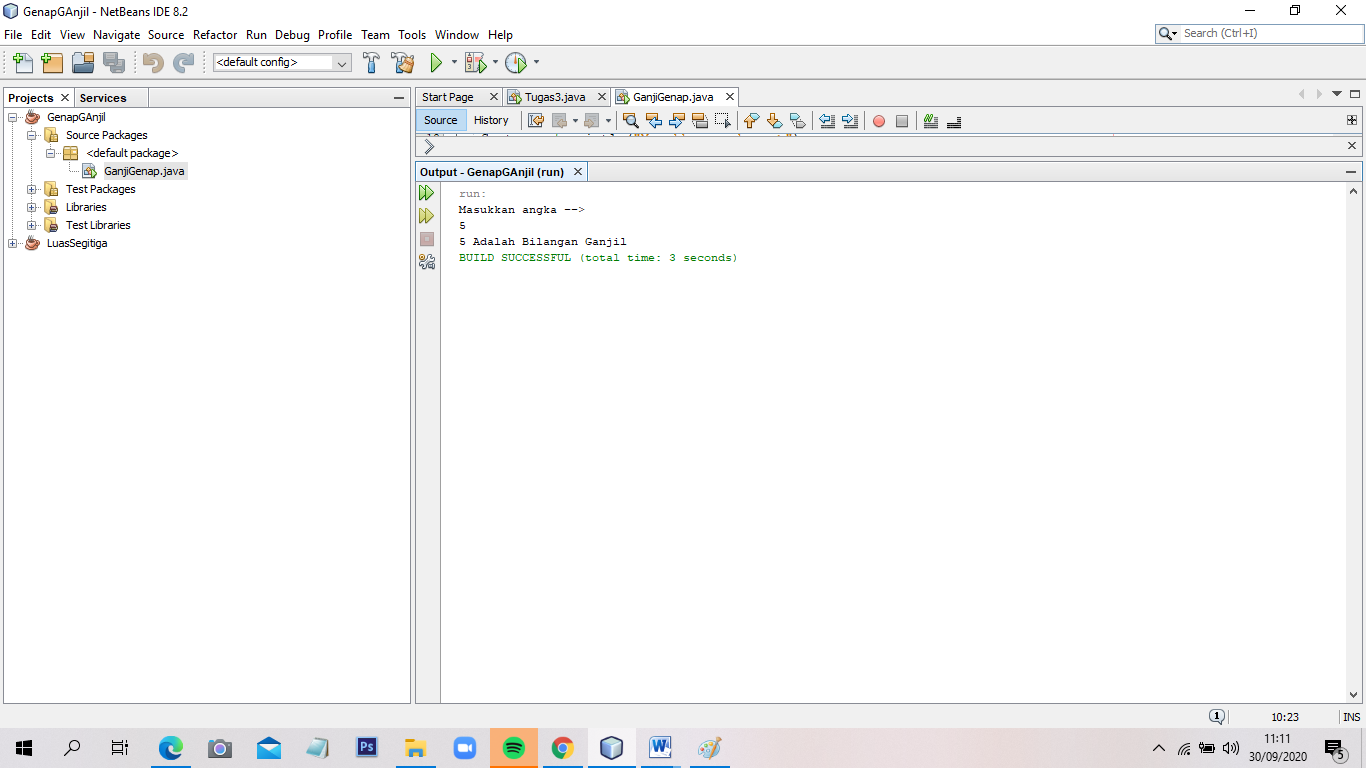
Outputnya seperti yang ada di bawah ini



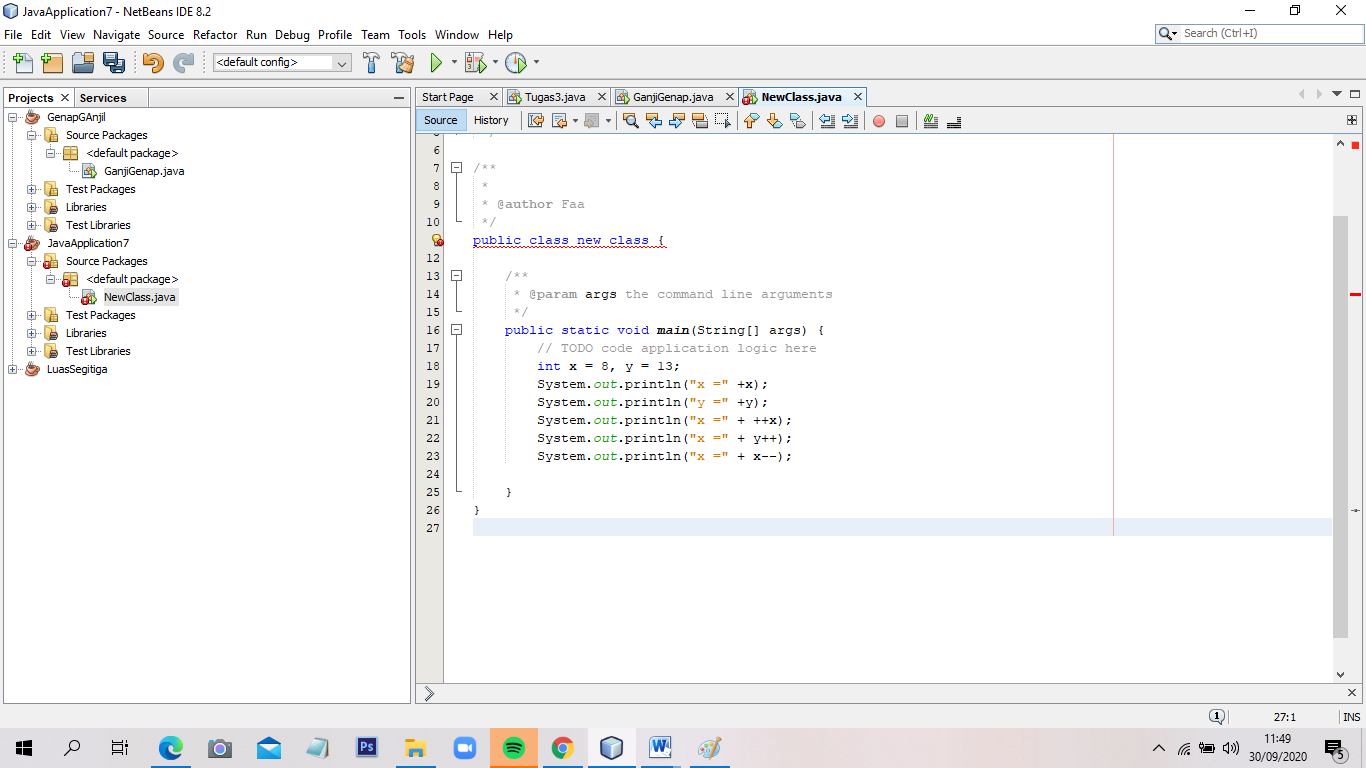
1. Inputan menggunakan Scanner. Membuat sebuah program untuk menampilkan bilangan genap dan ganjil.



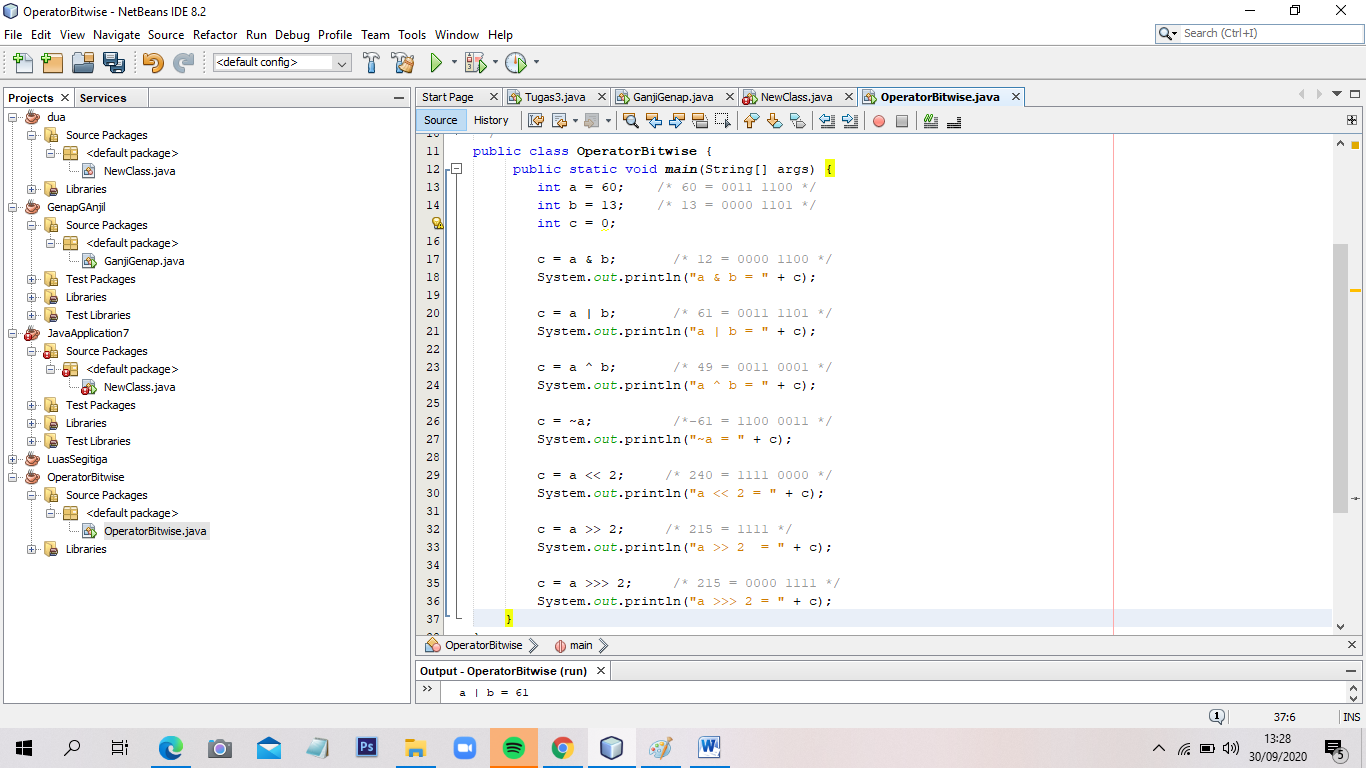
Outputnya seperti yang terlihat di gambar ini.



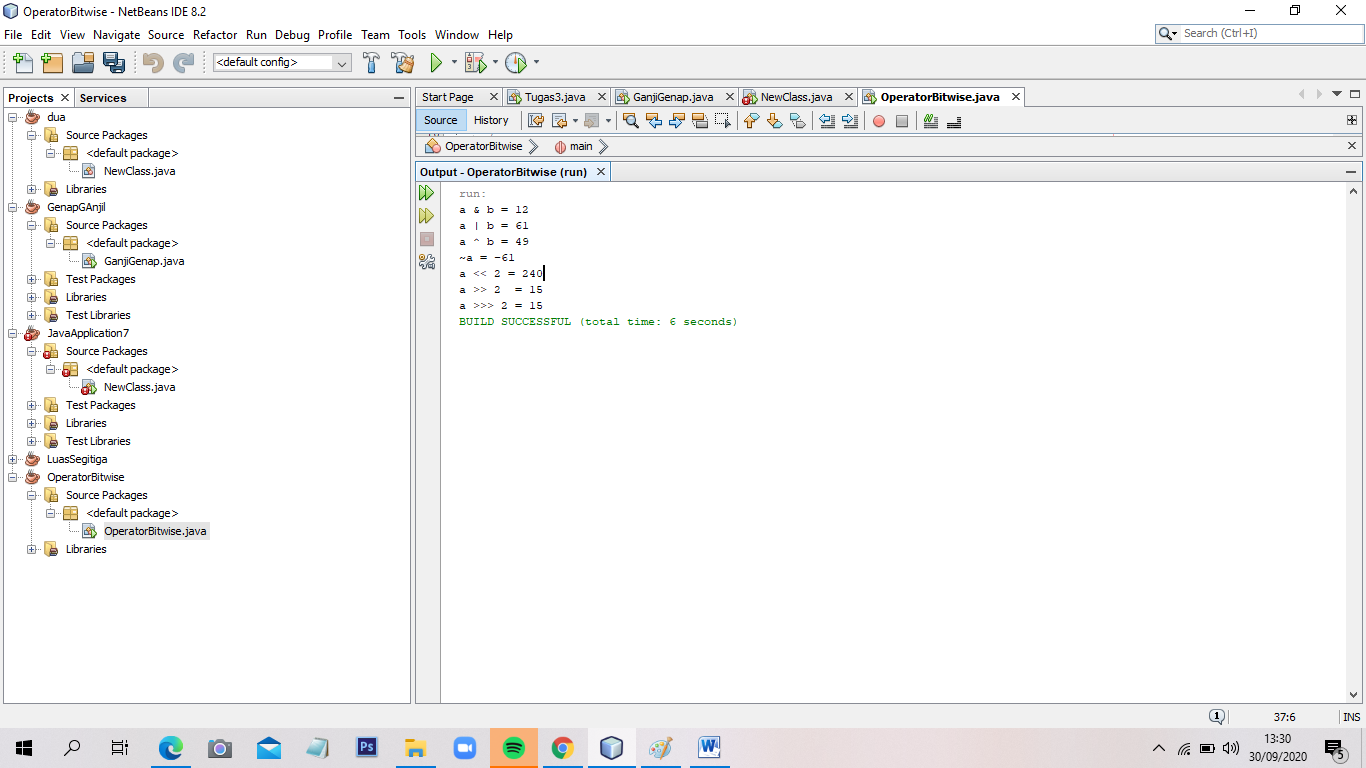
1. Percobaan 1 : Melakukan increment dan decrement nilai



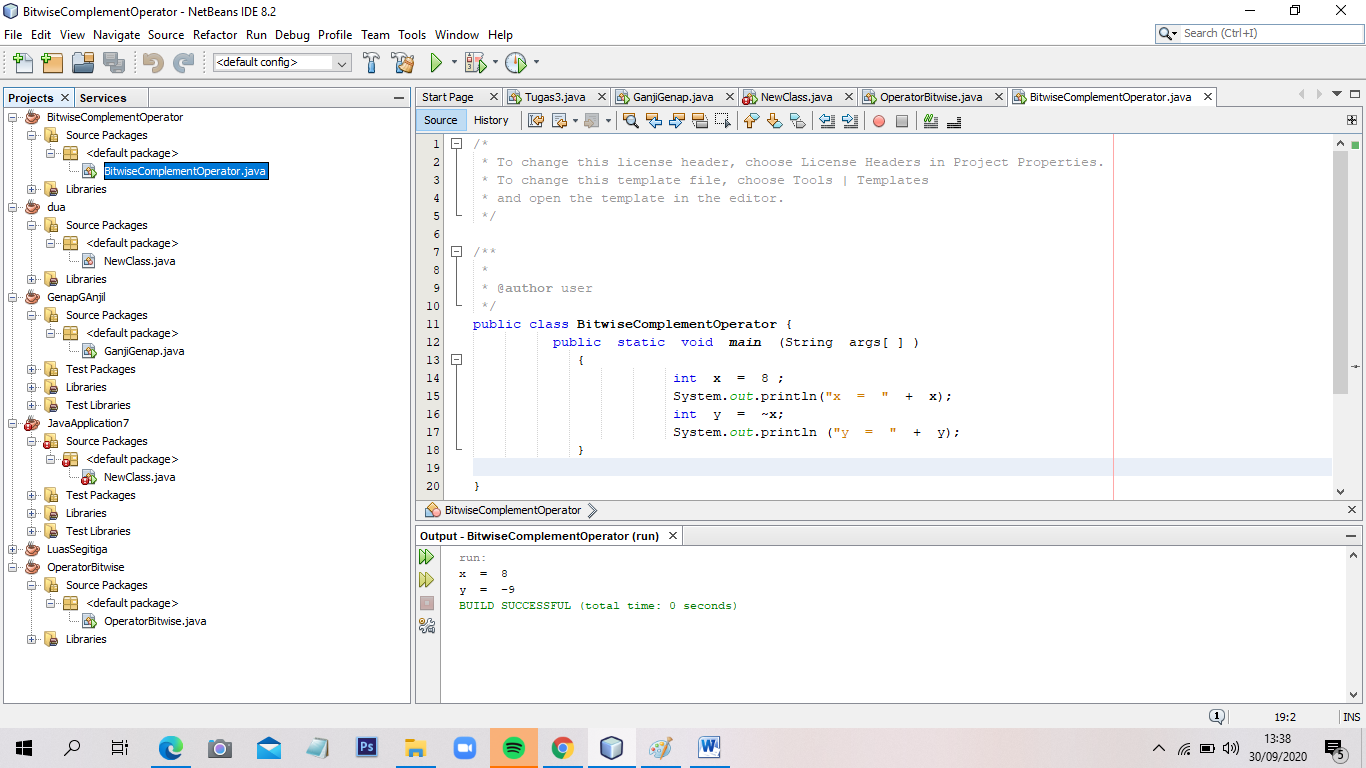
1. Percobaan 2 : Melakukan operasi bit.



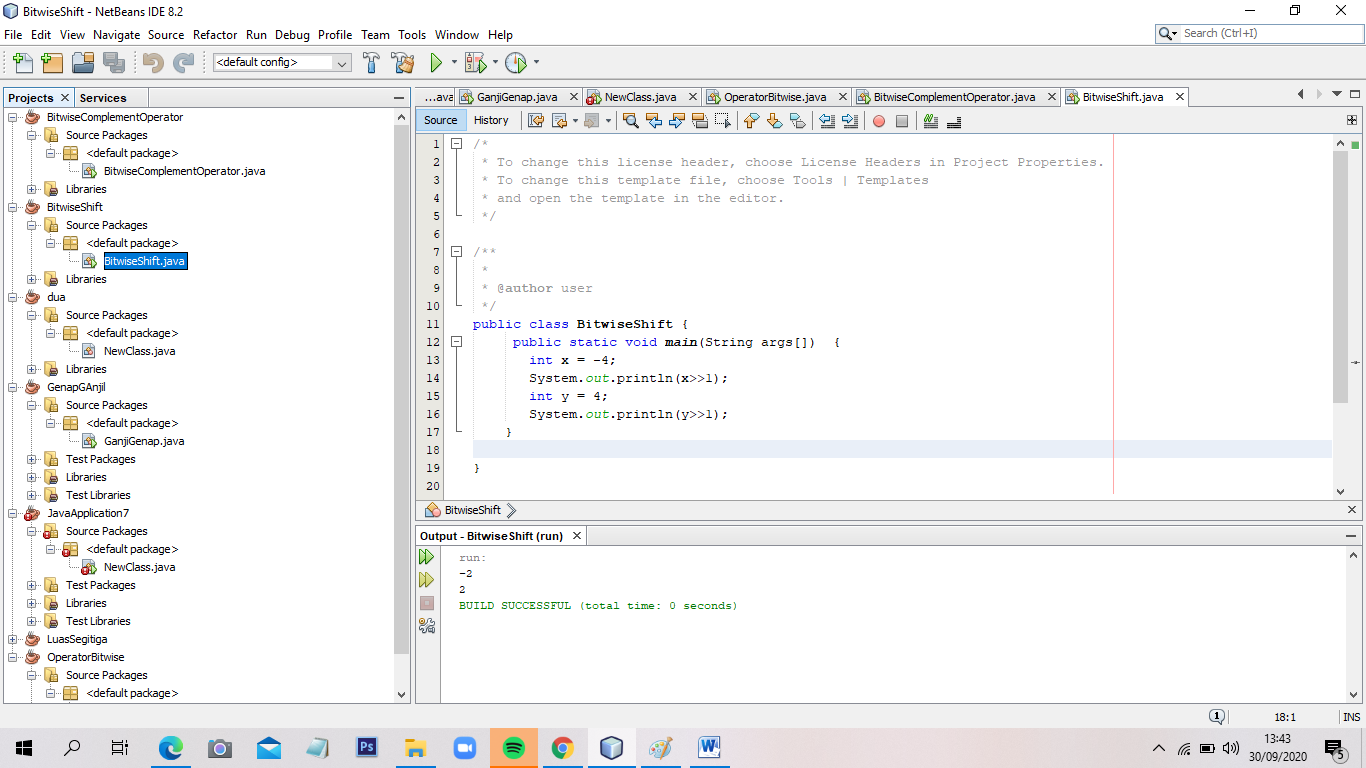
Hasil outpynya keetika di run seperti gambar di bawah ini



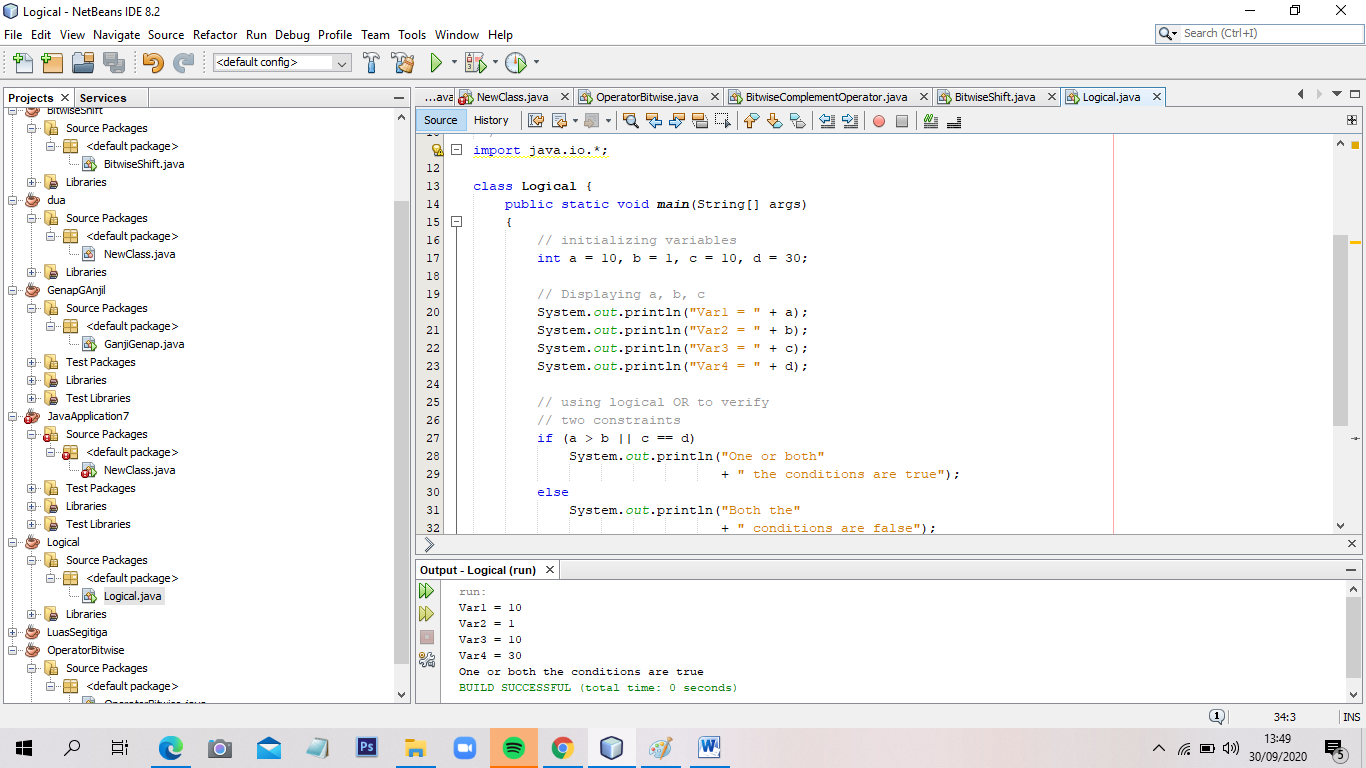
1. Percobaan 3 : Melakukan operasi komplemen beserta outputnya.



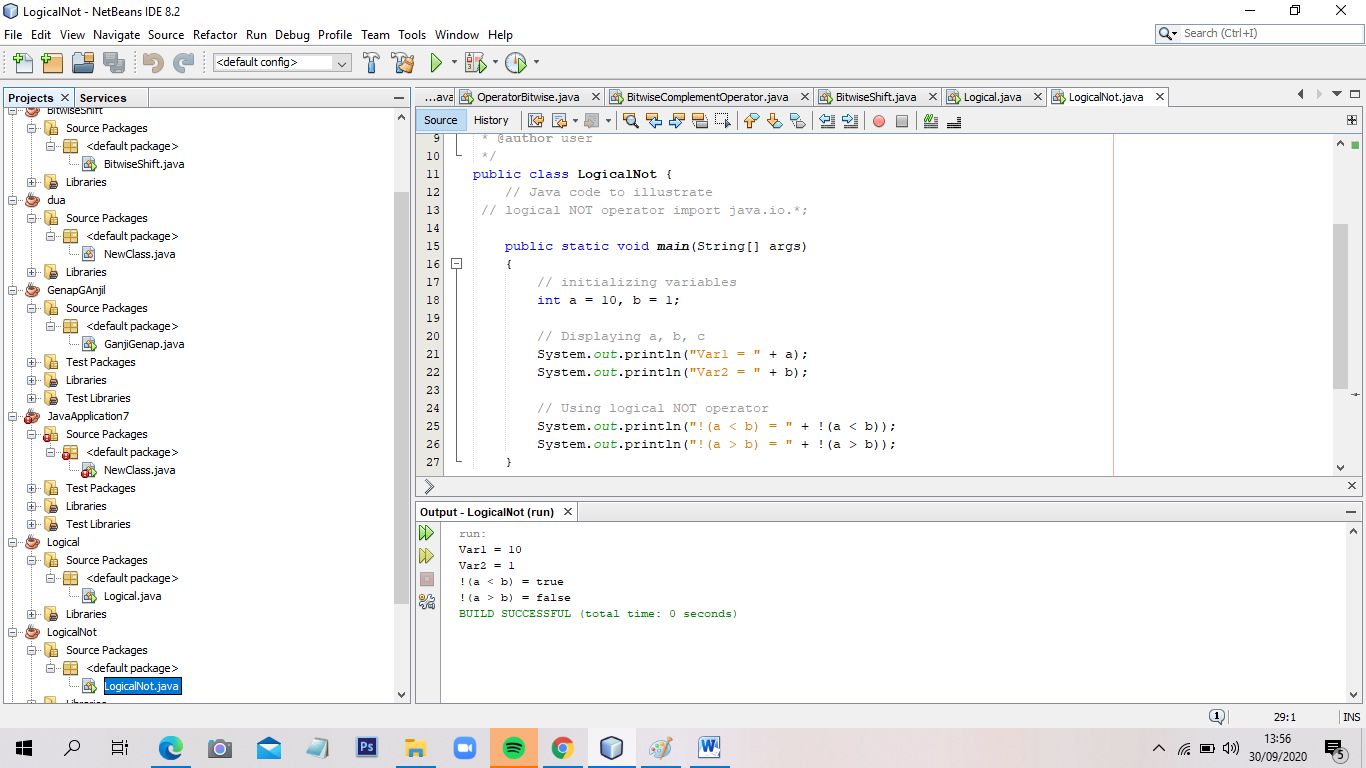
1. Percobaan 4 : Melakukan operasi shift beserta outputnya.



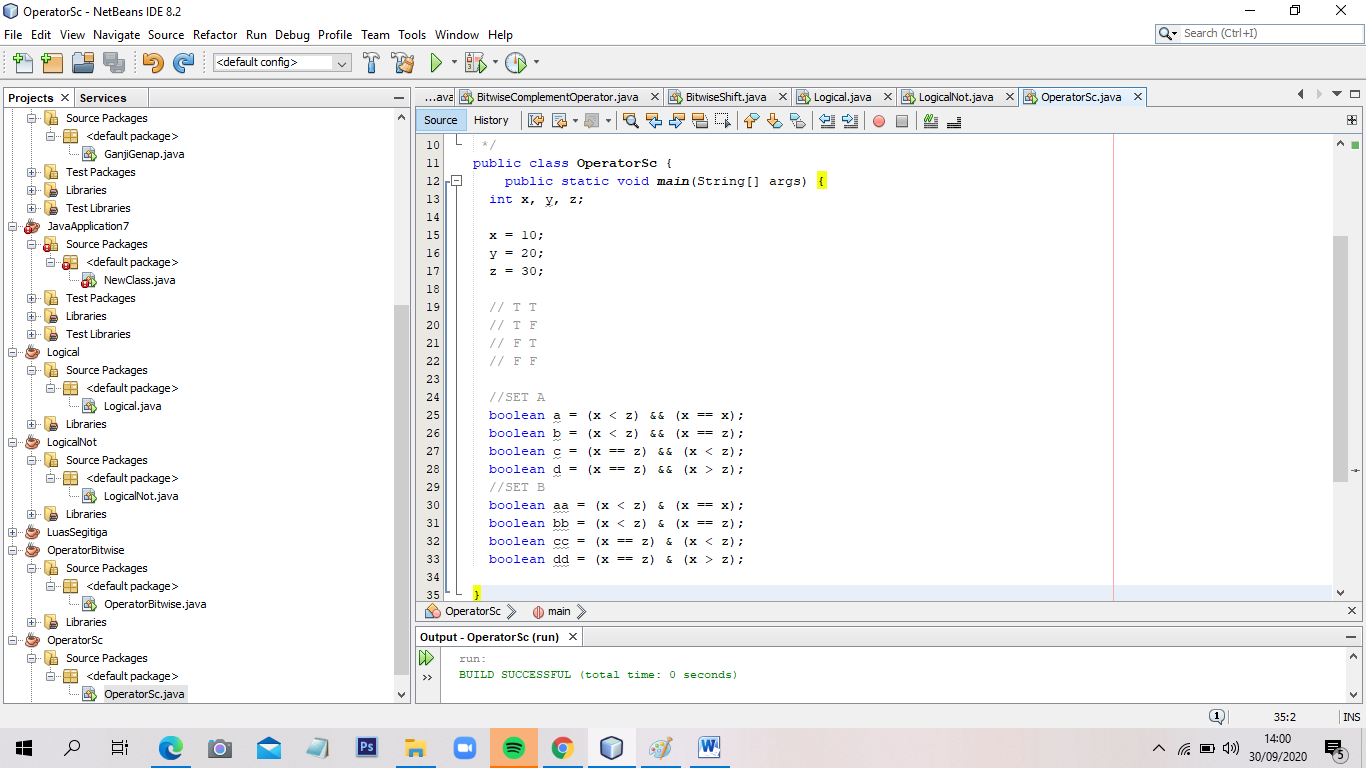
1. Percobaan 5 : Menggunakan logical OR operator beserta outputnya.



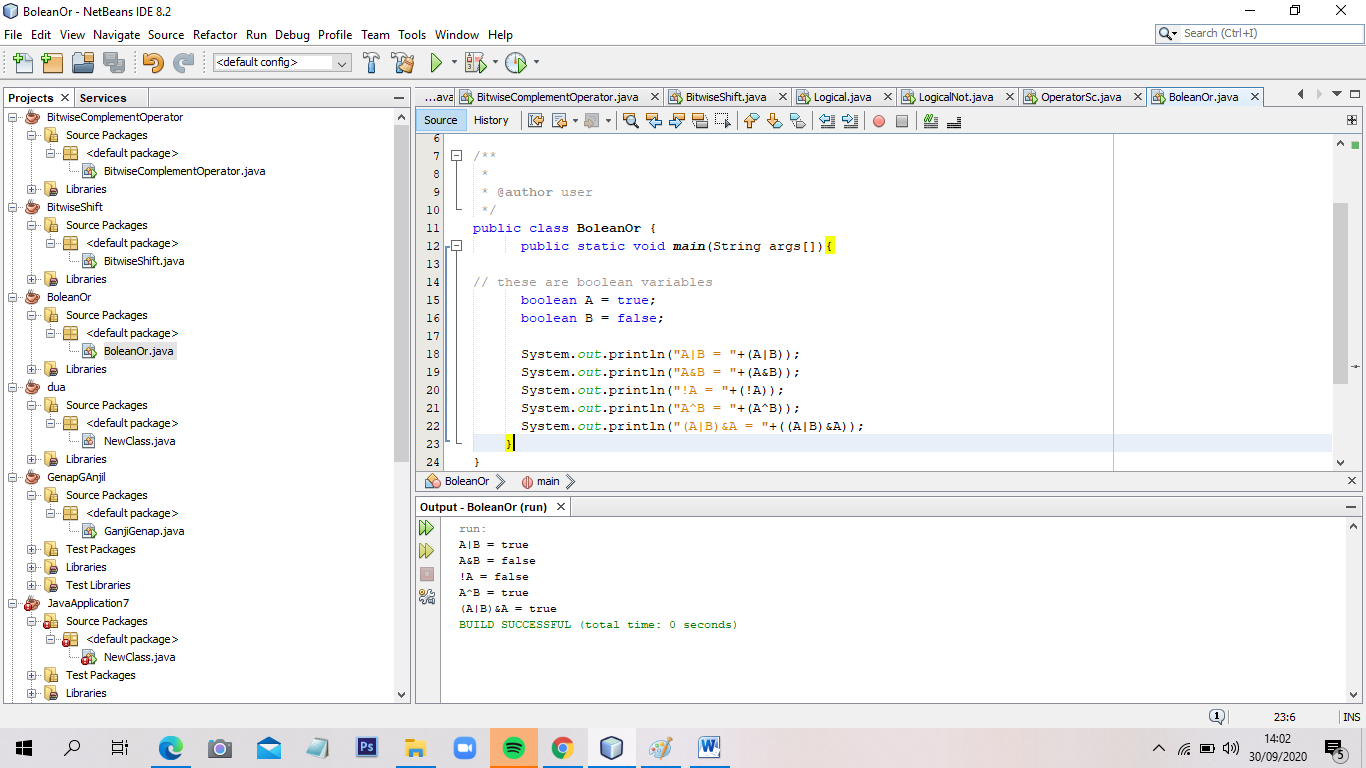
1. Percobaan 5 : Menggunakan logical Not(!) operator dengan outputnya.



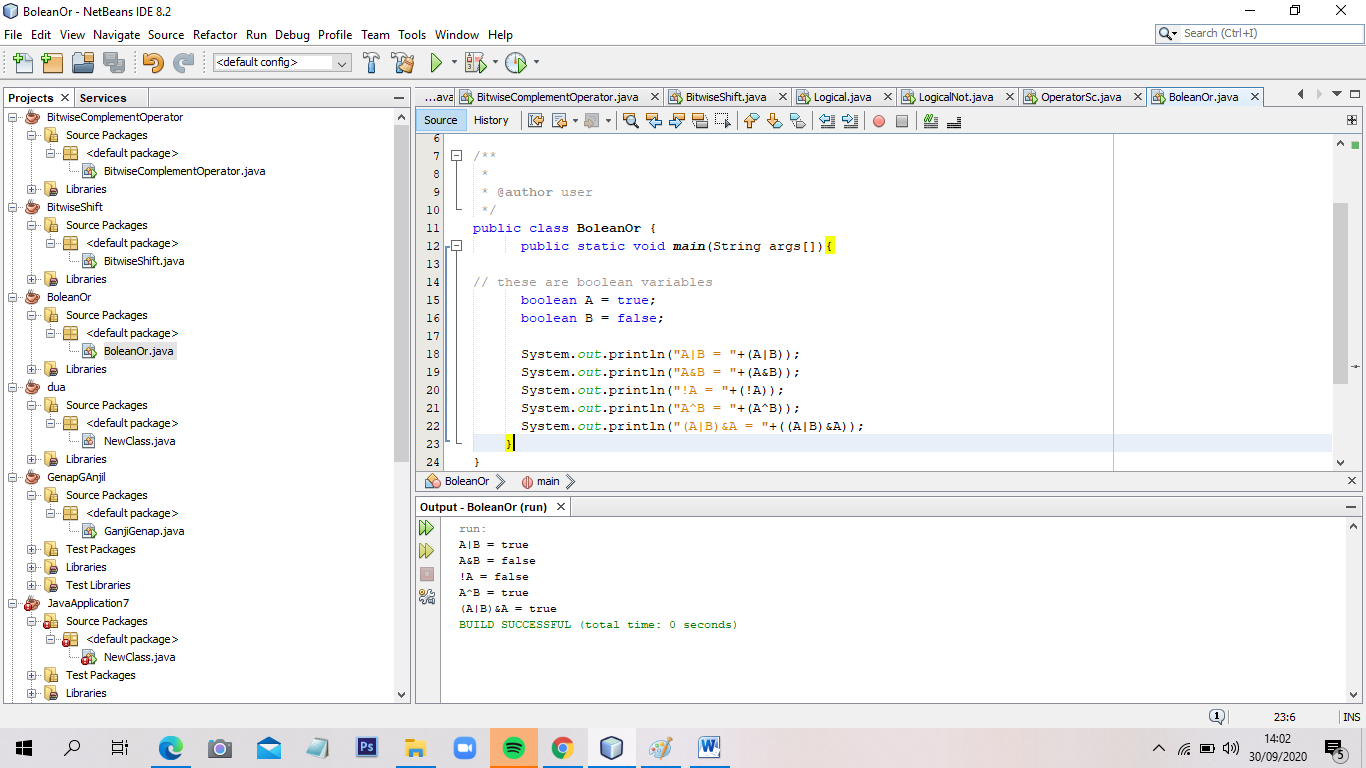
1. Percobaan 7 : Menggunakan operator boolean and short-circuit.

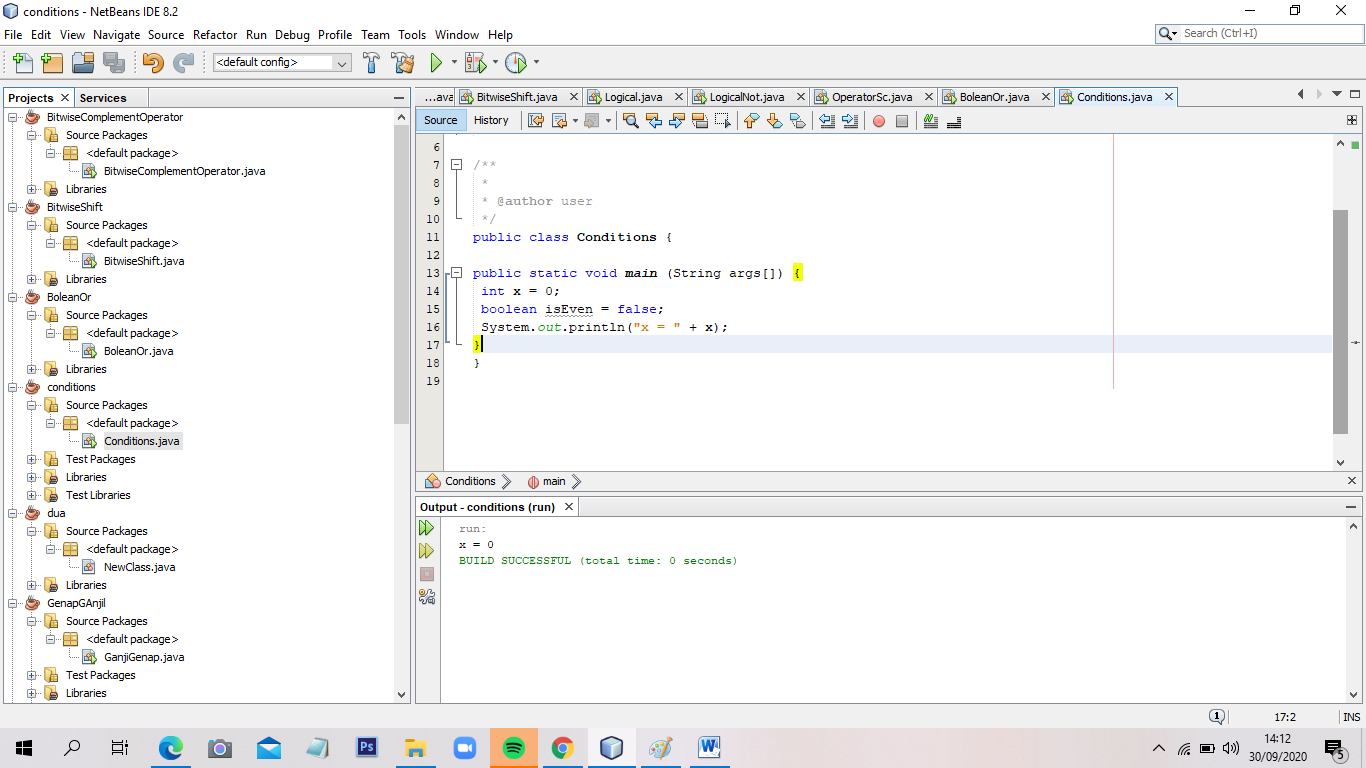


1. Percobaan 8 : Menggunakan boolean or beserta outputnya.



1. Percobaan 9 : Menggunakan boolean or short-circuit



1. Percobaan 10 : Menggunakan operator kondisi
2. **Tugas Pengumpulan di Akun Githup.**

Akses Akun Githup saya klik link berikut

<https://github.com/nurafifah7/vivi>

1. **Analisa dan Kesimpulan**

Mengambil Inputan yang lebih mudah dan sesuai yang dibutuhkan agar tidak terjadi error.

1. **Sumber Materi yang di Ambil**

<http://beginer-java.blogspot.com/2014/11/metode-inputan-menggunakan-buffered.html>

<https://www.petanikode.com/java-variabel-dan-tipe-data/>

<https://www.petanikode.com/java-input-output/>

<https://norkandirblog.wordpress.com/2016/08/24/tiga-metode-sangat-penting-dalam-pemrograman-java-bagi-pemula/>

<https://www.petanikode.com/java-operator/>

<https://stackoverflow.com/questions/8759868/java-logical-operator-short-circuiting>

<https://www.freejavaguide.com/boolean_operators.htm>

<https://books.trinket.io/thinkjava/chapter5.html>

<https://ecomputernotes.com/java/what-is-java-operators-and-expressions/bitwise-complement-operator>

<https://www.geeksforgeeks.org/bitwise-shift-operators-in-java/>

<https://www.geeksforgeeks.org/java-logical-operators-with-examples/>

<https://www.duniailkom.com/tutorial-belajar-java-jenis-jenis-operator-bitwise-bahasa-java/>

<https://www.wildantechnoart.net/2017/05/tutorial-5-java-Pengertian-dan-jenis-Operator-Logika-serta-penggunaan-boolean-pada-java.html>