**LAPORAN PRAKTIKUM**

**DASAR-DASAR PEMROGRAMAN PERTEMUAN KE-3**

Pengenalan Gitlab dan pengoperasian operator pada java

****

Disusun Oleh:

Muhammad Fitroh Amrilla

(11190910000022)

Program Studi Teknik Informatika

Fakultas Sains dan Teknologi

2019/2020

1. **Pembahasan**
2. Mengenali Gitlab dan cara penggunaannya.
3. Membahas lebih lanjut mengenai penggunaan Gitlab melalui aplikasi Git.
4. Membuat program dengan aturan operator dan tipe data dengan benar.
5. Menjalankan program java dengan aturan operator dan tipe data dengan benar.
6. Membuat program java dari pseudo-code yang telah dikerjakan pada latihan sebelumnya.
7. **Dasar Teori**

Pengembang aplikasi dan web tentu tidak asing dengan yang namanya Git. Git adalah tools yang berfungsi sebagai Version Control System (VCS) dan kalau diartikan ke bahasa kita artinya sebuah sistem pelacak perubahan pada file. Ini tentu berguna bagi para pengembang yang bekerja sendiri untuk projeknya atau berkolaborasi dengan pengembang lain.

Dengan menggunakan Git, setiap pengembang yang berkolaborasi dapat melakukan perubahan pada source-code tanpa harus takut terjadi bentrok ataupun kesulitan dalam menggabungkan hasil perubahan yang mereka lakukan. Dengan menggunakan Git, setiap perubahan pada source-code akan terlacak pesan perubahannya, apa saja yang diubah, siapa yang mengubah dan kapan waktunya. Tentu saja, kolaborasi antar pengembang tersebut tidak bisa berjalan jika kita tidak menggunakan layanan untuk penyimpanan dan berbagi kode Git yang telah kita buat. Saat ini sudah tersedia beberapa layanan web untuk menyimpan dan berbagi kode Git, seperti Github, Bitbucket, dan Gitorous.

Layanan penyimpanan Git lainnya yang mulai terkenal dan naik daun adalah Gitlab. Layanan Git ini dirilis pertama kali pada tahun 2011 dan mulai populer digunakan sebagai media penyimpanan Git di awal 2014, ketika mereka merilis versi Community dan Cloud nya secara gratis untuk pendamping versi enterprise nya.

Gitlab adalah sebuah manajer repositori [Git](https://id.wikipedia.org/wiki/Git) berbasis web dengan fitur [wiki](https://id.wikipedia.org/wiki/Wiki) dan [pelacakan masalah](https://id.wikipedia.org/w/index.php?title=Pelacakan_masalah&action=edit&redlink=1), menggunakan lisensi [sumber terbuka](https://id.wikipedia.org/wiki/Sumber_terbuka) dan dikembangkan oleh GitLab Inc. Perangkat lunak ini ditulis oleh [Dmitriy Zaporozhets](https://id.wikipedia.org/w/index.php?title=Dmitriy_Zaporozhets&action=edit&redlink=1) dan [Valery Sizov](https://id.wikipedia.org/w/index.php?title=Valery_Sizov&action=edit&redlink=1) dari [Ukraina](https://id.wikipedia.org/wiki/Ukraina). Kode yang ditulis adalah [Ruby](https://id.wikipedia.org/wiki/Ruby_(bahasa_pemrograman)). Kemudian, beberapa bagian telah ditulis ulang di Go.

Operator dalam pemrograman digunakan untuk melakukan operasi tertentu. Misalkan kita ingin menjumlahkan nilai dari variabel x dan y, maka kita bisa menggunakan operator penjumlahan (+).

Ada enam jenis kelompok operator dalam pemrograman Java, yaitu:

1. Operator Artimatika

Operator aritmatika digunakan untuk melakukan operasi aritmatika. Operator ini terdiri dari:

|  |  |
| --- | --- |
| Nama | Simbol |
| Penjumlahan | + |
| Pengurangan | - |
| Perkalian | \* |
| Pembagian | / |
| Sisa Bagi (mod) | % |

1. Operator Penugasan

Operator penugasan *(Assignment Operator)* fungsinya untuk memberikan tugas pada variabel tertentu. Biasanya untuk mengisi nilai.

Contoh:

int a = 10;

Variabel a ditugaskan untuk menyimpan nilai 10.

Operator Penugasan terdiri dari:

|  |  |
| --- | --- |
| Nama Operator | Sombol |
| Pengisian Nilai | = |
| Pengisian dan Penambahan | += |
| Pengisian dan Pengurangan | -= |
| Pengisian dan Perkalian | \*= |
| Pengisian dan Pembagian | /= |
| Pengisian dan Sisa bagi | %= |

1. Operator Pembanding

Seperti namanya, tugas operator ini untuk membandingkan. Operator ini juga dikenal dengan operator relasi. Nilai yang dihasilkan dari operator ini berupa boolean, yaitu: true dan false.

|  |  |
| --- | --- |
| Nama | Simbol |
| Lebih Besar | > |
| Lebih Kecil | < |
| Sama Dengan | == |
| Tidak Sama dengan | != |
| Lebih Besar Sama dengan | >= |
| Lebih Kecil Sama dengan | <= |

Operator ini terdiri dari:

contoh:

boolean x = 10 < 12;

Maka x akan bernilai *true*, karena 10 lebih kecil dari 12.

1. Operator Logika

Operator Logika digunakan untuk membuat operasi logika. Kalau kamu pernah belajar logika matematika, pasti tidak akan asing dengan operator ini. Tabel untuk operator logika yaitu:

|  |  |
| --- | --- |
| Nama | Simbol di Java |
| Logika AND | && |
| Logika OR | || |
| Negasi/kebalikan | ! |

1. Operator Bitwise

Operator bitwise merupakan operator yang digunakan untuk operasi bit (biner). Operator bitwise terdiri dari:

|  |  |
| --- | --- |
| Nama | Simbol di Java |
| AND | & |
| OR | | |
| XOR | ^ |
| Negasi/kebalikan | ~ |
| Left Shift | << |
| Right Shift | >> |
| Left Shift (unsigned) | <<< |
| Right Shift (unsigned) | >>> |

Operator ini berlaku untuk tipe data int, long, short, char, dan byte. Operator ini akan menghitung dari bit-ke-bit.

Misalnya, kita punya variabel a = 60 dan b = 13. Bila dibuat dalam bentuk biner, akan menjadi seperti ini:

a = 00111100

b = 00001101

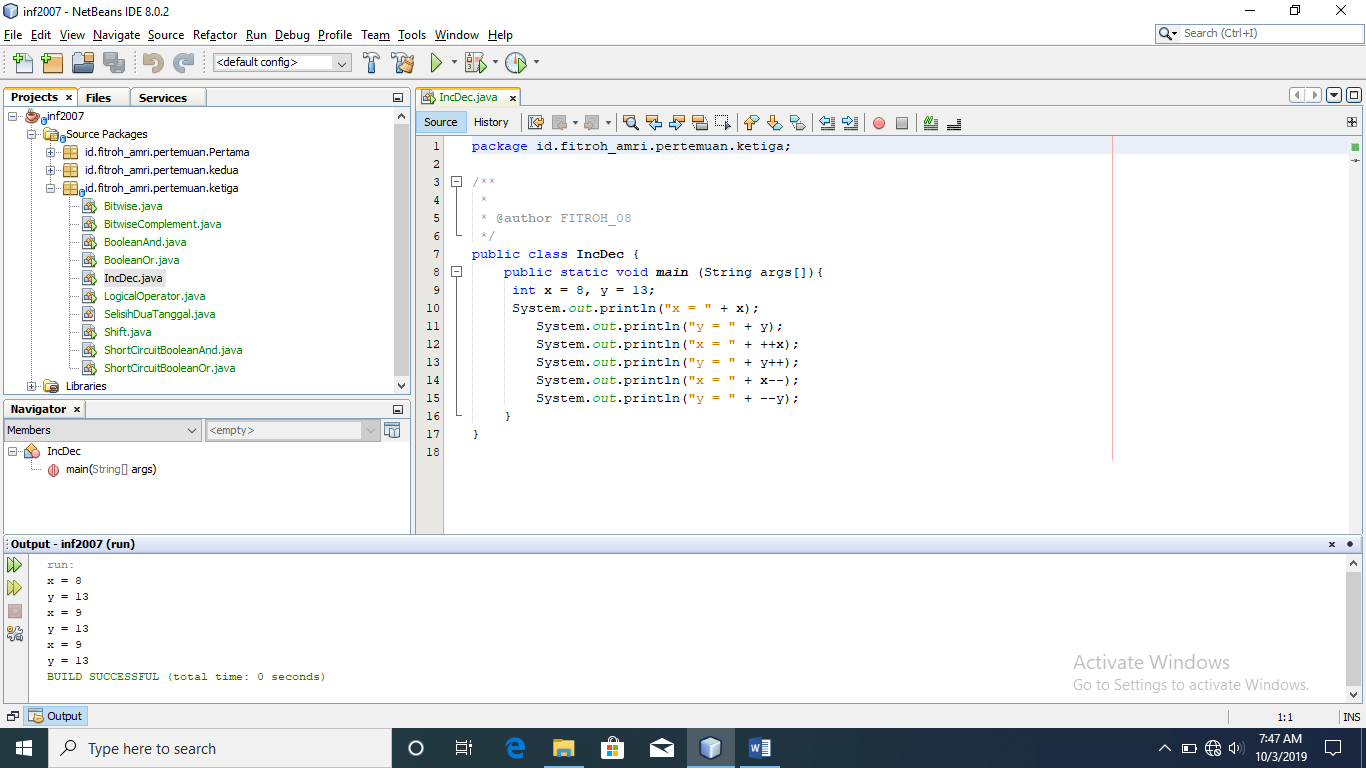
Kemudian, dilakukan operasi bitwise.

1. Operator Ternary

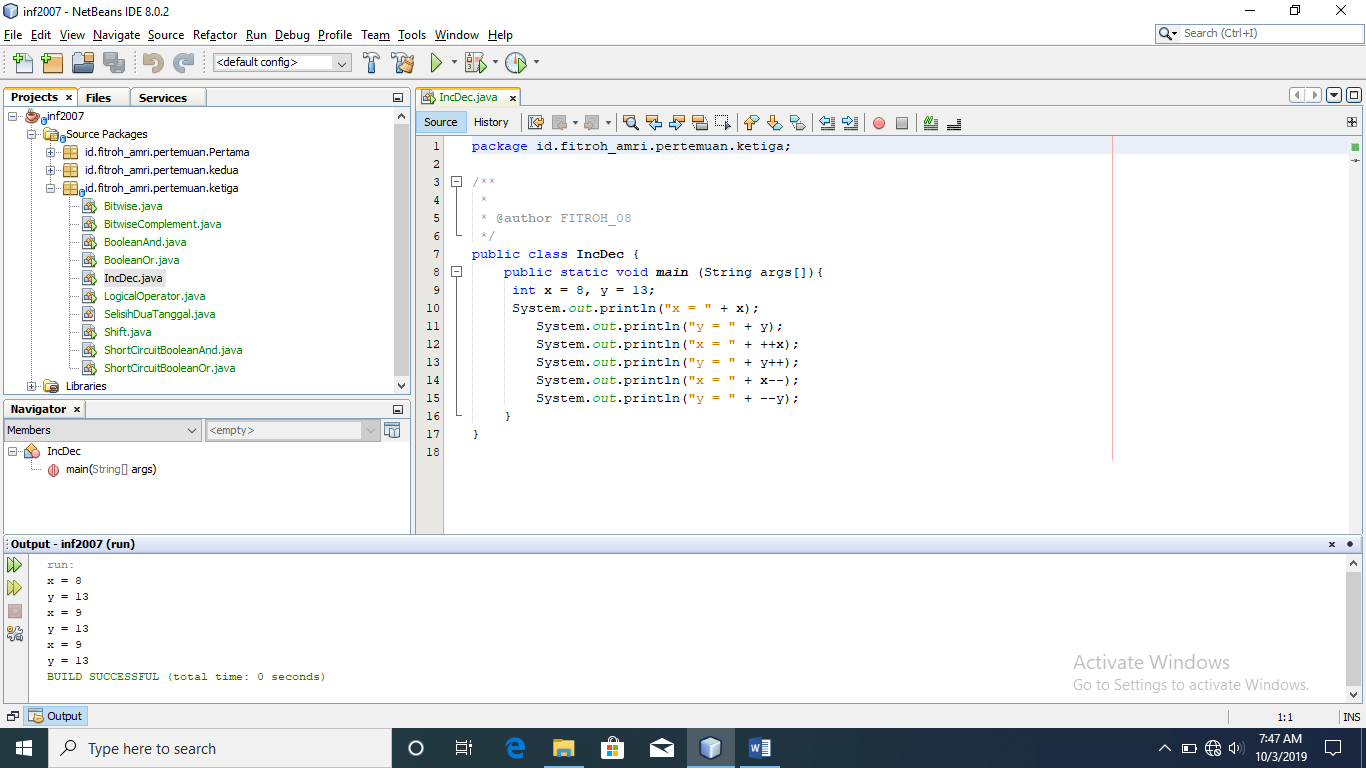
Operator ini unik, seperti membuat pertanyaan. Simbolnya menggunakan tanda tanya (?) dan titik-dua (:) untuk memisah jawabannya.

1. **Latihan**
2. Membuat program increment dan decrement nilai.

Source Code :

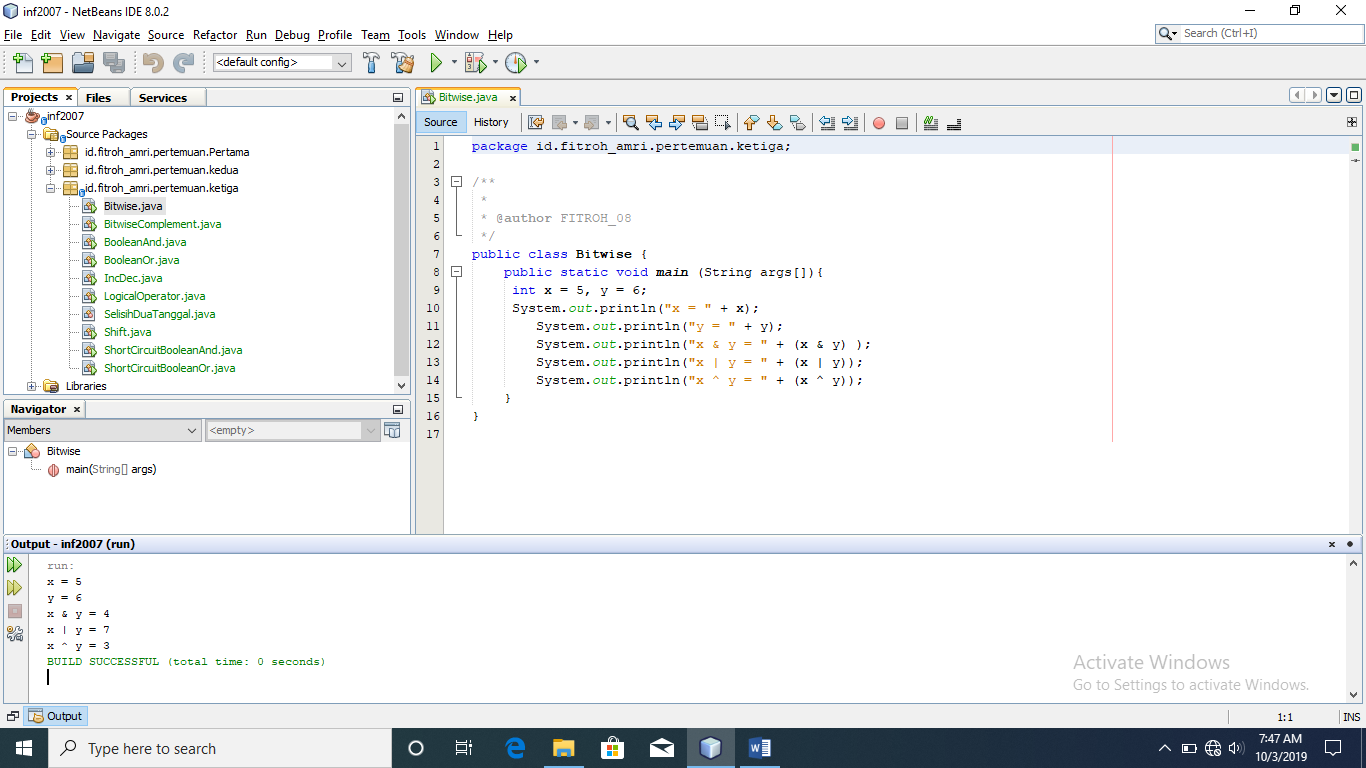


Output :

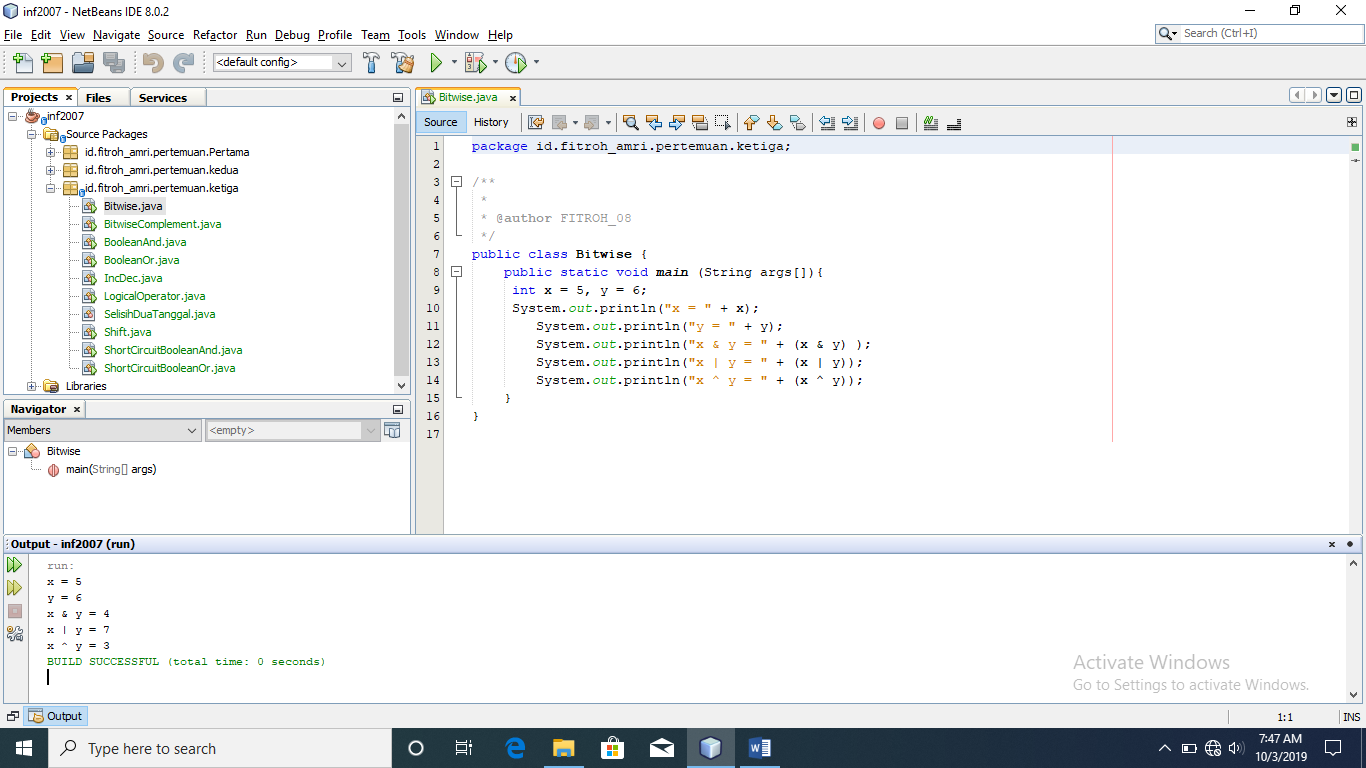


1. Melakukan operasi bit.

Source Code :

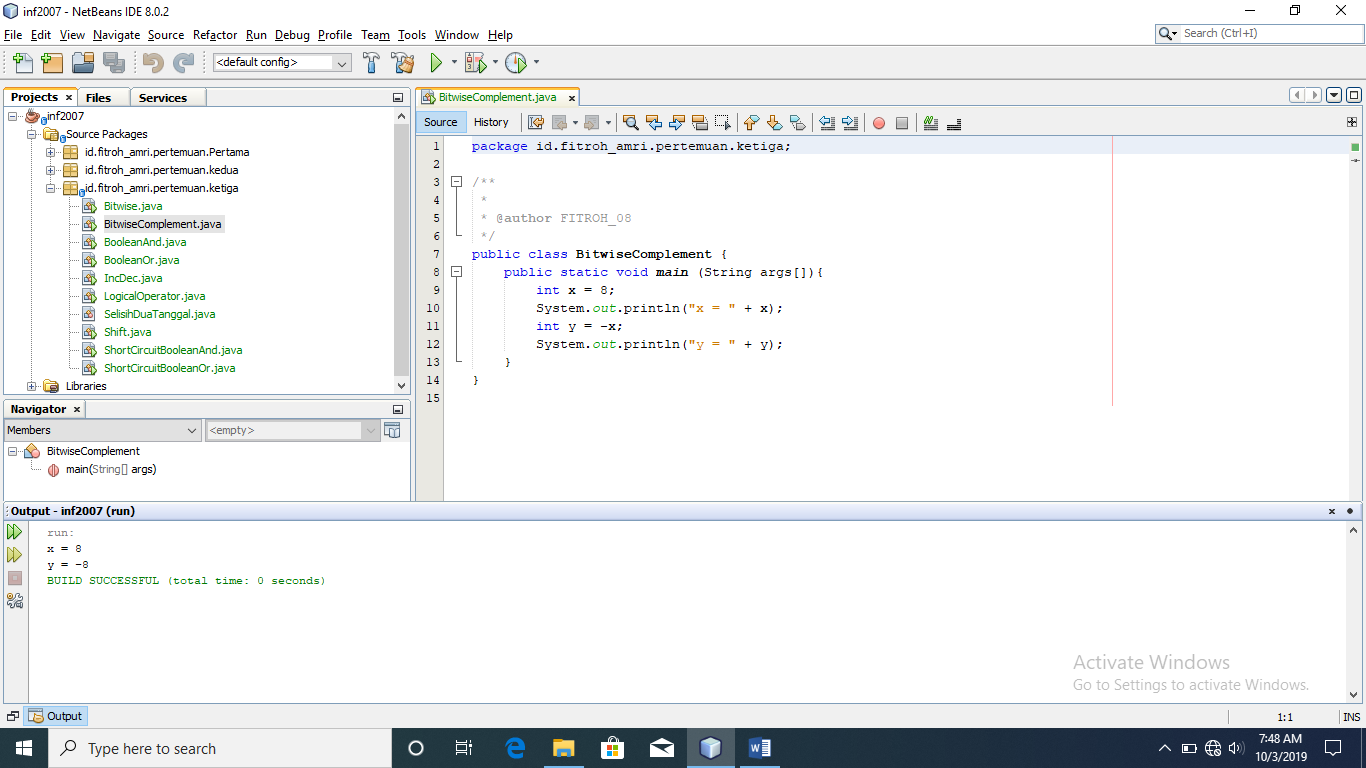


Output :

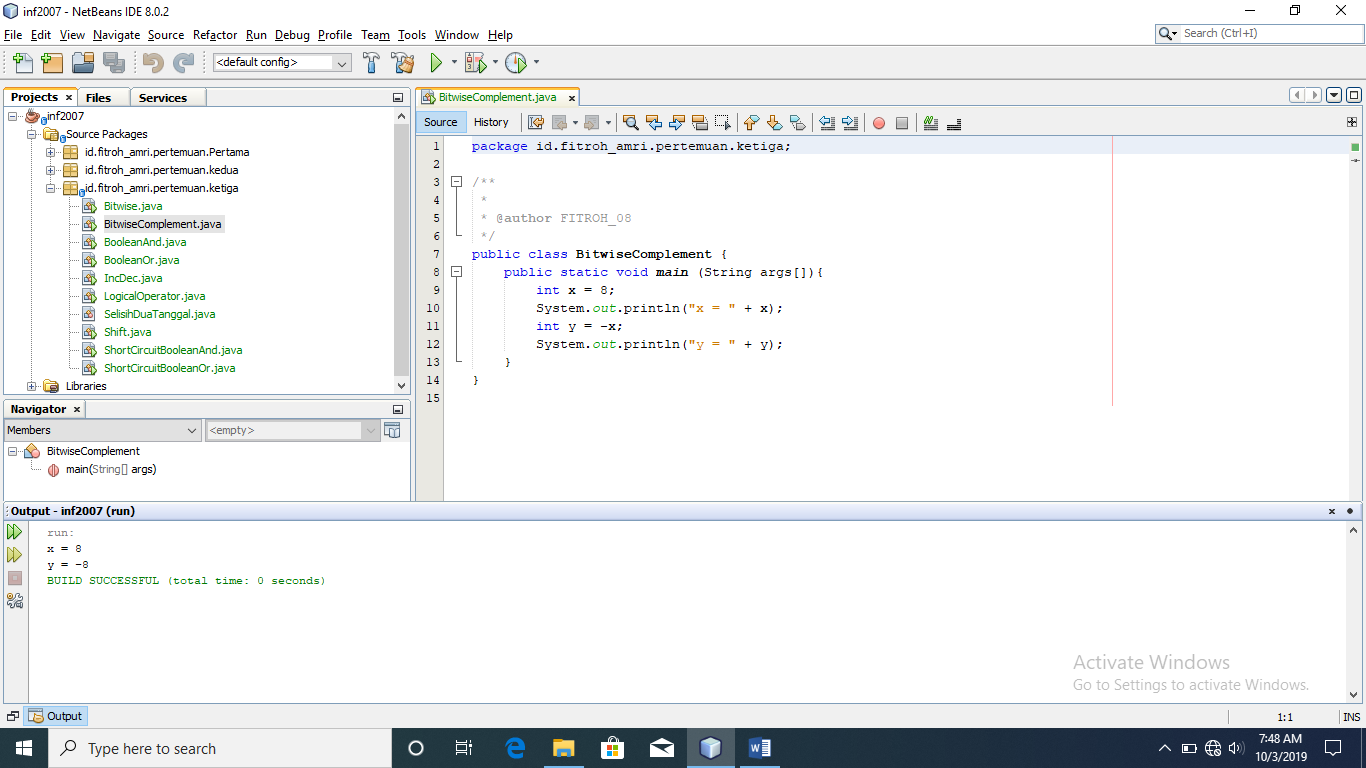


1. Melakukan operasi komplement.

Source Code :

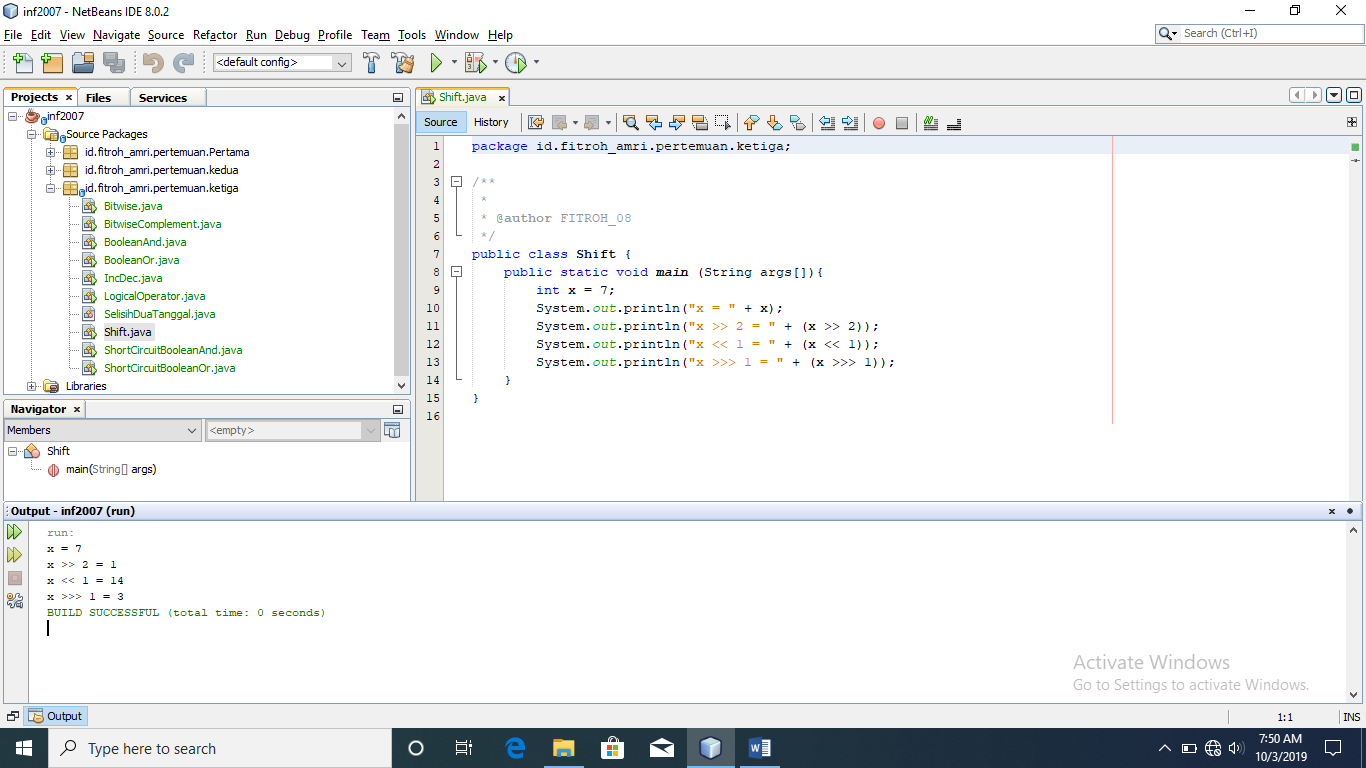


Output :

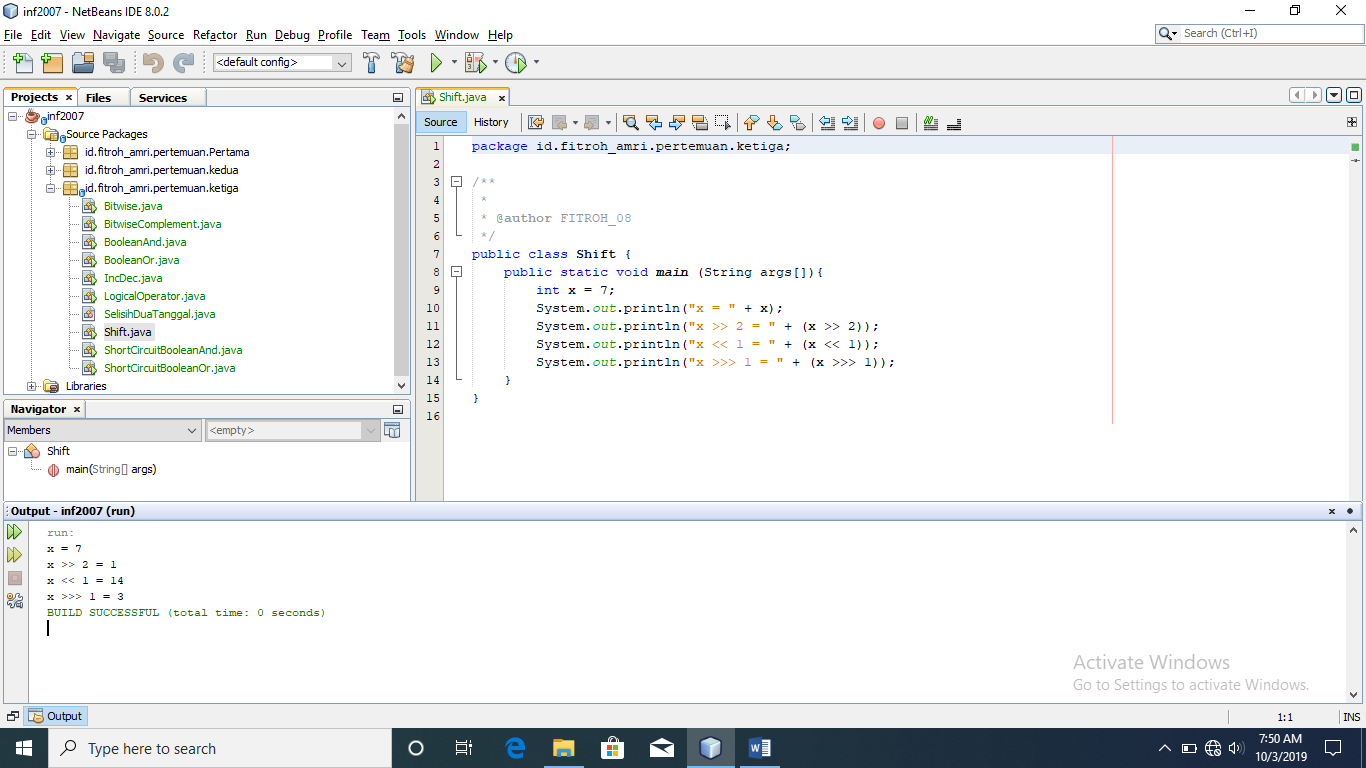


1. Melakukan operasi shift.

Source Code :

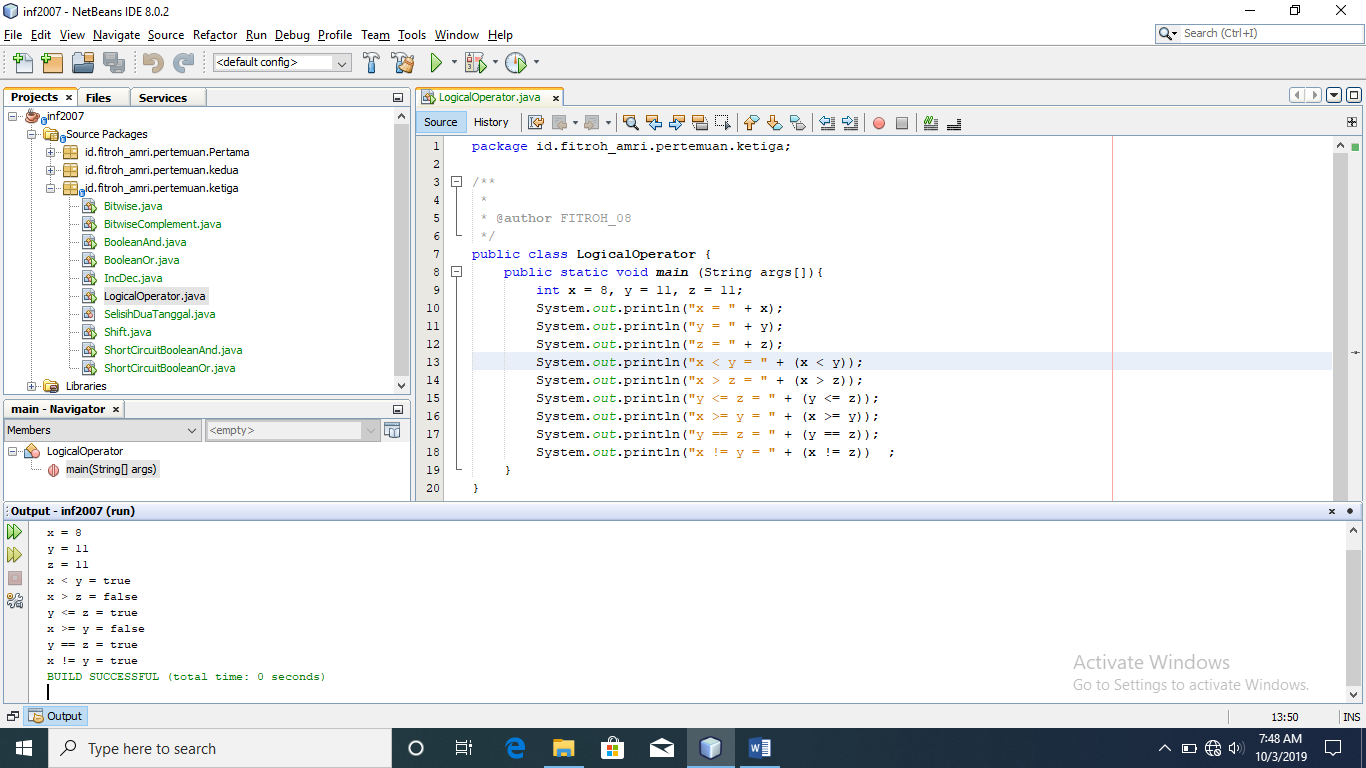


Output :

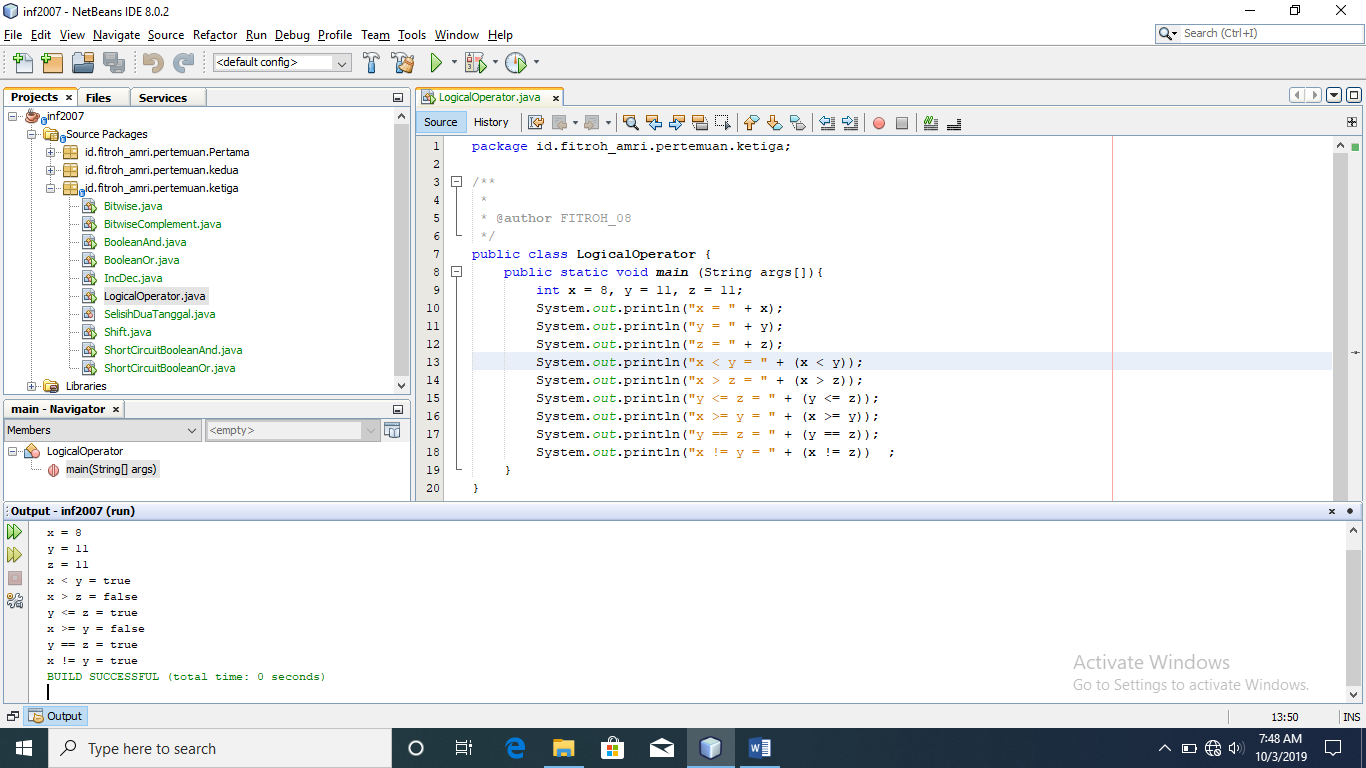


1. Menggunakan logical operator.

Source Code :

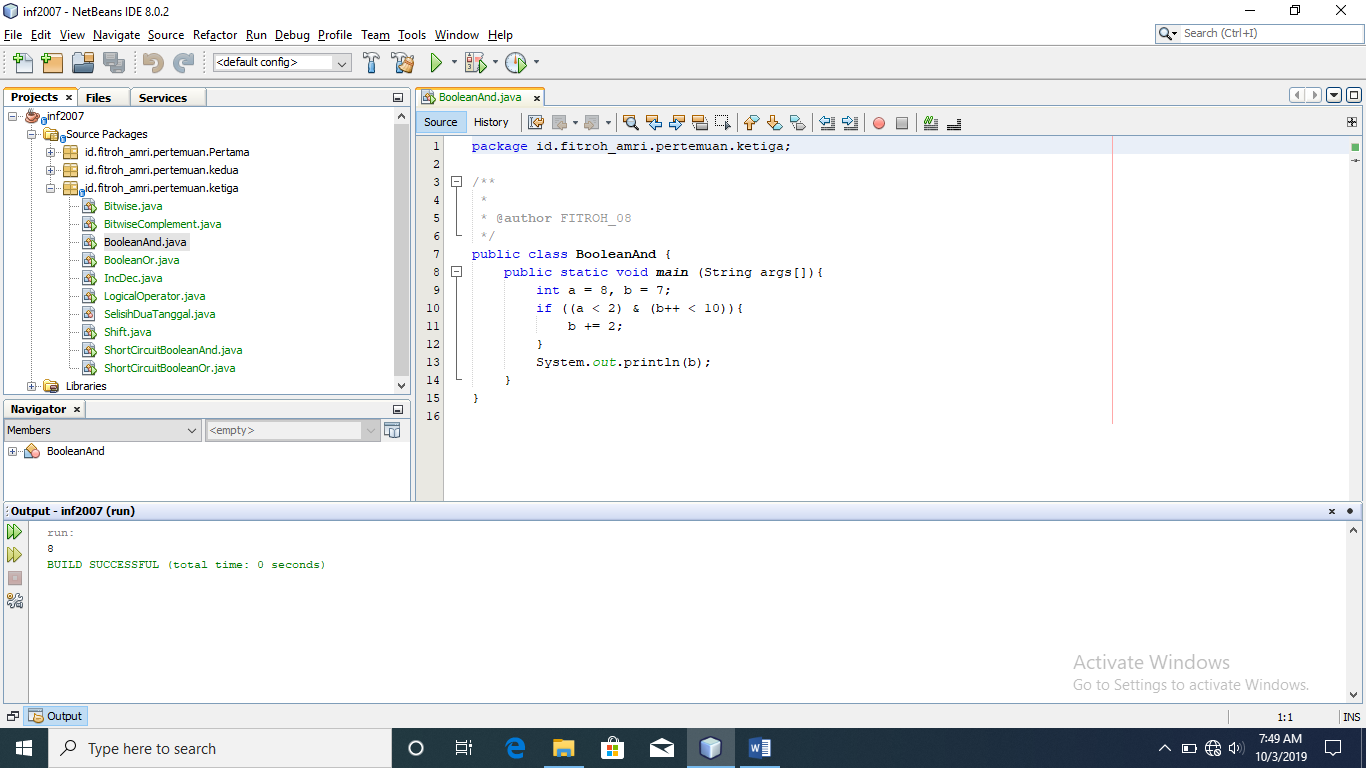


Output :

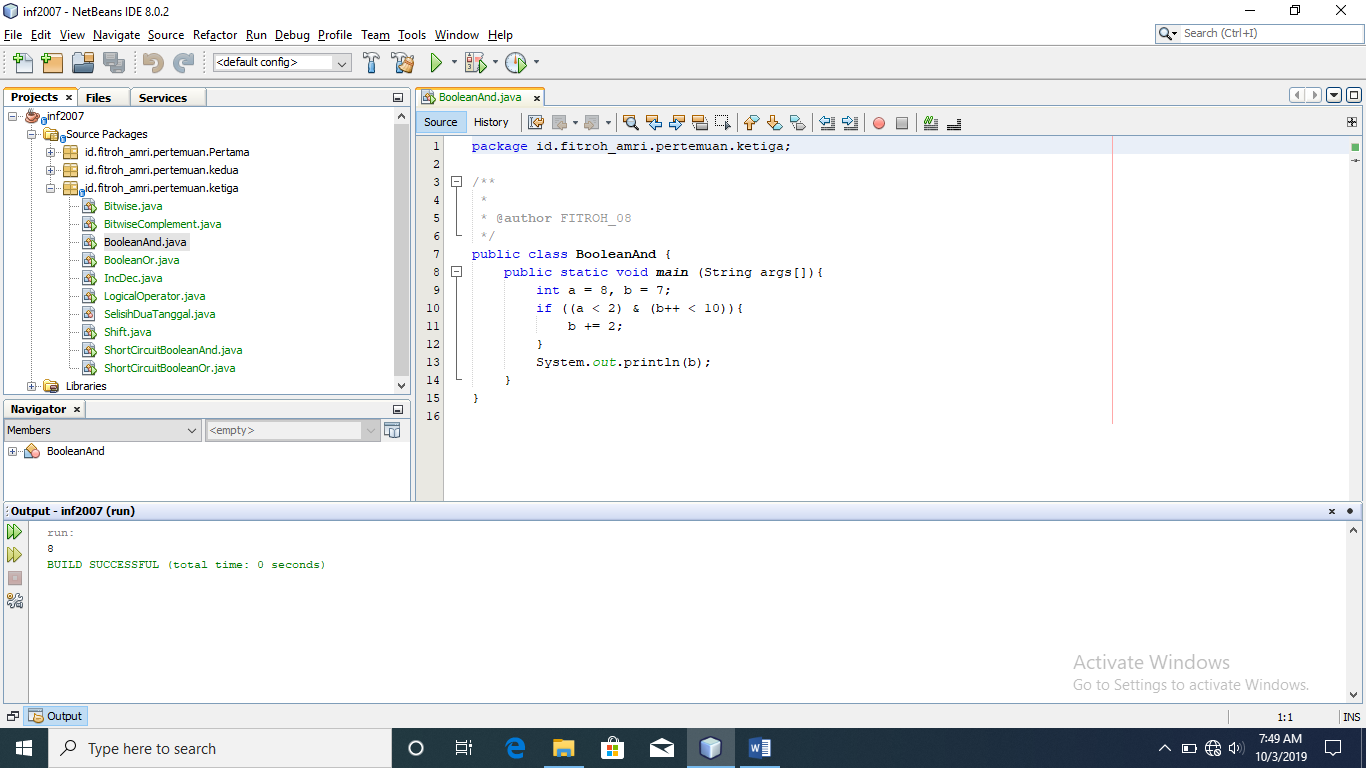


1. Menggunakan operator boolean AND.

Source Code :

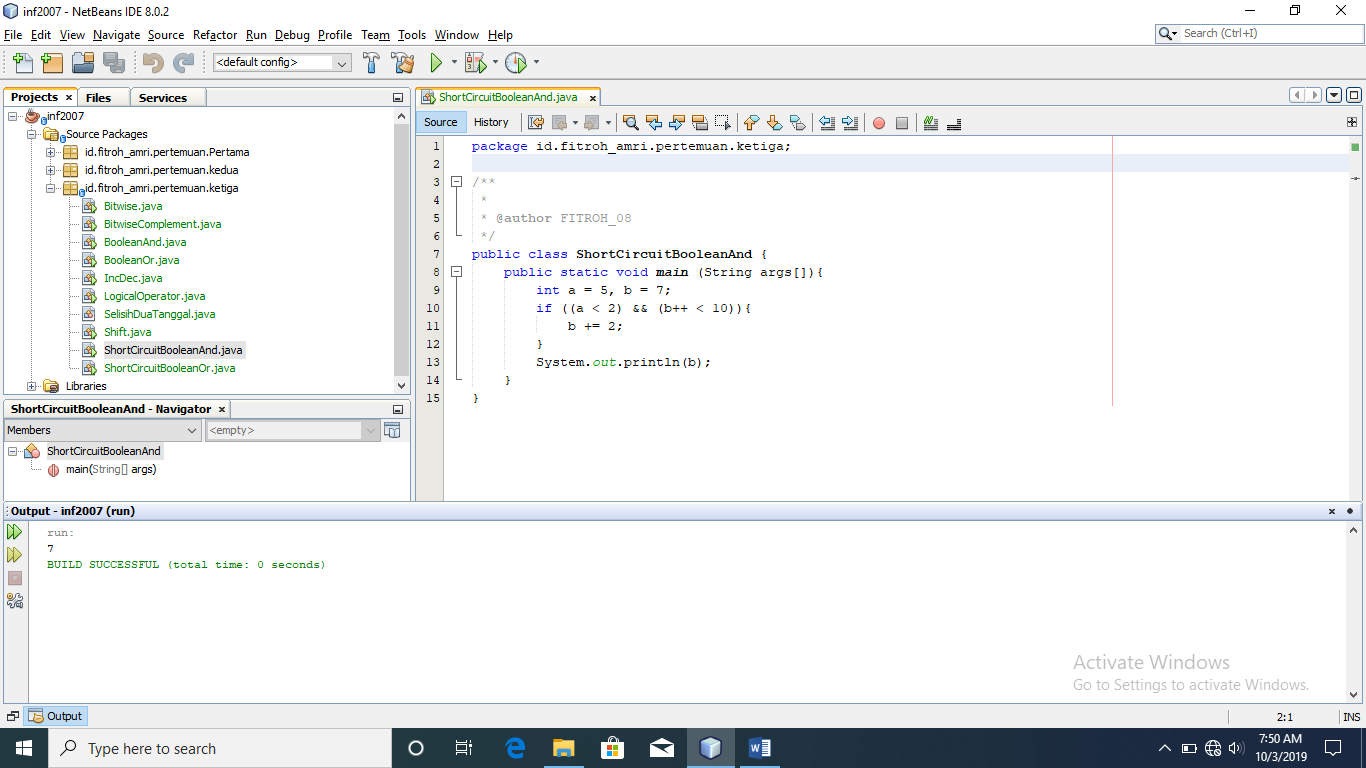


Output :

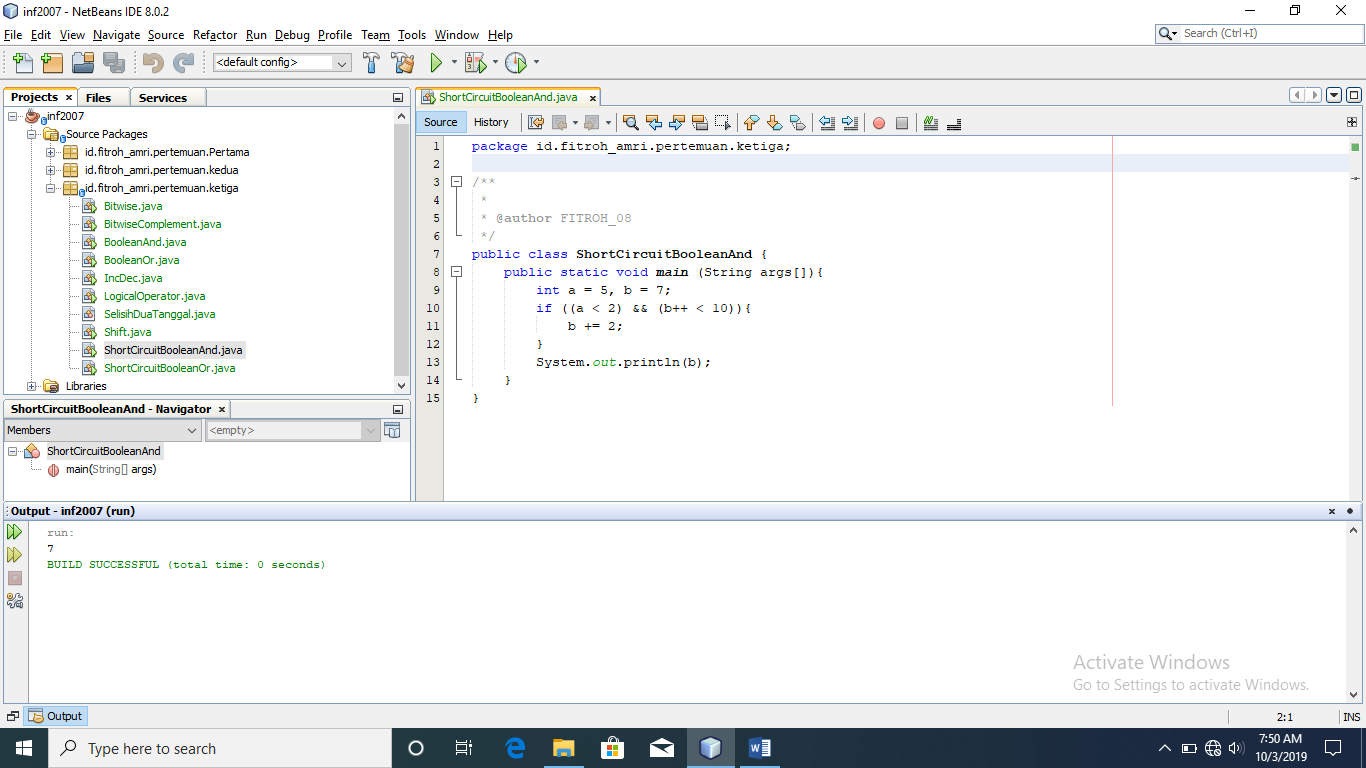


1. Menggunakan operator boolean and short-circuit.

Source Code :

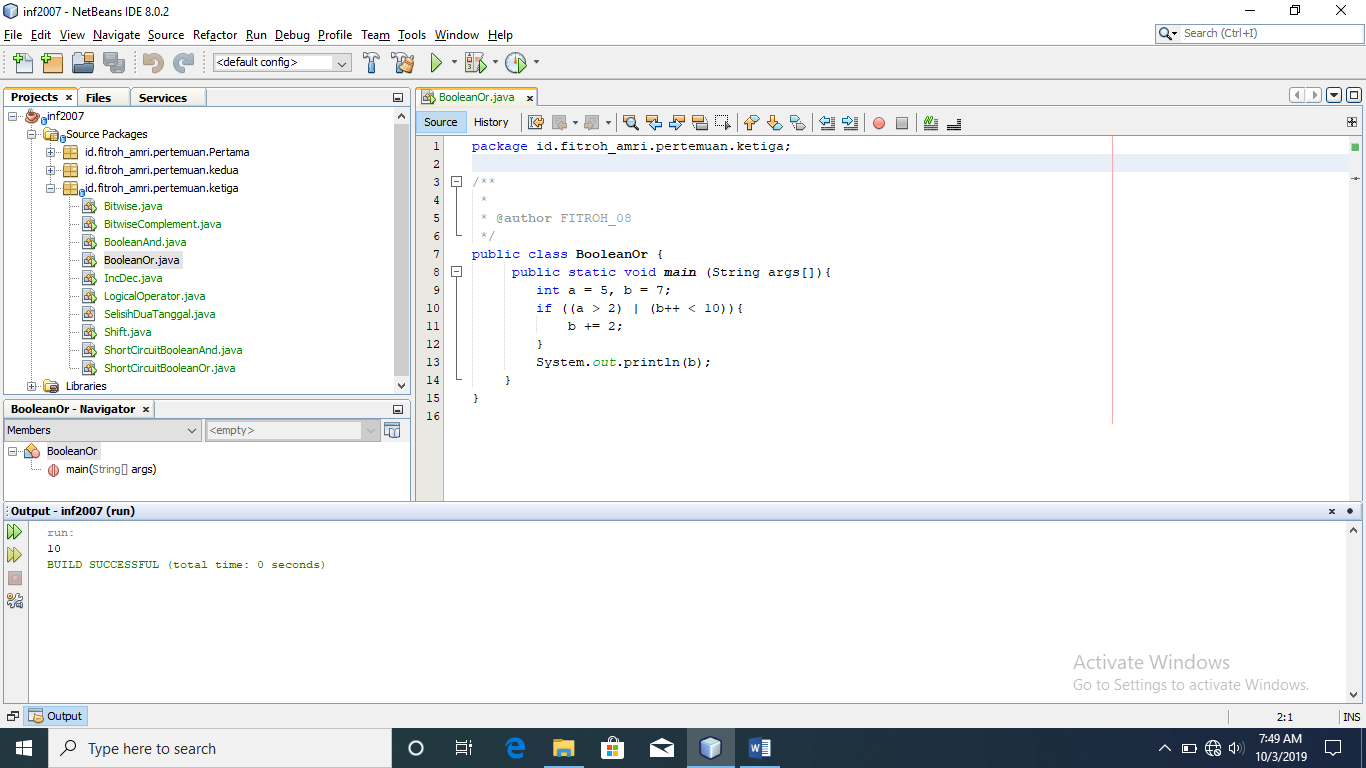


Output :

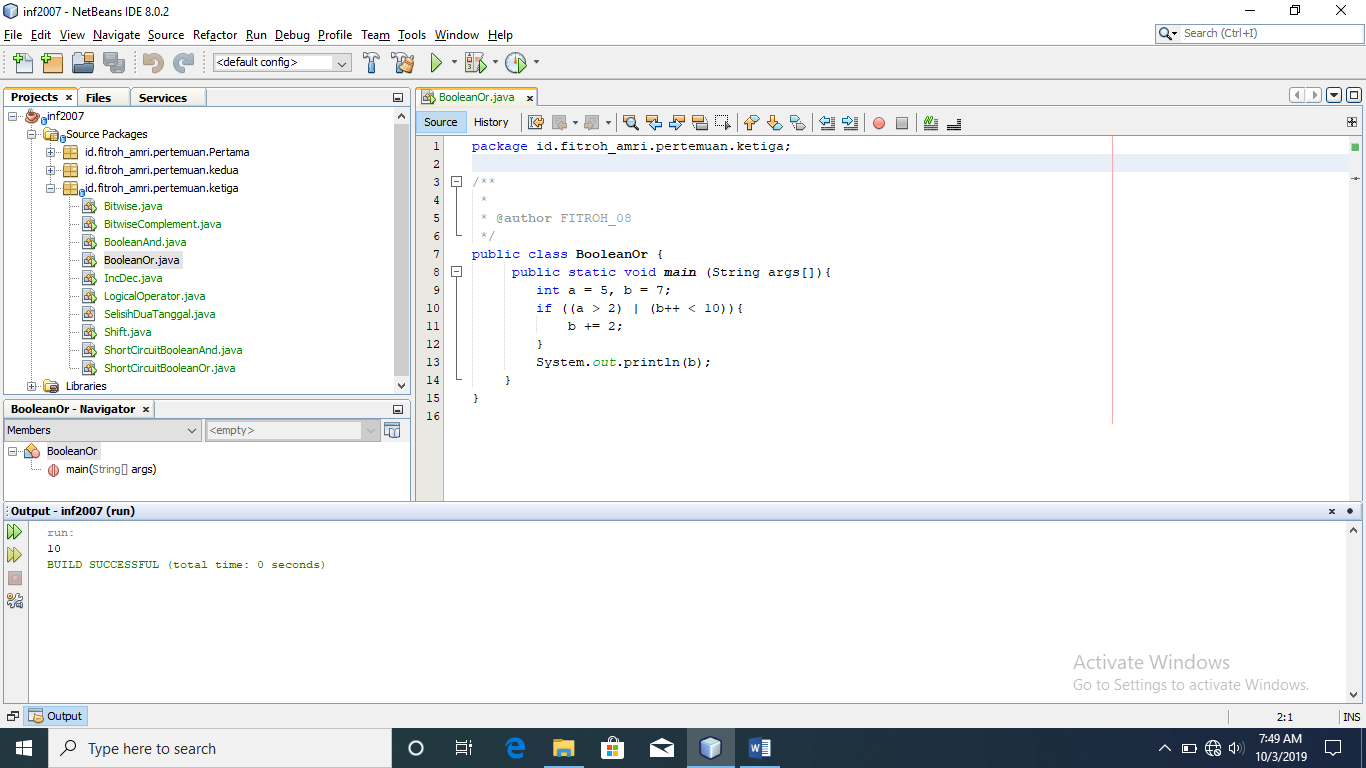


1. Menggunakan boolean OR.

Source Code :

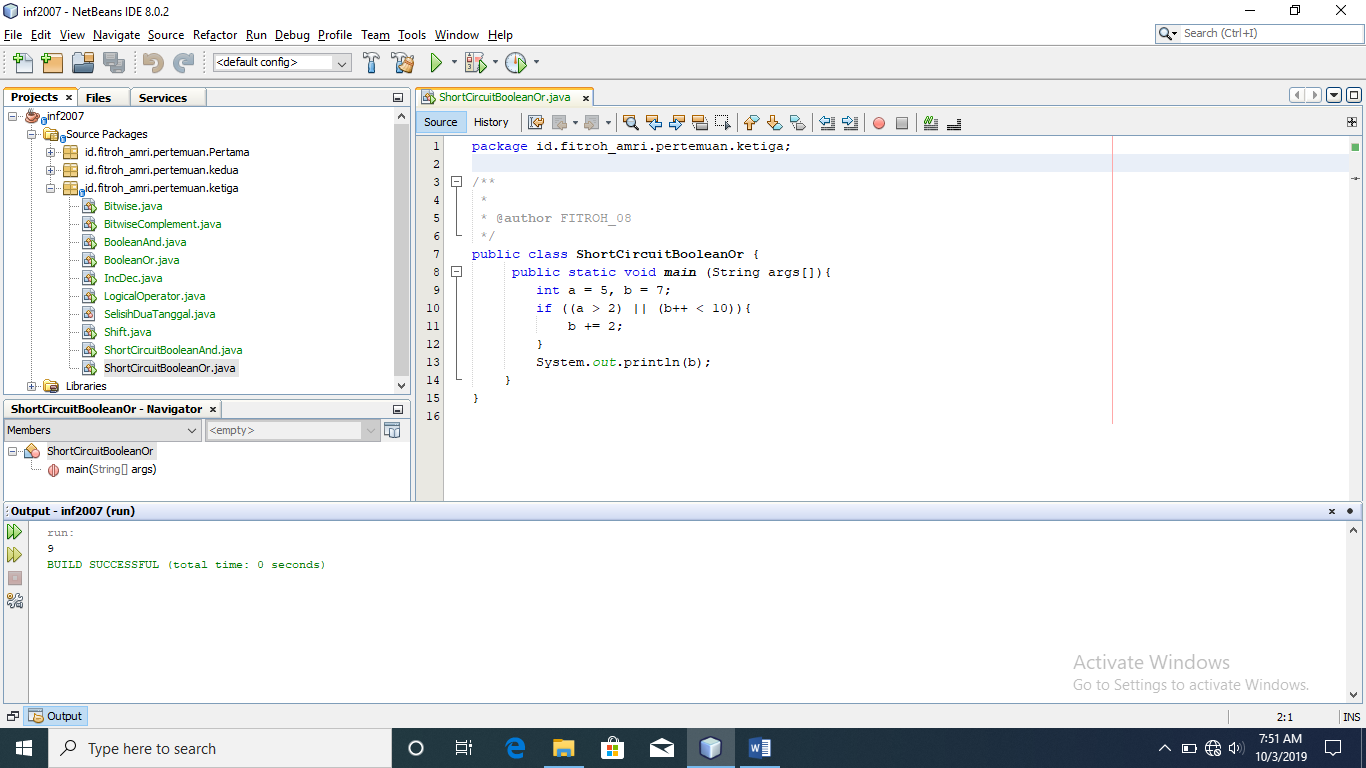


Output :

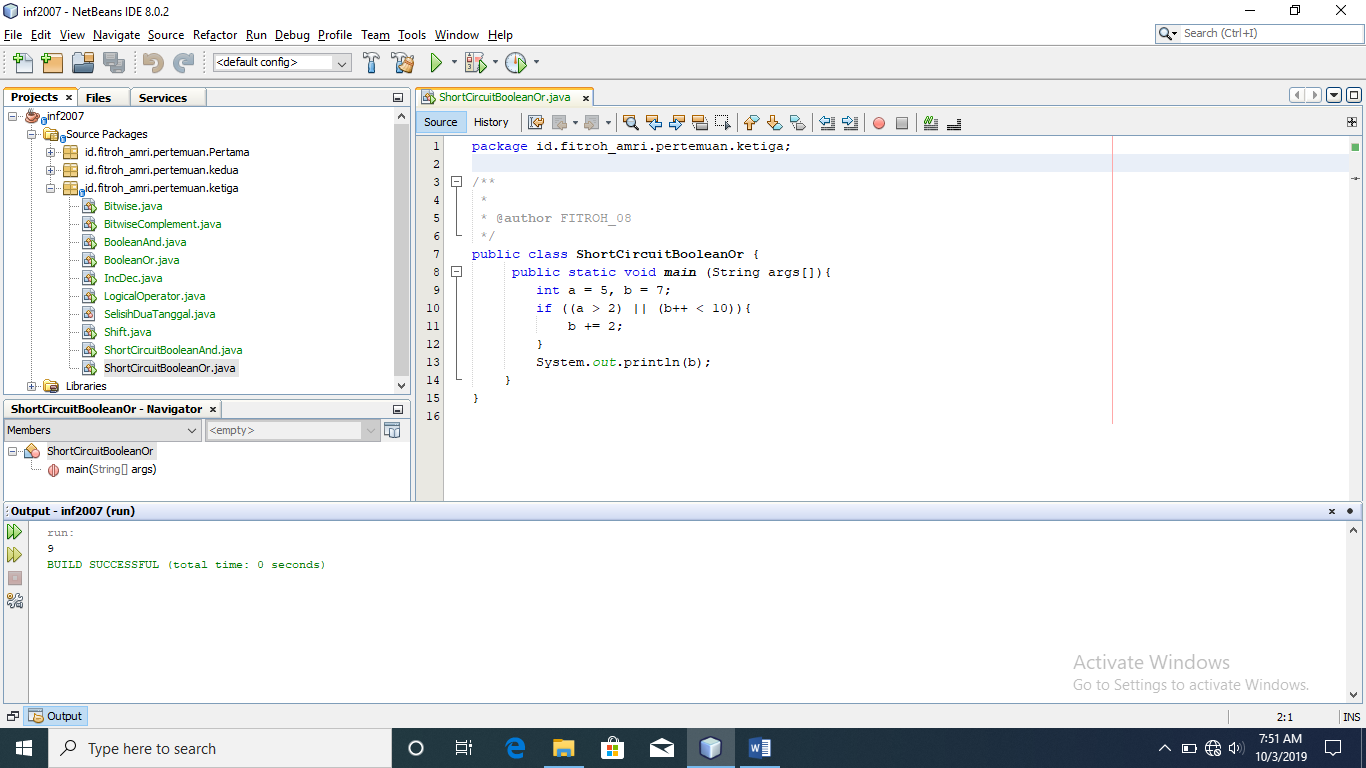


1. Menggunakan operator boolean or short-circuit.

Source Code :

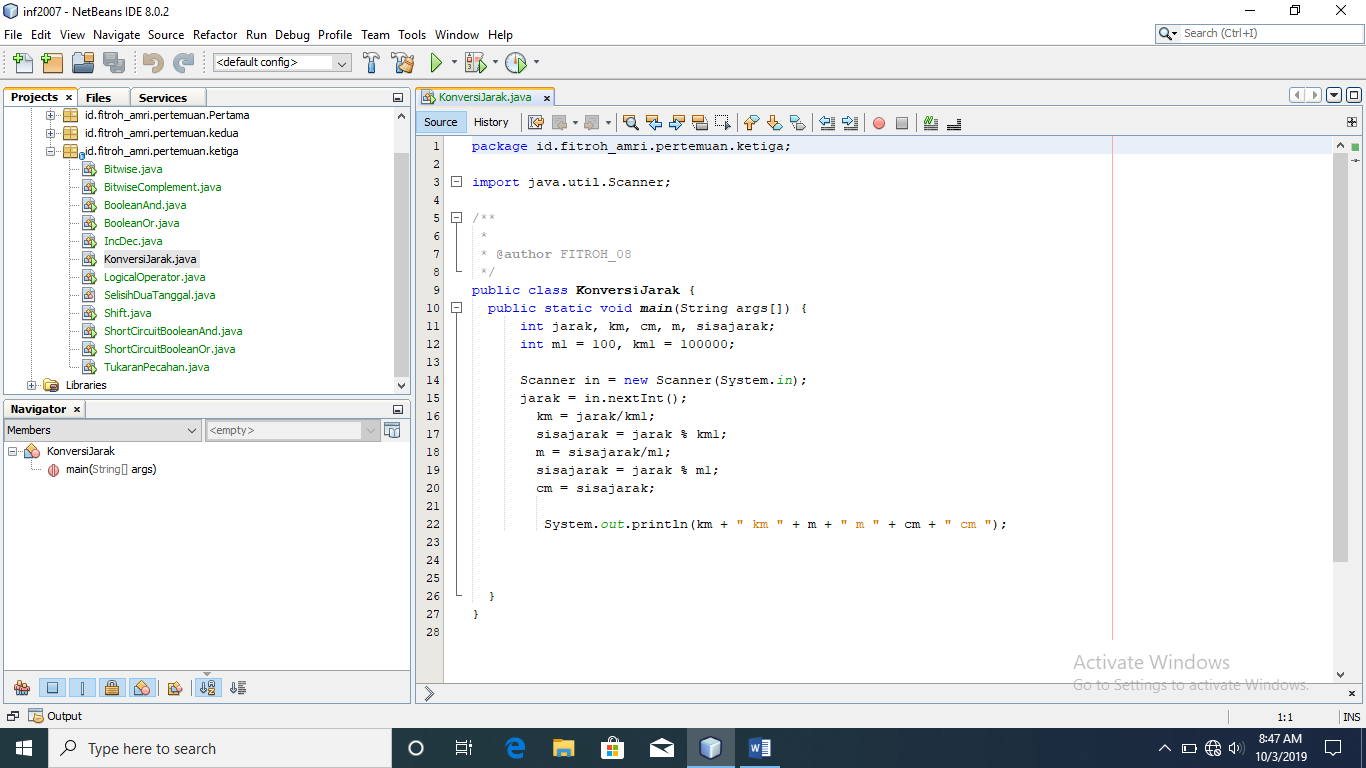


Output :

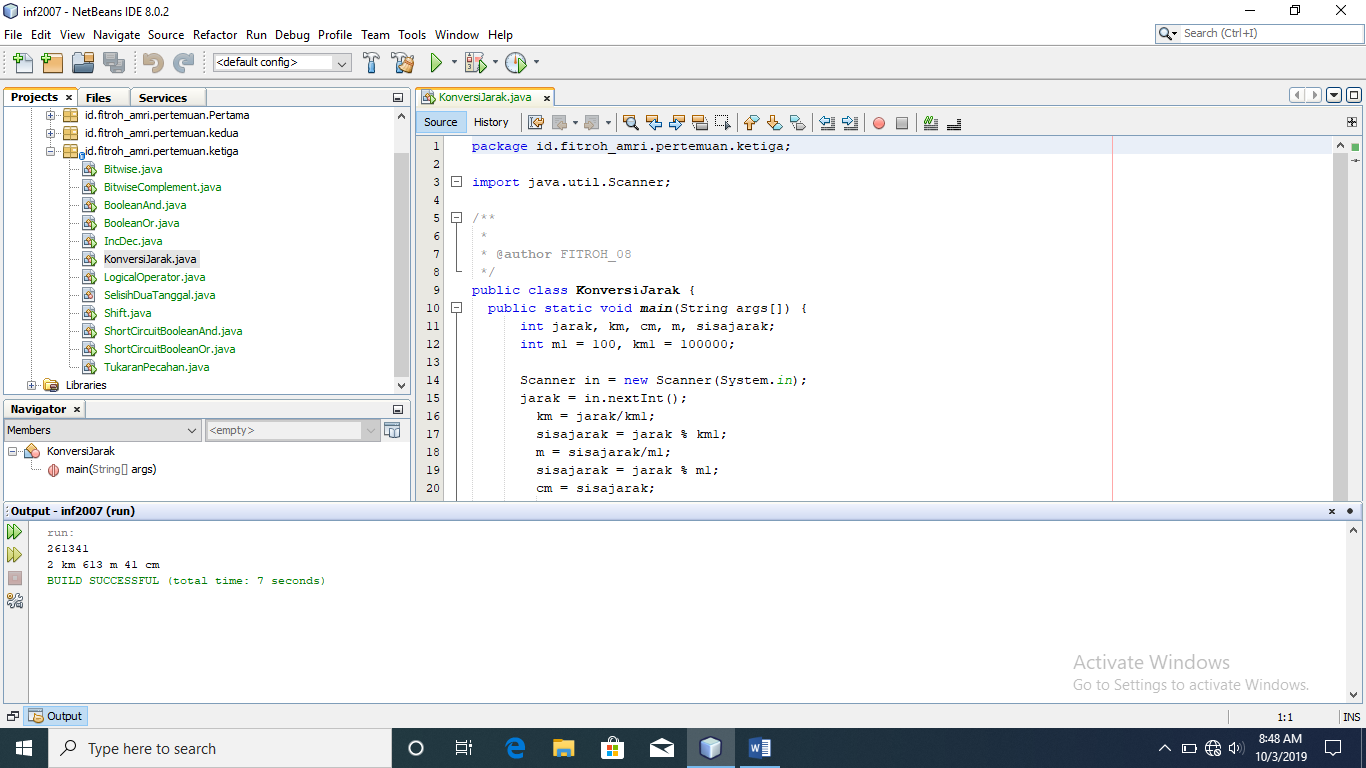


1. **Tugas**
2. Membuat program untuk menghitung selisih dari dua tanggal.

Source Code :

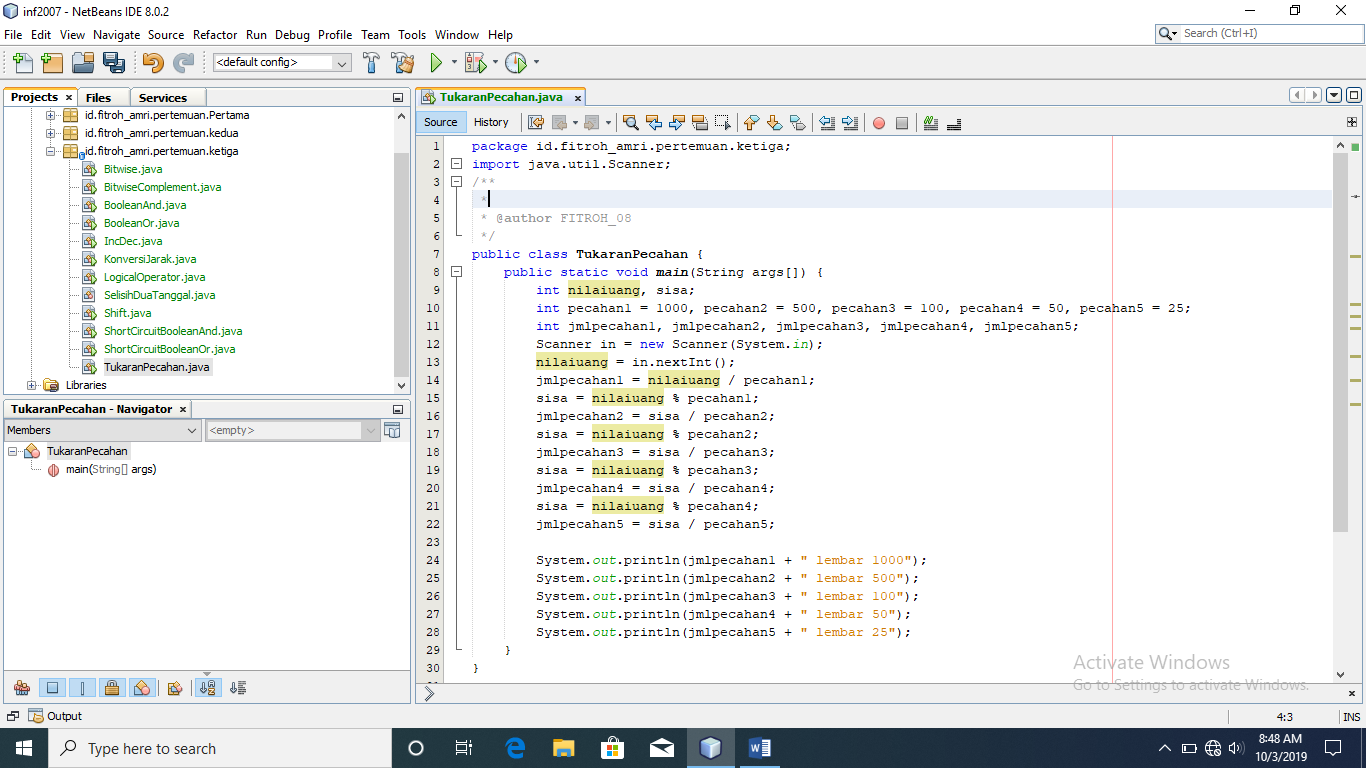


Output :

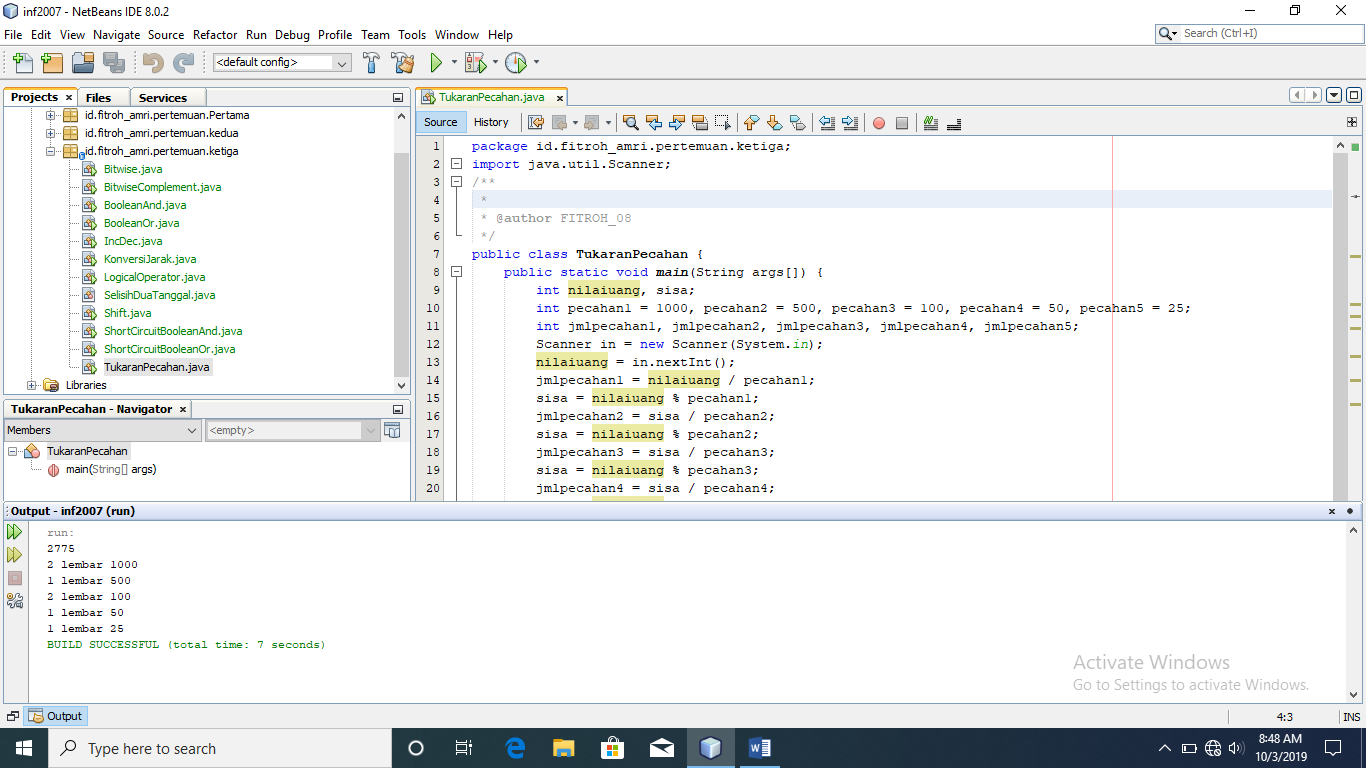


1. Membuat program yang menentukan nilai tukar pecahan.

Source Code :

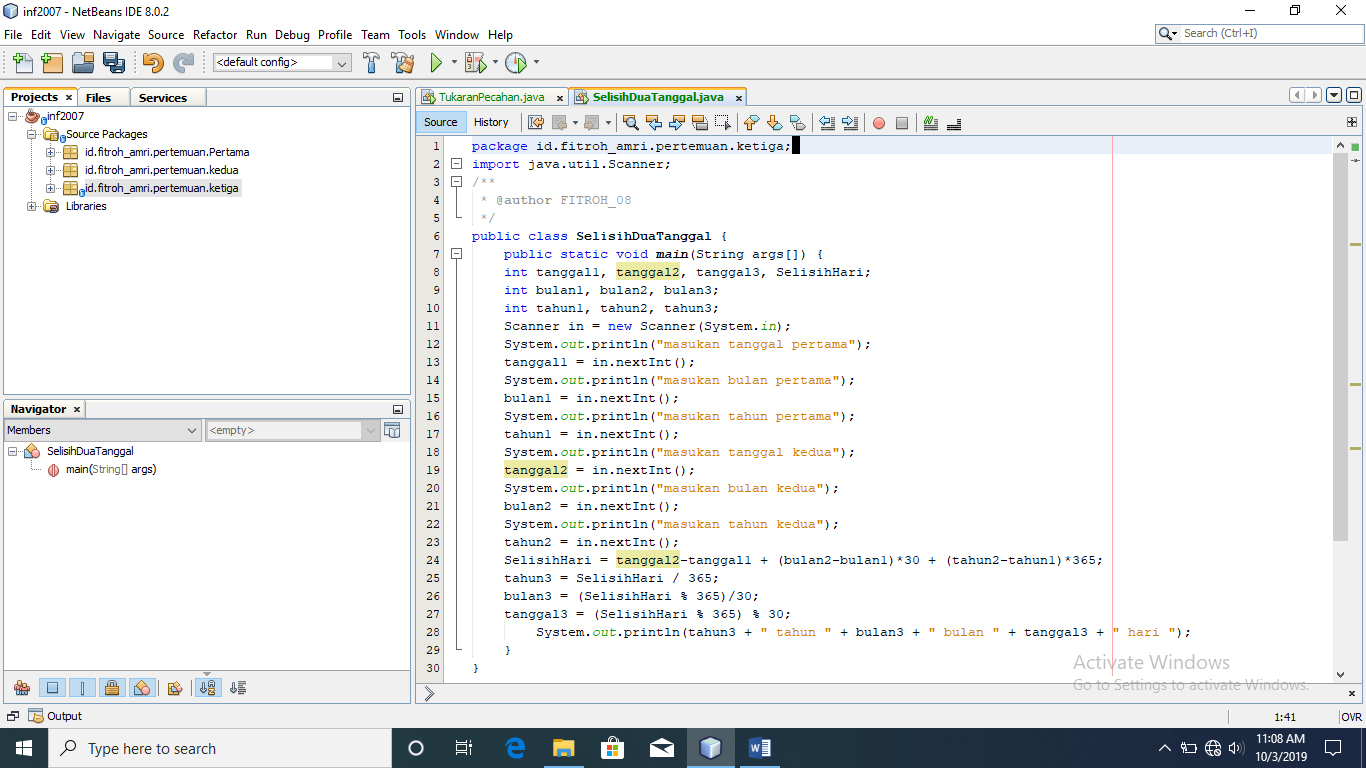


Output :



1. Membuat program untuk mengkonversi jarak.

Source Code :



Output :

