# Rangkuman

## Nama : Hikmatiar Ramadhan

# 1 .Algoritma

Algoritma adalah urutan langkah - langkah untuk menyelesaikan masalah yg di susun secara sistematis dan logis , sitematis dan logis artinya langkah dalam algoritma harus masuk akal serta tersusun rapi ,serta algoritma juga harus efektif pengertian efektif disini adalah setiap langkah harus sederhana sehingga tidak memakan waktu yg cukup lama dan tetap masuk akal. Contohnya kita aka memasak telur dadar :

1. Siapkan bahan2nya terlebih dahulu, telur,minyak goreng ,wajan,kompor dan garam
2. Panaskan minyak dalam wajan dengan kompor
3. Masukan telur ke wajan diamkan selama 4 menit lalu masukan garam ke telur
4. Matikan kompor
5. Bila terlihat sudah matang tiriskan telur
6. Telur dadar siap di hidangkan.

3 cara penulisan algoritma :

**- Structured English (SE)**  
SE merupakan alat yang cukup baik untuk menggambarkan suatu algoritma. Dasar dari SE adalah Bahasa Inggris, namun kita dapat memodifikasi dengan Bahasa Indonesia sehingga kita boleh menyebutnya sebagai Structured Indonesian (SI).  
SE atau SI lebih tepat untuk menggambarkan suatu algoritma yang akan dikomunikasikan kepada pemakai perangkat lunak.

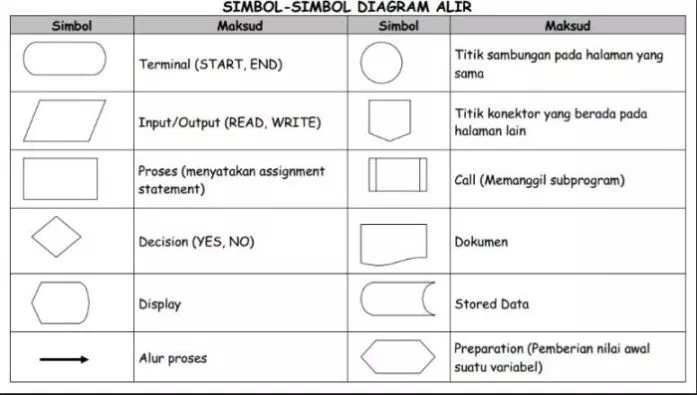
**- Pseudocode**  
Pseudocode adalah kode yang mirip dengan instruksi kode program sebenarnya.Pseudo berarti imitasi atau tiruan atau menyerupai, sedangkan code menunjuk pada kode program.  
**Contoh Pseudocode :**

1. Start
2. read alas, tinggi
3. Luas = 0.5 \* alas \* tinggi
4. Print Luas
5. Stop

**- Flowchart**

*flowchart* atau bagan alir adalah skema / bagan (*chart*) yang menunjukkan aliran (*flo*w) di dalam suatu program secara logika. *Flowchart merupakan alat yang banyak digunakan untuk menggambarkan algoritma dalam bentuk notasi-notasi tertentu.*

Simbol dalam Flowchart :



**2 .JavaScript**

JavaScript adalah bahasa pemrograman web yang bersifat *Client Side Programming Language*. Client Side Programming Language adalah tipe bahasa pemrograman yang pemrosesannya dilakukan oleh *client*. Aplikasi *client* yang dimaksud merujuk kepada *web browser* seperti Google Chrome dan Mozilla Firefox.

Dasar - dasar materi pemograman :

**- Tipe data**

Tipe data merupakan setiap himpunann yang ada pada semua data tanpa terkecuali

1. **Integer**, tipe data integer ini digunakan untuk mendeklarasikan data/variable sebagai bilangan bulat. Misalkan nilai 10, 7,29,100, adn 22. Penggunaan tipe data integer biasanya digunakan untuk data yang akan dilakukan pengoprasian penjumlahan, pengurangan, pembagian, atau perkalian.

*Contoh : Console.log(10);*

1. ***Float***, seperti halnya integer, tipe data ini digunakan untuk data angka, namun untuk angka/nilai yang berupa pecahan. Jadi apabila ingin mendapatkan hasil yang terbaik pada suatu proses pengoprasian angka, maka kita bisa menggunakan tipe data float. Contohnya nilai 1.5, 3.5, 3.14, dan lain bilangan pecahan lainnya.

*Contoh : Console.log(2,9);*

1. ***Char***, char adalah tipe data yang digunakan untuk deklarasi variable berupa karakter. Karakter tersebut bisa berupa simbol, alfabet, atau angka( 1 karakter)

*Contoh : Console.log(“A”);*

1. ***String***, khusus untuk string, tipe data yang digunakan untuk mendeklarasikan data yang berisi kalimat atau kata.

*Contoh : Console.log(“ Hallo, Selamat Datang. ”);*

1. ***Boolean***, tipe data boolean ini hanya digunakan untuk data yang mempunyai 2 nilai saja, yaitu **true** atau **false**.

*Contoh : Console.log(true);*

*Contoh : Console.log(false);*

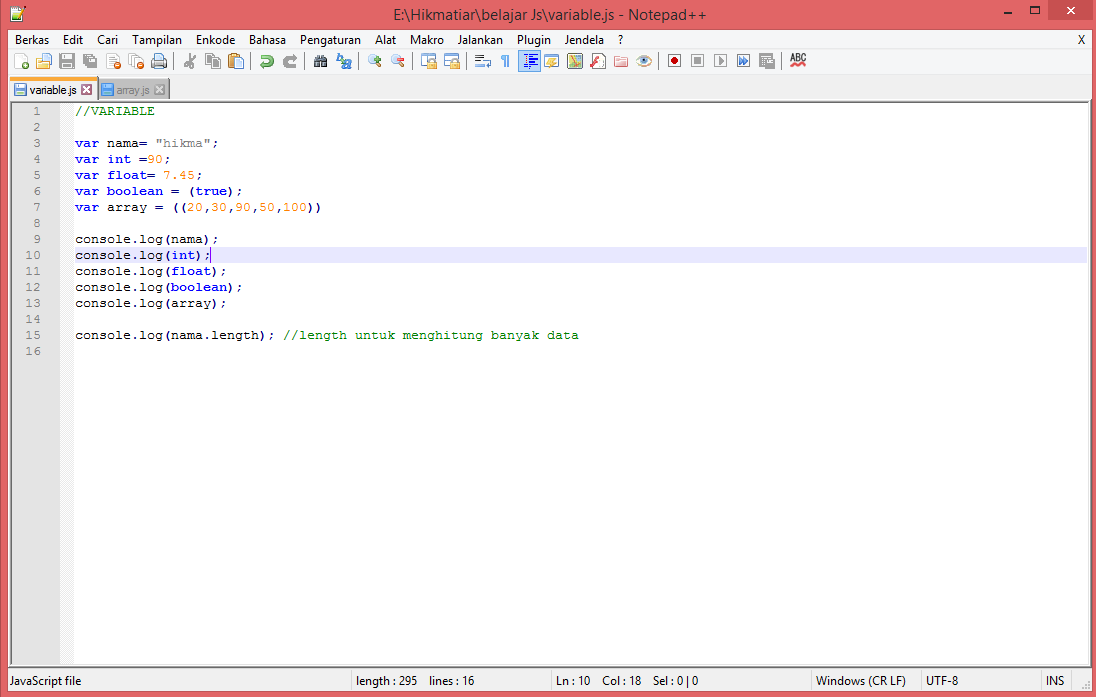
# 2 .Variable dan konstanta

**- Variable** adalah suatu tempat untuk menampung data yang nilainya selalu berubah

**- Konstanta** adalah suatu tempat untuk menanpung data yang nilainya selalu tetap Dan tidak pernah berubah .

Di dalam bahasa pemrograman seperti C++, C, C#, pembuatan variabel harus ditentukan tipe data, tipe data digunkan untuk menentukan jenis nilai atau value yang akan disimpan ke dalam variable untuk kemudian di proses dlam program .

Perbedann antara variable dan konstanta yaitu pada wadah tersebut .jika variable memliki yg dapat berubah rubah maka konstanta bersifat tetap.



**- Array**

Satu tipe data yang di dalamnya memuat banyak data dan memiliki sebuah *Index : sebagai pembeda antara data 1 dan data yang lain,dan juga sebagai petunjuk data.*Penulisan array menggunakan simbol [ ] , dan tanda ( , ) adalah pemisah antara data 1 dan data yang lain.

Array di bagi menjadi 2 ;

***Array 1 Dimensi***

adalah sekelompok data yang memiliki nama variable dan tipe data yang sama yang dapat diakses menggunakan 1 buah index saja.

Contoh array ;

***var data = [10,50,20,70];***

***console.log(data[2]);***

***Array Multidimensi***

adalah kumpulan data yang memiliki nama dan tipe data yang sama dimana elemennya dapat diakses menggunakan 2 buah index, bisa juga disebut sebagai index baris dan index kolom.

Contoh array :

***var data\_2 = [[10,10],[100,20],140,500];***

***console.log(data\_2[1][1]);***

Array Assosiative

Associative array adalah sebuah array yang indexnya dapat didefinisikan sendiri berupa label (string). Index pada associative array sering disebut dengan key (bukan berupa angka tapi berupa string).

# - ****Operator****

Ada beberapa jenis operator, yaitu :

1. **Assignment Operator / Operator Penugasan**

  Assignment Operator (operator penugasan) adalah operator yang menggunakan tanda sama dengan (=) untuk mengisi sebuah nilai dalam suatu variabel.

1. **Arithmetic Operator / Operator Aritmatika**

***Arithmetic Operator*** (operator aritmatika) adalah operator yang digunakan untuk melaksanakan operasi aritmatika.

Beberapa operator aritmatika antara lain:

\* : untuk perkalian

+ : untuk penjumlahan

- : untuk pengurangan

/ : untuk pembagian

% : untuk sisa pembagian (modulus)

**1. Operator Tambah (+)**

contoh :

x=5

x+5

hasil :

x=10

**2. Operator Kurang (-)**

contoh :

x=5

x-2

hasil :

x=3

**3. Operator Kali (\*)**

contoh :

x=3

x\*5

hasil :

x=15

**4. Operator Bagi (/)**

contoh :

x=6

x/3

hasil :

x=2

**5. Operator Modulus / Sisa Hasil Bagi (%)**

contoh :

20%9

hasil :

2

contoh lain :

40%5

hasil :

0

**6. Operator Increment (++)**

contoh :

x=5

x++

hasil :

x=6

**7. Operator Decrement (–)**

contoh :

x=6

x–

hasil :

x=5

1. **Operator Logika / Boolean Operator**

**Operator Boolean** atau **Operator Logika** adalah operator yang digunakan untuk melakukan operasi logika yaitu operator yang menghasilkan nilai TRUE (benar) atau FALSE (salah).

Bebarapa macam operator logika antara lain:

**1**. **and** : menghasilkan nilai TRUE jika kedua karakter bernilai TRUE

**2.** **or** : menghasilkan nilai TRUE jika salah satu karakter bernilai TRUE

**3.** **xor** : menghasilkan nilai TRUE jika salah satu karakter bernilai TRUE tetapi bukan keduaduanya bernilai TRUE

**4**. **!** : mengasilkan nilai tidak TRUE

**5. &&**  : menghasilkan nilai TRUE jika kedua karakter bernilai TRUE

**6**. **||** : menghasilkan nilai TRUE jika salah satu karakter bernilai TRUE

**1. Operator AND (&&)**

contoh :

x=6

y=3

(x < 10 && y > 1)

hasil :

TRUE

**2. Operator OR (||)**

contoh :

x=5

y=3

(x == 5 || y == 1)

hasil :

TRUE

**3. Operator NOT (!)**

contoh :

x=3

y=5

!(x==y)

hasil :

TRUE

1. ***Operator Perbanding***

**Operator Pembanding** adalah operator yang digunakan untuk membandingkan dua buah nilai atau karakter .

Operator perbandingan ini antara lain :

**<** : untuk kurang dari

**>** : untuk lebih dari

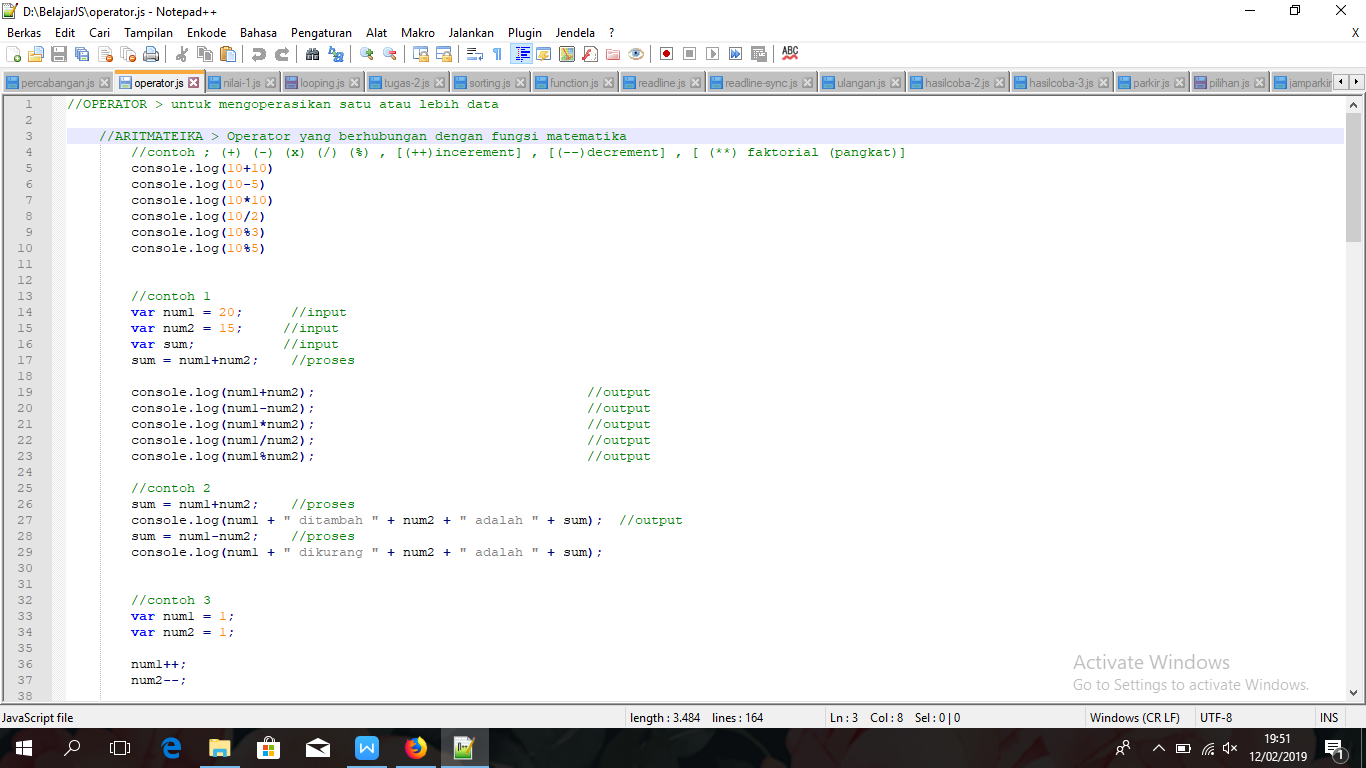
**<=** : untuk kurang dari atau sama dengan

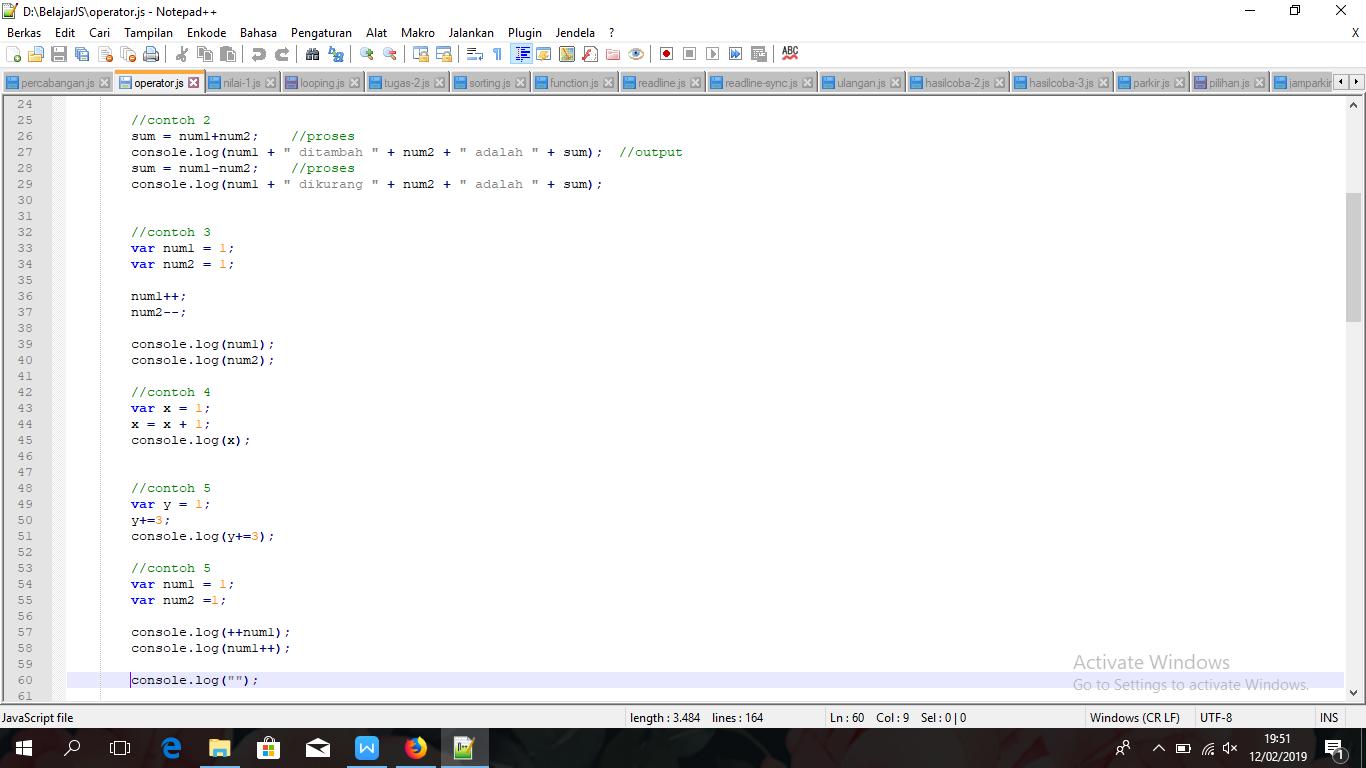
**>=** : untuk lebiih dari atau sama dengan

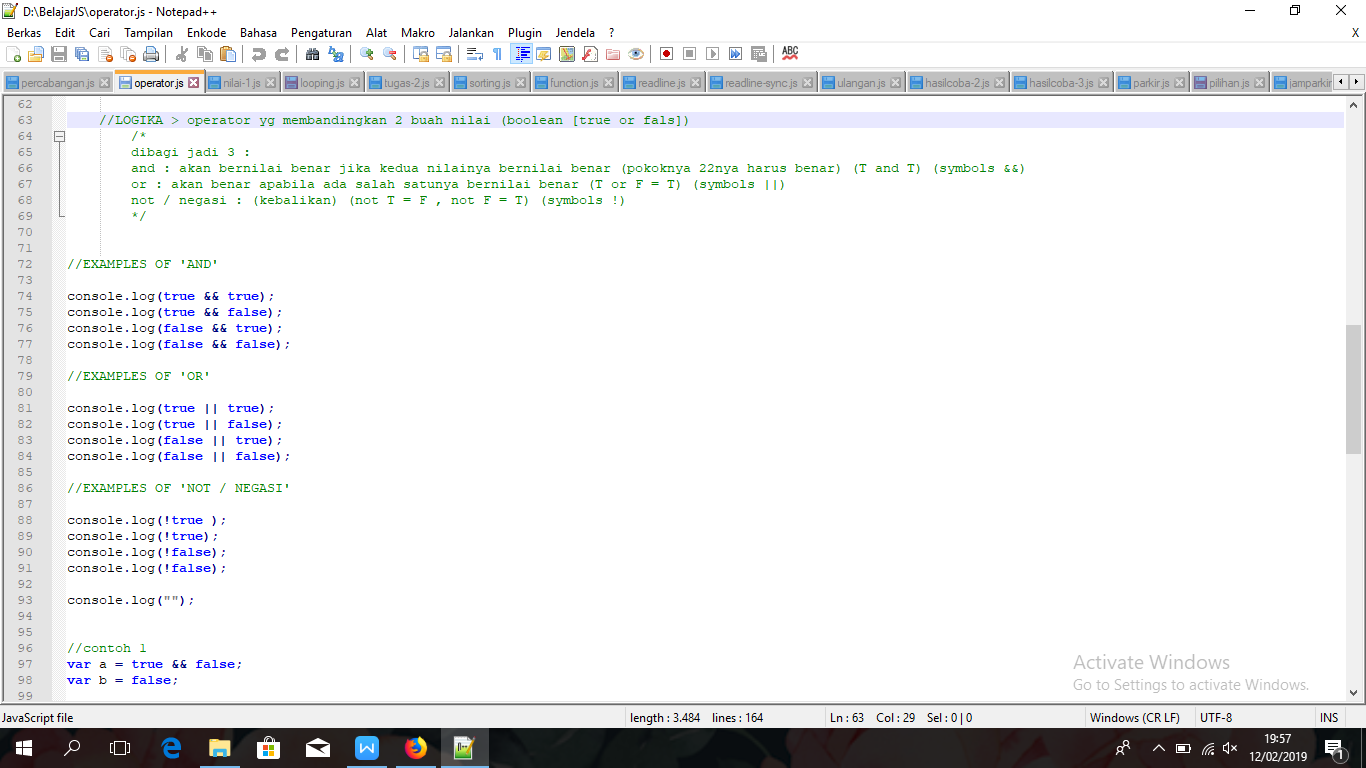
**==** : untuk sama dengan

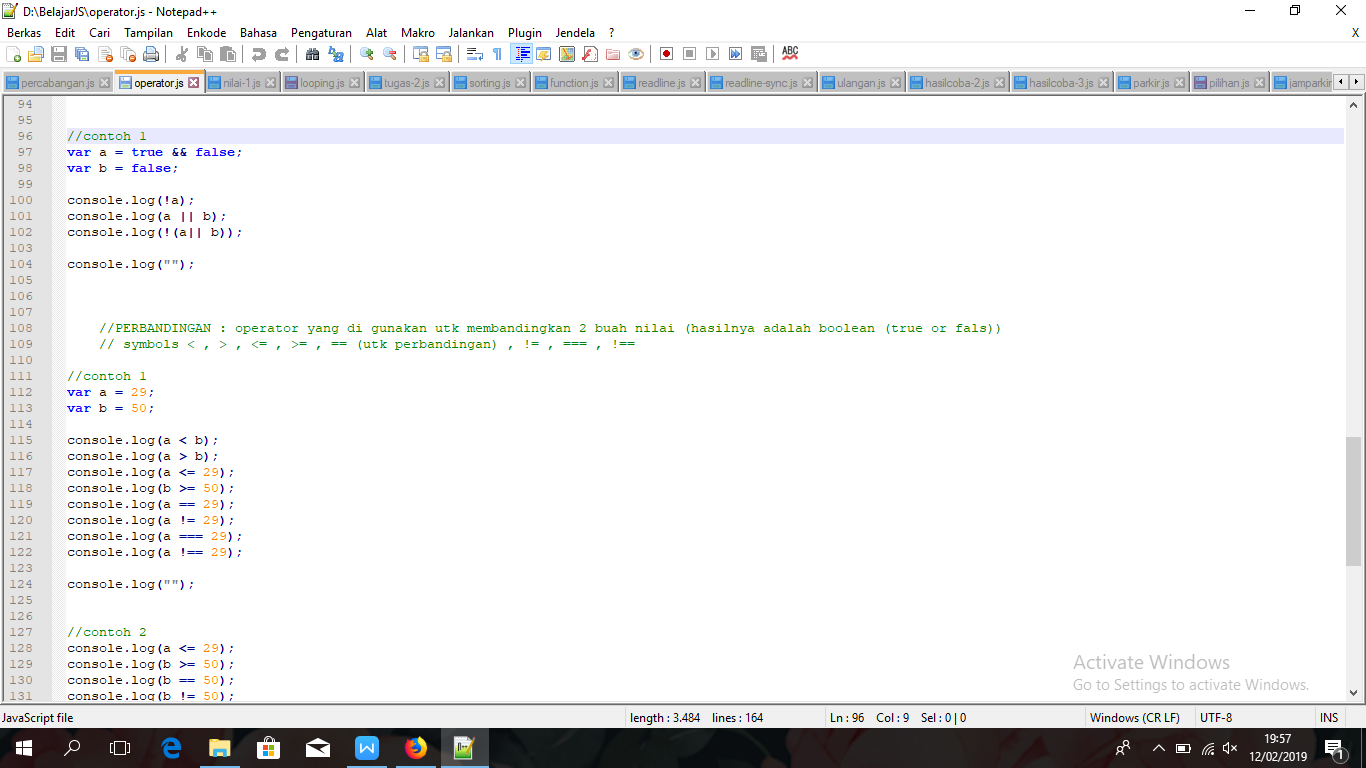
**!=** : untuk tidak sama dengan

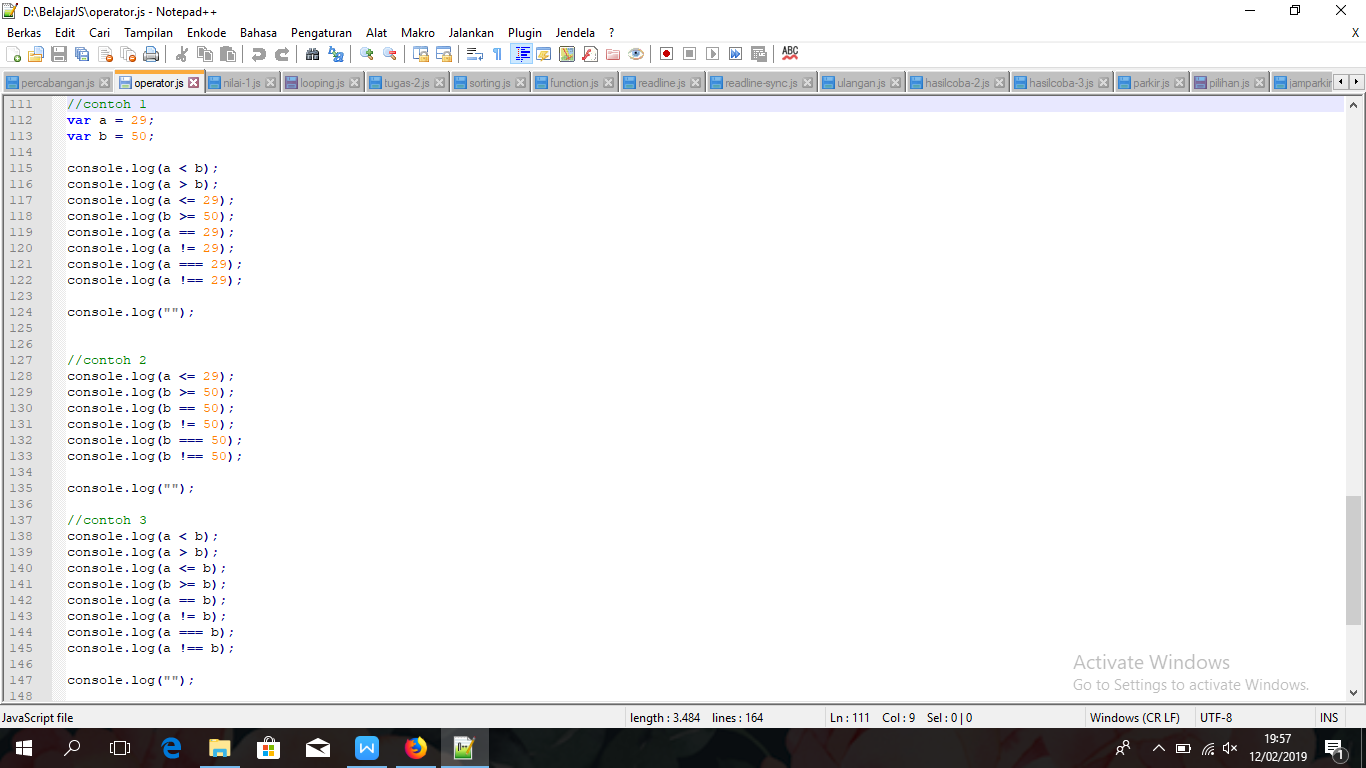
**<>** : untuk tidak sama dengan

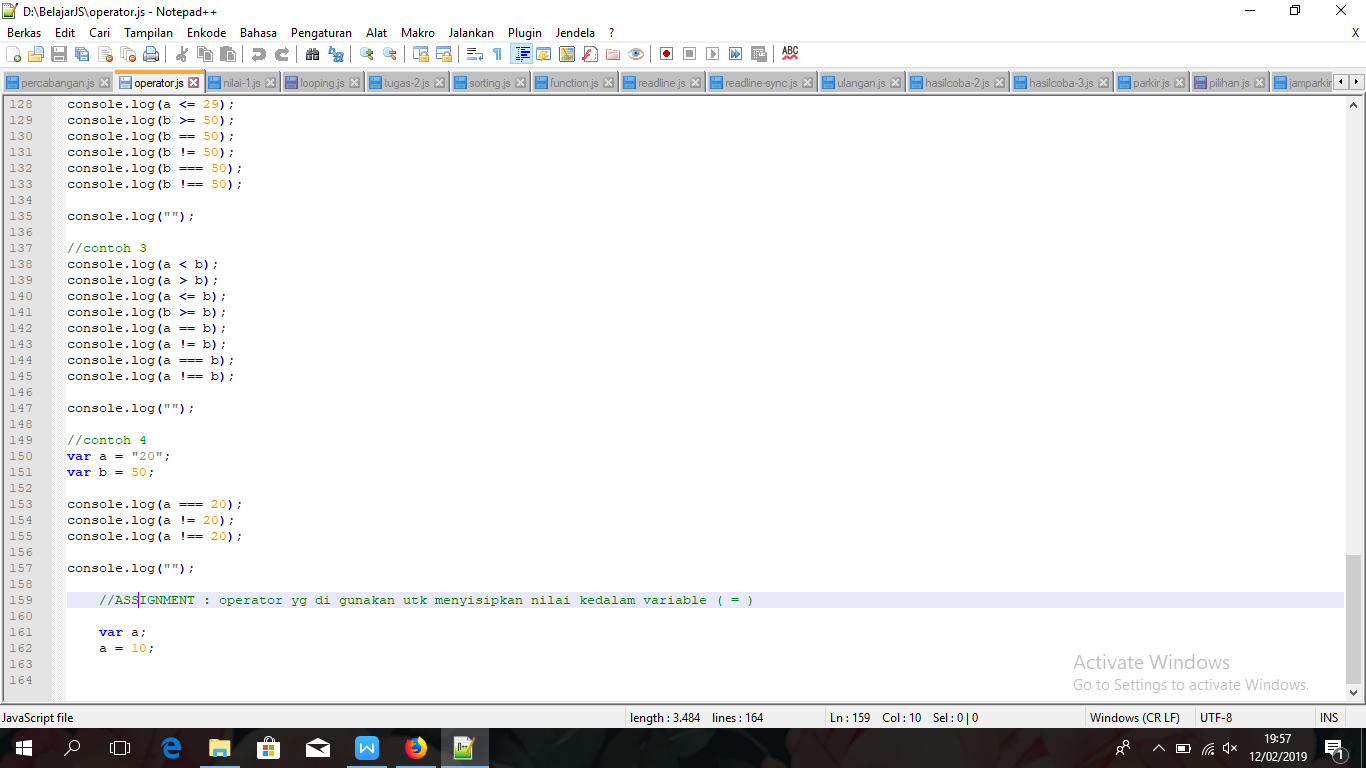
****











# ****- Struktur control/Percabangaan/Flow control****

**Stuktur control** adalah bagian yang sangat penting dan tidak bisa di pisahkan di dalam semua bahasa pemrograman karena memiliki fungsi dan tugas utama , yaitu mengatur bagaimana baris - baris akan di eksekusi.

Pada dasarnya, baris-baris program akan dieksekusi secara sekuensial (berurutan dimulai dari baris paling atas, kemudian ke baris berikutnya). Dengan adanya flow control, kita bisa mengatur baris program dieksekusi, baik berulang-ulang meloncat, maupun berhenti.

Kondisional  : if, if-else, Switch-Case

•  Pengulangan : while, do-while, for

•  Lain-lain : break, return, continue

Dalam struktur kontrol percabangan akan diberikan beberapa pilihan dimana pilihan tersebut harus diseleksi berdasarkan kondisi yang ada pada pilihan. Kondisi yang sudah dipilih itu yang akan menunjukkan proses seperti apa yang harus diambil oleh program untuk step selanjutnya. Apabila kondisinya mendapat nilai benar yang diperoleh adalah program akan dilanjutkan hingga keinginan tercapai. Namun bila kondisi memperoleh nilai salah maka program tidak akan melakukan atau memberikan perintah akses apapun untuk ketingkat selanjutnya.

Percabangan berbeda dengan pengulangan(Looping). Syarat suatu program akan dieksekusi adalah kondisi harus bernilai true. Sedangkan pada perulangan program akan terus dijalankan secara berkali-kali sampai menemukan kondisi yang terpenuhi.

1. **Operasi Kondisional**

****- If****

Logika percabangan ini akan memberikan beberapa pilihan yang apabila pernyataannya sesuai dengan kondisi makan kondisi tersebut akan bernilai TRUE. Percabangan IF memiliki tindak lanjutan yaitu dengan menambahkan ELSE pada perintah selanjutnya. Percabangan ini disebut percabangan IF-ELSE. Pada if-else akan ada kondisi bernilai false. Dimana apabila kondisi tersebut mendapat nilai false tindakan selanjutnya adalah program tidak akan tereksekusi. Maka presentasinya adalah apabila pernyataan sesuai dengan kondisi pada if yang sudah ditentukan maka kondisi akan bernilai true. Namun apabila pernyataan seperti yang terdapat pada kondisi di if akan diproses melalui perintah else. Dan jika masih tidak seperti yang tertulis pada kondisi maka kondisi akan bernilai false. Ada pula percabangan if-else-if. Percabangan if dapat dimodifikasi menurut apa yang dibutuhkan dari pembuatnya.

**- Switch-case**

Statement switch-case berfungsi mengeksekusi blok program dengan membandingkan nilai suatu

variabel sebagai pilihan dalam menentukan blok yang akan dieksekusi.

1. Pngulangan (Looping)