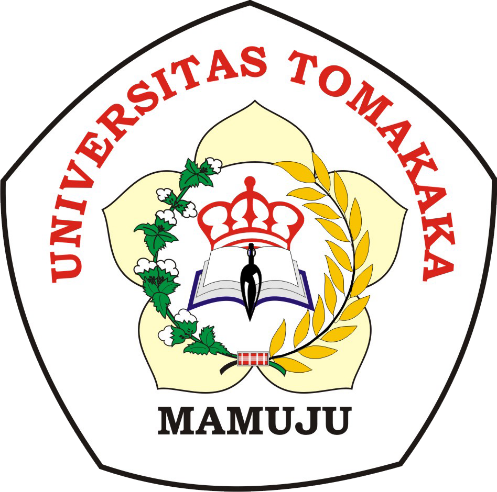
**PEMBAHASAN TENTANG BERBAGAI**

**MACAM LOOP DI JAVA**



**DISUSUN OLEH :**

**ADE IRMA ( 190250501001 )**

**PROGRAM STUDI SISTEM INFORMASI**

**FAKULTAS ILMU KOMPUTER**

**UNIVERSITAS TOMAKAKA MAMUJU**

**PENGERTIAN LOOPING, MACAM-MACAM DAN CONTOH**

**PENGERTIAN LOOPING**

Looping Berasal Dari Kata Loop Artinya Lingkaran, memiliki ciri melingkar atau kembali atau mengulang.

Looping artinya perulangan .

Adapun looping berarti suatu instruksi untuk membentuk aksi kerja secara berulang pada blok yang sama, dimana didalam blok yang akan di ulang ada terdapat statement atau pernyataan .

Didalam looping, ada 3 bagian yang mutlak harus ada agar terjadi perulangan yang diinginkan yang berlaku pada 3 perulangan yang sering digunakan yaitu For, Do While, While yaitu ;

* Inisialisasi : adalah untuk menyatakan keadaan awal dari variable control pada perulangan .
* Kondisi (Ekspresi Perbandingan) : adalah merupakan ekspresi yang berguna untuk mengakhiri perulangan.
* Iterasi : adalah bagian untuk mengatur/perubahan nilai dari variable control, iteriasi bisa berupa increment ( nilai dari inisialisai bertambah/menaik) atau Decrement (nilai dari inisialisasi berkurang/menurun)

**MACAM-MACAM LOOPING/LOOP**

1.   Loop/looping for

Perulangan for digunakan untuk mengerjakan pernyataan atau sekelompok pernyataan secara berulang, namun pernyataan perulangan for akan dijalankan jika sudah dapat dipastikan kapan perulangan berhenti. Dengan kata lain, jumlah perulangan yang dibutuhkan sudah diketahui sebelumnya.

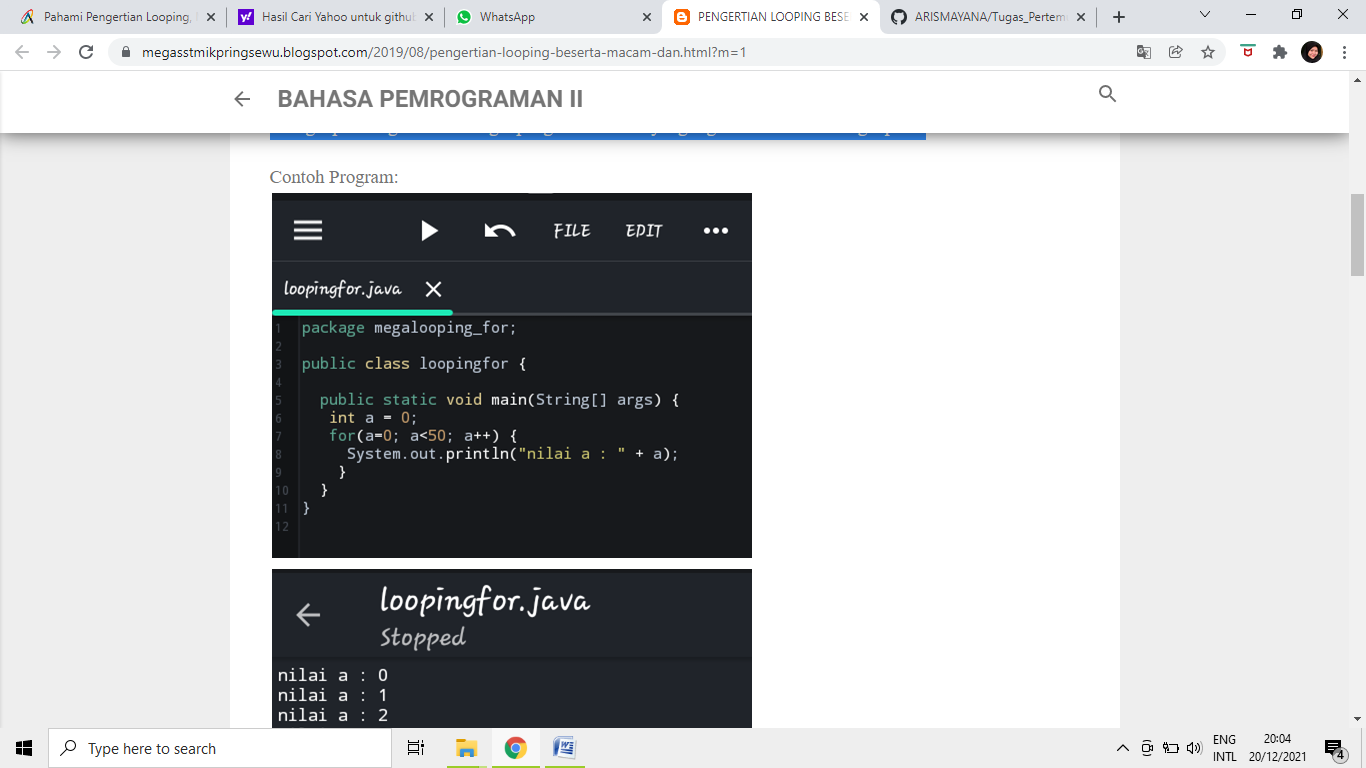
Keterangan :

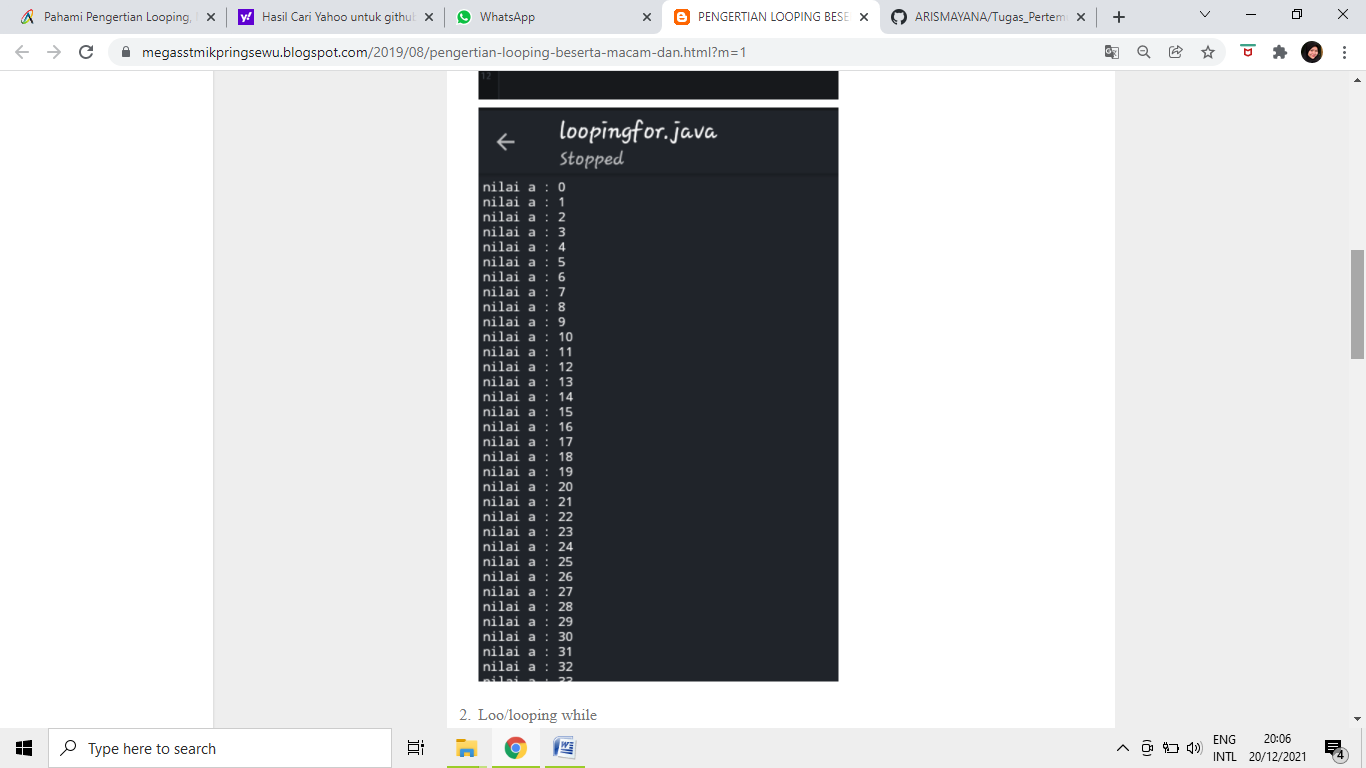
o   ungkapan1 merupakan pernyataan inisialisasi.

o   ungkapan2 sebagai kondisi  yang menentukan pengulangan terhadap pernyataan atau tidak.

o   ungkapan3 digunakan sebagai pengatur variabel yang digunakan didalam ungkapan1.

Contoh Program :

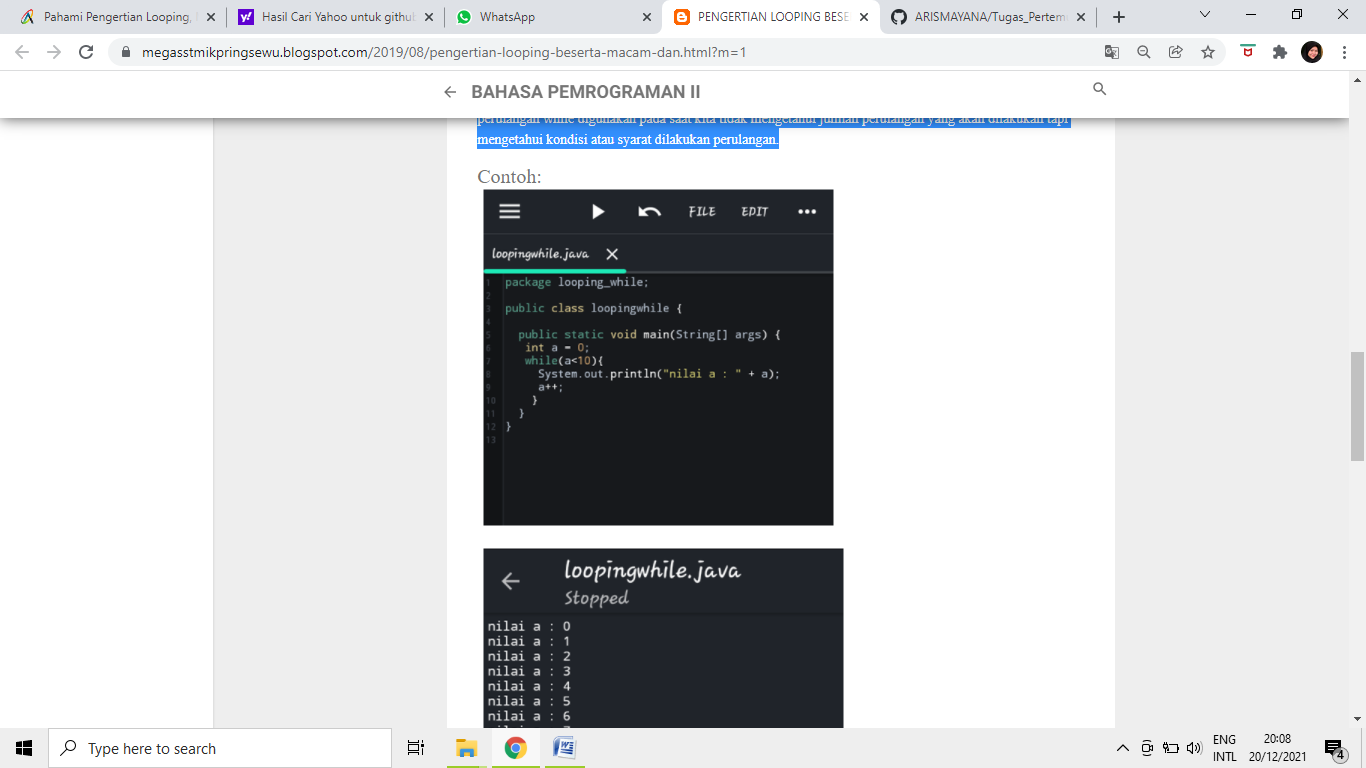


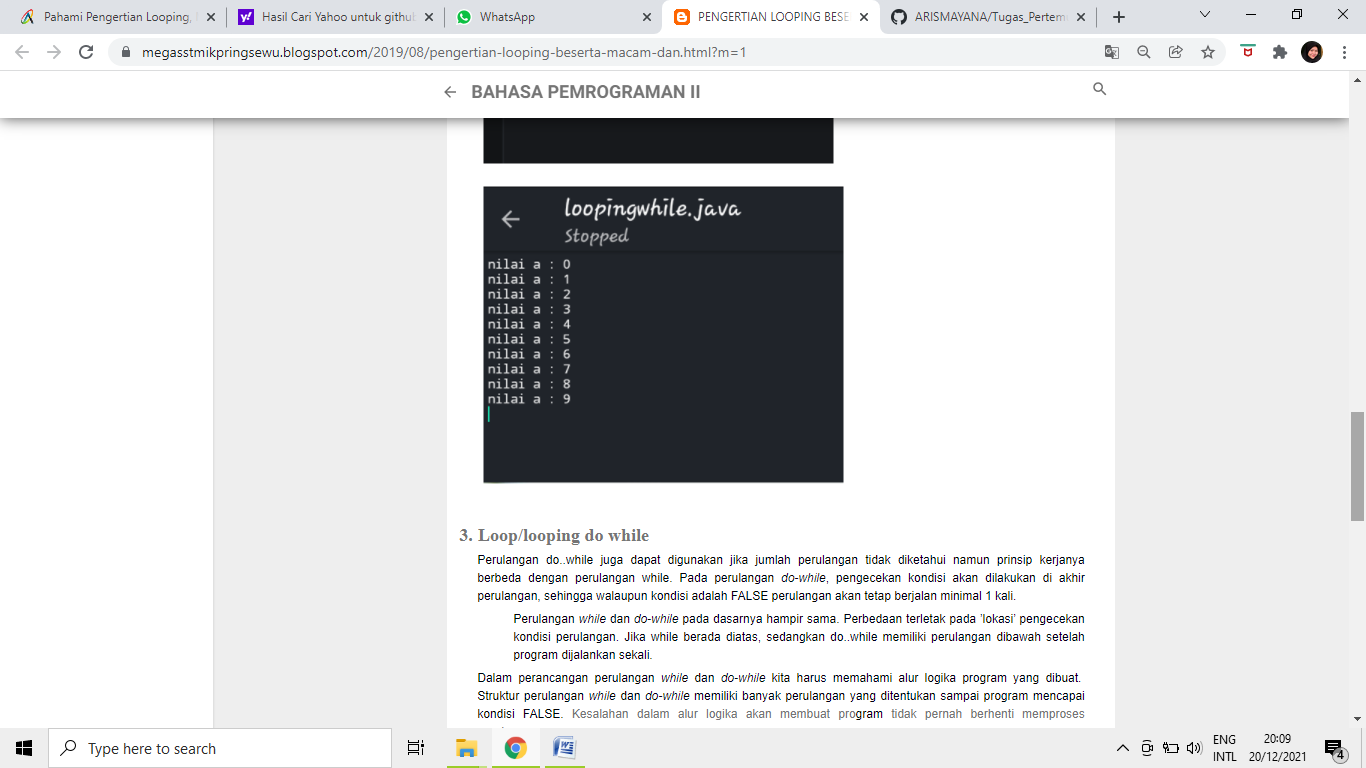


2.  Loop/looping while

Perulangan while digunakan pada saat kita tidak mengetahui jumlah perulangan yang akan dilakukan tapi mengetahui kondisi atau syarat dilakukan perulangan.

Contoh:





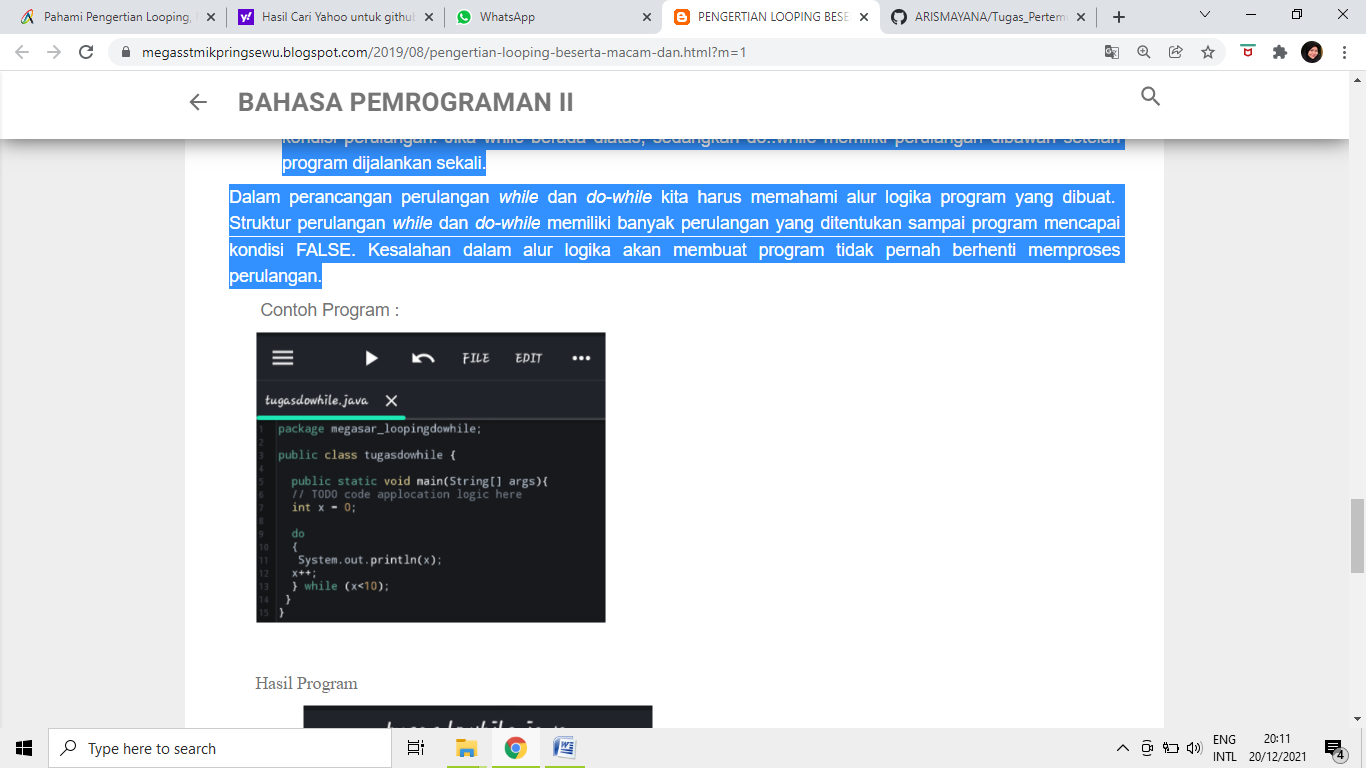
3.   Loop/looping do while

Perulangan do..while juga dapat digunakan jika jumlah perulangan tidak diketahui namun prinsip kerjanya berbeda dengan perulangan while. Pada perulangan *do-while*, pengecekan kondisi akan dilakukan di akhir perulangan, sehingga walaupun kondisi adalah FALSE perulangan akan tetap berjalan minimal 1 kali.

Perulangan *while* dan *do-while* pada dasarnya hampir sama. Perbedaan terletak pada ’lokasi’ pengecekan kondisi perulangan. Jika while berada diatas, sedangkan do..while memiliki perulangan dibawah setelah program dijalankan sekali.

Dalam perancangan perulangan *while* dan *do-while* kita harus memahami alur logika program yang dibuat.  Struktur perulangan *while* dan *do-while* memiliki banyak perulangan yang ditentukan sampai program mencapai kondisi FALSE. Kesalahan dalam alur logika akan membuat program tidak pernah berhenti memproses perulangan.

Contoh:



Hasil program :

