Static Varable atau juga disebut sebagai Static Duration Variable adalah variabel yang menggunakan keyword static, membuat variabel tersebut hanya dapat di inisialisasi satu kali, memiliki nilai yang tetap dan tidak terpengaruh oleh lingkungan, meskipun function tersebut telah berakhir, variabel tersebut tetap akan memiliki nilai yang sama seperti sebelumnya.

Keyword static dapat digunakan untuk global variabel, variabel local, dan Member Class. fungsi static tersebut tergantung dimana keyword static digunakan. Jika static digunakan pada global variabel maka variabel terebut akan bersifat local atau tidak bisa di akses dari luar file. Jika digunakan pada local variabel maka variabel tersebut akan bersifat hanya bisa di inisialisasi satu kali, variabel tidak dilepas ketikan function tersebut berakhir. Dan jika digunakan pada class, hal itu akan kami jelaskan di artikel berikutnya.

Berbeda dengan variabel biasa, Kita mengetahui bahwa deklarasi seperti variabel (Automatic duration) yang berada di dalam function (bukan main function) akan dibuat di saat dunction tersebut dijalankan, dan semua deklarasi (memori) akan dilepaskan saat berakhirnya function. Itu berarti variabel biasa yang didirikan di dalam function hanya akan ada di saat function itu dijalankan. berbeda dengan static, jika kita gunakan keyword static pada deklarasi variabel, saat kita melakukan deklarasi bersamaan inisialisasi di dalam function, static variabel hanya akan di inisialisasi satu kali meskipun jalur eksekusi compiler bertemu kembali dengan deklarasi static variabel tersebut dan akan tetap ada bersama nilai yang variabel punya sebelumnya.

Static Variabel sangat berguna bagi programmer c/c++, yang biasanya digunakan sebagai penghitung berapa kali function itu dijalankan, atau berapa kali pengulangan tersebut dilakukan, static variabel melakukan hal yang tidak bisa dilakukan oleh variable biasa.

Bentuk Umum penulisan

static tipe\_data identifier ;

Contoh penulisan

static int id;

Contoh program

#include <iostream>

using namespace std;

void myFunction();

int main(){

myFunction();

myFunction();

myFunction();

myFunction();

myFunction();

return 0;

}

void myFunction(){

int hitung = 0;

static int s\_hitung = 0; //Static Variable

cout<<"int hitung = "<<++hitung<<endl;

cout<<"static int s\_hitung = "<<++s\_hitung<<endl<<endl;

}

Di atas adalah contoh program penggunaa keyword static beserta bukti mengenai sifat dari static. Di atas didirikan function “myFunction” yang di dalamnya didirikan dua variabel (hitung dan s\_hitung) dan salah satunya adalah Static Variable (s\_hitung). dan pada baris berikutnya masih di dalam definisi “myFunction” terdapat perintah penambahaan di dalam perintah keluaran.

Pada main function kita tuliskan perintah pemanggilan function “myFunction” sebanyak 5 kali. Dan hasil dari program di atas adalah variabel biasa yang kita dirikan tanpa static di dalam function “myFunction”, akan memiliki nilai yang sama di setiap pemanggilan, berbeda dengan static variabel mempunyai nilai yang selalu meningkat, lebih besar dari pemanggilan sebelumnya.

Hal itu terjadi karena static variabel bersifat tetap, ketika jalur eksekusi bertemu akhir dari function maka semua deklarasi kecuali static akan dilepaskan, dan ketika jalur eksekusi memasuki function itu kembali maka semua deklarasi akan didirikan kembali bersama nilai awal pada inisialisasi (jika diinisialisasi), kecuali static variable yang memang masih ada dari pertama kali pemanggian function tersebut, bersama nilai sebelumnya. static variable hanya akan dilepaskan ketika jalur eksekusi memang sudah selesai dalam pembacaan program (akhir main function).

Local Variabel di lepas otomatis bukan hanya saat jalur kompilasi selesai dalam pembacaan function tapi juga selesai dalam pembacaan suatu scope “{ . . . }”. Jika global variabel sama seperti static duration variable yang akan dilepas saat selesainya program.

Static juga akan otomatis melakukan inisialisasi pada variable yang menggunakan static keyword menjadi memiiki nilai awal “0”, jika memang static variable yang kita dirikan tidak kita berikan nilai awal.

Dan juga static variable akan bersifat sama seperti halnya pada contoh program di atas jika kita lakukan deklarasi di dalam pengulangan. Di bawah ini adalah contoh static variable di dalam pengulangan.

Contoh program

#include <iostream>

using namespace std;

int main(){

for (int i=0;i<10;i++){

int hitung=0;

static int s\_hitung;

cout<<"int hitung = "<<++hitung<<endl;

cout<<"static int s\_hitung = "<<++s\_hitung<<endl<<endl;

}

return 0;

}