|  |
| --- |
| /\* |
|  | filename: storageclassess.cpp |
|  | details: Program to demonstrate different storage classes |
|  | author: Decca Bhagyaraju |
|  | date: 11/04/2020 |
|  | \*/ |
|  | #include<iostream> |
|  | #include<string.h> |
|  | using namespace std; |
|  | //global variable |
|  | int ia=10; |
|  | int ib; |
|  | int ic='a'; |
|  | void autoStorageClass(auto int a); |
|  | void registerStorageClass(register int b); |
|  | void externStorageClass(int iValue); |
|  | void staticStorageClass(int iValue); |
|  | //variable scope function |
|  | void scope() |
|  | { |
|  | //local variable |
|  | int ia=20; |
|  | cout<<"--------------------VARIABLE SCOPE -------------------------"<<endl; |
|  | cout<<" global variable a:"<<::ia<<endl; |
|  | cout<<" local variable b:"<<ia<<endl; |
|  | cout<<"-----------------------------------------------------------"<<endl; |
|  | } |
|  | void modifier() |
|  | { |
|  | int iNumb; |
|  | short int a; |
|  | long int b; |
|  | unsigned short c; |
|  | signed short d; |
|  | unsigned int e; |
|  | cout << "enter value: "; |
|  | cin>>iNumb; |
|  | a=c=d=b=e=iNumb; |
|  | cout << " ---------------- Modifiers--------------------------- "<<endl; |
|  | // values based on modifiers |
|  | cout << " short int = " << a << endl; |
|  | cout << " long int = " << b << endl; |
|  | cout << " unsigned short = " <<c << endl; |
|  | cout << " signed short = " <<d << endl; |
|  | cout << " unsignes int = " <<e << endl; |
|  |  |
|  | cout << "------------------------------------------------------- "<< endl; |
|  | } |
|  | class mutable\_test |
|  | { |
|  | public :mutable\_test(int m, int n) |
|  | { |
|  | x = m; |
|  | y = n; |
|  | } |
|  | mutable int x; |
|  | int y; |
|  | }; |
|  | int main(int argc,char \*argv[]) |
|  | { |
|  | auto int a; |
|  | register int b; |
|  | static int p; |
|  | int q; |
|  | int iValue; |
|  |  |
|  | if(argc>=2) |
|  | { |
|  | if(strcmp(argv[1],"-h")==0) |
|  | { |
|  | cout<<" USAGE "<<endl; |
|  | cout<<"This program explains the usage of variable scope,modifier types & storage classes"<<endl; |
|  | } |
|  | } |
|  | else |
|  | { |
|  | scope(); |
|  | modifier(); |
|  | // x = 5 and y = 3 |
|  | const mutable\_test temp(5, 3); |
|  | cout<<"----------------Mutable storage -----------------------"<<endl; |
|  | cout << " x : " << temp.x << " y : " << temp.y << endl; |
|  | // because 'x' is mutable |
|  | temp.x = 6; |
|  | cout << " x : " <<temp.x << " y : " <<temp.y << endl; |
|  | cout <<"------------------------------------------------------"<<endl; |
|  | cout<<"Enter a value: "; |
|  | cin>>iValue; |
|  | a=b=p=q=iValue; |
|  | cout<<"\n------------------Storage Classes --------------------\n"<<endl; |
|  |  |
|  | // To demonstrate auto Storage Class |
|  | autoStorageClass(a); |
|  |  |
|  | // To demonstrate register Storage Class |
|  | registerStorageClass(b); |
|  |  |
|  | // To demonstrate extern Storage Class |
|  | externStorageClass(iValue); |
|  |  |
|  | // To demonstrate static Storage Class |
|  | staticStorageClass(iValue); |
|  |  |
|  | return 0; |
|  | } |
|  | } |
|  | void autoStorageClass(auto int a) |
|  | { |
|  | //auto int a = 'a'; |
|  | cout<<" Value of the auto variable : "<<a<<endl; |
|  | cout<<"--------------------------------"<<endl; |
|  | } |
|  |  |
|  | void registerStorageClass(register int b) |
|  | { |
|  | //register int b = 'a'; |
|  | cout<<" Value of the register variable : "<<b<<endl; |
|  | cout<<"--------------------------------"<<endl; |
|  | } |
|  |  |
|  | void externStorageClass(int iValue) |
|  | { |
|  |  |
|  | // telling the compiler that the variable is an extern variable |
|  | extern int ic; |
|  |  |
|  | // print extern variables |
|  | cout<<" Value of the extern variable: "<<ic<<endl; |
|  |  |
|  | // modified extern |
|  | ic = iValue; |
|  |  |
|  | // print modified extern |
|  | cout<<" Value of the modified extern variable : "<<ic<<endl; |
|  | cout<<"--------------------------------"<<endl; |
|  | } |
|  |  |
|  |  |
|  | void staticStorageClass(int iValue) |
|  | { |
|  | // printing a static variable & non-static variables with in a loop |
|  | cout<<" Declaring static & non-static variable inside a loop \n"<<endl; |
|  |  |
|  | for (int i = 1; i <= 3; i++) { |
|  |  |
|  | // Declaring the static variable |
|  | static int p = iValue; |
|  |  |
|  | // Declare a non-static variable |
|  | int q = iValue; |
|  |  |
|  | // Incrementing the values |
|  | p++; |
|  | q++; |
|  |  |
|  | cout<<" Static variable = "<<p<<endl; |
|  | cout<<" Non-Static variable = "<<q<<endl; |
|  | } |
|  | cout<<"-----------------------------------------------------------------"<<endl; |
|  | } |