Content 22

Static Data Members & Methods in C++ OOPS

#### Static Data Members in C++

When a static data member is created, there is only a single copy of the data member which is shared between all the objects of the class. As we have discussed in our previous lecture that if the data members are not static then every object has an individual copy of the data member and it is not shared.

#### Static Methods in C++

When a static method is created, they become independent of any object and class. Static methods can only access static data members and static methods. Static methods can only be accessed using the scope resolution operator. An example program is shown below to demonstrate static data members and static methods in C++.

**Code:**

#include <iostream>

using namespace std;

class employe

{

    int empId;

    static int count;

public:

    void getdata(void)

    {

        cout << "Enter your Employe Id: " << endl;

        cin >> empId;

        count++;

    }

    void show(void)

    {

        cout << "Your emoploye Id is: " << empId << " and the value of count is: " << count << endl;

    }

    void show\_count(void);

};

void employe :: show\_count(void)

{

    cout << "The value of count is: " << count << endl;

}

//cout is the static data membe of class employ

int employe::count; //it is defaultly initialized by 0

int main()

{

    employe e1, e2, e3;

    e1.getdata();

    e1.show();

    e2.getdata();

    e2.show();

    e3.getdata();

    e3.show();

    return 0;

}

**Output:**

Enter your Employe Id:

98

Your emoploye Id is: 98 and the value of count is: 1

Enter your Employe Id:

78

Your emoploye Id is: 78 and the value of count is: 2

Enter your Employe Id:

67

Your emoploye Id is: 67 and the value of count is: 3