C++ static

In C++, static is a keyword or modifier that belongs to the type not instance. So instance is not required to access the static members. In C++, static can be field, method, constructor, class, properties, operator and event.

Advantage of C++ static keyword

Memory efficient: Now we don't need to create instance for accessing the static members, so it saves memory. Moreover, it belongs to the type, so it will not get memory each time when instance is created.

C++ Static Field

A field which is declared as static is called static field. Unlike instance field which gets memory each time whenever you create object, there is only one copy of static field created in the memory. It is shared to all the objects.

It is used to refer the common property of all objects such as rateOfInterest in case of Account, companyName in case of Employee etc.

C++ static field example

Let's see the simple example of static field in C++.

```
#include <iostream>
using namespace std;
class Account {
   public:
       int accno; //data member (also instance variable)
       string name; //data member(also instance variable)
       static float rateOfInterest;
       Account(int accno, string name)
        {
             this->accno = accno;
            this->name = name;
        }
       void display()
        {
            cout<<accno<< "<<name<< " "<<rateOfInterest<<endl;
```

```
      }
};
float Account::rateOfInterest=6.5;
int main(void) {
    Account a1 =Account(201, "Sanjay"); //creating an object of Employee
    Account a2=Account(202, "Nakul"); //creating an object of Employee
    a1.display();
    a2.display();
    return 0;
}
```

Output:

```
201 Sanjay 6.5
202 Nakul 6.5
```

C++ static field example: Counting Objects

Let's see another example of static keyword in C++ which counts the objects.

```
#include <iostream>
using namespace std;
class Account {
   public:
       int accno; //data member (also instance variable)
       string name;
       static int count;
       Account(int accno, string name)
        {
             this->accno = accno;
            this->name = name;
            count++;
        }
       void display()
        {
            cout<<accno<<" "<<name<<endl;
        }
};
int Account::count=0;
int main(void) {
    Account a1 =Account(201, "Sanjay"); //creating an object of Account
    Account a2=Account(202, "Nakul");
```

```
     Account a3=Account(203, "Ranjana");
    a1.display();
    a2.display();
    a3.display();
    cout<<"Total Objects are: "<<Account::count;
    return 0;
 }
```

Output:

```
201 Sanjay
202 Nakul
203 Ranjana
Total Objects are: 3
```

← prev next →

Please Share

Join Javatpoint Test Series

Placement Papers
TCS
HCL
Infosys
IBM
Accenture

AMCAT
eLitmas
Java
Python
C Programming
Networking

Bank PO/Clerk
UPSSSC
Government Exams
SSC
Civil Services
SBI

GATE
NEET
CAT
Railway
CTET
IIT JEE

Learn Latest Tutorials