

Aim:

To understand the concept of **Static Members** with the help of an example.

Theory:

A static member is shared by all objects of the class. When we declare a member of a class as static it means no matter how many objects of the class are created, there is only one copy of the static member. All static data is initialized to zero when the first object is created, if no other initialization is present. We can't put it in the class definition but it can be initialized outside the class as done in the following example by redeclaring the static variable, using the scope resolution operator `::` to identify which class it belongs to.

By declaring a function member as static, you make it independent of any particular object of the class. A static member function can be called even if no objects of the class exist and the static functions are accessed using only the class name and the scope resolution operator `::`.

A static member function can only access static data member, other static member functions and any other functions from outside the class.

Syntax :

```
class A {
    static int a ;
    public :
        static void test(){ }
};

// Using Static Members

A::a ;
A::test() ;
```

Code :

```
#include <iostream>

using namespace std;

class Test {
    int code ;
    static int count ;

    public :
        void setcode(){
            code = ++count ;
        }
        void showcode(){
            cout << "Obj No : " << code << endl;
        }
}
```

```

        static void showcount(){
            cout << "Object Count : " << count << endl ;
        }
};

int Test::count ;

int main (){
    Test T1, T2 ;
    T1.setcode() ;
    T2.setcode() ;

    Test::showcount() ;

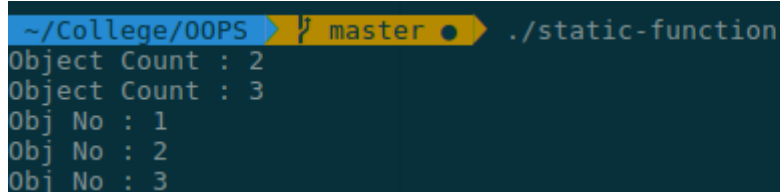
    Test T3 ;
    T3.setcode() ;

    Test::showcount() ;
    T1.showcode();
    T2.showcode();
    T3.showcode();

    return 0;
}

```

Output :



```

~/College/OOPS } master ● ./static-function
Object Count : 2
Object Count : 3
Obj No : 1
Obj No : 2
Obj No : 3

```

Discussion :

The above program illustrates the concept of static members. The static member `count` of class `Test` is declared as static and is used to store the number of objects created. A static member function `showcount()` is used to display that count and is called using the class name.

Learning Outcomes :

- It eliminates the need for an object as the members can be accessed using class.
- Static Data Members can be Encapsulated. A static member can be a private member, but a global variable cannot.
- It acts as a way to interact between different objects of the same class.