

Static Members of Class –

- **Class objects as static** - Just like variables, objects also when declared as static have a scope till the lifetime of program.

```
#include<iostream.h>
#include<conio.h>
```

```
class GfG
{
    int i = 0;
public:
    GfG()
    {
        i = 0;
        cout << "Inside Constructor\n";
    }

    ~GfG()
    {
        cout << "Inside Destructor\n";
    }
};
```

```
void main()
{
    int x = 0;
    if (x==0)
    {
        static GfG obj;
    }
    cout << "End of main\n";
}
```

Static functions in a class -

Just like the static data members or static variables inside the class, static member functions also does not depend on object of class. We are allowed to invoke a static member function using the object and the '.' operator but it is recommended to invoke the static members using the class name and the scope resolution operator. Static member functions are allowed to access only the static data members or other static member functions, they can not access the non-static data members or member functions of the class.

```
#include<iostream.h>
#include<conio.h>

class GGI
{
public:

    static void printMsg()
    {
        cout<<"Welcome to GGI...";
    }
};

void main()
{
    GfG::printMsg();
}
```