Hurry! Try our new Interactive Courses for FREE. 🥳 🚀

Types of Class Member Functions in C++

ADVERTISEMENT

We already know what member functions are, what they do, how to define member functions and how to call them using class objects. Now lets learn about some special member functions which can be defined in C++ classes. Following are the different types of Member functions:

☰ Index

1. Simple functions
2. Static functions
3. Const functions
4. Inline functions
5. Friend functions

Simple Member functions in C++

These are the basic member function, which dont have any special keyword like static etc as prefix. All the general member functions, which are of below given form, are termed as simple and basic member functions.

```
return_type functionName(parameter_list)
{
    function body;
}
```

Static Member functions in C++

Static is something that holds its position. Static is a keyword which can be used with data members as well as the member functions. We will discuss this in details later. As of now we will discuss its usage with member functions only.

☰ Index

A function is made static by using `static` keyword with function name. These functions work for the class as whole rather than for a particular object of a class.

It can be called using the object and the direct member access `.` operator. But, its more typical to call a static member function by itself, using class name and scope resolution `::` operator.

ADVERTISEMENT

For example:

```
class X
{
    public:
    static void f()
    {
        // statement
    }
};

int main()
{
    X::f();   // calling member function directly
}
```

These functions cannot access ordinary data members and member functions, but only `static` data members and `static` member functions can be called inside them.

It doesn't have any "this" keyword which is the reason it cannot access ordinary members. We will study about "this" keyword later.

☰ Index

Const Member functions in C++

We will study **Const** keyword in detail later(Const Keyword), but as an introduction, Const keyword makes variables constant, that means once defined, there values can't be changed.

When used with member function, such member functions can never modify the object or its related data members.

```
// basic syntax of const Member Function

void fun() const
{
    // statement
}
```

Inline functions in C++

All the member functions defined inside the class definition are by default declared as Inline. We will study Inline Functions in details in the next topic.

Friend functions in C++

☰ Index

Friend functions are actually not class member function. Friend functions are made to give **private** access to non-class functions. You can declare a global function as friend, or a member function of other class as friend.

For example:

```
class WithFriend
{
    int i;
```

```
    public:
    friend void fun(); // global function as fri
};

void fun()
{
    WithFriend wf;
    wf.i=10;  // access to private data member
    cout << wf.i;
}

int main()
{
    fun(); //Can be called directly
}
```

Hence, friend functions can access private data members by creating object of the class. Similarly we can also make function of some other class as friend, or we can also make an entire class as **friend class**.

☰ Index

```
class Other
{
    void fun();
};

class WithFriend
{
    private:
    int i;
```

```
    public:
    void getdata();  // Member function of class

    // making function of class Other as friend
    friend void Other::fun();

    // making the complete class as friend
    friend class Other;
};
```

When we make a class as friend, all its member functions automatically become friend functions.

Friend Functions is a reason, why C++ is not called as a **pure Object Oriented language**. Because it violates the concept of **Encapsulation**.

← Prev

Next →

☰ Index

ADVERTISEMENT