Content 24

Friend Function in C++

#### Friend Function in C++

Friend functions are those functions that have the right to access the private data members of class even though they are not defined inside the class. It is necessary to write the prototype of the friend function. One main thing to note here is that if we have written the prototype for the friend function in the class it will not make that function a member of the class. An example program to demonstrate the concept of friend function is shown below.

#### Properties of Friend Function

* Not in the scope of the class
* Since it is not in the scope of the class, it cannot be called from the object of that class, for example, **sumComplex()**is invalid
* A friend function can be invoked without the help of any object
* Usually contain objects as arguments
* Can be declared under the public or private access modifier, it will not make any difference
* It cannot access the members directly by their names, it needs (object\_name.member\_name) to access any member.

#include <iostream>

using namespace std;

class complex

{

    int a;

    int b;

public:

    friend complex sum\_complex(complex o1, complex o2); //using the private members in functions

    void setnum(int n1, int n2)

    {

        a = n1;

        b = n2;

    }

    void getnum(void)

    {

        cout << "The Complex number is: " << a << " + i" << b << endl;

    }

};

complex sum\_complex(complex o1, complex o2)

{

    complex o3;

    o3.setnum((o1.a + o2.a), (o1.b + o2.b));

    return o3;

}

int main()

{

    complex c1, c2, sum;

    c1.setnum(2, 3);

    c1.getnum();

    c2.setnum(3, 2);

    c2.getnum();

    sum = sum\_complex(c1, c2);

    sum.getnum();

    return 0;

}

**Output:**

The Complex number is: 2 + i3

The Complex number is: 3 + i2

The Complex number is: 5 + i5