

# Supplement: Compile and Link Object Files

For Introduction to C++ Programming  
By Y. Daniel Liang

If you are a software vendor to sell software, you wish to hide the source code from the client. You can separate declarations and implementation into two files. The declaration file simply lists all the function prototypes. The implementation file implements the functions. Both files should have the same name, but with different extension names. The declaration file has an extension name `.h` and the implementation file has an extension name `.cpp`, as shown in Listings 1 and 2.

Listing 1 MyLib.h

**\*\*\*PD: Please add line numbers in the following code\*\*\***  
**<Side Remark line 1: function prototype>**  
`bool isEven(int number);`

Listing 2 MyLib.cpp

**\*\*\*PD: Please add line numbers in the following code\*\*\***  
**<Side Remark line 1: implement function>**  
`bool isEven(int number)`  
`{`  
 `return (number % 2 == 0);`  
`}`

For the client to use `MyLib`, you only need to provide the client with the header file and object code generated from `MyLib.cpp`. You don't need to provide the client with `MyLib.cpp`. This protects software vendor's intellectual properties.

Here is a test program that uses `MyLib.h`.

Listing 3 UseMyLib.cpp

**\*\*\*PD: Please add line numbers in the following code\*\*\***  
**<Side Remark line 2: include MyLib.h>**  
**<Side Remark line 8: invoke isEven>**  
**<Side Remark line 9: invoke isEven>**

```
#include <iostream>
#include "MyLib.h"

using namespace std;

int main()
{
    cout << isEven(4) << endl;
    cout << isEven(5) << endl;

    return 0;
}
```

**<Output>**

1

0

**<End Output>**

To create an object file, use

```
g++ MyLib.cpp -c  
g++ UseMyLib.cpp -c
```

You will see MyLib.o and UseMyLib.o created. Now you can link object files to create an executable file, as follows:

```
g++ MyLib.o UseMyLib.o -o UseMyLib.exe
```

You can now run

```
UseMyLib.exe
```