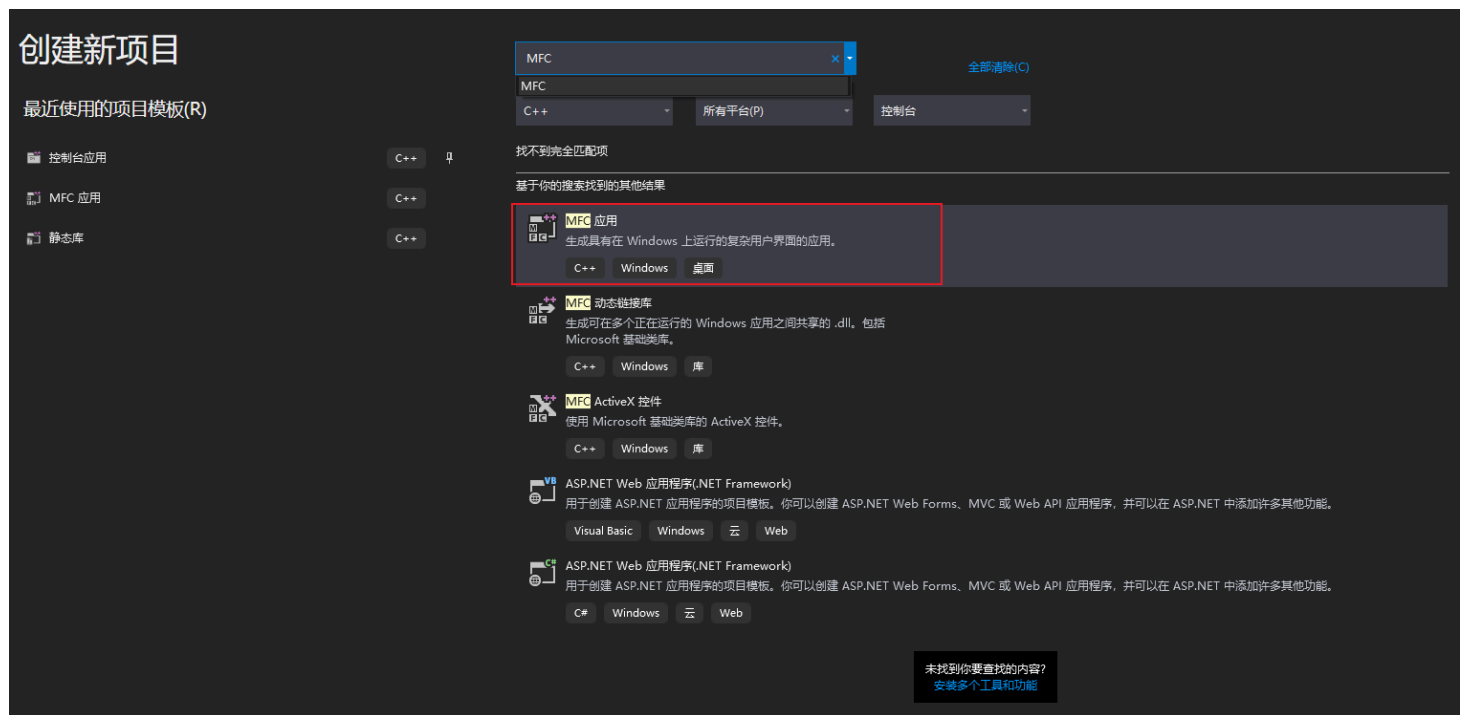


CEF 在 MFC 中的使用

工程配置

1、首先创建一个MFC对话框工程



配置新项目

MFC 应用

C++

Windows

桌面

项目名称(I)

CEFExample

位置(L)

E:\git_open_my\Todo\cefExample\

解决方案名称(M) ⓘ

CEFExample

☐ 将解决方案和项目放在同一目录中(D)

MFC 应用程序

应用程序类型选项

应用程序类型

文档模板属性

用户界面功能

高级功能

生成的类

应用程序类型(T)

基于对话框

单个文档

多个文档

基于对话框

多个顶层文档

基于对话框的选项(I)

<无>

复合文档支持

<无>

文档支持选项:

☐ 活动文档服务器(A)

☐ 活动文档容器(D)

☐ 支持复合文件(U)

项目样式

MFC standard

视觉样式和颜色(Y)

Windows Native/Default

☐ 启用视觉样式切换(C)

资源语言(L)

zh-CN

使用 MFC

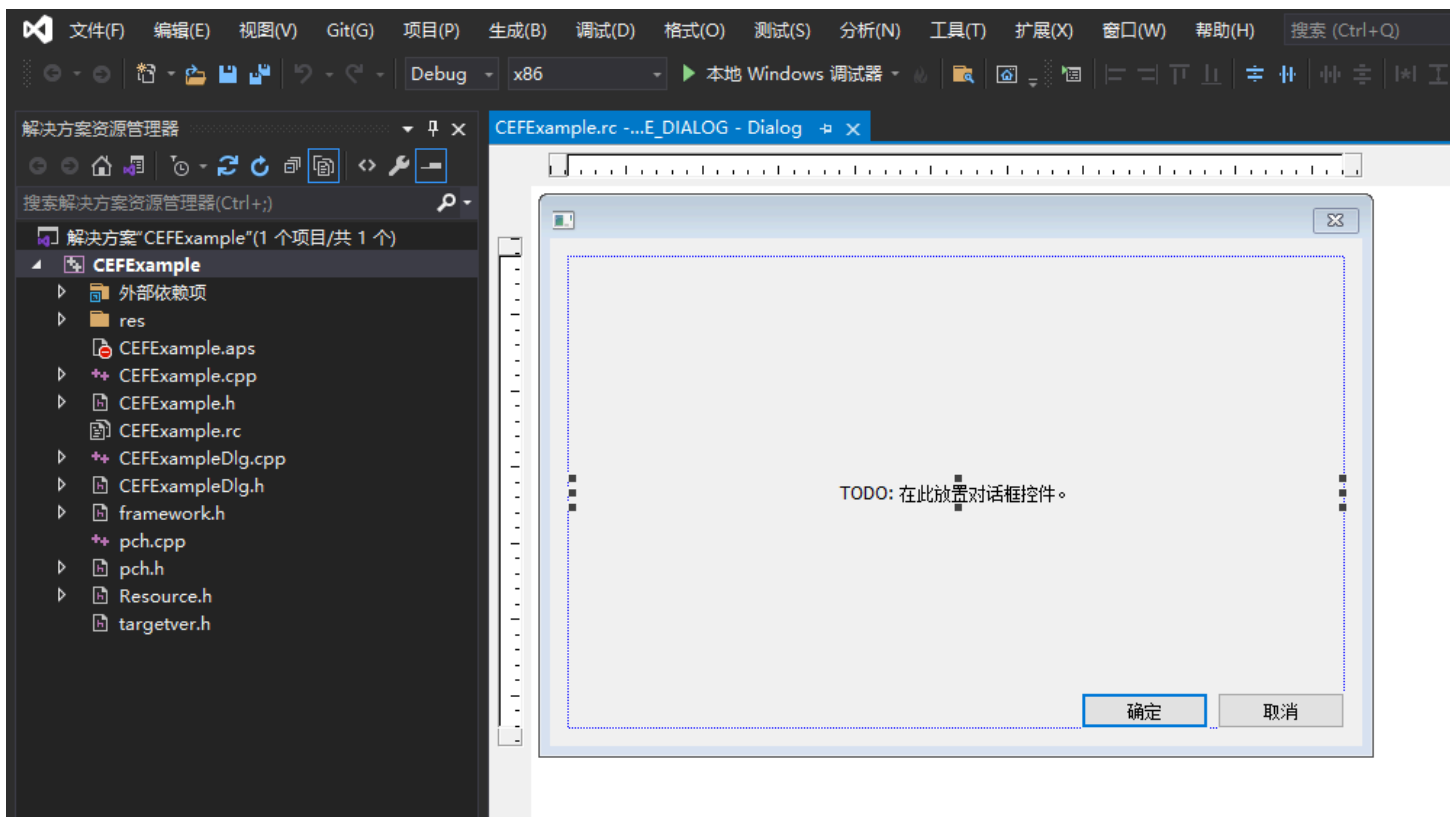
在共享 DLL 中使用 MFC

上一步

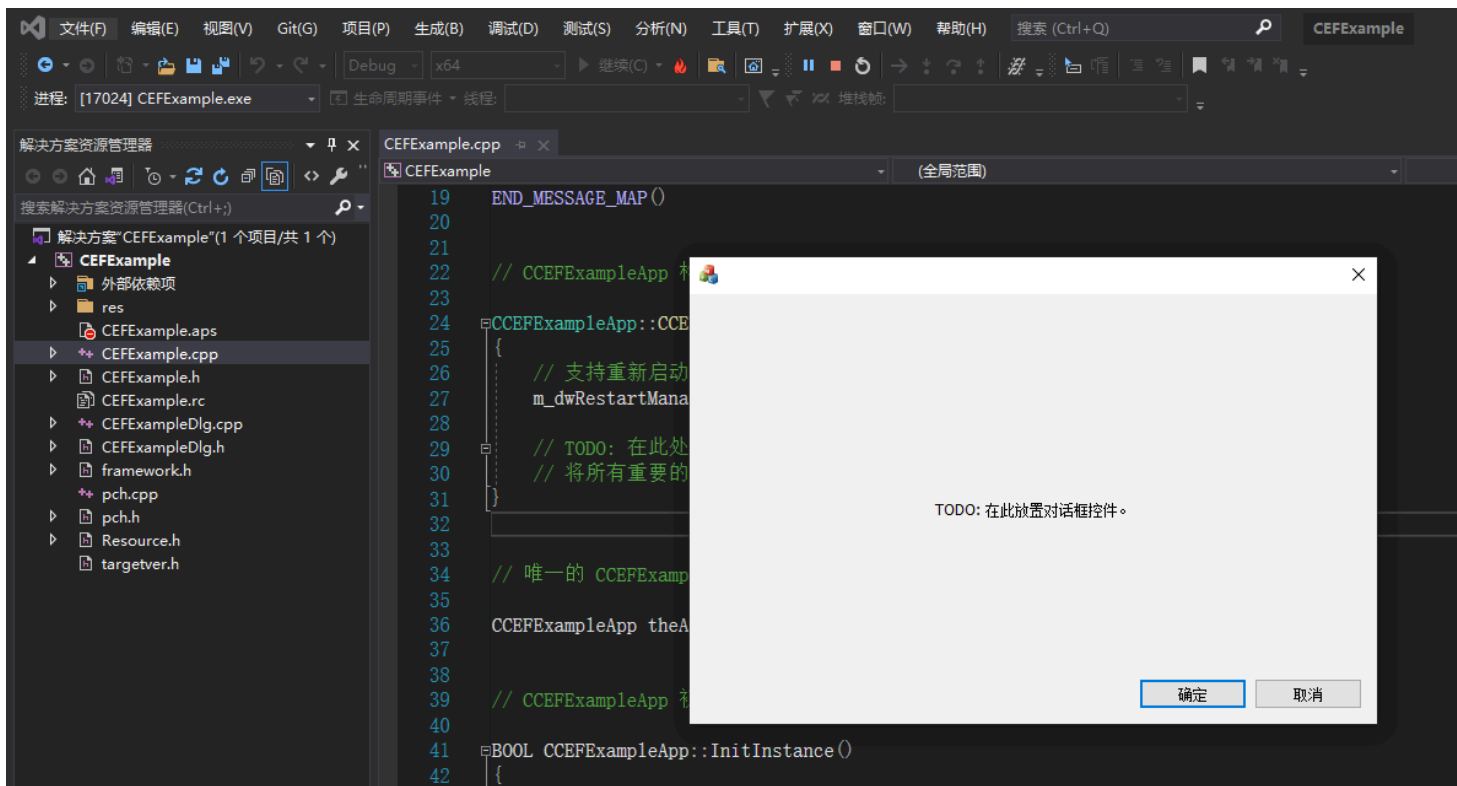
下一步

完成

取消



创建完运行测试效果如下

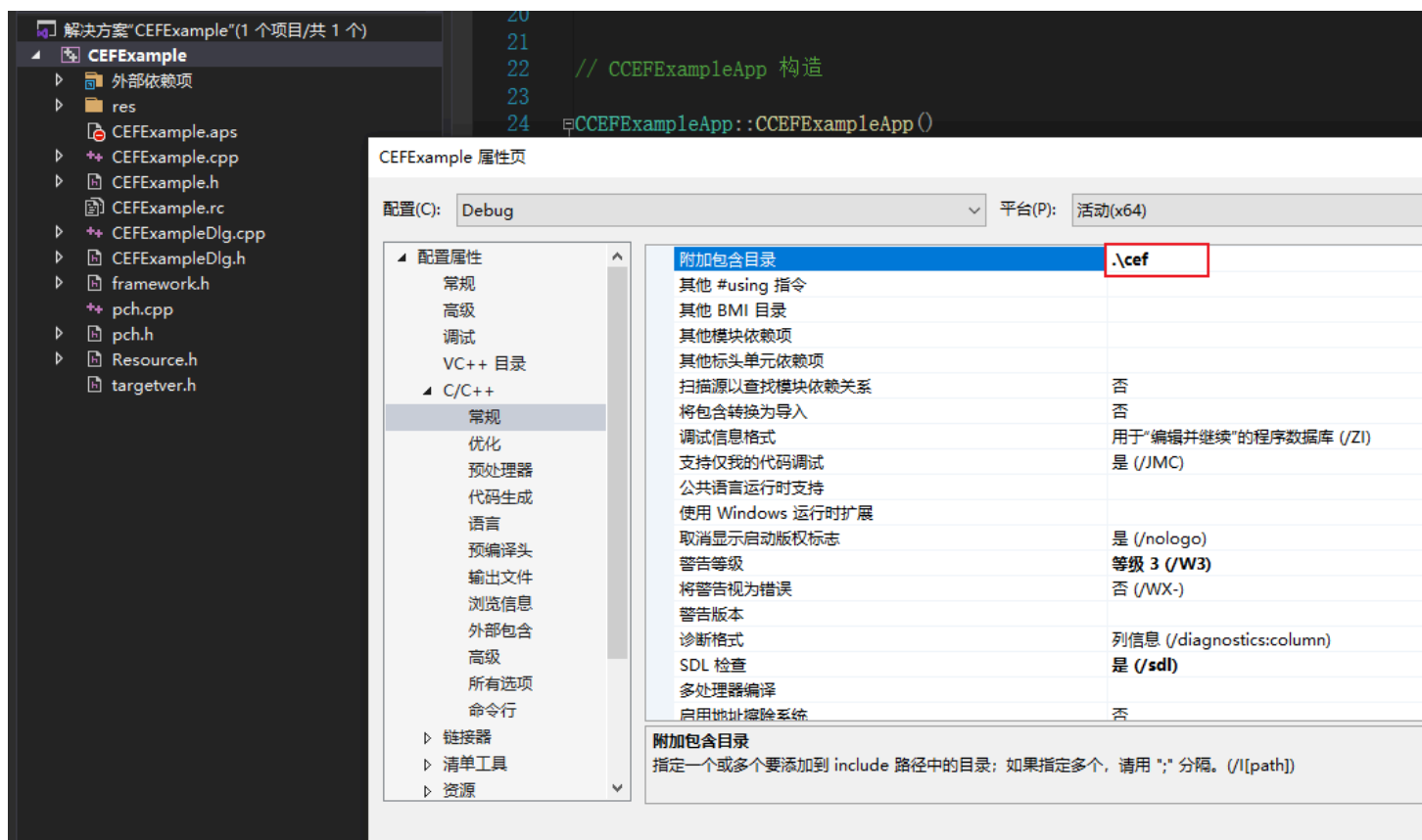


2、MFC工程引入CEF库

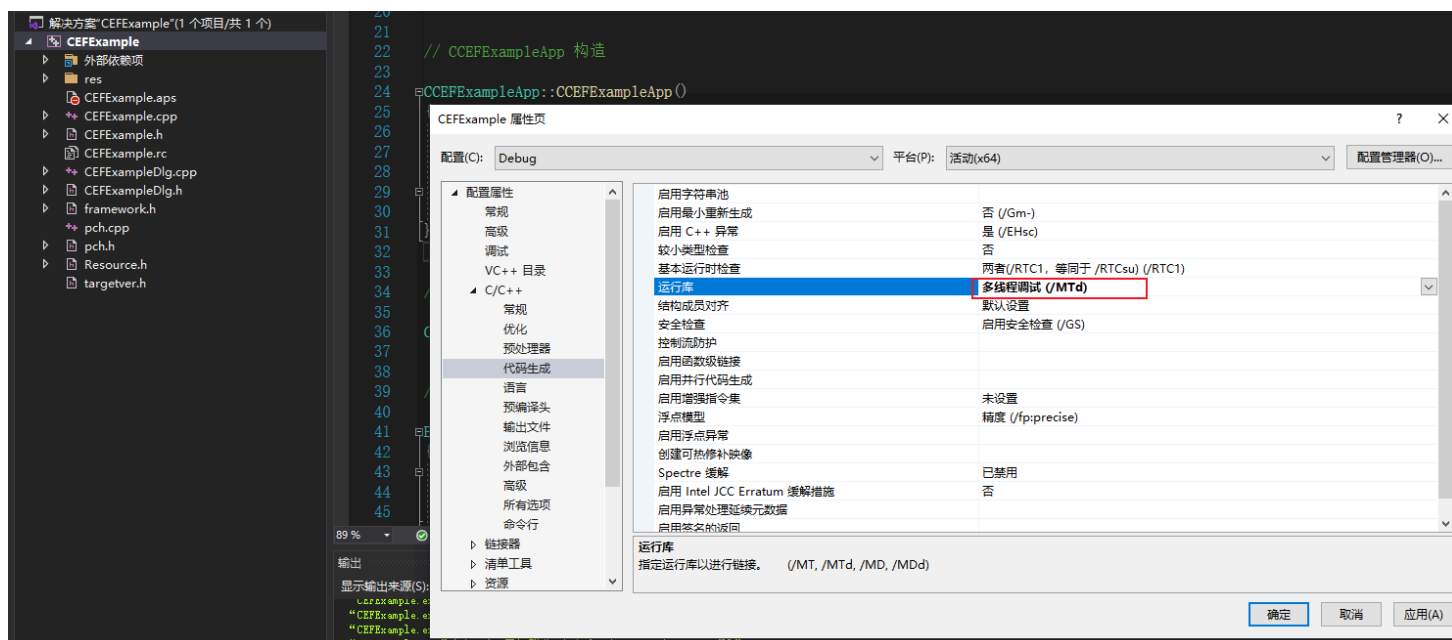
将 CEF 目录下的 cef子目录下载解压后放到MFC工程中：

名称	修改日期	类型	大小
cef	2025/3/8 20:08	文件夹	
res	2025/3/8 20:04	文件夹	
x64	2025/3/8 20:05	文件夹	
CEFExample.apx	2025/3/8 20:04	APX 文件	90 KB
CEFExample.cpp	2025/3/8 20:04	C++ Source file	3 KB
CEFExample.h	2025/3/8 20:04	C/C++ Header	1 KB
CEFExample.rc	2025/3/8 20:04	Resource Source...	10 KB
CEFExample.vcxproj	2025/3/8 20:04	VC++ Project	11 KB
CEFExample.vcxproj.filters	2025/3/8 20:04	VC++ Project Fil...	3 KB
CEFExample.vcxproj.user	2025/3/8 20:05	Per-User Project...	1 KB
CEFExampleDlg.cpp	2025/3/8 20:04	C++ Source file	4 KB
CEFExampleDlg.h	2025/3/8 20:04	C/C++ Header	1 KB
framework.h	2025/3/8 20:04	C/C++ Header	2 KB
pch.cpp	2025/3/8 20:04	C++ Source file	1 KB
pch.h	2025/3/8 20:04	C/C++ Header	1 KB
Resource.h	2025/3/8 20:04	C/C++ Header	1 KB
targetver.h	2025/3/8 20:04	C/C++ Header	1 KB

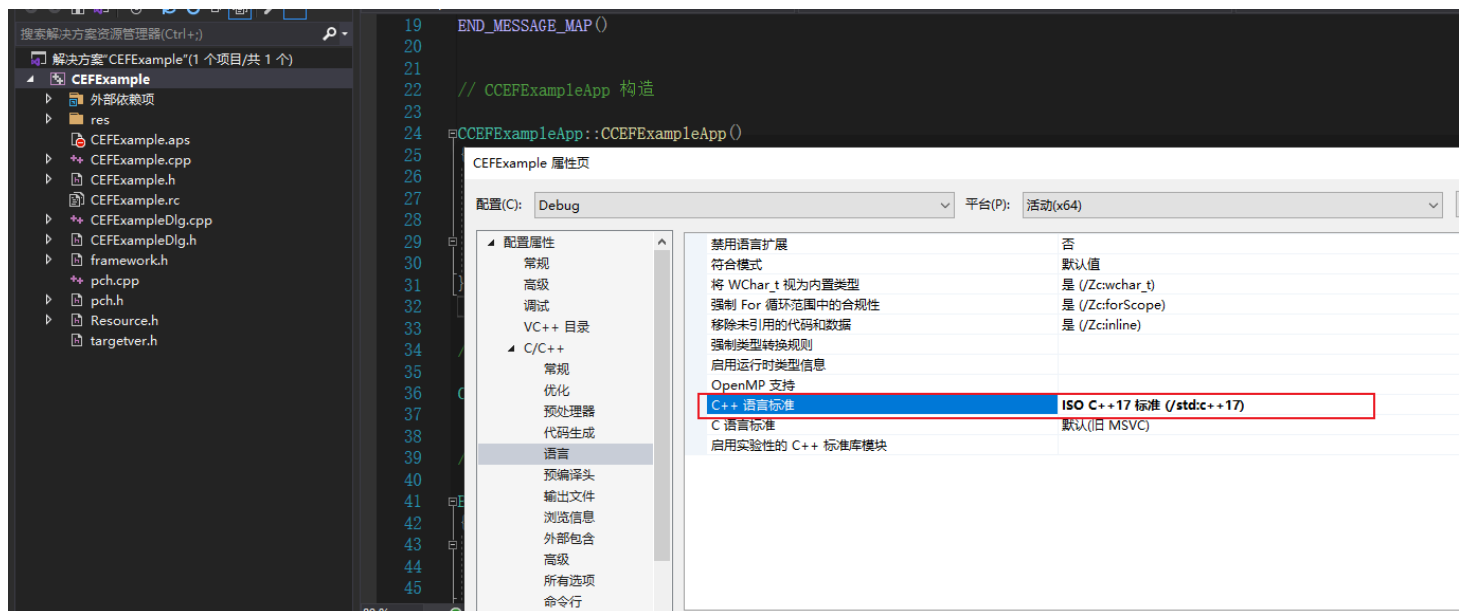
然后在VS中对工程右键 -> 属性 -> C/C++ -> 常规 -> 附加包含目录，添加“.\cef”



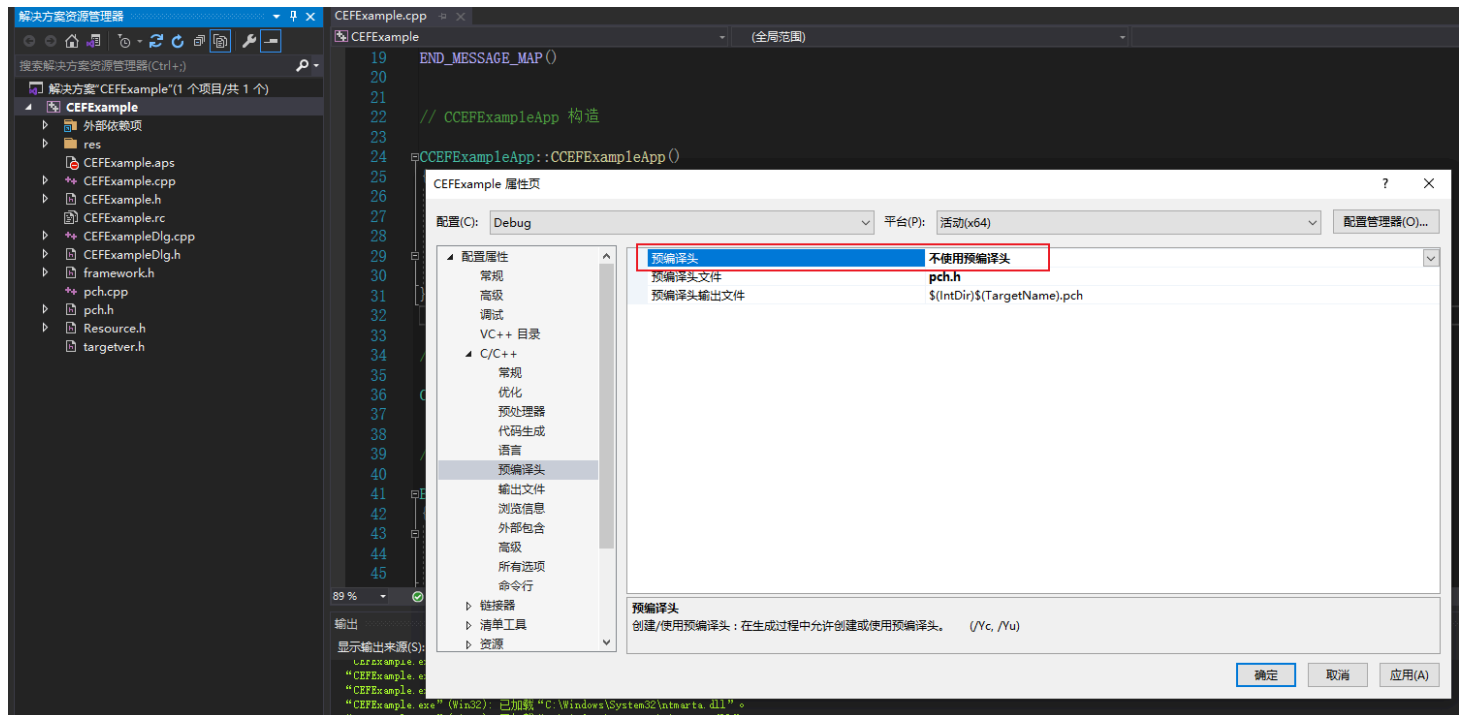
然后在VS中对工程右键 -> 属性 -> C/C++ -> 代码生成 -> 运行库，调整为 MT (debug调整为MTd，Release为MT)



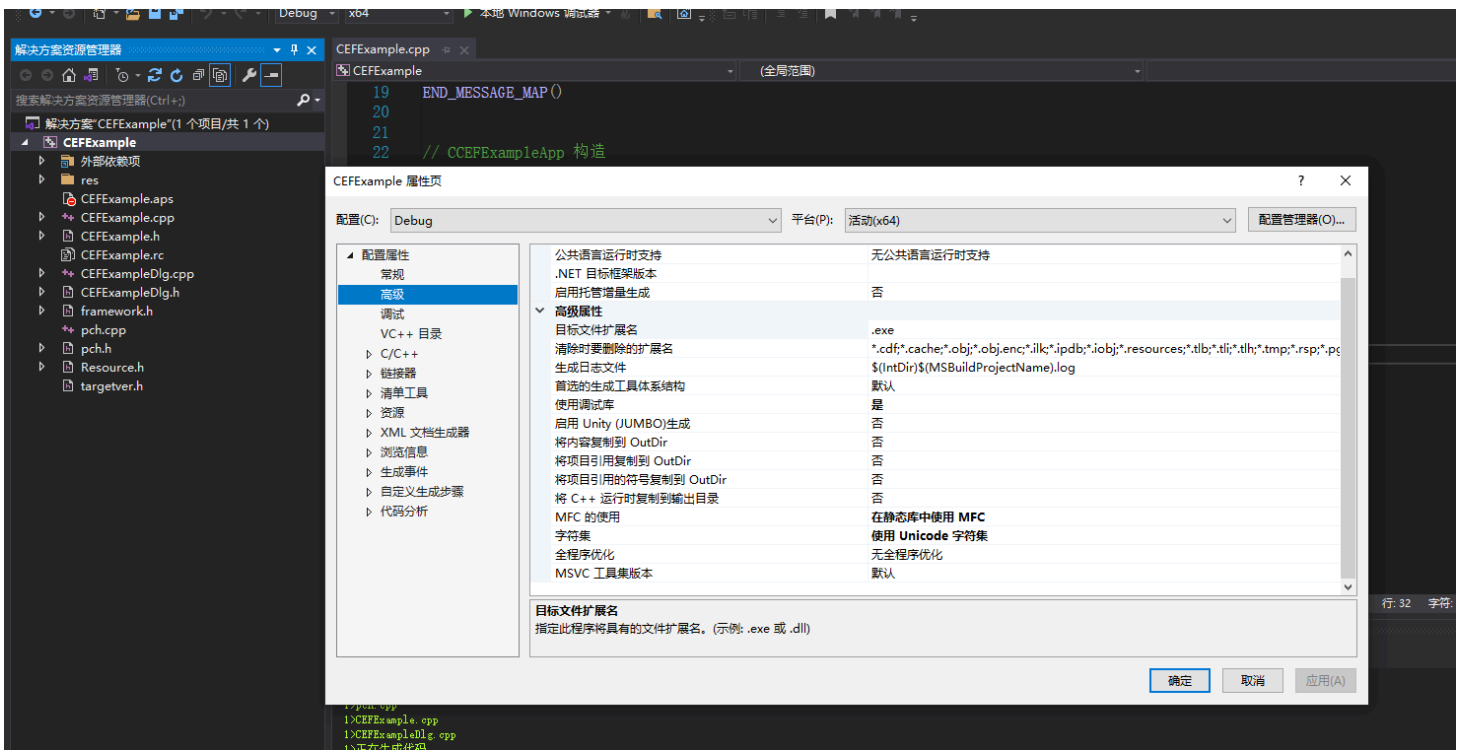
然后在VS中对工程右键 -> 属性 -> C/C++ -> 语言 -> c++语言标准，调整为 MT (debug调整为MTd，Release为MT)



然后在VS中对工程右键 -> 属性 -> C/C++ -> 预编译头 -> 预编译头，调整为不适用预编译头

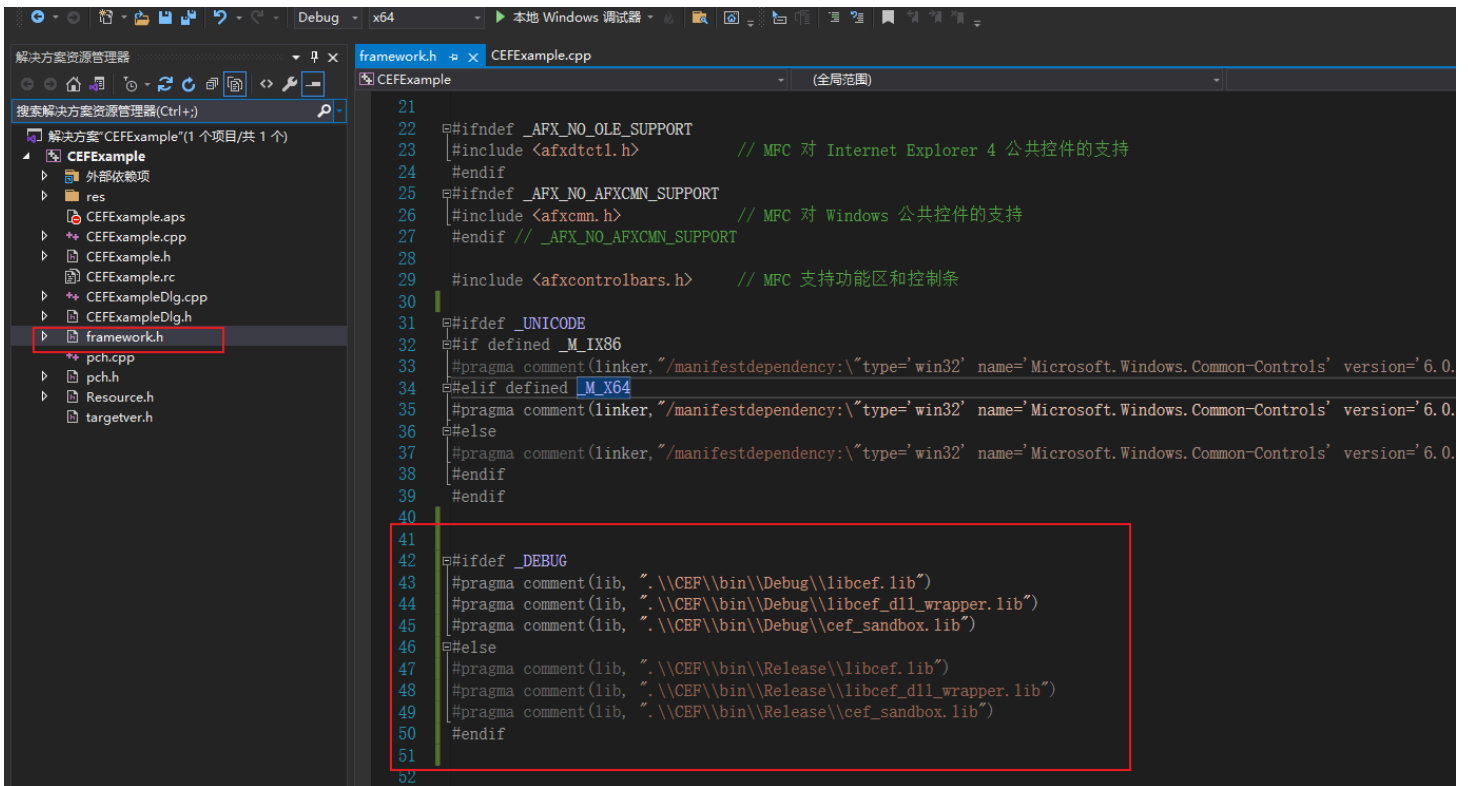


然后在VS中对工程右键 -> 属性 -> 高级 -> MFC的使用，改为“在静态库中使用MFC”



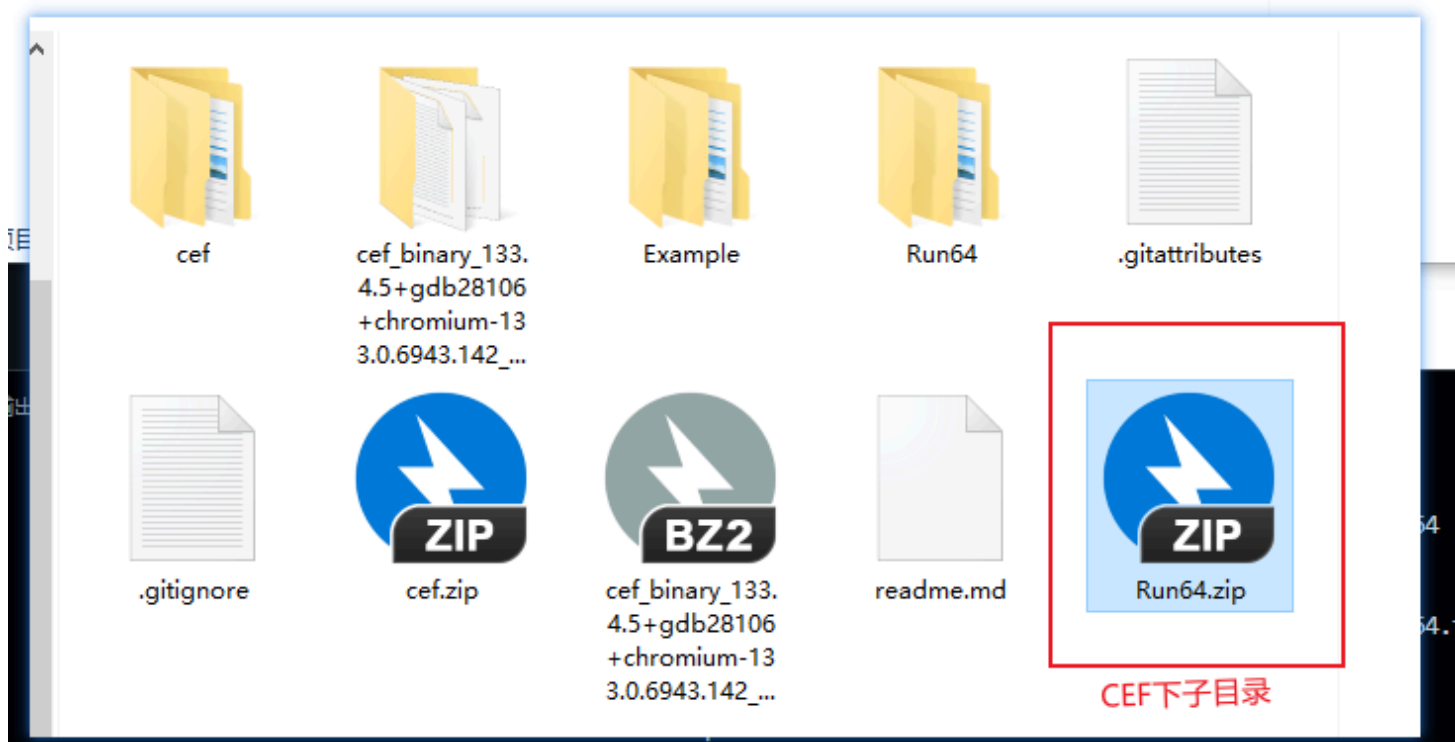
然后在工程中找到 `framework.h`，在最下面添加：

```
#ifdef _DEBUG
#pragma comment(lib, ".\\CEF\\bin\\Debug\\libcef.lib")
#pragma comment(lib, ".\\CEF\\bin\\Debug\\libcef_dll_wrapper.lib")
#pragma comment(lib, ".\\CEF\\bin\\Debug\\cef_sandbox.lib")
#else
#pragma comment(lib, ".\\CEF\\bin\\Release\\libcef.lib")
#pragma comment(lib, ".\\CEF\\bin\\Release\\libcef_dll_wrapper.lib")
#pragma comment(lib, ".\\CEF\\bin\\Release\\cef_sandbox.lib")
#endif
```



然后对工程右键生成，生成信息中会显示工程exe生成位置，一般是代码的解决方案文件所在目录同级的 x64\\Debug\\ 中，定位到exe所在位置，在CEF目录的子目录中下载 Run64.zip，然后解压后将对应子项目下的文件全部拷贝到exe所在目录：

locales	2025/3/8 20:27	文件夹	
CEFExample.exe	2025/3/8 20:23	应用程序	13,203 KB
CEFExample.pdb	2025/3/8 20:23	Program Debug...	32,564 KB
chrome_100_percent.pak	2025/3/8 15:24	PAK 文件	656 KB
chrome_200_percent.pak	2025/3/8 15:24	PAK 文件	1,034 KB
chrome_elf.dll	2025/3/8 15:24	应用程序扩展	1,806 KB
icudtl.dat	2025/3/8 15:24	DAT 文件	10,222 KB
libcef.dll	2025/3/8 15:24	应用程序扩展	338,567 KB
libEGL.dll	2025/3/8 15:24	应用程序扩展	515 KB
libGLESv2.dll	2025/3/8 15:24	应用程序扩展	13,428 KB
resources.pak	2025/3/8 15:24	PAK 文件	8,879 KB
v8_context_snapshot.bin	2025/3/8 15:24	BIN 文件	1,451 KB
vk_swiftshader.dll	2025/3/8 15:24	应用程序扩展	5,416 KB
vk_swiftshader_icd.json	2025/3/8 15:24	JSON File	1 KB
vulkan-1.dll	2025/3/8 15:24	应用程序扩展	909 KB



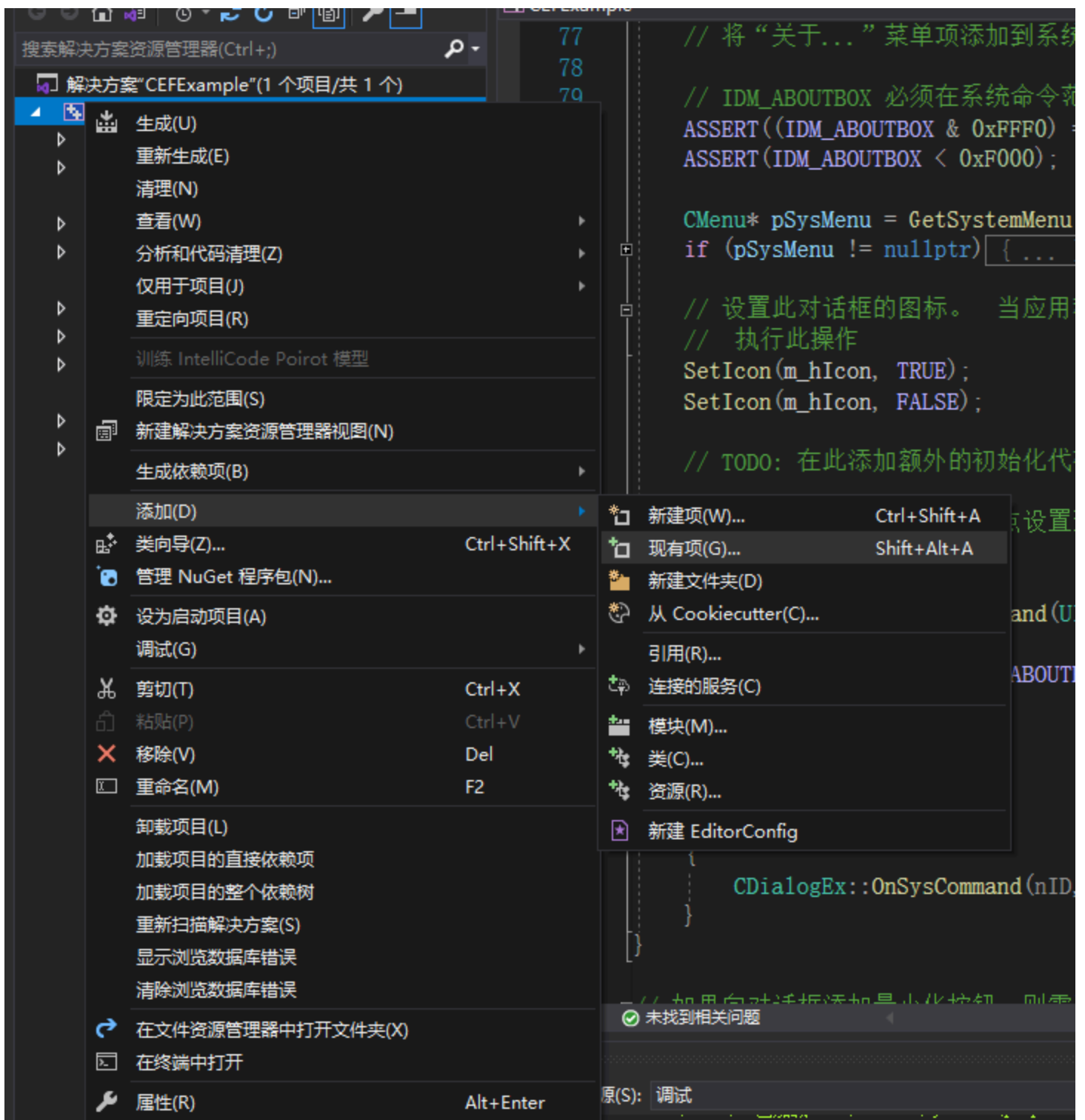
至此，CEF库导入完成

测试代码

首先去将cef目录下的子目录 test 中将测试类拷贝到工程下，这几个类其实来自于CEF源码中自带的 cefsimple工程，只是把simple_app.cc 中的默认路径改成了百度，其他无任何修改：

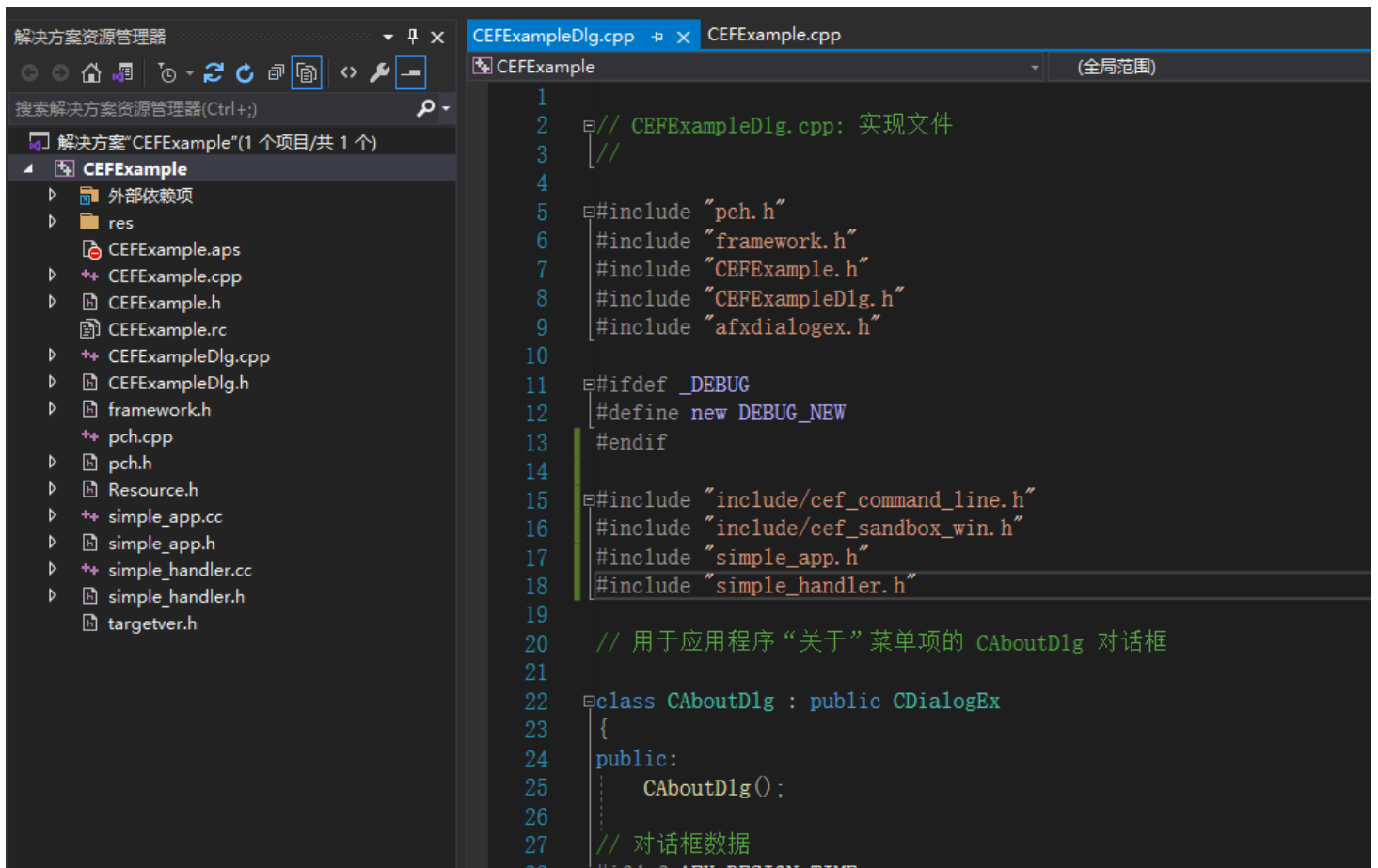
cef	2025/3/8 20:08	文件夹	
res	2025/3/8 20:04	文件夹	
x64	2025/3/8 20:05	文件夹	
CEFExample.aps	2025/3/8 20:04	APS 文件	90 KB
CEFExample.cpp	2025/3/8 20:04	C++ Source file	3 KB
CEFExample.h	2025/3/8 20:04	C/C++ Header	1 KB
CEFExample.rc	2025/3/8 20:04	Resource Sourc...	10 KB
CEFExample.vcxproj	2025/3/8 20:21	VC++ Project	11 KB
CEFExample.vcxproj.filters	2025/3/8 20:04	VC++ Project Fil...	3 KB
CEFExample.vcxproj.user	2025/3/8 20:05	Per-User Project...	1 KB
CEFExampleDlg.cpp	2025/3/8 20:29	C++ Source file	4 KB
CEFExampleDlg.h	2025/3/8 20:04	C/C++ Header	1 KB
framework.h	2025/3/8 20:20	C/C++ Header	2 KB
pch.cpp	2025/3/8 20:04	C++ Source file	1 KB
pch.h	2025/3/8 20:04	C/C++ Header	1 KB
Resource.h	2025/3/8 20:04	C/C++ Header	1 KB
simple_app.cc	2025/3/8 14:19	C++ Source file	7 KB
simple_app.h	2025/3/1 7:56	C/C++ Header	1 KB
simple_handler.cc	2025/3/8 14:30	C++ Source file	6 KB
simple_handler.h	2025/3/1 7:56	C/C++ Header	3 KB
targetver.h	2025/3/8 20:04	C/C++ Header	1 KB

在工程导入这几个文件，工程右键 -> 添加 -> 现有项，选择刚才考过来的文件

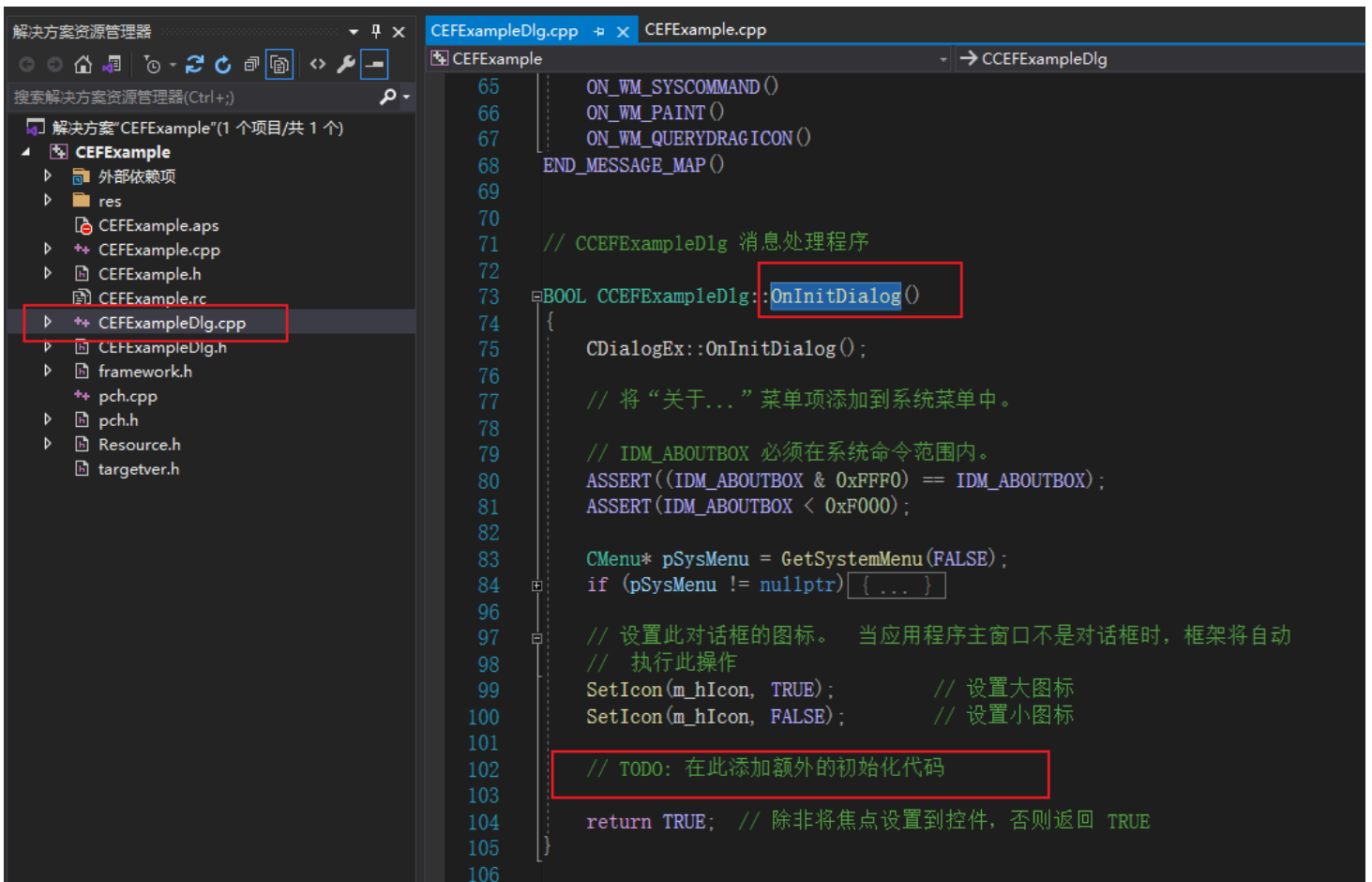


找到工程名称+Dlg 的类,

```
#include "include/cef_command_line.h"
#include "include/cef_sandbox_win.h"
#include "simple_app.h"
#include "simple_handler.h"
```



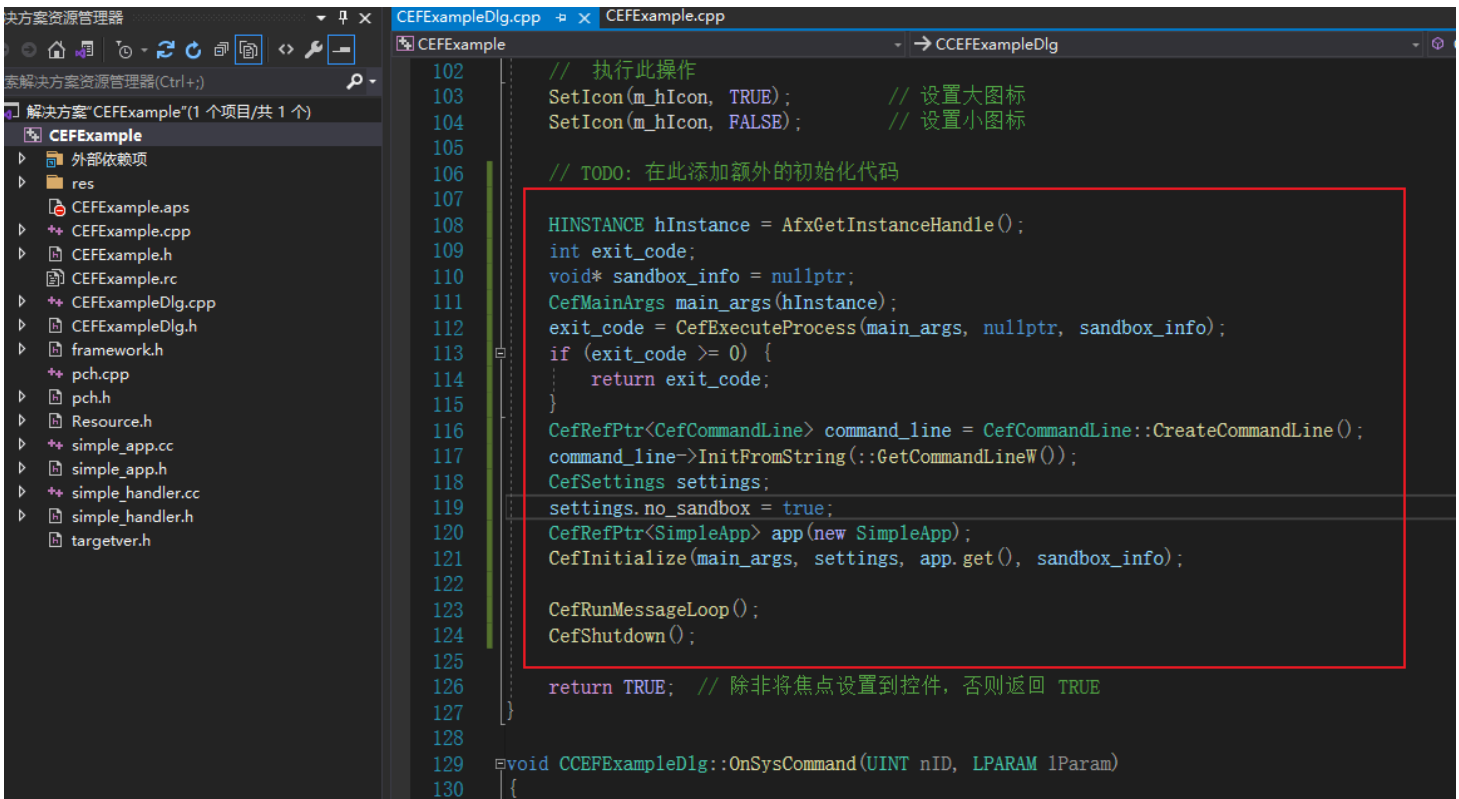
在当前类定位到 OnInitDialog 函数



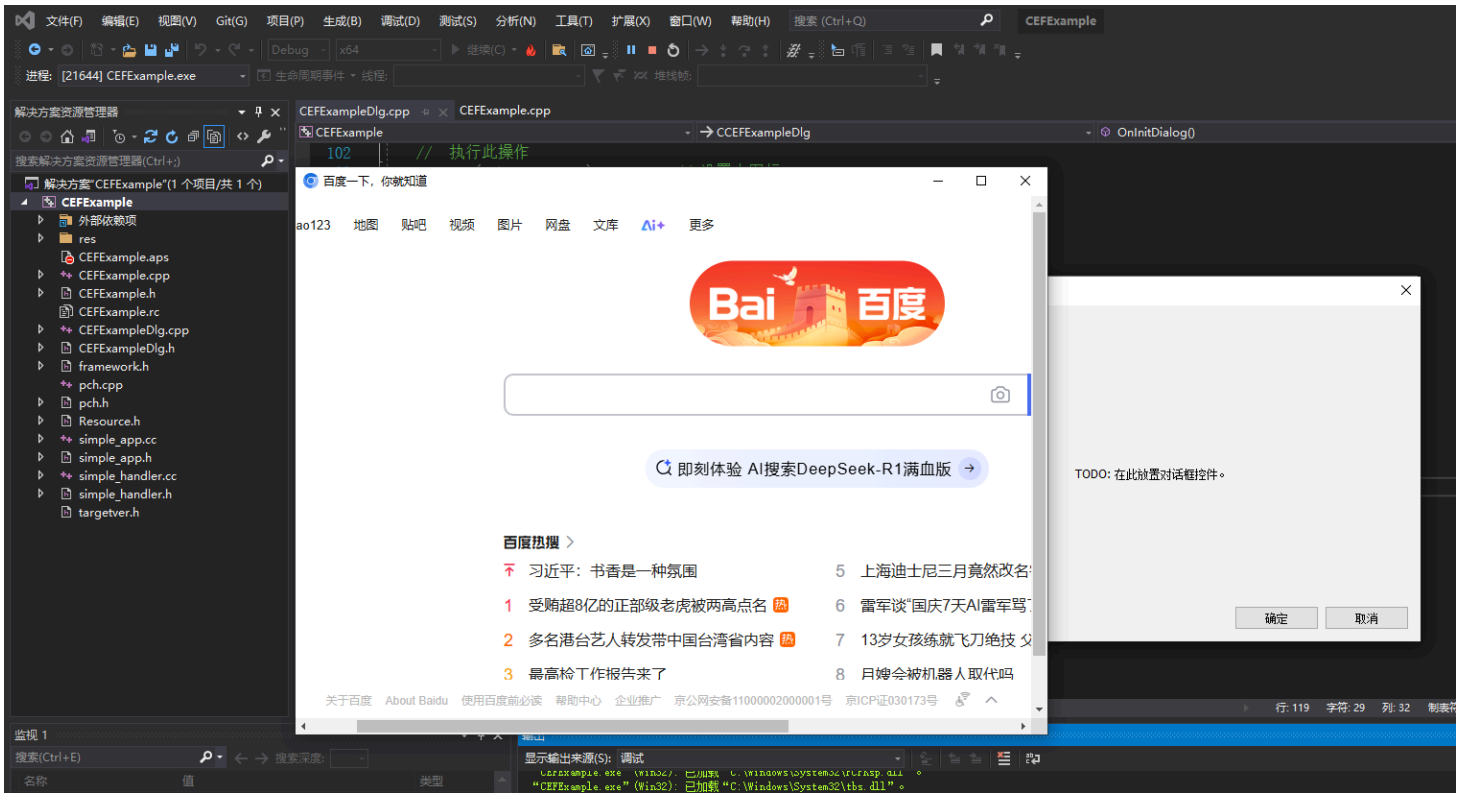
在TODO 后添加:

```
HINSTANCE hInstance = AfxGetInstanceHandle();
int exit_code;
void* sandbox_info = nullptr;
CefMainArgs main_args(hInstance);
exit_code = CefExecuteProcess(main_args, nullptr, sandbox_info);
if (exit_code >= 0) {
    return exit_code;
}
CefRefPtr<CefCommandLine> command_line = CefCommandLine::CreateCommandLine();
command_line->InitFromString(::GetCommandLineW());
CefSettings settings;
settings.no_sandbox = true;
CefRefPtr<SimpleApp> app(new SimpleApp);
CefInitialize(main_args, settings, app.get(), sandbox_info);

CefRunMessageLoop();
CefShutdown();
```



然后运行



会看到一个有浏览器并且打开了百度的页面，以及两个空的对话框。

这份测试代码是有bug的，不过只是为了测试CEF库是否可以正常使用，相关bug暂不定位处理；