Windows运作机制的核心是一个称作“动态连结”的概念。

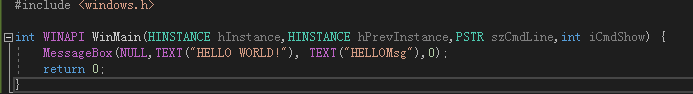
动态连结程式库（dynamic linking library）DLL

Windows的主要部分仅通过三个动态连结程式库实作。这代表了windows的三个主要子系统，他们被称作kernel,user和GDI

Kernel处理所有在传统上由作业系统核心处理的事物——记忆体管理、档案I/O和多工管理

Use指使用者界面，实作所有视窗运作机制。

GDI是一个图形装置界面，允许程式在荧幕和印表机上显示文字和图形。

这是第一个程序

Windows.h是主要的含入档案，它包含了其他windows表头档案，这些表头档案中最重要和最基本的是：

Windef.h,基本形态定义

winnt.h,支援Unicode的型态定义

winbase.h,kernel函式

winuser.h,使用者界面函式

wingdi.h,图形装置界面函式

正如在C程式中的进入点是函数main一样，windows程式的进入点是winmain

变量字首i表示int，sz表示“以零结束的字串”

Winmain函式宣告为返回一个int值。Winapi识别字在windef.h定义，语句为:

#define WINAPI \_\_stdcall

该语句指定了一个呼叫约定，包括如何生产机械码以在堆叠中放置函式呼叫的参数，许多windows函式呼叫宣告为WINAPI

Winmain的第一个参数被称作“执行体代号”，在windows程式设计中，代号仅是一个应用程式用来识别某些东西的数字，需要它在其他windows函式呼叫中作为参数。

在32位元windows版本中，传给winmain的第二个参数总是NULL。

Winmain的第三个参数是用于执行程式的命令列。某些windows应用程式利用它在程式启动时将档案载入记忆体。

Winmain的第四个参数指出程式最初显示的方式，可以是正常的或者是最大化地充满整个画面，或者是最小化显示在工作列中。

MessageBox函式：

用于显示短资讯——对话方块

第一个参数通常是视窗代号

第二个参数是在讯息方块主体中显示的字串

第三个参数是出现在讯息方块标题上的字串，这些文字字串的每一个都被封装在一个TEXT巨集中。

MessageBox的第四个参数可以使在winuser.h中定义的一组字首MB\_开始的常数的组合，

