WM\_MENUSELECT消息。随着使用者在菜单项中移动光标或者鼠标，程序会收到。这对实作那些包含对菜单项的文字描述的状态列是很有帮助的。WM\_MENUSELECT的参数如下所示：

LOWORD (wParam)：被选中项目：菜单ID或者弹出式菜单句柄

HIWORD (wParam)：选择旗标

lParam: 包含被选中项目的菜单句柄

当Windows准备显示一个弹出式菜单时，它给窗口消息处理程序发送一个WM\_INITMENUPOPUP消息，参数如下：

wParam： 弹出式菜单句柄

LOWORD (lParam)：弹出式菜单索引

HIWORD (lParam)： 系统菜单为1，其它为0

WM\_COMMAND，它表示使用者已经从菜单中选中了一个被启用的菜单项。WM\_COMMAND消息也可以由子窗口控件产生。如果您碰巧为菜单和子窗口控件使用同一ID码，那么您可以通过lParam的值来区别它们，菜单项的lParam其值为0，请参见表10-1。

|  |
| --- |
| 表10-1 |

|  |  |  |
| --- | --- | --- |
|  | **菜单** | **控件** |
| LOWORD (wParam): | 菜单ID | 控件ID |
| HIWORD (wParam): | 0 | 通知码 |
| lParam: | 0 | 子窗口句柄 |

WM\_SYSCOMMAND消息类似于WM\_COMMAND消息，只是WM\_SYSCOMMAND表示使用者从系统菜单中选择一个启用的菜单项：

wParam: 菜单ID

lParam: 0

**然而，如果WM\_SYSCOMMAND消息是由按鼠标按键产生的，LOWORD（lParam）和HIWORD（lParam）将包含鼠标光标位置的x和y屏幕坐标。**

对于WM\_SYSCOMMAND，菜单ID指示系统菜单中的哪一项被选中。对于预先定义的系统菜单项，较低的那四个位应该和0xFFF0进行AND运算来屏蔽掉，结果值应该为下列之一：SC\_SIZE、SC\_MOVE、SC\_MINIMIZE、SC\_MAXIMIZE、SC\_NEXTWINDOW、SC\_PREVWINDOW、SC\_CLOSE、SC\_VSCROLL、SC\_HSCROLL、SC\_ARRANGE、SC\_RESTORE和SC\_TASKLIST。此外，wParam可以是SC\_MOUSEMENU或SC\_KEYMENU。

**如果您在系统菜单中添加菜单项，那么wParam的低字组将是您定义的菜单ID。为了避免与预先定义的菜单ID相冲突，应用程序应该使用小于0xF000的值**，这对于将一般的WM\_SYSCOMMAND消息发送给DefWindowProc是很重要的。如果您不这样做，那么您实际上就是禁用了正常的系统菜单命令。

WM\_MENUCHAR。实际上，它根本不是菜单消息。如果使用者按下Alt和一个与菜单项不匹配的字符时，或者在显示弹出式菜单而使用者按下一个与弹出式菜单里的项目不匹配的字符键时。随WM\_MENUCHAR消息一起发送的参数如下所示：

LOWORD (wParam): 字符代码（ASCII或Unicode）

HIWORD (wParam): 选择码

lParam: 菜单句柄

选择码是：

* 0 不显示弹出式菜单
* MF\_POPUP 显示弹出式菜单
* MF\_SYSMENU 显示系统弹出式菜单

Windows程序通常把该消息传递给DefWindowProc，它一般给Windows传回0，这会使Windows发出哔声。