ZeroMemory(&wndclass,Sizeof(WNDCLASS));

wndclass.style = CS\_HREDRAW | CS\_VREDRAW;

wndclass.lpfnWndProc = WndProc;

wndclass.hInstance = hinstance;

wndclass.hIcon = LoadIcon(NULL,IDI\_APPLICATION);

wndclass.hCursor = LoadCursor(NULL,IDC\_ARROW);

wndclass.lpszClassName = szappname;

wndclass.lpszMenuName = NULL;

wndclass.hbrBackground = (HBRUSH) GetStockObject (WHITE\_BRUSH);

**style**:指定类风格

CS\_BYTEALIGNCLIENT: 在字节边界上(在x方向上)定位窗口的用户区域的位置

CS\_BYTEALIGNWINDOW: 在字节边界上(在x方向上)定位窗口的位置

CS\_CLASSDC: 该[窗口类](http://baike.haosou.com/doc/1541156-1629239.html" \t "_blank)的所有窗口实例都共享一个窗口类DC

CS\_DBLCLKS: 允许向窗口发送双击鼠标键的消息

CS\_GLOBALCLASS: 当调用CreateWindow 或 CreateWindowEx 函数来创建窗口时允许它的hInstance参数和注册窗口类时传递给RegisterClass 的 hInstance参数不同。如果不指定该风格，则这两个 hInstance 必须相同。

CS\_HREDRAW: 当水平长度改变或移动窗口时，重画整个窗口

CS\_NOCLOSE: 禁止系统菜单的关闭选项

CS\_OWNDC: 给予每个窗口实例它本身的DC。注意，尽管这样是很方便，但它必须慎重使用，因为每个DC大约要占800个字节的内存。

CS\_PARENTDC: 将子窗口的裁剪区域设置到父窗口的DC中去，这样子窗口便可以在父窗口上绘制自身。注意，这是子窗口还是从[系统缓存](http://baike.haosou.com/doc/5714968-5927694.html" \t "_blank)中获取DC，而不是使用父窗口的DC。使用该风格可以提高系统性能。

CS\_SAVEBITS: 以位图形式保存被该窗口遮挡的屏幕部分，这样当给窗口移动以后，系统便可以用该保存的位图恢复屏幕移动的相应部分，从而系统不用向被该窗口遮挡的窗口发送 WM\_PAINT 消息。该特性对于菜单类型的窗口比较合适，因为它通常是简短的显示一下之后便消失。设置该特性将增加显示该窗口的时间，因为它通常要先分配保存位图的内存。

CS\_VREDRAW: 当垂直长度改变或移动窗口时，重画整个窗口