|  |  |  |
| --- | --- | --- |
| **消息名** | **消息值** | **说明** |
| **WM\_CREATE** | 0x0001 | 应用程序创建一个窗口 |
| **WM\_DESTROY** | 0x0002 | 一个窗口被销毁 |
| **WM\_MOVE** | 0x0003 | 移动一个窗口 |
| **WM\_SIZE** | 0x0005 | 改变一个窗口的大小 |
| **WM\_ACTIVATE** | 0x0006 | 一个窗口被激活或失去激活状态 |
| **WM\_SETFOCUS** | 0x0007 | 获得焦点后 |
| **WM\_KILLFOCUS** | 0x0008 | 失去焦点 |
| **WM\_ENABLE** | 0x000A | 改变enable状态 |
| **WM\_SETREDRAW** | 0x000B | 设置窗口是否能重画 |
| **WM\_SETTEXT** | 0x000C | 应用程序发送此消息来设置一个窗口的文本 |
| **WM\_GETTEXT** | 0x000D | 应用程序发送此消息来复制对应窗口的文本到缓冲区 |
| **WM\_GETTEXTLENGTH** | 0x000E | 得到与一个窗口有关的文本的长度（不包含空字符） |
| **WM\_PAINT** | 0x000F | 要求一个窗口重画自己 |
| **WM\_CLOSE** | 0x0010 | 当一个窗口或应用程序要关闭时发送一个信号 |
| **WM\_QUERYENDSESSION** | 0x0011 | 当用户选择结束对话框或程序自己调用ExitWindows函数 |
| **WM\_QUIT** | 0x0012 | 用来结束程序运行或当程序调用postquitmessage函数 |
| **WM\_QUERYOPEN** | 0x0013 | 当用户窗口恢复以前的大小位置时，把此消息发送给某个图标 |
| **WM\_ERASEBKGND** | 0x0014 | 当窗口背景必须被擦除时（例在窗口改变大小时） |
| **WM\_SYSCOLORCHANGE** | 0x0015 | 当系统颜色改变时，发送此消息给所有顶级窗口 |
| **WM\_ENDSESSION** | 0x0016 | 当系统进程发出WM\_QUERYENDSESSION消息后,此消息发送给应用程序,通知它对话是否结束 |
| **WM\_SHOWWINDOW** | 0x0018 | 当隐藏或显示窗口是发送此消息给这个窗口 |
| **WM\_ACTIVATEAPP** | 0x001C | 发此消息给应用程序哪个窗口是激活的，哪个是非激活的 |
| **WM\_FONTCHANGE** | 0x001D | 当系统的字体资源库变化时发送此消息给所有顶级窗口 |
| **WM\_TIMECHANGE** | 0x001E | 当系统的时间变化时发送此消息给所有顶级窗口 |
| **WM\_CANCELMODE** | 0x001F | 发送此消息来取消某种正在进行的摸态（操作） |
| **WM\_SETCURSOR** | 0x0020 | 如果鼠标引起光标在某个窗口中移动且鼠标输入没有被捕获时，就发消息给某个窗口 |
| **WM\_MOUSEACTIVATE** | 0x0021 | 当光标在某个非激活的窗口中而用户正按着鼠标的某个键发送此消息给当前窗口 |
| **WM\_CHILDACTIVATE** | 0x0022 | 发送此消息给MDI子窗口当用户点击此窗口的标题栏，或当窗口被激活，移动，改变大小 |
| **WM\_QUEUESYNC** | 0x0023 | 此消息由基于计算机的训练程序发送，通过WH\_JOURNALPALYBACK的hook程序分离出用户输入消息 |
| **WM\_GETMINMAXINFO** | 0x0024 | 此消息发送给窗口当它将要改变大小或位置 |
| **WM\_PAINTICON** | 0x0026 | 发送给最小化窗口当它图标将要被重画 |
| **WM\_ICONERASEBKGND** | 0x0027 | 此消息发送给某个最小化窗口，仅当它在画图标前它的背景必须被重画 |
| **WM\_NEXTDLGCTL** | 0x0028 | 发送此消息给一个对话框程序去更改焦点位置 |
| **WM\_SPOOLERSTATUS** | 0x002A | 每当打印管理列队增加或减少一条作业时发出此消息 |
| **WM\_DRAWITEM** | 0x002B | 当button，combobox，listbox，menu的可视外观改变时发送此消息给这些控件的所有者 |
| **WM\_MEASUREITEM** | 0x002C | 当button, combo box, list box, list view control, or menu item 被创建时发送此消息给控件的所有者 |
| **WM\_DELETEITEM** | 0x002D | 当the list box 或 combo box 被销毁 或 当 某些项被删除通过LB\_DELETESTRING, LB\_RESETCONTENT,CB\_DELETESTRING, or CB\_RESETCONTENT消息 |
| **WM\_VKEYTOITEM** | 0x002E | 此消息有一个LBS\_WANTKEYBOARDINPUT风格的发出给它的所有者来响应WM\_KEYDOWN消息 |
| **WM\_CHARTOITEM** | 0x002F | 此消息由一个LBS\_WANTKEYBOARDINPUT风格的列表框发送给他的所有者来响应WM\_CHAR消息 |
| **WM\_SETFONT** | 0x0030 | 当绘制文本时程序发送此消息得到控件要用的颜色 |
| **WM\_GETFONT** | 0x0031 | 应用程序发送此消息得到当前控件绘制文本的字体 |
| **WM\_SETHOTKEY** | 0x0032 | 应用程序发送此消息让一个窗口与一个热键相关连 |
| **WM\_GETHOTKEY** | 0x0033 | 应用程序发送此消息来判断热键与某个窗口是否有关联 |
| **WM\_QUERYDRAGICON** | 0x0037 | 此消息发送给最小化窗口，当此窗口将要被拖放而它的类中没有定义图标，应用程序能返回一个图标或光标的句柄，当用户拖放图标时系统显示这个图标或光标 |
| **WM\_COMPAREITEM** | 0x0039 | 发送此消息来判定combobox或listbox新增加的项的相对位置 |
| **WM\_COMPACTING** | 0x0041 | 显示内存已经很少了 |
| **WM\_WINDOWPOSCHANGING** | 0x0046 | 发送此消息给那个窗口的大小和位置将要被改变时，来调用setwindowpos函数或其它窗口管理函数 |
| **WM\_WINDOWPOSCHANGED** | 0x0047 | 发送此消息给那个窗口的大小和位置已经被改变时，来调用setwindowpos函数或其它窗口管理函数 |
| **WM\_POWER** | 0x0048 | （适用于16位的windows）当系统将要进入暂停状态时发送此消息 |
| **WM\_COPYDATA** | 0x004A | 当一个应用程序传递数据给另一个应用程序时发送此消息 |
| **WM\_CANCELJOURNAL** | 0x004B | 当某个用户取消程序日志激活状态，提交此消息给程序 |
| **WM\_NOTIFY** | 0x004E | 当某个控件的某个事件已经发生或这个控件需要得到一些信息时，发送此消息给它的父窗口 |
| **WM\_INPUTLANGCHANGEREQUEST** | 0x0050 | 当用户选择某种输入语言，或输入语言的热键改变 |
| **WM\_INPUTLANGCHANGE** | 0x0051 | 当平台现场已经被改变后发送此消息给受影响的最顶级窗口 |
| **WM\_TCARD** | 0x0052 | 当程序已经初始化windows帮助例程时发送此消息给应用程序 |
| **WM\_HELP** | 0x0053 | 此消息显示用户按下了F1，如果某个菜单是激活的，就发送此消息个此窗口关联的菜单，否则就发送给有焦点的窗口，如果当前都没有焦点，就把此消息发送给当前激活的窗口 |
| **WM\_USERCHANGED** | 0x0054 | 当用户已经登入或退出后发送此消息给所有的窗口，当用户登入或退出时系统更新用户的具体设置信息，在用户更新设置时系统马上发送此消息 |
| **WM\_NOTIFYFORMAT** | 0x0055 | 公用控件，自定义控件和他们的父窗口通过此消息来判断控件是使用ANSI还是UNICODE结构在WM\_NOTIFY消息，使用此控件能使某个控件与它的父控件之间进行相互通信 |
| **WM\_CONTEXTMENU** | 0x007B | 当用户某个窗口中点击了一下右键就发送此消息给这个窗口 |
| **WM\_STYLECHANGING** | 0x007C | 当调用SETWINDOWLONG函数将要改变一个或多个 窗口的风格时发送此消息给那个窗口 |
| **WM\_STYLECHANGED** | 0x007D | 当调用SETWINDOWLONG函数一个或多个 窗口的风格后发送此消息给那个窗口 |
| **WM\_DISPLAYCHANGE** | 0x007E | 当显示器的分辨率改变后发送此消息给所有的窗口 |
| **WM\_GETICON** | 0x007F | 此消息发送给某个窗口来返回与某个窗口有关连的大图标或小图标的句柄 |
| **WM\_SETICON** | 0x0080 | 程序发送此消息让一个新的大图标或小图标与某个窗口关联 |
| **WM\_NCCREATE** | 0x0081 | 当某个窗口第一次被创建时，此消息在WM\_CREATE消息发送前发送 |
| **WM\_NCDESTROY** | 0x0082 | 此消息通知某个窗口，非客户区正在销毁 |
| **WM\_NCCALCSIZE** | 0x0083 | 当某个窗口的客户区域必须被核算时发送此消息 |
| **WM\_NCHITTEST** | 0x0084 | 移动鼠标，按住或释放鼠标时发生 |
| **WM\_NCPAINT** | 0x0085 | 程序发送此消息给某个窗口当它（窗口）的框架必须被绘制时 |
| **WM\_NCACTIVATE** | 0x0086 | 此消息发送给某个窗口 仅当它的非客户区需要被改变来显示是激活还是非激活状态 |
| **WM\_GETDLGCODE** | 0x0087 | 发送此消息给某个与对话框程序关联的控件，widdows控制方位键和TAB键使输入进入此控件,通过响应WM\_GETDLGCODE消息，应用程序可以把他当成一个特殊的输入控件并能处理它 |
| **WM\_NCMOUSEMOVE** | 0x00A0 | 当光标在一个窗口的非客户区内移动时发送此消息给这个窗口 //非客户区为：窗体的标题栏及窗的边框体 |
| **WM\_NCLBUTTONDOWN** | 0x00A1 | 当光标在一个窗口的非客户区同时按下鼠标左键时提交此消息 |
| **WM\_NCLBUTTONUP** | 0x00A2 | 当用户释放鼠标左键同时光标某个窗口在非客户区十发送此消息 |
| **WM\_NCLBUTTONDBLCLK** | 0x00A3 | 当用户双击鼠标左键同时光标某个窗口在非客户区十发送此消息 |
| **WM\_NCRBUTTONDOWN** | 0x00A4 | 当用户按下鼠标右键同时光标又在窗口的非客户区时发送此消息 |
| **WM\_NCRBUTTONUP** | 0x00A5 | 当用户释放鼠标右键同时光标又在窗口的非客户区时发送此消息 |
| **WM\_NCRBUTTONDBLCLK** | 0x00A6 | 当用户双击鼠标右键同时光标某个窗口在非客户区十发送此消息 |
| **WM\_NCMBUTTONDOWN** | 0x00A7 | 当用户按下鼠标中键同时光标又在窗口的非客户区时发送此消息 |
| **WM\_NCMBUTTONUP** | 0x00A8 | 当用户释放鼠标中键同时光标又在窗口的非客户区时发送此消息 |
| **WM\_NCMBUTTONDBLCLK** | 0x00A9 | 当用户双击鼠标中键同时光标又在窗口的非客户区时发送此消息 |
| **WM\_KEYDOWN** | 0x0100 | 按下一个键 |
| **WM\_KEYUP** | 0x0101 | 释放一个键 |
| **WM\_CHAR** | 0x0102 | 按下某键，并已发出WM\_KEYDOWN，WM\_KEYUP消息 |
| **WM\_DEADCHAR** | 0x0103 | 当用translatemessage函数翻译WM\_KEYUP消息时发送此消息给拥有焦点的窗口 |
| **WM\_SYSKEYDOWN** | 0x0104 | 当用户按住ALT键同时按下其它键时提交此消息给拥有焦点的窗口 |
| **WM\_SYSKEYUP** | 0x0105 | 当用户释放一个键同时ALT 键还按着时提交此消息给拥有焦点的窗口 |
| **WM\_SYSCHAR** | 0x0106 | 当WM\_SYSKEYDOWN消息被TRANSLATEMESSAGE函数翻译后提交此消息给拥有焦点的窗口 |
| **WM\_SYSDEADCHAR** | 0x0107 | 当WM\_SYSKEYDOWN消息被TRANSLATEMESSAGE函数翻译后发送此消息给拥有焦点的窗口 |
| **WM\_INITDIALOG** | 0x0110 | 在一个对话框程序被显示前发送此消息给它，通常用此消息初始化控件和执行其它任务 |
| **WM\_COMMAND** | 0x0111 | 当用户选择一条菜单命令项或当某个控件发送一条消息给它的父窗口，一个快捷键被翻译 |
| **WM\_SYSCOMMAND** | 0x0112 | 当用户选择窗口菜单的一条命令或当用户选择最大化或最小化时那个窗口会收到此消息 |
| **WM\_TIMER** | 0x0113 | 发生了定时器事件 |
| **WM\_HSCROLL** | 0x0114 | 当一个窗口标准水平滚动条产生一个滚动事件时发送此消息给那个窗口，也发送给拥有它的控件 |
| **WM\_VSCROLL** | 0x0115 | 当一个窗口标准垂直滚动条产生一个滚动事件时发送此消息给那个窗口也，发送给拥有它的控件 |
| **WM\_INITMENU** | 0x0116 | 当一个菜单将要被激活时发送此消息，它发生在用户菜单条中的某项或按下某个菜单键，它允许程序在显示前更改菜单 |
| **WM\_INITMENUPOPUP** | 0x0117 | 当一个下拉菜单或子菜单将要被激活时发送此消息，它允许程序在它显示前更改菜单，而不要改变全部 |
| **WM\_MENUSELECT** | 0x011F | 当用户选择一条菜单项时发送此消息给菜单的所有者(一般是窗口） |
| **WM\_MENUCHAR** | 0x0120 | 当菜单已被激活用户按下了某个键（不同于加速键），发送此消息给菜单的所有者 |
| **WM\_ENTERIDLE** | 0x0121 | 当一个模态对话框或菜单进入空载状态时发送此消息给它的所有者，一个模态对话框或菜单进入空载状态就是在处理完一条或几条先前的消息后没有消息它的列队中等待 |
| **WM\_CTLCOLORMSGBOX** | 0x0132 | 在windows绘制消息框前发送此消息给消息框的所有者窗口，通过响应这条消息，所有者窗口可以通过使用给定的相关显示设备的句柄来设置消息框的文本和背景颜色 |
| **WM\_CTLCOLOREDIT** | 0x0133 | 当一个编辑型控件将要被绘制时发送此消息给它的父窗口；通过响应这条消息，所有者窗口可以通过使用给定的相关显示设备的句柄来设置编辑框的文本和背景颜色 |
| **WM\_CTLCOLORLISTBOX** | 0x0134 | 当一个列表框控件将要被绘制前发送此消息给它的父窗口；通过响应这条消息，所有者窗口可以通过使用给定的相关显示设备的句柄来设置列表框的文本和背景颜色 |
| **WM\_CTLCOLORBTN** | 0x0135 | 当一个按钮控件将要被绘制时发送此消息给它的父窗口；通过响应这条消息，所有者窗口可以通过使用给定的相关显示设备的句柄来设置按纽的文本和背景颜色 |
| **WM\_CTLCOLORDLG** | 0x0136 | 当一个对话框控件将要被绘制前发送此消息给它的父窗口；通过响应这条消息，所有者窗口可以通过使用给定的相关显示设备的句柄来设置对话框的文本背景颜色 |
| **WM\_CTLCOLORSCROLLBAR** | 0x0137 | 当一个滚动条控件将要被绘制时发送此消息给它的父窗口；通过响应这条消息，所有者窗口可以通过使用给定的相关显示设备的句柄来设置滚动条的背景颜色 |
| **WM\_CTLCOLORSTATIC** | 0x0138 | 当一个静态控件将要被绘制时发送此消息给它的父窗口；通过响应这条消息，所有者窗口可以通过使用给定的相关显示设备的句柄来设置静态控件的文本和背景颜色 |
| **WM\_MOUSEMOVE** | 0x0200 | 移动鼠标 |
| **WM\_LBUTTONDOWN** | 0x0201 | 按下鼠标左键 |
| **WM\_LBUTTONUP** | 0x0202 | 释放鼠标左键 |
| **WM\_LBUTTONDBLCLK** | 0x0203 | 双击鼠标左键 |
| **WM\_RBUTTONDOWN** | 0x0204 | 按下鼠标右键 |
| **WM\_RBUTTONUP** | 0x0205 | 释放鼠标右键 |
| **WM\_RBUTTONDBLCLK** | 0x0206 | 双击鼠标右键 |
| **WM\_MBUTTONDOWN** | 0x0207 | 按下鼠标中键 |
| **WM\_MBUTTONUP** | 0x0208 | 释放鼠标中键 |
| **WM\_MBUTTONDBLCLK** | 0x0209 | 双击鼠标中键 |
| **WM\_MOUSEWHEEL** | 0x020A | 当鼠标轮子转动时发送此消息个当前有焦点的控件 |
| **WM\_PARENTNOTIFY** | 0x0210 | 当MDI子窗口被创建或被销毁,或用户按了一下鼠标键而光标在子窗口上时发送此消息给它的父窗口 |
| **WM\_ENTERMENULOOP** | 0x0211 | 发送此消息通知应用程序的主窗口that已经进入了菜单循环模式 |
| **WM\_EXITMENULOOP** | 0x0212 | 发送此消息通知应用程序的主窗口that已退出了菜单循环模式 |
| **WM\_SIZING** | 532 | 当用户正在调整窗口大小时发送此消息给窗口；通过此消息应用程序可以监视窗口大小和位置也可以修改他们 |
| **WM\_CAPTURECHANGED** | 533 | 发送此消息给窗口当它失去捕获的鼠标时 |
| **WM\_MOVING** | 534 | 当用户在移动窗口时发送此消息，通过此消息应用程序可以监视窗口大小和位置也可以修改他们 |
| **WM\_POWERBROADCAST** | 536 | 此消息发送给应用程序来通知它有关电源管理事件 |
| **WM\_DEVICECHANGE** | 537 | 当设备的硬件配置改变时发送此消息给应用程序或设备驱动程序 |
| **WM\_MDICREATE** | 0x0220 | 应用程序发送此消息给多文档的客户窗口来创建一个MDI 子窗口 |
| **WM\_MDIDESTROY** | 0x0221 | 应用程序发送此消息给多文档的客户窗口来关闭一个MDI 子窗口 |
| **WM\_MDIACTIVATE** | 0x0222 | 应用程序发送此消息给多文档的客户窗口通知客户窗口激活另一个MDI子窗口，当客户窗口收到此消息后，它发出WM\_MDIACTIVE消息给MDI子窗口（未激活）激活它 |
| **WM\_MDIRESTORE** | 0x0223 | 程序发送此消息给MDI客户窗口让子窗口从最大最小化恢复到原来大小 |
| **WM\_MDINEXT** | 0x0224 | 程序发送此消息给MDI客户窗口激活下一个或前一个窗口 |
| **WM\_MDIMAXIMIZE** | 0x0225 | 程序发送此消息给MDI客户窗口来最大化一个MDI子窗口 |
| **WM\_MDITILE** | 0x0226 | 程序发送此消息给MDI客户窗口以平铺方式重新排列所有MDI子窗口 |
| **WM\_MDICASCADE** | 0x0227 | 程序发送此消息给MDI客户窗口以层叠方式重新排列所有MDI子窗口 |
| **WM\_MDIICONARRANGE** | 0x0228 | 程序发送此消息给MDI客户窗口重新排列所有最小化的MDI子窗口 |
| **WM\_MDIGETACTIVE** | 0x0229 | 程序发送此消息给MDI客户窗口来找到激活的子窗口的句柄 |
| **WM\_MDISETMENU** | 0x0230 | 程序发送此消息给MDI客户窗口用MDI菜单代替子窗口的菜单 |
| **WM\_CUT** | 0x0300 | 程序发送此消息给一个编辑框或combobox来删除当前选择的文本 |
| **WM\_COPY** | 0x0301 | 程序发送此消息给一个编辑框或combobox来复制当前选择的文本到剪贴板 |
| **WM\_PASTE** | 0x0302 | 程序发送此消息给editcontrol或combobox从剪贴板中得到数据 |
| **WM\_CLEAR** | 0x0303 | 程序发送此消息给editcontrol或combobox清除当前选择的内容 |
| **WM\_UNDO** | 0x0304 | 程序发送此消息给editcontrol或combobox撤消最后一次操作 |
| **WM\_DESTROYCLIPBOARD** | 0x0307 | 当调用ENPTYCLIPBOARD函数时发送此消息给剪贴板的所有者 |
| **WM\_DRAWCLIPBOARD** | 0x0308 | 当剪贴板的内容变化时发送此消息给剪贴板观察链的第一个窗口；它允许用剪贴板观察窗口来显示剪贴板的新内容 |
| **WM\_PAINTCLIPBOARD** | 0x0309 | 当剪贴板包含CF\_OWNERDIPLAY格式的数据并且剪贴板观察窗口的客户区需要重画 |
| **WM\_SIZECLIPBOARD** | 0x030B | 当剪贴板包含CF\_OWNERDIPLAY格式的数据并且剪贴板观察窗口的客户区域的大小已经改变是此消息通过剪贴板观察窗口发送给剪贴板的所有者 |
| **WM\_ASKCBFORMATNAME** | 0x030C | 通过剪贴板观察窗口发送此消息给剪贴板的所有者来请求一个CF\_OWNERDISPLAY格式的剪贴板的名字 |
| **WM\_CHANGECBCHAIN** | 0x030D | 当一个窗口从剪贴板观察链中移去时发送此消息给剪贴板观察链的第一个窗口 |
| **WM\_HSCROLLCLIPBOARD** | 0x030E | 此消息通过一个剪贴板观察窗口发送给剪贴板的所有者,它发生在当剪贴板包含CFOWNERDISPALY格式的数据并且有个事件在剪贴板观察窗的水平滚动条上；所有者应滚动剪贴板图象并更新滚动条的值 |
| **WM\_QUERYNEWPALETTE** | 0x030F | 此消息发送给将要收到焦点的窗口,此消息能使窗口在收到焦点时同时有机会实现他的逻辑调色板 |
| **WM\_PALETTEISCHANGING** | 0x0310 | 当一个应用程序正要实现它的逻辑调色板时发此消息通知所有的应用程序 |
| **WM\_PALETTECHANGED** | 0x0311 | 此消息在一个拥有焦点的窗口实现它的逻辑调色板后发送此消息给所有顶级并重叠的窗口，以此来改变系统调色板 |
| **WM\_HOTKEY** | 0x0312 | 当用户按下由REGISTERHOTKEY函数注册的热键时提交此消息 |