



This is part of **Win16 API** which allow to create versions of program from one source code to run under OS/2 and Win16. Under OS/2 program can be running under Win-OS/2 if program is Windows NE executable, and with help on Windows Libraries for OS/2, if it is OS/2 NE executable. [Here](#) is a WLO to OS/2 API mapping draft

2021/09/01 04:23 · prokushev · [0 Comments](#)

Ordinal	Name	Description	Status	Version
1	<a href="#">MessageBox</a>	Displays a modal dialog box that contains a system icon, a set of buttons, and a brief application-specific message		
2	<a href="#">OldExitWindows</a>		Done	
5	<a href="#">InitApp</a>	Initializes an application instance		
6	<a href="#">PostQuitMessage</a>	Indicates that the application has requested to terminate		
7	<a href="#">ExitWindows</a>	Shuts down Windows and restarts the operating system		
10	<a href="#">SetTimer</a>	Creates a timer with the specified time-out value		
11	<a href="#">BEAR11</a>			
12	<a href="#">KillTimer</a>	Destroys the specified timer		
13	<a href="#">GetTickCount</a>	Retrieves the number of milliseconds that have elapsed since the system was started	Done	
14	<a href="#">GetTimerResolution</a>	Retrieves the minimum timer resolution for the system		
15	<a href="#">GetCurrentTime</a>	Retrieves the current Windows time	Done	
16	<a href="#">ClipCursor</a>	Confines the cursor to a rectangular area on the screen		
17	<a href="#">GetCursorPos</a>	Retrieves the position of the mouse cursor, in screen coordinates	Done	
18	<a href="#">SetCapture</a>	Sets the mouse capture to the specified window	Done	
19	<a href="#">ReleaseCapture</a>	Releases the mouse capture from a window and restores normal mouse input processing	Done	
20	<a href="#">SetDoubleClickTime</a>	Sets the double-click time for the mouse		
21	<a href="#">GetDoubleClickTime</a>	Retrieves the current double-click time for the mouse		
22	<a href="#">SetFocus</a>	Sets the keyboard focus to the specified window	Done	
23	<a href="#">GetFocus</a>	Retrieves the handle to the window that has the keyboard focus	Done	
24	<a href="#">RemoveProp</a>	Removes a property from a window's property list		
25	<a href="#">GetProp</a>	Retrieves a data handle from a window's property list		
26	<a href="#">SetProp</a>	Adds a new entry or changes an existing entry in a window's property list		
27	<a href="#">EnumProps</a>	Enumerates all entries in a window's property list		

Ordinal	Name	Description	Status	Version
28	<a href="#">ClientToScreen</a>	Converts the client-area coordinates of a specified point to screen coordinates		
29	<a href="#">ScreenToClient</a>	Converts the screen coordinates of a specified point to client coordinates		
30	<a href="#">WindowFromPoint</a>	Retrieves the handle of the window that contains the specified point		
31	<a href="#">IsIconic</a>	Determines whether the specified window is minimized (iconic)		
32	<a href="#">GetWindowRect</a>	Retrieves the dimensions of the bounding rectangle of the specified window		
33	<a href="#">GetClientRect</a>	Retrieves the coordinates of a window's client area		
34	<a href="#">EnableWindow</a>	Enables or disables mouse and keyboard input to the specified window		
35	<a href="#">IsWindowEnabled</a>	Determines whether the specified window is enabled for mouse and keyboard input		
36	<a href="#">GetWindowText</a>	Copies the text of the specified window's title bar into a buffer	Done	
37	<a href="#">SetWindowText</a>	Changes the text of the specified window's title bar	Done	
38	<a href="#">GetWindowTextLength</a>	Retrieves the length of the specified window's title bar text	Done	
39	<a href="#">BeginPaint</a>	Prepares the specified window for painting and fills a PAINTSTRUCT structure with painting information	Done	
40	<a href="#">EndPaint</a>	Marks the end of painting in the specified window	Done	
41	<a href="#">CreateWindow</a>	Creates an overlapped, pop-up, or child window	Done	
42	<a href="#">ShowWindow</a>	Sets the specified window's show state	Done	
43	<a href="#">CloseWindow</a>	Minimizes (but does not destroy) the specified window		
44	<a href="#">OpenIcon</a>	Restores a minimized (iconic) window to its previous size and position		
45	<a href="#">BringWindowToTop</a>	Brings the specified window to the top of the Z order		
46	<a href="#">GetParent</a>	Retrieves the handle of the specified window's parent window		
47	<a href="#">IsWindow</a>	Determines whether the specified handle is a window handle	Done	
48	<a href="#">IsChild</a>	Determines whether a window is a child window of the specified parent window		
49	<a href="#">IsWindowVisible</a>	Determines whether the specified window is visible		
50	<a href="#">FindWindow</a>	Retrieves the handle to the top-level window whose class name and window name match the specified strings		
52	<a href="#">AnyPopup</a>	Indicates whether an owned, visible, top-level pop-up, or overlapped window exists on the screen		

Ordinal	Name	Description	Status	Version
53	<a href="#">DestroyWindow</a>	Destroys the specified window		
54	<a href="#">EnumWindows</a>	Enumerates all top-level windows on the screen		
55	<a href="#">EnumChildWindows</a>	Enumerates the child windows that belong to the specified parent window		
56	<a href="#">MoveWindow</a>	Changes the position and dimensions of the specified window	Done	
57	<a href="#">RegisterClass</a>	Registers a window class for subsequent use in calls to the CreateWindow function	Done	
58	<a href="#">GetClassName</a>	Retrieves the name of the class to which the specified window belongs		
59	<a href="#">SetActiveWindow</a>	Sets the specified window to the active window		
60	<a href="#">GetActiveWindow</a>	Retrieves the handle to the active window		
61	<a href="#">ScrollWindow</a>	Scrolls the contents of the specified window's client area		
62	<a href="#">SetScrollPos</a>	Sets the position of the scroll box (thumb) in the specified scroll bar		
63	<a href="#">GetScrollPos</a>	Retrieves the current position of the scroll box (thumb) in the specified scroll bar		
64	<a href="#">SetScrollRange</a>	Sets the minimum and maximum scroll box positions for the specified scroll bar		
65	<a href="#">GetScrollRange</a>	Retrieves the current minimum and maximum scroll box positions for the specified scroll bar		
66	<a href="#">GetDC</a>	Retrieves a handle to a device context for the client area of the specified window	Done	
67	<a href="#">GetWindowDC</a>	Retrieves a handle to a device context for the entire window, including title bar, menus, and scroll bars	Done	
68	<a href="#">ReleaseDC</a>	Releases a device context, freeing it for use by other applications	Done	
69	<a href="#">SetCursor</a>	Sets the cursor shape	Done	
70	<a href="#">SetCursorPos</a>	Moves the cursor to the specified screen coordinates	Done	
71	<a href="#">ShowCursor</a>	Displays or hides the cursor	Done	
72	<a href="#">SetRect</a>	Sets the coordinates of the specified rectangle	Done	
73	<a href="#">SetRectEmpty</a>	Creates an empty rectangle	Done	
74	<a href="#">CopyRect</a>	Copies the coordinates of one rectangle to another	Done	
75	<a href="#">IsRectEmpty</a>	Determines whether the specified rectangle is empty	Done	
76	<a href="#">PtInRect</a>	Determines whether the specified point is inside the specified rectangle	Done	
77	<a href="#">OffsetRect</a>	Moves the specified rectangle by the specified offsets	Done	
78	<a href="#">InflateRect</a>	Increases or decreases the width and height of the specified rectangle	Done	

Ordinal	Name	Description	Status	Version
79	<a href="#">IntersectRect</a>	Calculates the intersection of two source rectangles and places the coordinates of the intersection rectangle into the destination rectangle	Done	
80	<a href="#">UnionRect</a>	Creates the union of two rectangles	Done	
81	<a href="#">FillRect</a>	Fills a rectangle by using the specified brush	Done	
82	<a href="#">InvertRect</a>	Inverts a rectangle in a window by performing a logical NOT operation on the color values for each pixel in the rectangle's interior	Done	
83	<a href="#">FrameRect</a>	Draws a border around the specified rectangle by using the specified brush	Done	
84	<a href="#">DrawIcon</a>	Draws an icon or cursor into the specified device context	Done	
85	<a href="#">DrawText</a>	Draws formatted text in the specified rectangle	Done	
86	<a href="#">BEAR86</a>			
87	<a href="#">DialogBox</a>	Creates a modal dialog box from a dialog box template resource	Done	
88	<a href="#">EndDialog</a>	Destroys a modal dialog box, causing the system to end any processing for the dialog box		
89	<a href="#">CreateDialog</a>	Creates a modeless dialog box from a dialog box template resource	Done	
90	<a href="#">IsDialogMessage</a>	Determines whether a message is intended for the specified dialog box and, if it is, processes the message		
91	<a href="#">GetDlgItem</a>	Retrieves the handle of a control in the specified dialog box		
92	<a href="#">SetDlgItemText</a>	Sets the title or text of a control in a dialog box	Done	
93	<a href="#">GetDlgItemText</a>	Retrieves the title or text associated with a control in a dialog box	Done	
94	<a href="#">SetDlgItemInt</a>	Sets the text of a control in a dialog box to the string representation of a specified integer value		
95	<a href="#">GetDlgItemInt</a>	Retrieves the text of the specified control by converting it to an integer value		
96	<a href="#">CheckRadioButton</a>	Adds a check mark to (checks) a specified radio button in a group and removes a check mark from (clears) all other radio buttons in the group		
97	<a href="#">CheckDlgButton</a>	Changes the check state of a button control	Done	
98	<a href="#">IsDlgButtonChecked</a>	Determines whether a button control is checked or not	Done	
99	<a href="#">DlgDirSelect</a>	Retrieves the current selection from a list box filled by the DlgDirList function	Done	
100	<a href="#">DlgDirList</a>	Fills a list box with a file or directory listing		

Ordinal	Name	Description	Status	Version
101	<a href="#">SendDlgItemMessage</a>	Sends a message to the specified control in a dialog box	Done	
102	<a href="#">AdjustWindowRect</a>	Calculates the required size of the window rectangle based on the desired client rectangle size		
103	<a href="#">MapDialogRect</a>	Converts the dialog-box coordinates of a specified rectangle to screen coordinates		
104	<a href="#">MessageBeep</a>	Plays a waveform sound corresponding to the specified alert type		
105	<a href="#">FlashWindow</a>	Flashes the specified window one time		
106	<a href="#">GetKeyState</a>	Retrieves the status of the specified virtual key		
107	<a href="#">DefWindowProc</a>	Calls the default window procedure to provide default processing for any window messages that an application does not process	Done	
108	<a href="#">GetMessage</a>	Retrieves a message from the calling thread's message queue	Done	
109	<a href="#">PeekMessage</a>	Checks the thread message queue for a message and returns immediately	Done	
110	<a href="#">PostMessage</a>	Places (posts) a message in the message queue associated with the thread that created the specified window and returns without waiting for the thread to process the message	Done	
111	<a href="#">SendMessage</a>	Sends the specified message to a window or windows and does not return until the window procedure has processed the message	Done	
112	<a href="#">WaitMessage</a>	Yields control to other threads when a thread has no other messages in its message queue		
113	<a href="#">TranslateMessage</a>	Translates virtual-key messages into character messages		
114	<a href="#">DispatchMessage</a>	Dispatches a message to a window procedure	Done	
115	<a href="#">ReplyMessage</a>	Used to reply to a message sent through the SendMessage function without returning control to the function that called SendMessage		
116	<a href="#">PostAppMessage</a>	Places a message in the message queue of all top-level windows and returns immediately		
118	<a href="#">RegisterWindowMessage</a>	Defines a new window message that is guaranteed to be unique throughout the system		
119	<a href="#">GetMessagePos</a>	Retrieves the cursor position for the last message retrieved by the GetMessage function		

Ordinal	Name	Description	Status	Version
120	<a href="#">GetMessageTime</a>	Retrieves the message time for the last message retrieved by the GetMessage function		
121	<a href="#">SetWindowsHook</a>	Installs an application-defined hook procedure into a hook chain	Done	
122	<a href="#">CallWindowProc</a>	Passes message information to the specified window procedure		
123	<a href="#">CallMsgFilter</a>	Passes a message and hook code to the hook procedures associated with the WH_SYSMSGFILTER and WH_MSGFILTER hooks		
124	<a href="#">UpdateWindow</a>	Updates the client area of the specified window by sending a WM_PAINT message to the window if the window's update region is not empty	Done	
125	<a href="#">InvalidateRect</a>	Adds a rectangle to the specified window's update region, invalidating the rectangle	Done	
126	<a href="#">InvalidateRgn</a>	Adds a region to the specified window's update region, invalidating the region	Done	
127	<a href="#">ValidateRect</a>	Removes a rectangle from the specified window's update region, validating the rectangle	Done	
128	<a href="#">ValidateRgn</a>	Removes a region from the specified window's update region, validating the region	Done	
129	<a href="#">GetClassWord</a>	Retrieves the address of the specified window class's menu name string		
130	<a href="#">SetClassWord</a>	Replaces the address of the menu name string for the specified window class		
131	<a href="#">GetClassLong</a>	Retrieves the specified 32-bit value from the window class structure		
132	<a href="#">SetClassLong</a>	Replaces the specified 32-bit value in the window class structure		
133	<a href="#">GetWindowWord</a>	Retrieves the specified 16-bit value from the window structure		
134	<a href="#">SetWindowWord</a>	Replaces the specified 16-bit value in the window structure		
135	<a href="#">GetWindowLong</a>	Retrieves the specified 32-bit value from the window structure		
136	<a href="#">SetWindowLong</a>	Replaces the specified 32-bit value in the window structure		
137	<a href="#">OpenClipboard</a>	Opens the clipboard for examination and prevents other applications from modifying the clipboard content		
138	<a href="#">CloseClipboard</a>	Closes the clipboard		
139	<a href="#">EmptyClipboard</a>	Empties the clipboard and frees handles to data in the clipboard		
140	<a href="#">GetClipboardOwner</a>	Retrieves the window handle of the current owner of the clipboard		

Ordinal	Name	Description	Status	Version
141	<a href="#">SetClipboardData</a>	Places data on the clipboard in a specified clipboard format		
142	<a href="#">GetClipboardData</a>	Retrieves data from the clipboard in a specified format		
143	<a href="#">CountClipboardFormats</a>	Retrieves the number of different data formats currently on the clipboard		
144	<a href="#">EnumClipboardFormats</a>	Enumerates the data formats currently available on the clipboard		
145	<a href="#">RegisterClipboardFormat</a>	Registers a new clipboard format		
146	<a href="#">GetClipboardFormatName</a>	Retrieves the name of the specified registered clipboard format		
147	<a href="#">SetClipboardViewer</a>	Adds the specified window to the chain of clipboard viewers		
148	<a href="#">GetClipboardViewer</a>	Retrieves the handle to the first window in the clipboard viewer chain		
149	<a href="#">ChangeClipboardChain</a>	Removes a window from the chain of clipboard viewers		
150	<a href="#">LoadMenu</a>	Loads the specified menu resource from the executable file associated with an application instance	Done	
151	<a href="#">CreateMenu</a>	Creates a menu	Done	
152	<a href="#">DestroyMenu</a>	Destroys the specified menu and frees any memory that the menu occupies	Done	
153	<a href="#">ChangeMenu</a>	Adds, deletes, or modifies menu items	Done	
154	<a href="#">CheckMenuItem</a>	Adds or removes a check mark from a menu item	Done	
155	<a href="#">EnableMenuItem</a>	Enables, disables, or grays a menu item	Done	
156	<a href="#">GetSystemMenu</a>	Allows the application to access the window menu (system menu) for copying and modification	Done	
157	<a href="#">GetMenu</a>	Retrieves the handle of the menu assigned to the specified window	Done	
158	<a href="#">SetMenu</a>	Assigns a new menu to the specified window	Done	
159	<a href="#">GetSubMenu</a>	Retrieves the handle of a submenu in the specified menu	Done	
160	<a href="#">DrawMenuBar</a>	Redraws the menu bar of the specified window	Done	
161	<a href="#">GetMenuString</a>	Copies the text string of the specified menu item into a buffer	Done	
162	<a href="#">HiliteMenuItem</a>	Highlights or removes the highlighting from a menu item	Done	
163	<a href="#">CreateCaret</a>	Creates a new shape for the system caret and assigns ownership of the caret to the specified window	Done	
164	<a href="#">DestroyCaret</a>	Destroys the current shape of the caret and frees the caret from the window	Done	
165	<a href="#">SetCaretPos</a>	Moves the caret to the specified coordinates	Done	

Ordinal	Name	Description	Status	Version
166	<a href="#">HideCaret</a>	Removes the caret from the screen	Done	
167	<a href="#">ShowCaret</a>	Makes the caret visible on the screen at the caret's current position	Done	
168	<a href="#">SetCaretBlinkTime</a>	Sets the caret blink time	Done	
169	<a href="#">GetCaretBlinkTime</a>	Retrieves the caret blink time	Done	
170	<a href="#">ArrangelconicWindows</a>	Arranges all the minimized (iconic) child windows of the specified parent window		
171	<a href="#">WinHelp</a>	Starts Windows Help		
173	<a href="#">LoadCursor</a>	Loads the specified cursor resource from the executable file associated with an application instance	Done	
174	<a href="#">LoadIcon</a>	Loads the specified icon resource from the executable file associated with an application instance	Done	
175	<a href="#">LoadBitmap</a>	Loads the specified bitmap resource from the executable file associated with an application instance	Done	
176	<a href="#">LoadString</a>	Loads a string resource from the executable file associated with an application instance	Done	
177	<a href="#">LoadAccelerators</a>	Loads the specified accelerator table	Done	
178	<a href="#">TranslateAccelerator</a>	Processes accelerator keys for menu commands		
179	<a href="#">GetSystemMetrics</a>	Retrieves the system metrics for the current display	Done	
180	<a href="#">GetSysColor</a>	Retrieves the current color of the specified display element	Done	
181	<a href="#">SetSysColors</a>	Sets the colors for one or more display elements	Done	
182	<a href="#">BEAR182</a>			
183	<a href="#">GetCaretPos</a>	Retrieves the current position of the caret	Done	
184	<a href="#">QuerySendMessage</a>	Determines whether the message was sent from another task using the SendMessage function		
185	<a href="#">GrayString</a>	Draws gray text at the specified location	Done	
186	<a href="#">SwapMouseButton</a>	Reverses or restores the meaning of the left and right mouse buttons		
188	<a href="#">SetSysModalWindow</a>	Causes the system to direct all user input to the specified window, regardless of which application the user is interacting with		
189	<a href="#">GetSysModalWindow</a>	Retrieves the system-modal window, if one exists		
190	<a href="#">GetUpdateRect</a>	Retrieves the coordinates of the rectangle that completely encloses the update region of the specified window		
191	<a href="#">ChildWindowFromPoint</a>	Determines which, if any, of the child windows belonging to the specified parent window contains the specified point		

Ordinal	Name	Description	Status	Version
192	<a href="#">InSendMessage</a>	Determines whether the current window procedure is processing a message sent from another task by the SendMessage function		
193	<a href="#">IsClipboardFormatAvailable</a>	Determines whether the clipboard contains data in the specified format		
194	<a href="#">DlgDirSelectComboBox</a>	Retrieves the current selection from a combo box filled by the DlgDirListComboBox function	Done	
195	<a href="#">DlgDirListComboBox</a>	Fills a combo box with a file or directory listing		
196	<a href="#">TabbedTextOut</a>	Writes a character string at a specified location, expanding tabs to the values specified in an array of tab-stop positions		
197	<a href="#">GetTabbedTextExtent</a>	Computes the width and height of a character string, which may include tab characters		
198	<a href="#">CascadeChildWindows</a>	Cascades the specified child windows of the specified parent window		
199	<a href="#">TileChildWindows</a>	Tiles the specified child windows of the specified parent window		
200	<a href="#">OpenComm</a>	Opens a communications device	Stub	
201	<a href="#">SetCommState</a>	Configures a communications device according to the specifications in a device-control block	Stub	
202	<a href="#">GetCommState</a>	Retrieves the current control settings for a communications device	Stub	
203	<a href="#">GetCommError</a>	Retrieves information about the most recent communications error for a communications device	Stub	
204	<a href="#">ReadComm</a>	Reads data from a communications device	Stub	
205	<a href="#">WriteComm</a>	Writes data to a communications device	Stub	
206	<a href="#">TransmitCommChar</a>	Transmits a character ahead of any pending data in the output buffer of a communications device	Stub	
207	<a href="#">CloseComm</a>	Closes a communications device	Stub	
208	<a href="#">SetCommEventMask</a>	Specifies a set of events to be monitored for a communications device	Stub	
209	<a href="#">GetCommEventMask</a>	Retrieves the event mask for a communications device	Stub	
210	<a href="#">SetCommBreak</a>	Suspends character transmission for a communications device and places the transmission line in a break state	Stub	
211	<a href="#">ClearCommBreak</a>	Restores character transmission for a communications device and places the transmission line in a nonbreak state	Stub	
212	<a href="#">UngetCommChar</a>	Places a character back into the input buffer of a communications device	Stub	
213	<a href="#">BuildCommDCB</a>	Fills a device-control block with values specified in a string	Stub	

Ordinal	Name	Description	Status	Version
214	<a href="#">EscapeCommFunction</a>	Directs a communications device to perform an extended function	Stub	
215	<a href="#">FlushComm</a>	Flushes the input or output buffer of a communications device	Stub	
216	<a href="#">UserSeeUserDo</a>		Done	
218	<a href="#">DialogBoxIndirect</a>	Creates a modal dialog box from a dialog box template in memory	Done	
219	<a href="#">CreateDialogIndirect</a>	Creates a modeless dialog box from a dialog box template in memory	Done	
220	<a href="#">LoadMenuIndirect</a>	Loads the specified menu template from memory	Done	
221	<a href="#">ScrollDC</a>	Scrolls a rectangle of bits horizontally and vertically		
222	<a href="#">GetKeyboardState</a>	Retrieves the status of each virtual key on the keyboard		
223	<a href="#">SetKeyboardState</a>	Sets the status of each virtual key on the keyboard		
224	<a href="#">GetWindowTask</a>	Retrieves the handle of the task that owns the specified window		
225	<a href="#">EnumTaskWindows</a>	Enumerates all windows associated with a task		
226	<a href="#">LockInput</a>	Locks or unlocks input to the specified task		
227	<a href="#">GetNextDlgGroupItem</a>	Retrieves the handle of the first control in a group of controls that precedes or follows the specified control		
228	<a href="#">GetNextDlgTabItem</a>	Retrieves the handle of the first control that has the WS_TABSTOP style that precedes or follows the specified control		
229	<a href="#">GetTopWindow</a>	Examines the Z order of the child windows associated with the specified parent window and retrieves the handle of the child window at the top of the Z order		
230	<a href="#">GetNextWindow</a>	Retrieves the handle of the next or previous window in the Z order		
231	<a href="#">GetSystemDebugState</a>	Retrieves information about the current state of the system for debugging purposes		
232	<a href="#">SetWindowPos</a>	Changes the size, position, and Z order of a child, pop-up, or top-level window		
233	<a href="#">SetParent</a>	Changes the parent window of the specified child window		
234	<a href="#">UnhookWindowsHook</a>	Removes a hook procedure installed in a hook chain by the SetWindowsHook function		
235	<a href="#">DefHookProc</a>	Passes a hook message to the next hook procedure in the current hook chain	Done	
236	<a href="#">GetCapture</a>	Retrieves the handle of the window that has captured the mouse	Done	
237	<a href="#">GetUpdateRgn</a>	Retrieves the update region of a window by copying it into the specified region		

Ordinal	Name	Description	Status	Version
238	<a href="#">ExcludeUpdateRgn</a>	Prevents drawing within the update region of the specified window by excluding the update region from the specified device context's clipping region		
239	<a href="#">DialogBoxParam</a>	Creates a modal dialog box from a dialog box template resource, passing application-defined data to the dialog box procedure		
240	<a href="#">DialogBoxIndirectParam</a>	Creates a modal dialog box from a dialog box template in memory, passing application-defined data to the dialog box procedure		
241	<a href="#">CreateDialogParam</a>	Creates a modeless dialog box from a dialog box template resource, passing application-defined data to the dialog box procedure	Done	
242	<a href="#">CreateDialogIndirectParam</a>	Creates a modeless dialog box from a dialog box template in memory, passing application-defined data to the dialog box procedure		
243	<a href="#">GetDialogBaseUnits</a>	Retrieves the dialog base units used to create the dialog box		
244	<a href="#">EqualRect</a>	Determines whether the two specified rectangles are equal by comparing their coordinates	Done	
245	<a href="#">EnableCommNotification</a>	Enables or disables notification for a communications device	Stub	
246	<a href="#">ExitWindowsExec</a>	Executes the specified application when Windows exits		
247	<a href="#">GetCursor</a>	Retrieves the handle of the current cursor		
248	<a href="#">GetOpenClipboardWindow</a>	Retrieves the handle of the window that currently has the clipboard open		
249	<a href="#">GetAsyncKeyState</a>	Determines whether a key is up or down at the time the function is called, and whether the key was pressed after a previous call to <a href="#">GetAsyncKeyState</a>		
250	<a href="#">GetMenuState</a>	Retrieves the menu item state	Done	
251	<a href="#">SendDriverMessage</a>	Sends a message to an installable driver		
252	<a href="#">OpenDriver</a>	Opens an installable driver		
253	<a href="#">CloseDriver</a>	Closes an installable driver		
254	<a href="#">GetDriverModuleHandle</a>	Retrieves the module handle of an installable driver		
255	<a href="#">DefDriverProc</a>	Provides default processing for any messages that an installable driver does not process		
256	<a href="#">GetDriverInfo</a>	Retrieves information about an installable driver		
257	<a href="#">GetNextDriver</a>	Retrieves the handle of the next installable driver in the list of installable drivers		

Ordinal	Name	Description	Status	Version
258	<a href="#">MapWindowPoints</a>	Converts (maps) a set of points from a coordinate space relative to one window to a coordinate space relative to another window		
259	<a href="#">BeginDeferWindowPos</a>	Allocates memory for a multiple-window-position structure and returns the handle to the structure		
260	<a href="#">DeferWindowPos</a>	Updates the specified multiple-window-position structure for the specified window		
261	<a href="#">EndDeferWindowPos</a>	Simultaneously updates the position and size of one or more windows in a single screen-refreshing cycle		
262	<a href="#">GetWindow</a>	Retrieves a handle to a window that has the specified relationship to the specified window		
263	<a href="#">GetMenuItemCount</a>	Retrieves the number of items in the specified menu	Done	
264	<a href="#">GetMenuItemID</a>	Retrieves the menu item identifier of a menu item located at the specified position in a menu	Done	
265	<a href="#">ShowOwnedPopups</a>	Shows or hides all pop-up windows owned by the specified window		
266	<a href="#">SetMessageQueue</a>	Creates a new message queue for the calling application	Done	
267	<a href="#">ShowScrollBar</a>	Shows or hides the specified scroll bar		
268	<a href="#">GlobalAddAtom</a>	Adds a character string to the global atom table and returns a unique value (an atom) identifying the string	Done	
269	<a href="#">GlobalDeleteAtom</a>	Decrements the reference count of a global string atom, and if the reference count reaches zero, removes the string from the global atom table	Done	
270	<a href="#">GlobalFindAtom</a>	Retrieves the atom associated with the specified character string from the global atom table	Done	
271	<a href="#">GlobalGetAtomName</a>	Retrieves a copy of the character string associated with the specified global atom	Done	
272	<a href="#">IsZoomed</a>	Determines whether a window is maximized		
277	<a href="#">GetDlgCtrlID</a>	Retrieves the identifier of the specified control		
278	<a href="#">GetDesktopHwnd</a>	Retrieves the handle of the desktop window	Done	
279	<a href="#">OldSetDeskPattern</a>	Sets the desktop pattern	Done	
282	<a href="#">SelectPalette</a>	Selects a logical palette into a device context		
283	<a href="#">RealizePalette</a>	Maps palette entries from the current logical palette to the system palette		
284	<a href="#">GetFreeSystemResources</a>	Retrieves the percentage of free system resources	Done	

Ordinal	Name	Description	Status	Version
286	<a href="#">GetDesktopWindow</a>	Retrieves the handle of the desktop window	Done	
287	<a href="#">GetLastActivePopup</a>	Determines which pop-up window owned by the specified window was most recently active		
288	<a href="#">GetMessageExtraInfo</a>	Retrieves the extra message information for the current thread		
290	<a href="#">RedrawWindow</a>	Updates the specified rectangle or region in a window's client area	Done	
291	<a href="#">SetWindowsHookEx</a>	Installs an application-defined hook procedure into a hook chain		
292	<a href="#">UnhookWindowsHookEx</a>	Removes a hook procedure installed in a hook chain by the SetWindowsHookEx function		
293	<a href="#">CallNextHookEx</a>	Passes the hook information to the next hook procedure in the current hook chain		
294	<a href="#">LockWindowUpdate</a>	Disables or enables drawing in the specified window		
299	<a href="#">mouse_event</a>	Synthesizes mouse motion and button clicks	Done	
308	<a href="#">DefDlgProc</a>	Provides default processing for any messages that a dialog box with a private window class does not process	Done	
309	<a href="#">GetClipCursor</a>	Retrieves the cursor confinement rectangle		
319	<a href="#">ScrollWindowEx</a>	Scrolls the content of the specified window's client area		
324	<a href="#">FillWindow</a>	Fills the client area of the specified window with the specified brush	Done	
325	<a href="#">PaintRect</a>	Paints the specified rectangle by using the specified brush	Done	
326	<a href="#">GetControlBrush</a>	Retrieves the brush for the specified control	Done	
331	<a href="#">EnableHardwareInput</a>	Enables or disables mouse and keyboard input to the system		
333	<a href="#">IsUserIdle</a>	Determines whether the system is idle	Done	
334	<a href="#">GetQueueStatus</a>	Retrieves the type of messages in the calling thread's message queue		
335	<a href="#">GetInputState</a>	Determines whether there are mouse-button, keyboard, or timer events in the message queue		
337	<a href="#">GetMouseEventProc</a>	Retrieves the address of the current mouse event procedure	Done	
358	<a href="#">IsMenu</a>	Determines whether a handle is a menu handle		
359	<a href="#">GetDCEx</a>	Retrieves a handle to a device context for the client area of the specified window	Done	
368	<a href="#">CopyIcon</a>	Copies the specified icon from another module to the current module	Done	
369	<a href="#">CopyCursor</a>	Copies the specified cursor from another module to the current module	Done	

Ordinal	Name	Description	Status	Version
370	<a href="#">GetWindowPlacement</a>	Retrieves the show state and the restored, minimized, and maximized positions of the specified window		
371	<a href="#">SetWindowPlacement</a>	Sets the show state and the restored, minimized, and maximized positions of the specified window		
373	<a href="#">SubtractRect</a>	Obtains the coordinates of a rectangle determined by subtracting one rectangle from another	Done	
397	<a href="#">RegisterClassEx</a>	Registers a window class for subsequent use in the CreateWindow or CreateWindowEx function	Done	
398	<a href="#">GetClassInfoEx</a>	Retrieves information about a window class	Done	
402	<a href="#">GetPriorityClipboardFormat</a>	Retrieves the first available clipboard format in the specified list		
403	<a href="#">UnregisterClass</a>	Unregisters a window class, freeing the memory required for the class	Done	
404	<a href="#">GetClassInfo</a>	Retrieves information about a window class	Done	
406	<a href="#">CreateCursor</a>	Creates a cursor having the specified size, bit patterns, and hot spot	Done	
407	<a href="#">CreateIcon</a>	Creates an icon having the specified size, colors, and bit patterns	Done	
408	<a href="#">CreateCursorIconIndirect</a>	Creates an icon or cursor from an icon or cursor resource data structure	Done	
410	<a href="#">InsertMenu</a>	Inserts a new menu item into a menu, moving other items down	Done	
411	<a href="#">AppendMenu</a>	Appends a new item to the end of the specified menu	Done	
412	<a href="#">RemoveMenu</a>	Deletes a menu item or detaches a submenu from the specified menu	Done	
413	<a href="#">DeleteMenu</a>	Deletes an item from the specified menu	Done	
414	<a href="#">ModifyMenu</a>	Changes an existing menu item	Done	
415	<a href="#">CreatePopupMenu</a>	Creates a drop-down menu, submenu, or shortcut menu	Done	
416	<a href="#">TrackPopupMenu</a>	Displays a shortcut menu at the specified location and tracks the selection of items on the shortcut menu	Done	
417	<a href="#">GetMenuCheckMarkDimensions</a>	Retrieves the dimensions of the default check mark bitmap	Done	
418	<a href="#">SetMenuItemBitmaps</a>	Associates the specified check mark bitmaps with a menu item	Done	
420	<a href="#">_wsprintf</a>	Writes formatted data to a string	Done	
421	<a href="#">wvsprintf</a>	Writes formatted data to a string using a variable argument list	Done	
422	<a href="#">DlgDirSelectEx</a>	Retrieves the current selection from a single-selection or multiple-selection list box filled by the DlgDirList function		
423	<a href="#">DlgDirSelectComboBoxEx</a>	Retrieves the current selection from a combo box filled by the DlgDirListComboBox function		

Ordinal	Name	Description	Status	Version
430	<a href="#">lstrcmp</a>	Compares two character strings	Done	
431	<a href="#">AnsiUpper</a>	Converts a character string or a single character to uppercase	Done	
432	<a href="#">AnsiLower</a>	Converts a character string or a single character to lowercase	Done	
433	<a href="#">IsCharAlpha</a>	Determines whether a character is an alphabetic character	Done	
434	<a href="#">IsCharAlphaNumeric</a>	Determines whether a character is either an alphabetic or a numeric character	Done	
435	<a href="#">IsCharUpper</a>	Determines whether a character is uppercase	Done	
436	<a href="#">IsCharLower</a>	Determines whether a character is lowercase	Done	
437	<a href="#">AnsiUpperBuff</a>	Converts a character string to uppercase	Done	
438	<a href="#">AnsiLowerBuff</a>	Converts a character string to lowercase	Done	
445	<a href="#">DefFrameProc</a>	Provides default processing for any window messages that a multiple-document interface (MDI) frame window does not process		
447	<a href="#">DefMDIChildProc</a>	Provides default processing for any window messages that an MDI child window does not process		
451	<a href="#">TranslateMDISysAccel</a>	Processes accelerator keystrokes for system menu commands of multiple-document interface (MDI) child windows		
452	<a href="#">CreateWindowEx</a>	Creates an overlapped, pop-up, or child window with an extended window style	Done	
454	<a href="#">AdjustWindowRectEx</a>	Calculates the required size of the window rectangle based on the desired client rectangle size and window styles		
457	<a href="#">DestroyIcon</a>	Destroys an icon and frees any memory the icon occupied	Done	
458	<a href="#">DestroyCursor</a>	Destroys a cursor and frees any memory the cursor occupied	Done	
462	<a href="#">CalcChildScroll</a>	Calculates the scrolling rectangle for a scroll bar control		
466	<a href="#">DrawFocusRect</a>	Draws a rectangle in the style used to indicate that the rectangle has the focus	Done	
471	<a href="#">lstrcmpi</a>	Compares two character strings, ignoring case	Done	
472	<a href="#">AnsiNext</a>	Returns a pointer to the next character in a string	Done	
473	<a href="#">AnsiPrev</a>	Returns a pointer to the previous character in a string	Done	
482	<a href="#">EnableScrollBar</a>	Enables or disables one or both arrows of a scroll bar		
483	<a href="#">SystemParametersInfo</a>	Retrieves or sets the value of one of the system-wide parameters	Done	
499	<a href="#">WNetErrorText</a>	Retrieves the error string for a network error	Stub	

Ordinal	Name	Description	Status	Version
501	<a href="#">WNetOpenJob</a>	Opens a print job on a network printer	Stub	
502	<a href="#">WNetCloseJob</a>	Closes a print job on a network printer	Stub	
503	<a href="#">WNetAbortJob</a>	Aborts a print job on a network printer	Stub	
504	<a href="#">WNetHoldJob</a>	Holds a print job on a network printer	Stub	
505	<a href="#">WNetReleaseJob</a>	Releases a held print job on a network printer	Stub	
506	<a href="#">WNetCancelJob</a>	Cancels a print job on a network printer	Stub	
507	<a href="#">WNetSetJobCopies</a>	Sets the number of copies for a print job on a network printer	Stub	
508	<a href="#">WNetWatchQueue</a>	Installs a queue watch for a network print queue	Stub	
509	<a href="#">WNetUnwatchQueue</a>	Removes a queue watch for a network print queue	Stub	
510	<a href="#">WNetLockQueueData</a>	Locks the queue data for a network print queue	Stub	
511	<a href="#">WNetUnlockQueueData</a>	Unlocks the queue data for a network print queue	Stub	
512	<a href="#">WNetGetConnection</a>	Retrieves the network resource name associated with a local device		
513	<a href="#">WNetGetCaps</a>	Retrieves the capabilities of the network provider	Done	
514	<a href="#">WNetDeviceMode</a>	Displays the configuration dialog box for a network device	Stub	
515	<a href="#">WNetBrowseDialog</a>	Displays a network browse dialog box	Stub	
516	<a href="#">WNetGetUser</a>	Retrieves the current network user name		
517	<a href="#">WNetAddConnection</a>	Connects a local device to a network resource	Stub	
518	<a href="#">WNetCancelConnection</a>	Disconnects a network connection	Stub	
519	<a href="#">WNetGetError</a>	Retrieves the most recent network error	Stub	
520	<a href="#">WNetGetErrorText</a>	Retrieves the error text for a network error	Stub	
521	<a href="#">WNetEnable</a>	Enables or disables the network provider		
522	<a href="#">WNetDisable</a>	Disables the network provider		
523	<a href="#">WNetRestoreConnection</a>	Restores a network connection	Stub	
524	<a href="#">WNetWriteJob</a>	Writes data to a network print job	Stub	
525	<a href="#">WNetConnectDialog</a>	Displays a network connection dialog box	Stub	
526	<a href="#">WNetDisconnectDialog</a>	Displays a network disconnection dialog box	Stub	
527	<a href="#">WNetConnectionDialog</a>	Displays a network connection dialog box	Stub	
528	<a href="#">WNetViewQueueDialog</a>	Displays a network queue view dialog box	Stub	
529	<a href="#">WNetPropertyDialog</a>	Displays a network property dialog box	Stub	
530	<a href="#">WNetGetDirectoryType</a>	Determines the type of a network directory		
531	<a href="#">WNetDirectoryNotify</a>	Notifies the network of a directory change	Stub	
532	<a href="#">WNetGetPropertyText</a>	Retrieves the property text for a network resource	Stub	
<b>Group</b>		<b>Functions</b>		

Group	Functions
<b>Module Management</b>	GetVersion GetModuleHandle GetModuleUsage GetModuleFileName GetProcAddress MakeProInstance FreeProInstance GetInstanceData Catch Throw GetCodeHandle LoadLibrary FreeLibrary
<b>Global Memory Management</b>	GlobalAlloc GlobalCompact GlobalDiscard GlobalFree GlobalLock GlobalReAlloc GlobalSize GlobalUnlock GlobalFlags
<b>Local Memory Management</b>	LocalInit LocalAlloc LocalCompact LocalDiscard LocalFree LocalLock LocalFreeze LocalMelt LocalReAlloc LocalSize LocalUnlock LocalHandleDelta LockData UnlockData LocalFlags
<b>Task Scheduling</b>	GetCurrentTask Yield SetPriority
<b>Resource Management</b>	AddFontResource RemoveFontResource LoadBitmap LoadCursor LoadIcon LoadMenu LoadString LoadAccelerators FindResource LoadResource AllocResource LockResource FreeResource AccessResource SizeofResource SetResourceHandler
<b>String Manipulation</b>	AnsiUpper AnsiLower AnsiNext AnsiPrev AnsiUpperBuff AnsiLowerBuff lstricmp lstrcmpi IsCharAlpha IsCharAlphaNumeric IsCharUpper IsCharLower _wsprintf wvsprintf
<b>Atom Management</b>	InitAtomTable AddAtom DeleteAtom FindAtom GetAtomName GlobalAddAtom GlobalDeleteAtom GlobalFindAtom GlobalGetAtomName
<b>Initialization Files</b>	GetProfileInt GetProfileString WriteProfileString
<b>Debugging</b>	FatalExit
<b>File Input/Output</b>	_lopen _lcreat _lseek _lread _lwrite _lclose OpenFile GetTempFileName GetTempDrive
<b>Registry</b>	RegOpenKey RegCreateKey RegCloseKey RegDeleteKey RegSetValue RegQueryValue RegEnumKey
<b>Window Management</b>	CreateWindow CreateWindowEx DestroyWindow ShowWindow CloseWindow OpenIcon MoveWindow SetWindowPos BringWindowToTop GetWindow GetParent SetParent GetNextWindow GetTopWindow GetWindowRect GetClientRect AdjustWindowRect AdjustWindowRectEx EnableWindow IsWindowEnabled IsWindowVisible IsIconic IsZoomed IsWindow IsChild FindWindow EnumWindows EnumChildWindows EnumTaskWindows GetDesktopWindow GetDesktopHwnd WindowFromPoint ChildWindowFromPoint GetWindowText SetWindowText GetWindowTextLength GetClassName GetWindowLong SetWindowLong GetWindowWord SetWindowWord GetWindowTask GetWindowPlacement SetWindowPlacement ArrangeIconicWindows CascadeChildWindows TileChildWindows ShowOwnedPopups AnyPopup GetLastActivePopup SetActiveWindow GetActiveWindow SetFocus GetFocus SetCapture GetCapture ReleaseCapture SetSysModalWindow GetSysModalWindow LockWindowUpdate RedrawWindow UpdateWindow InvalidateRect InvalidateRgn ValidateRect ValidateRgn GetUpdateRect GetUpdateRgn ExcludeUpdateRgn BeginDeferWindowPos DeferWindowPos EndDeferWindowPos FlashWindow
<b>Message Handling</b>	GetMessage PeekMessage PostMessage PostAppMessage SendMessage ReplyMessage InSendMessage WaitMessage TranslateMessage DispatchMessage GetMessagePos GetMessageTime GetMessageExtraInfo PostQuitMessage RegisterWindowMessage SetMessageQueue GetQueueStatus GetInputState QuerySendMessage
<b>Painting and Drawing</b>	BeginPaint EndPaint GetDC GetDCEX GetWindowDC ReleaseDC DrawText GrayString TabbedTextOut GetTabbedTextExtent FillRect InvertRect FrameRect DrawFocusRect DrawIcon ScrollDC ScrollWindow ScrollWindowEx FillWindow PaintRect GetControlBrush

Group	Functions
<b>Menus</b>	CreateMenu CreatePopupMenu DestroyMenu GetMenu SetMenu GetSystemMenu GetSubMenu AppendMenu InsertMenu ModifyMenu DeleteMenu RemoveMenu ChangeMenu CheckMenuItem EnableMenuItem HiliteMenuItem GetMenuState GetMenuString GetMenuItemCount GetMenuItemID DrawMenuBar TrackPopupMenu GetMenuCheckMarkDimensions SetMenuItemBitmaps IsMenu
<b>Clipboard</b>	OpenClipboard CloseClipboard EmptyClipboard GetClipboardOwner GetOpenClipboardWindow SetClipboardData GetClipboardData CountClipboardFormats EnumClipboardFormats RegisterClipboardFormat GetClipboardFormatName IsClipboardFormatAvailable GetPriorityClipboardFormat SetClipboardViewer GetClipboardViewer ChangeClipboardChain
<b>Caret</b>	CreateCaret DestroyCaret SetCaretPos GetCaretPos HideCaret ShowCaret SetCaretBlinkTime GetCaretBlinkTime
<b>Cursor and Icon</b>	SetCursor GetCursor SetCursorPos GetCursorPos ShowCursor ClipCursor GetClipCursor LoadCursor LoadIcon CreateCursor CreateIcon CopyCursor CopyIcon DestroyCursor DestroyIcon CreateCursorIconIndirect
<b>Scroll Bars</b>	SetScrollPos GetScrollPos SetScrollRange GetScrollRange ShowScrollBar EnableScrollBar CalcChildScroll
<b>Dialog Boxes</b>	DialogBox DialogBoxIndirect DialogBoxParam DialogBoxIndirectParam CreateDialog CreateDialogIndirect CreateDialogParam CreateDialogIndirectParam EndDialog GetDlgItem SetDlgItemText GetDlgItemText SetDlgItemInt GetDlgItemInt CheckDlgButton IsDlgButtonChecked CheckRadioButton SendDlgItemMessage GetNextDlgGroupItem GetNextDlgTabItem GetDlgCtrlID IsDialogMessage MapDialogRect GetDialogBaseUnits DefDlgProc DlgDirList DlgDirListComboBox DlgDirSelect DlgDirSelectEx DlgDirSelectComboBox DlgDirSelectComboBoxEx
<b>Device Context Management</b>	GetDC GetDCEx GetWindowDC ReleaseDC SaveDC RestoreDC CreateDC CreateCompatibleDC DeleteDC ResetDC
<b>Coordinates and Transformations</b>	SetMapMode GetMapMode SetWindowOrg GetWindowOrg SetWindowExt GetWindowExt SetViewportOrg GetViewportOrg SetViewportExt GetViewportExt OffsetWindowOrg OffsetViewportOrg ScaleWindowExt ScaleViewportExt DPToLP LPtoDP
<b>Drawing Functions</b>	SetPixel GetPixel MoveTo LineTo Polyline Polygon PolyPolygon Rectangle RoundRect Ellipse Arc Chord Pie FloodFill ExtFloodFill DrawFocusRect PatBlt BitBlt StretchBlt StretchDIBits SetDIBits GetDIBits SetDIBitsToDevice
<b>Regions</b>	CreateRectRgn CreateRectRgnIndirect CreateRoundRectRgn CreateEllipticRgn CreateEllipticRgnIndirect CreatePolygonRgn CreatePolyPolygonRgn CombineRgn EqualRgn OffsetRgn FillRgn FrameRgn InvertRgn PaintRgn SelectClipRgn ExcludeClipRect IntersectClipRect OffsetClipRgn GetClipBox GetRgnBox PtInRegion RectInRegion SetRectRgn
<b>Bitmaps</b>	CreateBitmap CreateBitmapIndirect CreateCompatibleBitmap CreateDiscardableBitmap CreateDIBitmap CreateDIBPatternBrush SetBitmapBits GetBitmapBits SetBitmapDimension GetBitmapDimension
<b>Brushes and Pens</b>	CreateSolidBrush CreateHatchBrush CreatePatternBrush CreateBrushIndirect CreatePen CreatePenIndirect SetBrushOrg GetBrushOrg SelectObject GetObject DeleteObject GetStockObject UnrealizeObject

Group	Functions
<b>Fonts and Text</b>	CreateFont CreateFontIndirect CreateScalableFontResource AddFontResource RemoveFontResource EnumFonts EnumFontFamilies GetTextMetrics GetTextExtent GetTextFace GetCharWidth GetCharABCWidths GetOutlineTextMetrics GetGlyphOutline GetKerningPairs SetTextColor GetTextColor SetBkColor GetBkColor SetBkMode GetBkMode SetTextAlign GetTextAlign SetTextCharacterExtra GetTextCharacterExtra SetTextJustification TextOut ExtTextOut TabbedTextOut GetTabbedTextExtent
<b>Metafiles</b>	CreateMetaFile CloseMetaFile DeleteMetaFile CopyMetaFile GetMetaFile GetMetaFileBits SetMetaFileBits PlayMetaFile PlayMetaFileRecord EnumMetaFile IsValidMetaFile
<b>Printing</b>	StartDoc EndDoc StartPage EndPage SetAbortProc AbortDoc QueryAbort OpenJob CloseJob DeleteJob WriteSpool StartSpoolPage EndSpoolPage SpoolFile
<b>Palettes</b>	CreatePalette SelectPalette RealizePalette GetPaletteEntries SetPaletteEntries ResizePalette AnimatePalette UpdateColors GetNearestPaletteIndex GetNearestColor SetSystemPaletteUse GetSystemPaletteUse GetSystemPaletteEntries
<b>Device Capabilities</b>	GetDeviceCaps GetRasterizerCaps GetEnvironment SetEnvironment GetAspectRatioFilter Escape CreateIC
<b>Miscellaneous GDI</b>	SetROP2 GetROP2 SetPolyFillMode GetPolyFillMode SetStretchBltMode GetStretchBltMode SetMapperFlags GetMapperFlags LineDDA MulDiv GetCurrentPosition GetBoundsRect SetBoundsRect FastWindowFrame
<b>Timer</b>	SetTimer KillTimer GetTickCount GetTimerResolution GetCurrentTime
<b>Hooks</b>	SetWindowsHook SetWindowsHookEx UnhookWindowsHook UnhookWindowsHookEx CallNextHookEx DefHookProc CallMsgFilter
<b>Communications</b>	OpenComm CloseComm ReadComm WriteComm SetCommState GetCommState GetCommError TransmitCommChar SetCommEventMask GetCommEventMask SetCommBreak ClearCommBreak UngetCommChar BuildCommDCB EscapeCommFunction FlushComm EnableCommNotification
<b>Networking (WNet)</b>	WNetErrorText WNetOpenJob WNetCloseJob WNetAbortJob WNetHoldJob WNetReleaseJob WNetCancelJob WNetSetJobCopies WNetWatchQueue WNetUnwatchQueue WNetLockQueueData WNetUnlockQueueData WNetGetConnection WNetGetCaps WNetDeviceMode WNetBrowseDialog WNetGetUser WNetAddConnection WNetCancelConnection WNetGetError WNetGetErrorText WNetEnable WNetDisable WNetRestoreConnection WNetWriteJob WNetConnectDialog WNetDisconnectDialog WNetConnectionDialog WNetViewQueueDialog WNetPropertyDialog WNetGetDirectoryType WNetDirectoryNotify WNetGetPropertyText
<b>System Information</b>	GetSystemMetrics GetSysColor SetSysColors SystemParametersInfo GetFreeSystemResources GetSystemDebugState GetDoubleClickTime SetDoubleClickTime SwapMouseButton GetKeyState GetAsyncKeyState GetKeyboardState SetKeyboardState mouse_event GetMouseEventProc EnableHardwareInput IsUserIdle LockInput MessageBeep WinHelp
<b>Window Classes</b>	RegisterClass RegisterClassEx UnregisterClass GetClassInfo GetClassInfoEx GetClassWord SetClassWord GetClassLong SetClassLong GetClassName
<b>Window Properties</b>	SetProp GetProp RemoveProp EnumProps
<b>MDI Support</b>	DefFrameProc DefMDIChildProc TranslateMDISysAccel
<b>Drivers</b>	OpenDriver CloseDriver SendDriverMessage GetDriverModuleHandle GetDriverInfo GetNextDriver DefDriverProc

Group	Functions
<b>Miscellaneous</b>	<a href="#">InitApp</a> <a href="#">ExitWindows</a> <a href="#">ExitWindowsExec</a> <a href="#">OLDEXITWINDOWS</a> <a href="#">BEAR11</a> <a href="#">BEAR86</a> <a href="#">BEAR182</a> <a href="#">UserSeeUserDo</a> <a href="#">OldSetDeskPattern</a> <a href="#">SetRect</a> <a href="#">SetRectEmpty</a> <a href="#">CopyRect</a> <a href="#">IsRectEmpty</a> <a href="#">PtInRect</a> <a href="#">OffsetRect</a> <a href="#">InflateRect</a> <a href="#">IntersectRect</a> <a href="#">UnionRect</a> <a href="#">EqualRect</a> <a href="#">SubtractRect</a>

2022/11/17 15:22 · [prokushev](#) · [0 Comments](#)

From:  
<http://ftp.osfree.org/doku/> - **osFree wiki**

Permanent link:  
<http://ftp.osfree.org/doku/doku.php?id=en:docs:win16:modules:user>

Last update: **2026/03/30 05:38**

