



2x

DEVELOP 10 TIMES FASTER

# TABLE OF CONTENTS

## Overview

1. Window and page management functions

1.1 Window management functions

1.2 Page management functions

1.3 Additional functions available

1.3.1 MDI functions

1.3.2 Menu management functions

1.3.3 Dialog box management functions

1.3.4 Toast management functions

2. Functions for managing different types of controls

2.1 Usual controls

2.2 Controls that handle data

2.3 Container controls

2.4 Graphic controls

11

15

18

18

18

19

21

22

22

22

22

24

28

32

34

35

36

36

38

38

38

38

39

40

40

41

41

42

43

45

45

46

48

52

52

53

54

55

55

56

58

58

Remember to visit the download section of [www.windev.com](http://www.windev.com) on a regular basis to check whether upgraded versions are available.

Email address of our Free Technical Support: [freetechnicalsupport@windev.com](mailto:freetechnicalsupport@windev.com).

This documentation is not contractually binding. Modifications may have been made to the software since this guide was published. Please check the online help.

All product names or other trademarks mentioned in this publication are registered trademarks of their respective owners.  
© PC SOFT 2023: This publication may not be reproduced in part or in full without the express consent of PC SOFT.

2.5	Other controls .....	59	4.20	Network management functions .....	110
2.5.1	HTML Display control functions .....	59	4.21	Socket management functions .....	111
2.5.2	HTML Editor control functions .....	60	4.22	WebSocket management functions .....	112
2.5.3	Action Bar control functions .....	61	4.23	WINDEV FTP functions .....	112
2.5.4	Sliding Banner control functions .....	61	4.24	SNMP functions .....	114
2.5.5	Captcha control functions .....	61	4.25	SSH functions .....	115
2.5.6	Breadcrumb control functions .....	61	4.26	Telephony functions .....	115
2.5.7	Progress Bar control functions .....	62	4.27	Fax functions .....	117
2.5.8	Repositionable Note control functions .....	62	4.28	Stream management functions (video and sound) .....	118
2.5.9	Xaml control functions .....	62	4.29	XML functions .....	118
2.6	General control functions .....	63	4.30	Lotus Notes, Outlook functions .....	121
2.6.1	Control management functions .....	63	4.30.1	Appointment management functions .....	121
2.6.2	Window control search functions .....	67	4.30.2	Contact management functions .....	122
2.6.3	Assisted input management functions .....	67	4.30.3	Contact groups management functions .....	123
2.6.4	RTF control functions .....	68	4.31	Notes document management functions .....	124
2.6.5	Token management functions .....	68	4.32	Task management functions .....	125
2.6.6	Animation management functions .....	69	4.33	LDAP functions .....	126
3.	Basic functions .....		4.34	Google functions .....	127
3.1	Character string management functions .....	71	4.34.1	Google Contacts management functions .....	127
3.2	Compression / encryption functions .....	76	4.34.2	Google document management functions .....	127
3.3	Hash functions .....	77	4.34.3	Google Calendar management functions .....	128
3.4	Date and Time management functions .....	78	4.34.4	Google Analytics functions .....	128
3.5	Array management functions .....	82	4.35	Google Maps management functions .....	129
3.6	External file management functions .....	87	4.36	RSS feed management functions .....	129
3.7	Shared memory area functions .....	92	4.37	Facebook functions .....	129
3.8	Queue, stack and list management functions .....	92	4.38	MQTT functions .....	130
3.9	Combination and Enumeration management functions .....	93	4.39	Geolocation/GPS functions .....	130
3.10	Java functions .....	93	4.40	In-App functions .....	131
3.11	Python functions .....	94	4.41	NFC functions .....	132
4.	Communication .....		4.42	Mobile phone management functions .....	132
4.1	Beacon functions .....	95	4.43	SMS functions .....	134
4.2	Bluetooth functions .....	95	4.44	iCloud functions .....	135
4.3	BTLE functions .....	96	4.45	Functions specific to iOS - Apple Watch communication .....	135
4.4	Wi-Fi functions .....	97	4.46	Sensor functions .....	136
4.5	Email management functions .....	98	4.47	Download management functions on an Android device .....	136
4.6	HTTP functions .....	101	5.	Multimedia management .....	
4.7	Standard FTP functions .....	103	5.1	Image and drawing management functions .....	137
4.8	SCP functions .....	104	5.2	Exif functions .....	143
4.9	J2EE functions .....	105	5.3	Image scrolling management functions .....	144
4.10	OBEX functions .....	105	5.4	TWAIN functions .....	144
4.11	.Net functions .....	106	5.5	Scanner functions .....	146
4.12	SOAP functions .....	106	5.6	Graphic string management functions .....	147
4.13	OAuth and OpenID protocol management functions .....	107	5.7	CD/DVD functions .....	148
4.14	DLNA/UPnP functions .....	107	5.8	Burn functions .....	149
4.15	Domotic management functions .....	108	5.9	MCI functions .....	149
4.16	OLE functions .....	108	5.10	OCR (Optical Character Recognition) management functions .....	151
4.17	Serial and parallel port management functions .....	109	5.11	Image object detection functions .....	151
4.18	USB functions .....	110	5.12	Video functions .....	152
4.19	DNS functions .....	110			

6.	Databases	
6.1	HFSQL management functions.....	153
6.2	Link functions.....	166
6.3	HFSQL Cluster management functions.....	168
6.4	HFSQL Client/Server management functions.....	169
6.5	SQL database access functions.....	177
6.6	HBase functions.....	179
6.7	HDFS functions.....	179
6.8	Replication management functions.....	180
6.9	Memcached functions.....	181
6.10	MongoDB functions.....	183
6.11	Redis management functions.....	184
6.12	3-Tier functions.....	184
6.13	SAP functions.....	185
6.14	Salesforce functions.....	185
6.15	Local storage management functions.....	186
7.	Print functions	
8.	Maths	
8.1	Financial functions.....	193
8.2	Chart management functions.....	194
8.3	Math functions.....	200
8.4	Binary values management functions.....	202
8.5	Matrix management functions.....	202
8.6	Statistical functions.....	203
8.7	MATLAB functions.....	204
8.8	Memory area management functions.....	204
9.	Other functions	
9.1	Certificate management functions.....	205
9.2	Registry management functions.....	206
9.3	Archiving functions.....	206
9.4	XLS file handling functions.....	209
9.5	PDF file management functions.....	210
9.6	HTML file management functions.....	211
9.7	Functions for managing multilingual applications.....	212
9.8	Clipboard management functions.....	213
9.9	Mouse management functions.....	213
9.10	Drag and Drop management functions.....	214
9.11	System functions.....	215
9.11.1	Recycle bin management functions.....	215
9.11.2	Windows Scheduler management functions.....	215
9.11.3	Windows event management functions.....	216
9.11.4	Windows interactive notification management functions.....	216
9.11.5	Miscellaneous Windows functions.....	217
9.11.6	System-specific functions.....	218
9.11.7	Thread management functions.....	222
9.11.8	Parallel task management functions.....	224

9.12	Font management functions.....	224
9.13	User Groupware management functions.....	225
9.14	Service management functions.....	227
9.15	AAF (Automatic Application Features) management functions.....	228
9.16	Miscellaneous WEBDEV functions.....	229
9.17	Miscellaneous WINDEV functions.....	231
9.18	Project management functions.....	238
9.19	Debugging functions.....	239
9.20	Functions for managing Executables in Console mode.....	240
9.21	Activation key management functions.....	240
9.22	WEBDEV scheduled task management functions.....	241
9.23	Action plan management functions (software factory).....	241
9.24	Hasp functions.....	242
9.25	Background task management functions.....	242
9.26	Telemetry management functions.....	243
9.27	Action functions.....	243
9.28	AJAX management functions.....	243
9.29	Photo album management functions.....	244
9.30	Application management functions with "Live update".....	244
9.31	Bar Code management functions.....	245
9.32	DDE functions.....	245
9.33	Continent and country management functions.....	246
9.34	Multi-touch management functions.....	246
9.35	Runtime error management functions.....	247
9.36	Euro functions.....	248
9.37	End-user UI management functions.....	248
9.38	Browser management functions.....	249
9.39	Speech recognition management functions.....	251
9.40	Speech synthesis management functions.....	251
9.41	Spelling check functions.....	252
9.42	Upload functions.....	252
9.43	URI functions.....	253
9.44	Lock functions.....	253
9.45	SaaS functions.....	254
9.46	UMC management functions.....	257
9.47	Test-specific functions.....	257
9.48	Emulation functions.....	258
9.49	Webservice management functions.....	259

# Overview

The integrated WINDEV language, WLanguage, is powerful, simple and intuitive. it's a fifth generation language (5GL), which means that its commands are highly sophisticated. A WLanguage command can replace dozens or even hundreds of 4GL commands, making programming easier and more reliable.

This manual presents all the functions currently available in WLanguage, classified by theme, with a short description. An index allows you to find all the WLanguage functions in alphabetical order.

You will find more details about these functions in the online help (<https://doc.windev.com>). The detailed syntax of each function and the code examples are also available in the online help.

# 1. Window and page management functions

## 1.1 Window management functions

The following functions can be used to handle WINDEV and WINDEV Mobile windows:

Abandon	Closes the current window by running the "Click" event of the CANCEL Button control.
ChangeSkinTemplate	Dynamically changes the skin template: <ul style="list-style-type: none"><li>• of all windows found in the current application,</li><li>• of the windows found in the WINDEV components used by the application,</li><li>• of a specific window found in the project.</li></ul>
Close	Closes a WINDEV window (and returns a value if necessary).
CurrentTitle	Modifies the title of the current window (returned by <b>CurrentWin</b> ) and the title bar.
CurrentWin	Identifies the window currently in edit.
DelayBeforeClosing	Limits the display duration of a window or dialog box. If no user action is performed, when the specified amount of time is reached: <ul style="list-style-type: none"><li>• for a window, the code of a window button will be run.</li><li>• for a dialog box, the dialog box is automatically closed.</li></ul>
EnumMenu	Returns: <ul style="list-style-type: none"><li>• the name of nth option or sub-option found in the main menu of window.</li><li>• the name of nth option or sub-option found in a custom context menu (created with WINDEV or WINDEV Mobile).</li></ul>
ExecuteRefreshUI	Immediately executes the "Request for refreshing the display" event of the window, page, internal window, internal page, control template or supercontrol.
ExecuteRefreshUIElement	Immediately executes the "Request for refreshing the display" event of the given element.
ExecuteRefreshUIParent	Immediately executes the "Request for refreshing the display" event of the parent.



FullScreen	Enables or disables the full-screen mode in the applications in Universal Windows 10 App mode.
Iconize	Minimizes a WINDEV window.
InitWindow	Initializes (or not) the controls to zero and runs the initialization processes of the controls and the window.
Maximize	Displays a WINDEV window in its maximum size.
MultitaskRedraw	Immediately redraws the window controls that will be graphically refreshed.
NextTitle	Modifies the title of next window to open.
Open	Opens a modal WINDEV window.
OpenAsynchronous	Opens a window whose result will be retrieved via a WLanguage procedure ("callback").
OpenChild	Opens a non-modal child window.
OpenMainMenu	Opens the main menu of the current window.
OpenMobileWindow	Opens a window in a mobile application.
OpenPopup	Opens a popup window and waits for this window to be closed.
OpenPopupAsynchronous	Opens a popup window whose result will be retrieved via a WLanguage procedure ("callback").
OpenPopupPosition	Opens a popup window by specifying its opening position and waits until it is closed.
OpenSister	Opens a non-modal sibling window.
PreviousWin	Identifies the window that was in edit before the window that is currently in edit (for the same project).
RequestRefreshUI	Requests the execution of the "Request for refreshing the display" event (and gives back control <b>without waiting for the execution of the event</b> ).
RequestRefreshUIElement	Requests the execution of the "Request for refreshing the display" event of the specified element (and gives back control <b>without waiting for the execution of the event</b> ).
RequestRefreshUIParent	Asks to run the "Request for refreshing the display" process of parent.
Restore	Displays a WINDEV window to its initial size.

Use	Opens a WINDEV window and closes all the other windows that were opened beforehand.
WaitWindowEnd	Closes the wait window of application opened by <b>WaitWindowStart</b> .
WaitWindowStart	Opens the wait window of application.
WinActivateDDW	Enables or disables the DDW (Dim Disabled Window) feature for a specific window. ou for all the application windows.
WinAdaptHeight	Adapts the window height to the content of controls.
WinAdaptSize	Adapts the window size to the content of controls.
WinAdaptWidth	Adapts the window width to the content of controls.
WinBackgroundImage	Modifies the background image of a window and specifies the display mode of this image.
WinChangeAlias	Modifies the alias of a window.
WinChangeLayout	Applies a specific layout to a window or an internal window.
WinCopyForm	Copies the content of a form into the clipboard.
WinCurrentLayout	Returns the index of the current layoutin the current window or in the specified window or internal window.
WinDisableEffect	Disables the visual effects of the graphic engine of WINDEV (for a use in TSE mode for example).
WindowBitmap	Creates the image of the specified window in a BMP file.
WinForceDDW	Forces the DDW (Dim Disabled Windows) feature on a window even if this window is not accessible.
WinGiveSuitableHeight	Returns the height of a window, adapted to the content of controls.
WinGiveSuitableWidth	Returns the width of a window, adapted to the content of controls.
WinIconBarHeight	Returns or modifies the current height of the "Icon Bar" area in the current MDI parent window.
WinInactiveEffect	Enables or disables the DDW (Dim Disabled Window) feature for a specific window or for all the application windows.

WinInHeight	Returns the internal height of a window (without title bar and border).
WinInitialized	Used to find out whether the "end of initialization" code was run for a window.
WinInput	Identifies the window containing the WLanguage code currently run.
WinInWidth	Returns the internal width of a window (without border).
WinInXPos	Returns the horizontal position of the top left corner of the internal area (without border and title bar) of a window in relation to the top left corner of the screen.
WinInYPos	Returns the vertical position of the top left corner of the internal area (without border and title bar) of a window in relation to the top left corner of the screen.
WinOrientation	Modifies or retrieves the display orientation of the current window.
WinOutHeight	Returns the full height of the window (including the border, menu bar and title).
WinOutWidth	Returns the full width of a window (including the border).
WinOutXPos	Returns the horizontal position of the upper-left corner of a window in relation to the upper-left corner of the screen.
WinOutYPos	Returns the vertical position of the upper-left corner of a window in relation to the upper-left corner of the screen.
WinPasteForm	Pastes the content of a form stored in the clipboard.
WinRateDDW	Gets and sets the dimming rate for windows that use the DDW (Dim Disabled Windows) feature.
WinRedraw	Immediately redraws the window controls for which a graphic update is required.
WinRefreshVisible	Modifies or returns the visibility state of the refresh bar in a window.
WinScreenRectangle	Returns the coordinates of the screen that contains a window.
WinSize	WinSize returns or modifies the window display mode (maximized, minimized or normal), modifies the window height and/or width, moves the window and modifies a window's height and/or width.

WinSlidingVisible	Displays or hides a sliding window (sliding menu).
WinStatus	Identifies or modifies the window status.

## 1.2 Page management functions

The following functions are used to manage pages:

CancelAWPContext	Deletes from the AWP context a variable that was added by <b>DeclareAWPContext</b> .
CellCloseDialog	Hides a Cell control previously displayed in the page via <b>CellDisplayDialog</b> .
CellDisplayDialog	Displays a Cell control in a page with a DDW effect (Dim Disabled Windows). This function is used to easily simulate a dialog box in browser code, by using a Cell control in a page.
ChangeAction	Used to specify the action to perform when the HTML page displayed in the browser is no longer synchronized with the page context on server.
ChangeSourcePage	Dynamically changes the page displayed in an Internal Page control
ChangeTarget	Changes the target of a button action.
ConfigureAWPContext	Configures the operating mode of the AWP context.
ContextClose	Closes a page context.
ContextExist	Allows you to find out whether a page context exists on the server (which means whether the page was opened).
ContextOpen	Opens a new page context without returning the information to the browser.
CurrentPage	Returns the name of the page containing the WLanguage code currently run.
DeclareAWPContext	Used to declare a list of variables whose value will be persistent between the successive displays of Active WEBDEV Pages.
DynamicSiteDisplay	Starts a dynamic WEBDEV site from a static page or or from a dynamic WEBDEV page.
FramesetDisplay	Displays a WEBDEV frameset in the user's browser.



FramesetRefresh	Refreshes a frameset displayed in the user's browser from the context on the server.
FramesetUse	Displays a WEBDEV frameset in the user's browser and closes all the current page and frameset contexts.
FreeAWPContext	Frees the AWP context in advance (on disk) to allow the other call on the same AWP context to be processed in parallel.
IdentifierAWPContext	Returns the identifier of AWP context.
PageActivateDDW	Enables or disables the DDW (Dim Disabled Window) when displaying a modal page (the pages used to communicate with the user for example).
PageAddress	Used to find out the Internet address of a page.
PageChangeLayout	Applies a specific layout to a page.
PageCloseDialog	Closes the current page. This page was opened by <b>PageDisplayDialog</b> . A return value can be returned to the calling page.
PageCurrentLayout	Returns the index of the current layout in the current page or in the specified page.
PageDisplay	Displays a web page in the user's browser.
PageDisplayDialog	Displays a page in modal mode. This function is used to establish a dialog with the user. The page is displayed in the foreground while the opening page is displayed in the background, grayed by the DDW mechanism.
PageExist	Checks whether the specified page is currently displayed in the browser of the Web user. Used to find out whether a page is displayed in a frame other than the current one.
PageInitialization	Resets to zero (or not) the controls found in the current page and starts the events for initializing the controls and the page.
PageParameter	Returns the value of a parameter passed to the current page.
PagePosition	Scrolls a page up to position a control in the visible section of the page (top) in the browser.
PageRateDDW	Allows you to find out and modify the rate of gray for the pages that use DDW (Dim disabled pages).

PageReadHeaderHTTP	Reads and returns one or more HTTP headers received by the current page (these headers are sent by the client to the server in the current request).
PageRefresh	Refreshes a page displayed in the user's browser from the context on the server.
PageSubmit	Validates the specified page and starts the execution of a button.
PageToASP	Sends the data found in a page currently displayed in the browser to an ASP server.
PageToEmail	Emails the data found in a page currently displayed in the browser.
PageToJSP	Sends the data found in a page currently displayed in the browser to a JSP server.
PageToPHP	Sends the data found in a page currently displayed in the browser to a PHP server.
PageUse	Displays a WEBDEV page in the user's browser and closes all page contexts currently open on the server.
PageVisible	Indicates whether the page is visible to the user.
PageWriteHeaderHTTP	Adds an additional HTTP header that will be returned to the browser: <ul style="list-style-type: none"><li>• during the next function for page display (<b>PageDisplay</b>, <b>PageRefresh</b>, ...),</li><li>• during the automatic return of the current page at the end of the current action.</li></ul>
PopupAnimate	Displays a popup in a page cell.
PopupClose	Hides a popup displayed in the page via <b>PopupDisplay</b> .
PopupClosePage	Hides a page displayed as a popup via <b>PopupDisplayPage</b> .
PopupDisplay	Displays a popup in a page with a DDW effect (Dim Disabled Windows).
PopupDisplayPage	Displays a WEBDEV page as a popup in the current page with a DDW (Dim Disabled Windows) effect.
PreviousPage	Returns the name of the previous page.
SemiDynamicPageDisplay	Displays a semi-dynamic page in the browser of the Web user from a dynamic or static WEBDEV page.

SiteAddress	Returns the Internet address for connecting to a dynamic WEBDEV site (in Session mode) found on the same server.
-------------	--

### 1.3 Additional functions available

#### 1.3.1 MDI functions

The MDI functions are as follows:

MDIActive	Identifies or displays an MDI child window in the foreground.
MDIEnumChild	Returns the alias of the specified MDI child window.
MDIMother	Identifies the name of the MDI parent window.
MDIOpen	Opens and enables an MDI child window.
MDIWindowMenu	Modifies the layout of the MDI child windows.

#### 1.3.2 Menu management functions

The following functions are used to manage the menus:

MenuAdd	Adds an existing popup menu into the main menu of the current window.
MenuAddMenu	Adds a new menu into a window or into a page.
MenuAddOption	Adds a new menu option at the end of a menu.
MenuAddPopup	Transforms a menu option of a page in order for this option to open a popup.
MenuAddSeparator	Adds a new separator into a menu.
MenuAddURLOption	Adds a new menu option at the end of a page menu. This menu option is used to display the page corresponding to the specified URL.
MenuClone	Clones a menu or a menu option as well as the associated code.
MenuDelete	Deletes a menu or a menu option.
MenuExist	Indicates whether a menu option exists in a menu.
MenuInsertMenu	Inserts a menu before another menu in a window.

MenuInsertOption	Inserts a new option at a specific position.
MenuInsertSeparator	Inserts a separator into a menu.
MenuInvisible	Makes a menu option invisible (popup menu or drop-down menu).
MenuIsMarked	Used to find out whether a checkmark is displayed in front of a menu option (popup menu or drop-down menu).
MenuLabel	Gets or changes the label of a menu option (context or drop-down menu).
MenuMark	Positions the checkmark in front of the menu option (popup menu or drop-down menu).
MenuSelectMinus	Disables (grays out) a menu option (popup menu or drop-down menu).
MenuSelectPlus	Enables a menu option (popup menu or drop-down menu).
MenuState	Identifies the state of a menu option (context menu or drop-down menu): active, inactive or invisible.
MenuUnMark	Removes the checkmark displayed in front of the menu option (popup menu or drop-down menu).

#### 1.3.3 Dialog box management functions

The following functions are used to manage the dialog boxes:

Confirm	Displays a message in a standard dialog box that proposes "Yes", "No", "Cancel" and returns the user's choice.
ConfirmAsynchronous	Displays a non-blocking message in a standard dialog box with the answers "Yes", "No", "Cancel" and calls a WLanguage procedure with the user's response.
Dialog	Displays a message box and returns the value of the button clicked by the user.
DialogAsynchronous	Displays a non-blocking message box and calls a WLanguage procedure with the value of the button clicked by the user.
Error	Displays a custom error message in a system error window.
ErrorAsynchronous	Displays a custom error message in a non-blocking system error window.

ErrorBuild	Displays a custom error message in a system error window.
ErrorWithTimeout	Displays a custom error message in a system error window for a set amount of time.
ErrorWithTimeout-Asynchronous	Displays a custom error message in a non-blocking system error window for a given amount of time.
Info	Displays a custom message in a system information window.
InfoAsynchronous	Displays a non-blocking custom message in a system information window.
InfoBuild	Displays a custom message in a system information window.
InfoWithTimeout	Displays an information message in a system information window for a set amount of time
InfoWithTimeout-Asynchronous	Displays a custom message in a non-blocking system information window for a given amount of time.
Input	Displays a message allowing the user to type an information.
Message	Displays or removes a message from the status bar of the current window or browser.
OKCancel	Displays a message in a standard dialog box that proposes "OK" and "Cancel" and returns the user's choice.
OKCancelAsynchronous	Displays a message in a standard non-blocking dialog box with the answers "OK" and "Cancel" and calls a WLanguage procedure with the user's response.
ProgressBar	Displays a progress bar in the status bar of current window.
StatusBarAddCell	Adds a new cell to the status bar of the current window.
StatusBarDeleteCell	Deletes a cell from the status bar of the current window.
Warning	Displays a custom message in a system warning window.
WarningAsynchronous	Displays a custom message in a non-blocking system warning window.
YesNo	Displays a message in a standard dialog box that proposes "Yes" and "No" and returns the user's choice.

YesNoAsynchronous	Displays a message in a standard non-blocking dialog box with the answers "Yes" and "No" and calls a WLanguage procedure with the user's response.
-------------------	--

1.3.4 Toast management functions

The following functions are used to manage toasts:

ToastDeleteAll	Deletes all the toasts displayed by <b>ToastDisplay</b> .
ToastDisplay	Displays a "Toast" message.
ToastDisplayIW	Opens a toast whose content is based on an internal window.
ToastDisplayPopup	Displays a popup page during a given duration in order to display a "Toast" message.

## 2. Functions for managing different types of controls

### 2.1 Usual controls

#### 2.1.1 Check Box control functions

The following functions are used to manage Check Box controls:

CheckBoxAdd	Adds a new option into a Check Box control.
CheckBoxCount	Returns the number of options found in a Check Box control.
CheckBoxDelete	Deletes an option from a Check Box control.
CheckBoxInsert	Inserts a new option into a Check Box control.

#### 2.1.2 Radio Button control functions

The following functions are used to manage Radio Button controls:

RadioButtonAdd	Adds a new option into a Radio Button control.
RadioButtonCount	Returns the number of options found in a Radio Button control.
RadioButtonDelete	Deletes an option from a Radio Button control.
RadioButtonInsert	Inserts a new option into a Radio Button control.

### 2.2 Controls that handle data

#### 2.2.1 List Box, ListView and Combo Box control functions

The following functions are used to manage List Box, ListView and Combo Box controls:

FileToMemoryList	Populates a List Box or Combo Box control programmatically with the records from a data file or query.
ListAdd	Adds an element to a List Box, ListView or Combo Box control populated programmatically.
ListCount	Returns the number of elements in a List Box, ListView or Combo Box control.

ListDelete	Deletes an element from a List Box, ListView or Combo Box control populated programmatically.
ListDeleteAll	Clears a List Box, ListView or Combo Box control populated programmatically.
ListDisplay	Refreshes a List Box, ListView or Combo Box control based on a data file
ListInfoXY	Returns the index of the row (or image) at a given position in a List Box or ListView control.
ListInsert	Inserts an element into a List Box, ListView or Combo Box control populated programmatically.
ListModify	Modifies an element in a List Box, ListView or Combo Box control populated programmatically.
ListMoveLine	Moves a row or swaps two rows in a List Box or Combo Box control.
ListPosition	Displays a List Box or ListView control from a specified element or returns the index of the first element displayed in a List Box control (if the scrollbar was used).
ListSeek	Searches for an element in a List Box, ListView or Combo Box control (including the values described by <b>gStoredValue</b> ).
ListSelect	Returns the index of the selected element in a List Box, ListView or Combo Box control.
ListSelectCount	Returns the number of elements selected in a List Box, ListView or Combo Box control.
ListSelectMinus	Deselects an element in a List Box, ListView or Combo Box control.
ListSelectPlus	Selects one or more elements in a List Box, ListView or Combo Box control.
ListSort	Sorts the elements in a List Box, ListView or Combo Box control populated programmatically.

The following functions are specific to the Combo Box controls:

ComboClose	Closes the drop-down list in a Combo Box control (simple or multi-column Combo Box control).
ComboOpen	Expands the list of a Combo Box control (single-column or multi-column Combo Box).

2.2.2 Table control functions

The following functions are used to manage Table controls:

BuildBrowsingTable	Creates a Table control from the items of a data file (view or query).
FileToMemoryTable	Populates a Table control programmatically with the records from a data file, HFSQL view or query (query created in the query editor or with <b>HExecuteSQLQuery</b> ).
TableAdd	Adds a row in: <ul style="list-style-type: none"><li>• a Table control,</li><li>• a TreeView Table control,</li><li>• a Combo Box control "with table" (Table control displayed in a Combo Box control).</li></ul>
TableAddColumn	Adds elements on the right of an Horizontal Table control.
TableAddInProgress	Modifies or returns the visibility status of internal wait window when adding additional elements into an infinite Table control.
TableAddLine	Adds a row in: <ul style="list-style-type: none"><li>• a Table control,</li><li>• a TreeView Table control.</li></ul>
TableAddLineWithAnimation	Adds a row to a Table or TreeView Table control with an animation.
TableAddTitleHeader	Adds a column title-header into a Table or TreeView Table control in a window.
TableAdjust	Adjusts the title and size of columns found in a Table or TreeView Table control.
TableBreakSubscript	Returns the index of the break header and footer for a given break in a Table control.
TableCollapseExpand	Collapses or expands an element in a Table or TreeView Table control.
TableColumnSubscript	Returns the index of a column in a Table or Treeview Table control.
TableCount	Returns: <ul style="list-style-type: none"><li>• the number of columns found in a Table or TreeView Table control.</li><li>• the number of rows found in a Table or TreeView Table control.</li></ul>

TableDelete	Deletes a row from: <ul style="list-style-type: none"><li>• a Table control,</li><li>• a TreeView Table control,</li><li>• a table displayed in a Combo Box control.</li></ul>
TableDeleteAll	Deletes all rows from: <ul style="list-style-type: none"><li>• a Table control populated programmatically,</li><li>• a TreeView Table control populated programmatically,</li><li>• a table displayed in a Combo Box control.</li></ul>
TableDeleteSelect	Deletes the selected rows from a Table or Treeview Table control.
TableDeleteWithAnimation	Deletes a row from a Table control, from a TreeView Table control (or from a table displayed in a Combo Box control) with an animation.
TableDisableFilter	Disables a user filter on a column or on all the columns found in a Table or TreeView Table control.
TableDisplay	Refreshes a Table or TeeeView Table control from a given position: <ul style="list-style-type: none"><li>• for a control linked to a file: changes made in the linked data file are reflected in the control.</li><li>• for a control handled programmatically: the calculated columns are re-calculated.</li></ul>
TableEnableFilter	Implements a user filter on a column found in a Table or TreeView Table control.
TableEnd	Stops the manual iteration of data file linked to a Table control based on a data file.
TableEndFilling	Ends the process to populate a Table or TreeView Table control programmatically.
TableEnumBreak	Enumerates the breaks found in a Table control.
TableEnumColumn	Returns the name of a column found in a Table or Treeview Table control.
TableFilteredColumn	Returns the list of columns filtered by the user in a Table or TreeView Table control.
TableFixColumn	Fixes a column to the left or frees a column fixed to the left.
TableFormulaAdd	Adds a calculated row to a Table control by providing custom calculation procedures.
TableFormulaDeleteAll	Deletes all formulas added with <b>TableFormulaAdd</b> .

TableInfoXY	Returns for a given position in a Table or TreeView Table control (coordinates of a control point): <ul style="list-style-type: none"><li>the name of the column displayed for the specified position.</li><li>the index of the row or column for the specified position.</li></ul>
TableInputInProgress	Indicates whether a Table or TreeView Table control is currently in edit.
TableInputSearch	Enables input in the the search area in the column title of the Table or Treeview Table control.
TableInsert	Inserts a row into: <ul style="list-style-type: none"><li>a Table control,</li><li>a TreeView Table control,</li><li>a table displayed in a Combo Box control.</li></ul>
TableInsertColumn	Inserts elements in the format of columns into a horizontal Table control.
TableInsertLine	Inserts a row into: <ul style="list-style-type: none"><li>a Table control,</li><li>a TreeView Table control.</li></ul>
TableInsertLineWith-Animation	Inserts a row into a Table control or into a TreeView Table control with an animation.
TableModify	Modifies a row in: <ul style="list-style-type: none"><li>a Table control,</li><li>a TreeView Table control,</li><li>a table displayed in a Combo Box control.</li></ul>
TableModifyColumn	Modifies a column in a horizontal Table control.
TableModifyLine	Modifies a row in: <ul style="list-style-type: none"><li>a Table control,</li><li>a TreeView Table control.</li></ul>
TableMoveColumn	Moves a column in a Table or TreeView Table control.
TableMoveLine	Moves a row or swaps two rows in a Table control.
TablePosition	Displays a Table or TreeView Table control from a specified row or returns the index of the first row displayed in a Table or TreeView Table control.
TablePrint	Prints the content of the specified Table control without having to create a report based on the Table control.

TableRecNum	Returns the number of the current record in a Table control based on a data file.
TableRefreshVisible	Modifies or returns the visibility status of refresh bar in a Table or TreeView Table control.
TableRestorePosition-AndSelection	Restores the scrollbar position in a Table or TreeView Table control as well as the selected elements.
TableSave	Updates or adds the record bound to the current row in the Table or TreeView Table control.
TableSavePosition-AndSelection	Saves: <ul style="list-style-type: none"><li>the scrollbar position in a Table or TreeView Table control,</li><li>the elements selected in the control.</li></ul>
TableSearch	Performs a search in: <ul style="list-style-type: none"><li>a Table control,</li><li>a TreeView Table control,</li><li>a table displayed in a Combo Box control.</li></ul>
TableSearchEverywhere	Searches for a value in all the columns of a Table control, a TreeView Table control, a Table control displayed in a Combo Box control.
TableSelect	Returns the index of the selected element in the Table or TreeView Table control.
TableSelectCount	Returns the number of selected elements in a Table or TreeView Table control (especially multi-selection controls).
TableSelectMinus	Deselects a row from a Table or TreeView Table control.
TableSelectPlus	Selects a row in a Table or TreeView Table control.
TableSelectToFile	Positions on the data file record corresponding to a row selected in the multiselection browsing Table control.
TableSetFocus	Gives focus to a row found in a Table or TreeView Table control: the control switches to edit for the specified row and column.
TableSort	Sorts a Table or TreeView Table control on one or more columns (lexicographical sort).
TableSortedColumn	Returns the list of sorted columns found in a Table control.



TableStartFilling	Starts the process to populate a Table or TreeView Table control programmatically.
TableSwapLine	Swaps: <ul style="list-style-type: none"><li>• two rows in a Table control populated programmatically.</li><li>• two branches in a Treeview Table control populated programmatically.</li></ul>
TableToClipboard	Copies the content of a Table or TreeView Table control to the clipboard.
TableToExcel	Creates an Excel file with the data from a Table or TreeView Table control.
TableToText	Creates a character string from the data found in a Table or TreeView Table control.
TableToWord	Creates a Word file (.RTF) from the data found in a Table or TreeView Table control.
TableToXML	Creates an XML file from the data found in a Table or TreeView Table control.

2.2.3 TreeView Table control functions

The following functions are used to manage TreeView Table controls:

TableDisplay	Refreshes a Table or TeeeeView Table control from a given position: <ul style="list-style-type: none"><li>• for a control linked to a file: changes made in the linked data file are reflected in the control.</li><li>• for a control handled programmatically: the calculated columns are re-calculated.</li></ul>
TableAdd	Adds a row in a Table control, a TreeView Table control, a Combo Box control "with table" (Table control displayed in a Combo Box control).
TableAddChild	Adds a row into a TreeView Table control, at the end of a given hierarchy level.
TableAddLine	Adds a row in a Table control, a TreeView Table control.
TableAdjust	Adjusts the title and size of columns found in a Table or TreeView Table control.
TableSearchChild	Searches for a value in a column of a TreeView Table control, in a specified branch.

TableFilteredColumn	Returns the list of columns filtered by the user in a Table or TreeView Table control.
TableMoveBranch	Moves the content of a row as well as its entire descending tree structure in a TreeView Table control.
TableMoveColumn	Moves a column in a Table or TreeView Table control.
TableExpandAll	Expands: <ul style="list-style-type: none"><li>• the entire hierarchy (all nodes) of a TreeView Table control.</li><li>• all breaks of a Table control.</li></ul>
TableSwapBranch	Swaps the content of two rows (as well as their respective tree structure) in a TreeView Table control.
TableSwapLine	Swaps: <ul style="list-style-type: none"><li>• two rows in a Table control populated programmatically.</li><li>• two branches in a Treeview Table control populated programmatically.</li></ul>
TableCollapse	Collapses: <ul style="list-style-type: none"><li>• a branch that was previously expanded in a TreeView Table control.</li><li>• a break that was previously expanded in a Table control.</li><li>• a detailed window displayed for a row.</li></ul>
TableCollapseAll	Collapses: <ul style="list-style-type: none"><li>• the entire hierarchy (all nodes) of a TreeView Table control.</li><li>• all breaks of a Table control.</li></ul>
TableEnumColumn	Returns the name of a column found in a Table or Treeview Table control.
TableItemStatus	Returns the status of a node in a Treeview Table control: collapsed, expanded or not found.
TableChildCount	Returns the number of direct children for an element in a TreeView Table control.
TableFixColumn	Fixes a column to the left or frees a column fixed to the left.
TableColumnSubscript	Returns the index of a column in a Table or Treeview Table control.

TableInfoXY	Returns for a given position in a Table or TreeView Table control (coordinates of a control point): <ul style="list-style-type: none"><li>the name of the column displayed for the specified position.</li><li>the index of the row or column for the specified position.</li></ul>
TableInsert	Inserts a row into a Table control, a TreeView Table control, a table displayed in a Combo Box control.
TableInsertChild	Inserts a row into a TreeView Table control, in a given hierarchy level.
TableInsertLine	Inserts a row into a Table control, a TreeView Table control.
TableListChild	Lists the "children" of a node and the "child" elements of these "children" in a TreeView Table control.
TableModify	Modifies a row in a Table control, a TreeView Table control, a table displayed in a Combo Box control.
TableModifyLine	Modifies a row in a Table control, a TreeView Table control.
TableCount	Returns: <ul style="list-style-type: none"><li>the number of columns found in a Table or TreeView Table control.</li><li>the number of rows found in a Table or TreeView Table control.</li></ul>
TablePosition	Displays a Table or TreeView Table control from a specified row or returns the index of the first row displayed in a Table or TreeView Table control.
TableRefreshVisible	Modifies or returns the visibility status of refresh bar in a Table or TreeView Table control.
TableGiveChild	Returns the "children" of an element (lower level) in a TreeView Table control.
TableGiveParent	Returns the "parent" of an element in a TreeView Table control.
TableRestorePosition-AndSelection	Restores the scrollbar position in a Table or TreeView Table control as well as the selected elements.
TableInputInProgress	Indicates whether a Table or TreeView Table control is currently in edit.

TableSavePosition-AndSelection	Saves: <ul style="list-style-type: none"><li>the scrollbar position in a Table or TreeView Table control,</li><li>the elements selected in the control.</li></ul>
TableSelect	Returns the index of the selected element in the Table or TreeView Table control.
TableSelectMinus	Deselects a row from a Table or TreeView Table control.
TableSelectCount	Returns the number of selected elements in a Table or TreeView Table control (especially multi-selection controls).
TableSelectPlus	Selects a row in a Table or TreeView Table control.
TableDelete	Deletes a row from a Table control, a TreeView Table control, a table displayed in a Combo Box control.
TableDeleteChild	Deletes all child rows from a branch of the hierarchy.
TableDeleteSelect	Deletes the selected rows from a Table or Treeview Table control.
TableDeleteAll	Deletes all rows from a Table control populated programmatically, a TreeView Table control populated programmatically, a table displayed in a Combo Box control.
TableSortChild	Sorts a branch in a TreeView Table control on one or more columns.
TableTypeItem	Checks the existence of an element in a TreeView Table and returns its type (node or leaf) if the element exists.
TableToExcel	Creates an Excel file with the data from a Table or TreeView Table control.
TableToClipboard	Copies the content of a Table or TreeView Table control to the clipboard.
TableToText	Creates a character string from the data found in a Table or TreeView Table control.
TableToWord	Creates a Word file (.RTF) from the data found in a Table or TreeView Table control.
TableToXML	Creates an XML file from the data found in a Table or TreeView Table control.

2.2.4 Looper control functions

The following functions are used to manage Looper controls

GalleryClosePopup	Closes the automatic zoom popup for the image gallery.
GalleryDisplayPopup	Displays the automatic zoom popup for the Image Gallery control.
GalleryPopupDisplayed	Indicates whether the automatic zoom popup of the Image Gallery control is displayed.
LooperAdd	Adds a row at the end of a Looper control.
LooperAddInProgress	Modifies or returns the visibility status of internal wait window when adding additional elements into an infinite Looper control.
LooperAddIP	Adds a row as an internal page at the end of a Looper control populated programmatically.
LooperAddIW	Adds a row as an internal window at the end of a Looper control populated programmatically.
LooperAddLine	Adds a row at the end of a Looper control.
LooperBreakSubscript	Returns the index of the break header and footer for a given break in a Looper control.
LooperCancelSwipe	Cancels the display of the internal window displayed in a Looper control further to the row swipe.
LooperCollapse	Collapses a break that was expanded beforehand in a Looper control.
LooperCollapseAll	Collapses all expanded breaks in a Looper control.
LooperCollapseExpand	On a given row of a Looper control, collapses a break (if it is expanded), or expands a break (if it is collapsed).
LooperCount	Returns the number of rows in a Looper control.
LooperDelete	Deletes a row from a Looper control.
LooperDeleteAll	Deletes all rows from a Looper control populated programmatically.
LooperDeleteSelect	Deletes the selected rows from a Looper control.
LooperDisplay	Refreshes a Looper control based on a data file from a given position: the modifications performed in the linked file are applied to the loopers.

LooperEndFilling	Ends the process to populate a Looper control programmatically.
LooperExpand	Expands a break in the Looper control on a given row.
LooperExpandAll	Expands all breaks in a Looper control with break.
LooperInfoXY	Used to find out the row or control located at specific position in the Looper control (X, Y).
LooperInsert	Inserts a row into a Looper control.
LooperInsertIP	Inserts a row as an internal page into a Looper control populated programmatically.
LooperInsertIW	Inserts a row as an internal window into a Looper control populated programmatically.
LooperInsertLine	Inserts a row into a Looper control.
LooperModify	Changes the values of the attributes in a row of a Looper control.
LooperModifyIP	Modifies a row corresponding to an internal page in a Looper control populated programmatically.
LooperModifyIW	Modifies a row corresponding to an internal window in a Looper control populated programmatically.
LooperModifyLine	Changes the values of the attributes in a row of a Looper control based on a data file or populated programmatically.
LooperMoveLine	Moves a row or swaps two rows in a Looper control.
LooperPosition	Moves the scrollbar to display a specific row or returns the index of the first row displayed in a Looper control populated programmatically.
LooperRecNum	Returns the number of the current record in a Looper control based on a data file.
LooperRefreshVisible	Changes or returns the visibility status of the refresh bar in a Looper control.
LooperRestorePosition AndSelection	Restores the scrollbar position in a Looper control as well as the selected elements.
LooperSave	Validates the modification or addition of the record associated with the current row in the Looper control.
LooperSavePosition- AndSelection	Saves the position of the scrollbar in a Looper control, or the selected element(s).

LooperSearch	Searches for a value in an attribute of a Looper control.
LooperSearchEverywhere	Searches for a value in all the attributes of a Looper control.
LooperSelect	Returns the index of the selected element in a Looper control.
LooperSelectCount	Returns the number of selected rows in a Looper control.
LooperSelectMinus	Deselects a row in a Looper control.
LooperSelectPlus	Selects a row in a Looper control.
LooperSort	Allows you to: <ul style="list-style-type: none"><li>Sort a Looper control according to one or more attributes.</li><li>Sort a Looper control on all its attributes.</li><li>Cancel a sort that was performed beforehand (which means ignore the sort).</li></ul>
LooperStartFilling	Starts the process to populate a Looper control programmatically.
LooperToExcel	Creates an Excel file from the data found in a Looper control (browsing Looper control or memory Looper control).
LooperToWord	Creates a Word file (.RTF) using the data from a Looper control.
LooperToXML	Creates an XML file with data from a Looper control (Looper control based on a data file or populated programmatically).

2.2.5 Break management functions (Tables, Loopers)

The following functions are used to manage breaks in the Table controls and in the Looper controls:

BreakAdd	Adds a break into a Table or Looper control.
BreakDelete	Deletes a break from a Table control or from a Looper control.
BreakDeleteAll	Deletes all breaks from a Table control or from a Looper control.

2.2.6 Pivot Table control functions

These functions are used to handle the Pivot Table controls:

PVTAxisXY	Used to: <ul style="list-style-type: none"><li>Specify the row and column headers displayed in a Pivot Table control.</li><li>Find out the row and column headers displayed in a Pivot Table control.</li></ul>
PVTCalculateAll	Entirely calculates or recalculates a Pivot Table control.
PVTCalculateUpdate	Updates some of the data found in a Pivot Table control from a given date.
PVTCollapse	Collapses the data found in a Pivot Table control according to a row header or to a column header.
PVTCollapseAll	Collapses all row and/or column headers in a Pivot Table control.
PVTCompareDateRange	Calculates and displays a date range comparison in a Pivot Table control.
PVTExpand	Expands the data found in a Pivot Table control according to a row header or to a column header.
PVTExpandAll	Expands all row and/or column headers in a Pivot Table control.
PVTFilter	Enables or disables a filter on a Pivot Table control.
PVTHeaderValue	Returns the value displayed in a header of Pivot Table control.
PVTInfoXY	Returns the values of headers corresponding to a position in a Pivot Table control in pixels.
PVTListPositionHeader	Returns all values associated with a row or column header in a Pivot Table control.
PVTLoad	Loads the result of the calculation performed in a Pivot Table control from a backup (file found on disk or backup in a buffer).
PVTSave	Saves the result of the calculation performed in a Pivot Table control (in a file or in memory).
PVTSelect	Allows you to find out the position of selected cells and to select cells in a Pivot Table control.

PVTSelectCount	Returns the number of selected cells in a Pivot Table control.
PVTToExcel	Creates an Excel file with data from a Pivot Table control.

2.2.7 Multiline Zone control functions

The following functions are used to manage Multiline Zone controls:

mlzAddLine	Adds a new dynamic row into a Multiline Zone control.
mlzDelete	Deletes a dynamic row from a Multiline Zone control.
mlzDeleteAll	Deletes all dynamic rows from a Multiline Zone control.
mlzInsertLine	Inserts a new dynamic row in a Multiline Zone control.

2.2.8 TreeView control functions

The following functions are used to manage the TreeView controls:

TreeAdd	Adds a "leaf" element (and all necessary nodes) into a TreeView control.
TreeCollapse	Collapses a branch previously expanded.
TreeCollapseAll	Collapses a branch previously expanded (as well as its child elements).
TreeCopyItem	Copies an element (leaf or node) into a TreeView control.
TreeCount	Returns the number of elements found in a branch of a TreeView control.
TreeDelete	Deletes an element and all its "children" from a TreeView control.
TreeDeleteAll	Deletes all elements from a TreeView control.
TreeDeleteChild	Deletes the children of an element from a TreeView control without deleting the element itself.
TreeDisplay	Refreshes the display of a TreeView control from a given position.
TreeExpand	Expands a node in a TreeView control.
TreeExpandAll	Expands a node and all its child elements in a TreeView control.

TreeFind	Finds an element (leaf or node) in a TreeView control and returns its path.
TreeGetSubscript	Returns the index of an element (node or leaf) in a TreeView control.
TreeGiveChild	Returns the "children" of an element found in a TreeView control.
TreeID	Used to build the path of the elements found in a TreeView control by differentiating the duplicates.
TreeIdentifier	Returns the identifier associated with an element (node or leaf) in a TreeView control.
TreeInfoXY	Returns the TreeView element found at a given position.
TreeInsert	Inserts a leaf into a TreeView control at a specific position.
TreeListItem	Lists the "children" of a node and the "child" elements of these "children" in a TreeView control.
TreeModify	Modifies an element (node or leaf) in a TreeView control.
TreeMoveItem	Moves an element (leaf or node) in a TreeView control.
TreePosition	Displays a TreeView control from a specified element (node or leaf) or returns the name of the first element displayed in a TreeView control.
TreeSelect	Returns the full path of the selected element in a TreeView control.
TreeSelectCount	Returns the number of selected elements in a TreeView control (for a multi-selection TreeView control).
TreeSelectMinus	Deselects an element in a TreeView control.
TreeSelectPlus	Selects an element in a TreeView control.
TreeSort	Sorts the elements found in a TreeView control.
TreeStatus	Returns the status of a node: collapsed, expanded or not found.
TreeTypeItem	Checks the existence of an element in a TreeView control and returns its type (node or leaf) if the element exists.

## 2.3 Container controls

### 2.3.1 Tab control functions

The following functions are used to handle the Tab controls:

ControlPane	Returns the name of static or dynamic pane found in a Tab control associated with a control.
ControlTab	Returns the name of the Tab control that contains the specified control.
TabClose	Closes a dynamic pane in a Tab control.
TabCloseAll	Closes all dynamic panes of a Tab control.
TabLoadConfiguration	Restores the configuration of dynamic panes in a Tab control.
TabOpen	Opens a new pane in a dynamic Tab control.
TabSaveConfiguration	Saves the configuration of dynamic panes found in a Tab control in character string format.
TabStatus	Returns the status of a dynamic tab pane (identified by its alias).

### 2.3.2 Ribbon control functions

The following functions are used to manage Ribbon controls:

RibbonAdaptSize	Adapts the size of the controls found in a Ribbon control according to the size of their caption.
RibbonCloseGrouping	Closes the group currently open in a Ribbon control.
RibbonEndExpandedMode	Collpases a collapsed ribbon that is temporarily expanded.
RibbonHighlightPane	Highlights a tab in a Ribbon control.

### 2.3.3 Toolbar control functions

The following functions are used to manage the Toolbar controls:

ToolbarDock	Docks a Toolbar control to a window or changes the anchoring position of a docked Toolbar control.
ToolbarPosition	Indicates the display mode of a Toolbar control.

ToolbarUndock	Transforms a docked Toolbar control into a floating Toolbar control.
---------------	--

### 2.3.4 Dashboard control functions

The following functions are used to manage Dashboard controls:

DashAddWidget	Adds and displays a widget (internal window) in a Dashboard control.
DashCount	Returns the number of widgets found in a Dashboard control.
DashDelete	Deletes a widget displayed in a Dashboard control. The Dashboard control is updated and a free location replaces the deleted widget.
DashDeleteAll	Deletes all widgets displayed in a Dashboard control.
DashDisplay	Refreshes the display of elements in a Dashboard control.
DashGetWidgetPosition	Returns the position of a widget found in a Dashboard control.
DashInfoXY	Used to find out: <ul style="list-style-type: none"><li>• whether a widget is found at a given position in a Dashboard control.</li><li>• which widget is found at a given position in a Dashboard control.</li></ul>
DashInitialConfiguration	Modifies the configuration of a Dashboard control (defined in the editor) before its first display.
DashLoadConfiguration	Restores the configuration of a Dashboard control.
DashMode	Opens or closes the Dashboard control edit mode.
DashMoveWidget	Changes the position of a widget found in a Dashboard control.
DashResizeWidget	Changes the dimensions of a widget found in a Dashboard control.
DashSaveConfiguration	Saves the configuration of a Dashboard control in string format.



2.3.5 Dockable Panel control functions

The following functions are used to manage Dockable Panel controls:

PanelActivate	Activates a panel when several panels are docked together.
PanelAttach	Attaches or changes the attach position of a Dockable Panel control.
PanelCreate	Creates a new Dockable Panel control in the current window.
PanelDetach	Detaches a Dockable Panel control to transform it into a floating window.
PanelLoadConfiguration	Restores the configuration of the Dockable Panel controls in a window.
PanelPosition	Retrieves the position of a Dockable Panel control.
PanelSaveConfiguration	Saves the current configuration of the Dockable Panel controls of a window.

2.3.6 Internal Window control functions

The following functions are used to manage the "Internal windows" controls:

ChangeSourceWindow	Dynamically changes the window displayed in an Internal Window control.  <b>Scrolling internal windows:</b> Modifies the window currently displayed. This modification is saved in the list of internal windows to scroll.
IWListAdd	Adds a new internal window to the list of internal windows browsed by an Internal Window control.
IWListCount	Returns the number of internal windows currently found in the list of internal windows browsed by an Internal Window control.
IWListDelete	Deletes an internal window found in the list of internal windows browsed by an Internal Window control.
IWListDeleteAll	Deletes all the internal windows from the list of internal windows browsed by an Internal Window control.
IWListDisplay	Refreshes the display of an "Internal window" control with automatic iteration on a data file or query.

IWListInsert	Inserts a new internal window in the list of internal windows used by an Internal Window control.
IWListNext	Displays the next internal window in the list of internal windows browsed by an Internal Window control.
IWListPosition	Displays the internal window corresponding to the specified position (in the list of internal windows browsed by an Internal Window control) or returns the position of the internal window currently displayed (in the list of internal windows browsed by an Internal Window control).
IWListPrevious	Displays the previous internal window in the list of internal windows browsed by an Internal Window control.
WinUsefulSize	Specifies the useful size of an internal window with automatic scrollbars. The useful size corresponds to the full size of the internal window: <ul style="list-style-type: none"><li>Width of the internal window.</li><li>Height of the internal window.</li></ul>

2.4 Graphic controls

2.4.1 Calendar control functions

These functions are used to handle the Calendar controls as well as the Date edit controls:

CalendarOpen	In a Date edit control, opens a popup window displaying a calendar.
CalendarPicker	Opens a modal dialog with a calendar to enter a date.
CalendarPosition	Returns the month displayed in a Calendar control.
CalendarSelect	Returns the date selected in a Calendar control.
CalendarSelectCount	Returns the number of days selected in a Calendar control.
CalendarSelectMinus	Deselects a date in a multi-selection Calendar control.
CalendarSelectPlus	Selects a date in a Calendar control.

2.4.2 Organizer control functions

The following functions are used to manage the Organizer controls:

iCalendarToOrganizer	Imports data in iCalendar format into an Organizer control.
OrganizerAddAppointment	Adds a new appointment to an Organizer control.
OrganizerBackgroundColor	Defines a background color for a time slot in an Organizer control.
OrganizerChangeMode	Modifies the time range displayed by an Organizer control (Day/Week/Month).
OrganizerDateVisible	Makes a day visible or invisible in an Organizer control.
OrganizerDeleteAll	Deletes all the appointments from an Organizer control.
OrganizerDelete-Appointment	Deletes an appointment from an Organizer control.
OrganizerDisplay	Reinitializes the content of an Organizer control from its data source (HFSQL data file for example).
OrganizerDisplayNextPeriod	Moves the time area displayed to the next area (next weeks for example) in an Organizer control.
OrganizerDisplayPreviousPeriod	Moves the time area displayed to the previous area (previous weeks for example) in an Organizer control.
OrganizerInfoXY	Returns for a given position in the Organizer control (coordinates of a point in the organizer): <ul style="list-style-type: none"><li>• the index of the appointment for the specified position.</li><li>• the date corresponding to the specified position.</li></ul>
OrganizerListAppointment	Returns: <ul style="list-style-type: none"><li>• the list of appointments found between two dates in an Organizer control,</li><li>• all appointments found in an Organizer control,</li><li>• a specific appointment (selected or hovered).</li></ul>
OrganizerMode	Returns the display mode of an Organizer control: day, week, month, ...
OrganizerPosition	Displays an Organizer control from a specified date or returns the date of the first day (or the date of the last day) displayed in an Organizer control.
OrganizerSelectedRange	Returns the start or end date and time of the time slot selected in an Organizer control.

OrganizerSelectMinus	Deselects an appointment from an Organizer control.
OrganizerSelectPlus	Selects an appointment in an Organizer control.
OrganizerTimeDisplayed	Defines the time range displayed by the Organizer control in Day mode or in Week mode.
OrganizerToiCalendar	Exports the appointments found in an Organizer control in iCalendar format.
OrganizerWorkingHour	Used to indicate the specific working hours per day in an Organizer control.
OrganizerZoom	Returns the time range displayed by an Organizer control.

2.4.3 Scheduler control functions

The following functions are used to manage Scheduler controls:

iCalendarToScheduler	Imports data in iCalendar format into a Calendar control.
SchedulerAddAppointment	Adds a new appointment into a Scheduler control.
SchedulerAddResource	Adds a new resource into a Scheduler control.
SchedulerBackgroundColor	Defines a background color for a time period in a Scheduler control (time period corresponding to lunch time, a public holiday, etc.).
SchedulerChangeMode	Modifies the period (which means the number of days) displayed by a Scheduler control.
SchedulerDateVisible	Makes a day visible or invisible in a Scheduler control.
SchedulerDeleteAll	Deletes all appointments and resources from a Scheduler control.
SchedulerDelete-Appointment	Deletes an appointment from a Scheduler control.
SchedulerDeleteResource	Deletes a resource from a Scheduler control.
SchedulerDisplay	Reinitializes the content of a Scheduler control from its data source (HFSQL file for example).
SchedulerDisplayNextPeriod	Moves the time area displayed on the next area (next weeks for example) in a Scheduler control.
SchedulerDisplay-PreviousPeriod	Moves the time area displayed on the previous area (previous weeks for example) in a Scheduler control.

SchedulerInfoXY	Returns the appointment, the resource or the date found at given position in a Scheduler control.
SchedulerListAppointment	Returns: <ul style="list-style-type: none"><li>• the list of appointments found between two dates in a Scheduler control,</li><li>• all the appointments found in a Scheduler control,</li><li>• a specific appointment (selected or hovered).</li></ul>
SchedulerMode	Returns the display mode of a Scheduler control: day, week, month, ...
SchedulerPosition	Displays a Scheduler control from a specified date or returns the date of the first day (or last day) displayed in a Scheduler control.
SchedulerPositionDateTime	Displays a Scheduler control from a specified date or returns the date of the first day (or last day) displayed in a Scheduler control.
SchedulerPositionResource	Displays a given resource in a Scheduler control or returns the position of first visible resource in a Scheduler control.
SchedulerSelectedRange	Returns the start or end date and time of the time slot selected in a Scheduler control.
SchedulerSelectMinus	Deselects an appointment from a Scheduler control.
SchedulerSelectPlus	Selects an appointment in a Scheduler control.
SchedulerTimeDisplayed	Defines the time range displayed by the Scheduler control in Day mode or in Week mode.
SchedulerToiCalendar	Exports the appointments from a Scheduler control in iCalendar format.
SchedulerWorkingHour	Used to indicate the specific working hours per day in a Scheduler control.
SchedulerWorking-HourResource	Used to indicate the specific working hours per day and per resource in a Scheduler control.

2.4.4 TimeLine control functions

The following functions are used to manage TimeLine controls:

TimelineAddEvent	Adds an event into a TimeLine control.
TimelineAddTrack	Adds a new track into a TimeLine control.
TimelineChangeMode	Modifies the period (which means the number of seconds, milliseconds, microseconds, ...) displayed by a TimeLine control.
TimelineDeleteAll	Deletes all events and tracks from a TimeLine control.
TimelineDeleteEvent	Deletes an event from a TimeLine control.
TimelineDeleteTrack	Deletes a track from a TimeLine control.
TimelineDisplay	Reinitializes the content of a TimeLine control from its data source (HFSQL data file for example).
TimelineListEvent	Returns: <ul style="list-style-type: none"><li>• the list of events found between two given times in a TimeLine control,</li><li>• all events of a TimeLine control,</li><li>• a specific event (selected or hovered).</li></ul>

2.4.5 Gantt Chart control functions

The following functions are used to handle a Gantt Chart column found in Table control (or in a TreeView Table control):

GanttAddHoliday	Declares a new holiday period for a row in a Gantt Chart column found in a Table (or TreeView Table) control.
GanttAddLink	Creates an "End to Start" link between two tasks displayed in a Gantt Chart column found in a Table (or TreeView Table) control.
GanttAddTask	Adds a task in a Gantt Chart column in a Table or TreeView Table control.
GanttChangeMode	Modifies the period (which means the number of days) displayed in a Gantt Chart column found in a Table (or TreeView Table) control.
GanttDateTimePosition	Configures the first day and the first time displayed in a Gantt Chart column (found in a Table or TreeView Table control).

GanttDeleteAll	Deletes all the tasks from a Gantt Chart column found in a Table (or TreeView Table) control.
GanttDeleteLink	Removes links from a Gantt Chart column in a Table or TreeView Table control.
GanttDeleteTask	Removes a task from a Gantt Chart column in a Table or TreeView Table control.
GanttGetTask	Retrieves a task from its identifier in a Gantt Chart column (found in a Table or TreeView Table control).
GanttListLink	Lists the links found in a Gantt Chart column found in a Table (or TreeView Table) control.
GanttListTask	Returns: <ul style="list-style-type: none"><li>• the list of tasks included between two given dates in a Gantt Chart column,</li><li>• all the tasks found in a Gantt Chart column,</li><li>• a specific task (selected or hovered).</li></ul>
GanttOrganizeTask	Reorganizes the tasks of a Gantt chart according to the constraints (dependencies and resources).
GanttTimeDisplayed	Defines the time range displayed in a column of a Table or TreeView Table control of type Gantt Chart.
GanttWorkingHour	Overloads the working hours of a row in a Gantt Chart column found in a Table (or TreeView Table) control.
GanttWorkingWeekDay	Indicates if a day of the week is a working day in a Gantt Chart column in a Table or TreeView Table control.

2.4.6 Word Processing control functions

The following functions are used to manage Word Processing controls:

DocAdd	Adds: <ul style="list-style-type: none"><li>• at the end of a Word Processing document: an element, a document, a paragraph, a text, an image or a fragment.</li><li>• at the end of a fragment: an element, a text or an image.</li></ul>
DocChangeZOrder	Changes the Z-order of an Image or Text Area element in a DocX document.
DocClose	Closes a docx file and frees the document.
DocDelete	Deletes an element from a document. This element is deleted from its associated document.

DocDeleteTableOfContents	Deletes a table of contents from a document.
DocEndModification	Signals the end of grouping for a set of operations in a single event (management of "undo/redo").
DocExitEdit	Exits from the current edit. The entry in edit was performed by <b>DocGoIntoEdit</b> .
DocFind	Finds a character string: <ul style="list-style-type: none"><li>• in an entire Word Processing document.</li><li>• in a fragment.</li></ul>
DocFindElementByName	Searches for an element by its name in a Word Processing document.
DocGoIntoEdit	Enters in edit in the Word Processing control in a specific document section: body, header or footer.
DocInfoXY	For a given position in a Word Processing control (coordinates of a control point), returns one of the following: <ul style="list-style-type: none"><li>• the number of the corresponding page,</li><li>• the corresponding X-position (in mm),</li><li>• the corresponding Y-position (in mm)</li></ul>
DocInsert	Inserts an object into a Word Processing document or replaces the content of the current fragment.
DocInsertTable	Inserts a table into a Word Processing document.
DocInsertTableOfContents	Inserts a table of contents in a Word Processing document.
DocOpen	Opens a Word Processing file (docx format) in Read/Write mode. The docx file is opened and locked until it is closed.
DocPosition	Triggers the scroll of Word Processing control in order for the requested text to be entirely displayed.
DocPositionTopPage	Returns the position of the first character in a given page in a Text Processing control.
DocPrint	Prints a document in "docx" format.
DocRedo	Restores the last undone action on a Word Processing document.
DocRedoCount	Returns the number of canceled actions that can be redone on a Word Processing document.

DocReplace	Finds all the instances of a character string and replaces them with another string: <ul style="list-style-type: none"><li>in an entire Word Processing document.</li><li>in a fragment.</li></ul>
DocSave	Saves a document as a "docx" file.
DocSelect	Selects a fragment in a Word Processing control.
DocStartModification	Signals the start of grouping for a set of operations on a document in a single event (management of "undo/redo").
DocToHTML	Converts a Word Processing document into an HTML file.
DocToImage	Exports a page found in a Word Processing document in image format.
DocToPDF	Converts a Word Processing document into a PDF file.
DocToText	Creates a character string from the data: <ul style="list-style-type: none"><li>found in a Word Processing control or in a Word Processing document.</li><li>found in a paragraph of a Word Processing control.</li></ul>
DocUndo	Undoes the last action performed on a Word Processing document.
DocUndoCount	Returns the number of actions that can be undone on a document.
DocUndoRedoDeleteAll	Deletes the entire history of Undo/Redo mechanism for a Word Processing document.
DocUpdateFormula	Updates the formulas in a document.
DocUpdateTableOfContents	Updates the table of the contents of a document.

2.4.7 Spreadsheet control functions

The following functions are used to handle the Spreadsheet control:

SpreadsheetAddData	Adds a data into a Spreadsheet control found in a window.
SpreadsheetAddFormula	Adds a calculation formula into a Spreadsheet control found in a window.
SpreadsheetAddImage	Adds an image into the current worksheet of a Spreadsheet control.

SpreadsheetAddName	Specifies a name to identify a cell found in a Spreadsheet control.
SpreadsheetAddWorksheet	Adds a worksheet into a Spreadsheet control found in a window.
SpreadsheetAllowProcedure	Allows the use of a WLanguage function/procedure in a formula of a Spreadsheet control.
SpreadsheetBorderSelection	Modifies the border of selected cells in a Spreadsheet control found in a window.
SpreadsheetClose	Closes the XLSX document that was opened by <b>SpreadsheetLoad</b> .
SpreadsheetControlFormula	Allows you to use a control in order to create a custom formula bar for the Spreadsheet control.
SpreadsheetDeleteAll	Deletes the entire data from a Spreadsheet control found in a window (cells, formulas, size and header layouts).
SpreadsheetDeleteColumn	Deletes a column from a Spreadsheet control.
SpreadsheetDeleteImage	Deletes an image from the current worksheet in a Spreadsheet control.
SpreadsheetDeleteName	Deletes one of the names used to identify a cell found in a Spreadsheet control.
SpreadsheetDeleteRow	Deletes a row from a Spreadsheet control.
SpreadsheetDelete-Worksheet	Deletes a worksheet from a Spreadsheet control found in a window.
SpreadsheetEndModification	Signals the end of grouping for a set of operations on a Spreadsheet control in a single event (management of "undo/redo").
SpreadsheetFreeze	Used to freeze the display of a row or column in a Spreadsheet control.
SpreadsheetGetFormula	Retrieves the calculation formula of a cell found in a Spreadsheet control of a window.
SpreadsheetGetMerge	Used to find out whether a cell is merged with other cells and to get the name of merged cells.
SpreadsheetGetSelection	Retrieves the list of selected cells in a Spreadsheet control found in a window.
SpreadsheetGetSortInfo	Returns various information about the sort performed in a Spreadsheet control.

SpreadsheetInputFunction	Inserts a function name at cursor position in the editable cell of a Spreadsheet control found in a window.
SpreadsheetInsertColumn	Inserts an empty column into a Spreadsheet control found in a window.
SpreadsheetInsertRow	Inserts an empty row into a Spreadsheet control found in a window.
SpreadsheetInsert-Worksheet	Inserts a worksheet into a Spreadsheet control found in a window.
SpreadsheetListComment	Lists all the comments on the current worksheet of a Spreadsheet control.
SpreadsheetListImage	Lists the images found in the current worksheet of a Spreadsheet control.
SpreadsheetListName	Returns the list of names that identify a cell in a Spreadsheet control.
SpreadsheetListWorksheet	Returns the list of worksheets found in a Spreadsheet control.
SpreadsheetLoad	Loads a file in XLSX format in a Spreadsheet control found in a window.
SpreadsheetMergeCell	Merges specified or selected cells in an Spreadsheet control.
SpreadsheetModifyName	Modifies the name used to identify a cell in a Spreadsheet control.
SpreadsheetMoveWorksheet	Moves a worksheet within a Spreadsheet control in a window.
SpreadsheetPosition	Displays the specified cell in a Spreadsheet control found in a window: <ul style="list-style-type: none"><li>• either the visible area of Spreadsheet control is moved to display the cell.</li><li>• or the cell is displayed in the upper-left corner of visible area of Spreadsheet control.</li></ul>
SpreadsheetRecalculateAll	Recalculates all formulas of a Spreadsheet control found in a window (including the formulas that use WLanguage functions/procedures).
SpreadsheetReplace	Finds and replaces a value in a Spreadsheet control.

SpreadsheetRestore-PositionAndSelection	Restores: <ul style="list-style-type: none"><li>• the scrollbar position in a Spreadsheet control,</li><li>• the current worksheet,</li><li>• the elements selected in the current worksheet of Spreadsheet control.</li></ul> These elements have been saved by <b>SpreadsheetSavePositionAndSelection</b> .
SpreadsheetSave	Saves the content of a Spreadsheet control in a file in XLSX format.
SpreadsheetSavePosition-AndSelection	Saves: <ul style="list-style-type: none"><li>• the scrollbar position in a Spreadsheet control for the current worksheet,</li><li>• the current worksheet in the Spreadsheet control,</li><li>• the selected elements in the Spreadsheet control for the current worksheet.</li></ul>
SpreadsheetSeek	Seeks a value in the cells of a Spreadsheet control found in a window.
SpreadsheetSelectComment	Allows you to: <ul style="list-style-type: none"><li>• know if a comment is selected in a Spreadsheet control,</li><li>• select or deselect a comment in a Spreadsheet control.</li></ul>
SpreadsheetSelectMinus	Deselects one or more cells in a Spreadsheet control found in a window.
SpreadsheetSelectPlus	Selects one or more cells in a Spreadsheet control found in a window.
SpreadsheetSelect-Worksheet	Selects a worksheet (identified by its name) in a Spreadsheet control.
SpreadsheetSortSelection	Sorts the values selected in a Spreadsheet control.
SpreadsheetStart-Modification	Indicates the start of a group of operations on a Spreadsheet control in a single event (management of "undo/redo").
SpreadsheetStyleSelection	Returns or modifies the style of selected cells in a Spreadsheet control found in a window.
SpreadsheetTypeAnd-MaskSelection	Returns or modifies the type and mask of selected cells in a Spreadsheet control.
SpreadsheetUnmergeCell	Supprime la fusion des cellules d'un champ Tableur.



### 2.4.8 PDF Reader control functions

The following functions are used to manage PDF Reader controls:

PDFReaderClose	Closes and frees the PDF file displayed in a PDF Reader control.
PDFReaderGoToBookmark	Sets the position directly on the desired bookmark in the PDF document.
PDFReaderHighlight	Finds a text in the document displayed by a PDF Reader control and highlights the occurrences of this text in the control.
PDFReaderInfoXY	For a given position in a PDF Reader control (coordinates of a control point), returns one of the following: <ul style="list-style-type: none"><li>• the page number for the specified position,</li><li>• the X-position (in mm) for the specified position,</li><li>• the Y-position (in mm) for the specified position.</li></ul>
PDFReaderOpen	Opens a PDF file displayed in a PDF Reader control.
PDFReaderRotatePage	Rotates a page displayed in the PDF Reader control.
PDFReaderSave	Saves the content of a PDF Reader to a file.
PDFReaderSeekFirst	Initializes the search for a text in a PDF file displayed by the PDF Reader control and positions on the first occurrence found (from the current page).
PDFReaderSeekNext	Sets the position on the next result of a search in the content of a PDF Reader control.
PDFReaderSeekPrevious	Positions on the previous result of a search initiated by <b>PDFReaderSeekFirst</b> in the content of a PDF Reader control.

### 2.4.9 Image Editor control functions

The following functions are used to manage the Image Editor control:

PicClose	Closes the image found in an Image Editor control.
PicCreate	Creates an image in the Image Editor control.
PicMergeLayer	Merges the layers of a WDPic image.
PicOpen	Opens an image in an Image Editor control.

PicRectangularSnip	Captures a rectangular area defined with the cursor and retrieves the image in a variable of type <b>Image</b> .
PicRectangularSnipOCR	Runs OCR and extracts the text from a rectangular area defined with the mouse.
PicResize	Resizes a WDPic image.
PicRotation	Rotates a WDPic image.
PicSave	Saves the image found in an Image Editor control or in a WDPic variable.

### 2.4.10 Kanban control functions

The following functions handle Kanban controls:

KanbanAddCard	Adds a card to a list in a Kanban control.
KanbanAddList	Adds a new list to a Kanban control.
KanbanCount	Returns the number of cards or lists in a Kanban control.
KanbanDeleteAllCard	Deletes all cards from a Kanban control or Kanban List control.
KanbanDeleteCard	Deletes a card from a Kanban control or from a Kanban List control.
KanbanDisplay	Refresh the content of a Kanban control or a card in a Kanban control.
KanbanFilter	Refresh the content of a Kanban control or a card in a Kanban control.
KanbanInfoXY	Indicates which element of a Kanban control is at a given position.
KanbanInsertCard	Inserts a card in a Kanban list.
KanbanListIndex	Returns the index of a list in a Kanban control.
KanbanMoveCard	Moves a card in a Kanban control.
KanbanMoveList	Move a list in a Kanban control.
KanbanPositionList	Allows you to scroll through a list of a Kanban control to display a specific card.

2.4.11 Camera control functions

The following functions are used to manage the Camera control:

CameraCapture	Captures the images received from the camera: <ul style="list-style-type: none"><li>• as an image: takes a photo.</li><li>• as a video: records a video.</li></ul>
CameraChangeDriver	Connects the "Camera" control to a specific driver for video capture.
CameraConfigure	Allows you to configure the video capture driver.
CameraDisplay	Displays the video being captured by a web camera or by the camera of a mobile device in a "Camera" control.
CameraGenerateThumbnail	Generates a thumbnail of the specified photo or video.
CameraListDriver	Lists the video capture drivers installed on the current computer.
CameraListParameter	Lists the values supported by the device camera for a given parameter.
CameraParameter	Returns or modifies the value of a parameter for the device camera.
CameraPhoto	Captures a photo via a Camera control in an Android or iOS application.
CameraRequestPermission	Requests permission to use the device's camera.
CameraRunApp	Starts the device's native camera application to record a video or to take a photo.
CameraStartDecoding	Starts decoding bar codes in a Camera control.
CameraStatus	Returns the status of a Camera control.
CameraStop	Stops the video preview or video recording in a Camera control.
CameraStopDecoding	Stops decoding bar codes in a Camera control.
CameraVideoStart	Starts recording a video via a Camera control in an Android or iOS application.
CameraVideoStop	Stops the video recording previously started by <b>CameraVideoStart</b> .

2.4.12 Multimedia control functions

The Multimedia functions are used to handle media files associated with Multimedia controls.

MultimediaBeginning	Positions at the beginning of the media file played in the Multimedia control.
MultimediaClose	Closes the media file currently played in the Multimedia control.
MultimediaControlVisible	Manages the visibility of control buttons (Play, Pause, Fast Forward, ...) found in the Multimedia control.
MultimediaDuration	Returns the size of the media file currently played (in milliseconds) in the Multimedia control.
MultimediaEnd	Positions at the end of media file played in the Multimedia control.
MultimediaHeight	Returns the height (in pixels) of the source image of the multimedia file played in the Multimedia control.
MultimediaOpen	Opens a media file in a Multimedia control.
MultimediaPause	Pauses a media file played in a Multimedia control.
MultimediaPlay	Plays a multimedia file in a Multimedia control.
MultimediaPosition	Returns or modifies the current position of the media file currently played in the Multimedia control.
MultimediaStatus	Indicates the status of the multimedia file played in the Multimedia control.
MultimediaStop	Stops playing a media file found in a Multimedia control.
MultimediaVolume	Returns or modifies the sound volume of the multimedia file played in a Multimedia control.
MultimediaWidth	Returns the width (in pixels) of the source image of the multimedia file played in the Multimedia control.

2.4.13 TreeMap control functions

The following functions are used to manage TreeMap controls:

TreeMapAdd	Adds an element into a TreeMap control.
TreeMapCount	Returns the number of elements found in a TreeMap control.

TreeMapDelete	Deletes an element from a TreeMap control.
TreeMapDeleteAll	Deletes all elements from a TreeMap control.
TreeMapInfoXY	Returns the index of the element displayed at a given position in a TreeMap control.
TreeMapSearch	Finds the index of an element in a TreeMap control from the element path.
TreeMapSeek	Finds the index of an element in a TreeMap control from the element path.
TreeMapSelectGroup	Visually selects a group of elements in a TreeMap control.

2.4.14 Map control functions

The following functions are used to manage Map controls:

MapAddImage	Adds a new image to the map displayed in a Map control.
MapAddItinerary	Adds an itinerary onto a Map control.
MapAddMarker	Adds a new marker onto a map displayed in a Map control.
MapAddShape	Adds a new shape (polygon, line, circle) to the map displayed in a Map control.
MapChangeStyle	Changes the style of the map displayed in the Map control.
MapClosePopup	Closes the popup of a marker displayed in the Map control.
MapConfigureGglAPI	Declares additional parameters for the Map control that will be transmitted to the Google API.
MapDeleteAll	Deletes the following from a Map control: all markers, all shapes, all images.
MapDeleteImage	Deletes an image drawn on the map displayed in a Map control.
MapDeleteItinerary	Deletes an itinerary from a Map control.
MapDeleteMarker	Deletes a marker from a map displayed by a Map control.
MapDeleteShape	Deletes a shape drawn on the map displayed in an Map control.
MapDisplayArea	Moves the map displayed in the Map control so that the specified geographic area can be seen in its entirety.

MapDisplayPopup	Opens the popup of a marker displayed in a Map control.
MapDisplayPosition	Centers the map displayed in a Map control on a geographical position or on a specific location.
MapEndOfMove	Stops following the device movement in a Map control.
MapExecuteJS	Runs the native code of Google Maps API in a Map control in JavaScript.
MapFollowMovement	Displays the current location of the device in a Map control and updates the location as it moves.
MapGetImage	Retrieves the image of a map displayed in a Map control. The markers and the itineraries displayed on this map will also be found in the image.
MapGetJSObject	Returns the instance of the API of Map control.
MapGetPosition	Returns the geographical position of the point located in the center of the map currently displayed in a Map control.
MapGetStyle	Returns the style used by the map displayed in the Map control.
MapInfoPosition	Returns the coordinates (in pixels) of the point corresponding to a geographical position (latitude and longitude).
MapInfoXY	Returns the geographical position (latitude and longitude) corresponding to a point in a Map control.
MapItineraryDistance	Calculates the distance of an itinerary in a Map control.
MapLicenseBing	Indicates the "Bing map" license key that will be used in the Map controls used in Universal Windows 10 App mode.
MapLicenseGgl	Indicates the "Google Maps API" license key that will be used in the Map controls (of WINDEV or WEBDEV).
MapLimitArea	Limits the geographic area displayed in the Map control.
MapModifyImage	Modifies an image drawn on the map displayed in a Map control.
MapModifyMarker	Modifies a marker displayed in a Map control.
MapModifyShape	Modifies a shape drawn on the map displayed in a Map control.
MapPopupDisplayed	Indicates whether the popup of a marker in a Map control is displayed.

2.4.15 Diagram management functions

The diagram management functions are as follows:

DiagramEndModification	Indicates the end of a series of actions performed in a Diagram document in a single event ("undo/redo").
DiagramGroup	Creates a group from the elements selected in the Diagram Editor control.
DiagramInfoXY	Indicates the presence of a shape at a given position in the Diagram Editor control.
DiagramLoad	Loads a diagram into a Diagram variable from a ".wddiag" file, a memo or a binary buffer.
DiagramLoadLibrary	Loads a diagram file and creates a library of preset shapes from it.
DiagramSave	Saves a diagram as a "diag" file.
DiagramSearch	Searches for shapes containing a specific text in a diagram.
DiagramSelectMinus	Deselects a shape in a Diagram Editor control.
DiagramSelectPlus	Selects a shape in a Diagram Editor control.
DiagramShapeToImage	Creates an image from a shape in a diagram (the shape can be displayed or not in the diagram).
DiagramStartModification	Indicates the start of a series of actions performed in a Diagram document in a single event ("undo/redo").
DiagramToImage	Exports a diagram to a variable of type <b>Image</b> .
DiagramToSVG	Exports a diagram to an SVG image.
DiagramUngroup	Dissociates the different shapes of a group in a Diagram Editor control.

2.4.16 Organization Chart control functions

The following functions are used to manage Organization Chart controls:

OrgAdd	Adds a new root element into an Organization Chart control.
OrgAddChild	Adds a child element into an Organization Chart control.
OrgDelete	Deletes an element and its "children" from an Organization Chart control.

OrgDeleteAll	Deletes all elements from an Organization Chart control.
OrgDisplay	Reinitializes the content of an Organization Chart control from its data source (HFSQL data file for example).
OrgInfoXY	Returns the index of the element found at a given position in an Organization Chart control.
OrgModify	Modifies an element in an Organization Chart control.
OrgOccurrence	Returns the number of elements found in an Organization Chart control.
OrgSaveImage	Creates an image representing the entire Organization Chart control.
OrgSelect	Returns the index of the selected element in an Organization Chart control.
OrgSelectMinus	Deselects an element from an Organization Chart control.
OrgSelectPlus	Selects an element in an Organization Chart control.

2.5 Other controls

2.5.1 HTML Display control functions

The following functions are used to manage HTML Display controls:

HTMLControlValue	Used to identify or modify a control in an HTML form contained in an HTML Display control.
HTMLExecuteBrowser-Procedure	Executes a WLanguage browser procedure in an HTML Display control that displays a page created with WEBDEV.
HTMLExecute-ControlProcess	Executes a WLanguage event associated with a control in an HTML page displayed by an HTML Display control.
HTMLFromEmail	Imports the HTML content of an email and its images into an HTML Display control. The images contained in the email are also displayed in the control.
HTMLNavigate	Starts navigation in an HTML Display control.
HTMLOnErrorJS	Specifies the procedure to call if an error occurs while executing JavaScript code in an HTML Display control.
HTMLValidateForm	Validates a form in an HTML page displayed in an HTML Display control.

2.5.2 HTML Editor control functions

The following WLanguage functions are used to manage HTML Editor controls:

HTMLEditorExecuteJS	Executes JavaScript code in an HTML Editor control.
HTMLEditorFormatSelection	Modifies the style of the selected elements in an HTML Editor control.
HTMLEditorFromEmail	Imports the HTML content of an email and its images into an HTML Editor control. The images contained in the email are also displayed in the control.
HTMLEditorGet	Asynchronously retrieves the value of one of the properties of an HTML Editor control.
HTMLEditorGetDocument	Asynchronously retrieves the HTML document being displayed in an HTML Edit control.
HTMLEditorGetSelection	Asynchronously retrieves the current selection from an HTML Editor control.
HTMLEditorInsert	Inserts text or HTML code at the cursor position in an HTML Editor control.
HTMLEditorInsertTable	Inserts a table at the cursor position in an HTML Editor control.
HTMLEditorReplace	Replaces one string with another in an HTML Editor control.
HTMLEditorSearch	Searches for a string in an HTML Editor control.
HTMLEditorSelect	Selects a text fragment in an HTML Editor control.
HTMLEditorToEmail	Exports the content of an HTML Editor control to an Email variable (to send the email).
HTMLEditorToPDF	Exports the content of an HTML Editor control to a PDF file.

2.5.3 Action Bar control functions

The following functions are used to manage the Action Bar:

ActionBarDelete-SearchHistory	Deletes the history from the search control found in the action bar of a window.
ActionBarSearchVisible	Allows you to get and change the visibility status of the search control associated with the Action Bar in the current window.

2.5.4 Sliding Banner control functions

The following functions are used to manage Sliding Banner controls:

BannerFirst	Displays the first plane of a Sliding Banner control.
BannerLast	Displays the last plane of a Sliding Banner control.
BannerNext	Displays the next plane of a Sliding Banner control.
BannerPrevious	Displays the previous plane of a Sliding Banner control.
BannerStartSliding	Enables the automatic slide on a Sliding Banner control.
BannerStopSliding	Stops the automatic slide on a Sliding Banner control.

2.5.5 Captcha control functions

The following functions are used to manage Captcha controls:

CaptchaDisplay	Displays a new Captcha in a Captcha control.
CaptchaVerify	Checks whether the value typed by the user corresponds to the string displayed in a Captcha control.

2.5.6 Breadcrumb control functions

The following functions are used to manage Breadcrumb controls:

BreadcrumbAddLink	Adds a link into a Breadcrumb control.
BreadcrumbDeleteAll	Deletes all links from a Breadcrumb control.

BreadcrumbDeleteLink	Deletes a link from a Breadcrumb control.
BreadcrumbInsertLink	Inserts a link into a Breadcrumb control.
BreadcrumbModifyLink	Modifies a link in a Breadcrumb control.

2.5.7 Progress Bar control functions

The following functions are used to manage Progress Bar controls in a WEBDEV site:

ProgressBarDisable	Stops the periodic refresh of a Progress Bar via Ajax.
ProgressBarEnable	Enables the refresh of a Progress Bar via Ajax.
ProgressBarExecute	Starts a long browser process and fill a Progress Bar via Ajax according to the progress of this process.

2.5.8 Repositionable Note control functions

The following functions are used to manage Repositionable Note controls through programming:

NoteCreate	Creates a Repositionable Note control in the curenrnt window through programming.
NoteDelete	Deletes a repositionable note found in the current window.
NoteLoadAll	Imports and displays the repositionable notes of the current window saved beforehand by <b>NoteSaveAll</b> .
NoteSaveAll	Saves the repositionable notes of the current window in a file.

2.5.9 Xaml control functions

The following functions are used to manage Xaml controls:

XamlDisable	Simulates the absence of .Net 3.0 framework.
XamlEvent	Associates a procedure written in WLanguage with an event of Xaml control.
XamlProperty	Used to find out and modify the value of a .Net property for an Xaml control or for a Carousel control.

2.6 General control functions

2.6.1 Control management functions

The following functions are used to manage the controls:

ActiveXEvent	Associates a procedure written in WLanguage with an event of ActiveX control.
AddLink	Adds a link to a text section in an edit control.
AssistedInputAdd	Adds a row to the list of values available in the autocomplete suggestions of an Edit control.
AssistedInputClose	Closes the list of values proposed by the assisted input for an edit control.
AssistedInputDeleteAll	Clears the list of values proposed for the assisted input of an edit control.
AssistedInputOpen	Opens the list of values proposed by the assisted input for an edit control.
AttachPointAdd	Defines an attach point for one of the two extremities of a Wire control.
AttachPointDeleteAll	Deletes all the connection points for a control.
ChangeStyle	Modifies the style of a control dynamically.
ControlAlias	Gets, modifies or restores the data binding to a control or group of controls.
ControlClone	Creates a new control from an existing control.
ControlCreate	Creates a new control (of any type) in a window through programming.
ControlCurrent	Returns the name of the control currently in edit.
ControlDelete	Permanently deletes a control from a window, page or report.
ControlEnabled	Enables a control or a group of controls in a window. An animation can be set on the controls during this operation.
ControlError	Customizes the value displayed in the controls when the value cannot be displayed.
ControlExist	Checks the existence of a control (or group of controls) in a window, in a page or in a report.



ControlFirst	Indicates the name of the first field specified in the tab order for the specified window.
ControlGrayed	Grays a control or a group of controls. An animation can be set on the controls during this operation.
ControlInOutReason	Indicates the origin of the entry in the control.
ControlInfoXY	Returns the name of the control located at a given position.
ControlInvisible	Makes a control (or a group of controls) invisible in a window. An animation can be set on the controls during this operation.
ControlNoSpace	Specifies whether the space characters on the right of the value found in the control are deleted.
ControlNext	Returns the name of the next control in input mode.
ControlOver	Identifies the control hovered over with the mouse cursor.
ControlPopupOwner	Identifies the control on which the popup menu was opened.
ControlPrevious	Specifies the name of control that was in edit before the current control.
ControlTypeInfo	Returns the icon corresponding to the specified type of control.
ControlVisible	Makes a control (or a group of controls) visible in a window. During this operation, an animation can be performed on the controls.
Count	Returns the number of occurrences of a control in a window, in a page or in a group of controls.
ConfigureSpellCheck	Configures the spelling checker for the edit controls and for the Table control columns in the whole application.
CurrentColumn	Returns: <ul style="list-style-type: none"><li>• the index of the current indexed column in a Table control (based on a data file or populated programmatically).</li><li>• the number of the column currently in edit mode in the specified Table control (based on a data file or populated programmatically).</li></ul>
CoordinateImage-ControlToImage	Converts the coordinates of a rectangle in the Image control to the corresponding coordinates in the image displayed in the control.

CoordinateScreenToEditor	Converts a size or a position specified in the native dpi of the screen on the device where the application is run to the reference dpi in the editor (160 dpi).
CoordinateEditorToScreen	Converts a size or a position specified in the reference dpi in the editor (160 dpi) to the native dpi of screen on the device where the application is run.
CoordinateImageTo-ImageControl	Converts the coordinates of a rectangle in the image to the corresponding coordinates in the Image control.
CurrentIndex	Depending on the specified control, returns: <ul style="list-style-type: none"><li>• the number of the current row displayed in the current Table control,</li><li>• the number of the option with focus in a Check Box control,</li><li>• the number of the current row in the current Looper control,</li><li>• the index of the current indexed control.</li></ul>
DeleteParameter	Deletes a parameter (or a set of parameters) saved either by <b>SaveParameter</b> , or automatically via the persistence of data in the controls.
DictionaryLoad	Indicates and loads the dictionary corresponding to a given language.
EditCaret	Modifies the standard hover cursor for all the edit controls of the application.
EnumControl	Returns the name of Nth control found in the specified element.
HTMLNavigate	Starts navigation in an HTML Display control.
InitParameter	Initializes the management of persistent values.
LinkColor	Modifies the color of links in the edit controls and in the Static controls.
LoadParameter	Reads a persistent value.
NextIndex	Returns the number of the next row displayed in the current Table control.
OpenPopupMenu	Automatically opens a popup menu for the current control or window.
ParentObject	Identifies the "parent" of a graphic object: control, column, window, page, ...

Reset	Reinitializes: <ul style="list-style-type: none"><li>• all the controls found in the current window or page (including the groups of controls, the supercontrols and the internal windows).</li><li>• the controls found in a group of controls, a supercontrol, an internal window or an internal page.</li></ul>
ReturnToCapture	Stops the current process and forces the input in the specified control, window or page.
SaveParameter	Saves a persistent value in the registry or in another file specified by <b>InitParameter</b> .
ScreenFirst	Defines the next control that will be in edit in the current window.
ScrollBarShow	Displays (or not) a scrollbar in a WINDEV control (edit control or list box) even if no scrollbar was specified during the control description.
ScrollbarIsVisible	Used to find out whether the scrollbar of a control in a WINDEV application is visible.
ScrollbarPosition	Used to manage the position of scrollbars: <ul style="list-style-type: none"><li>• Modifies or retrieves the position of one of the horizontal or vertical scrollbars found in a control of a WINDEV or WINDEV Mobile application.</li><li>• Retrieves the position of one of the horizontal or vertical scrollbars found in a control of a WEBDEV site.</li></ul>
ScrollbarMaxPosition	Retrieves the maximum position of one of the horizontal or vertical scrollbars found in a control of a WINDEV application.
SelectionColor	Changes the characteristics of the selection bar in a Table control, Combo Box control or List Box control.
SetFocus	Set focus: <ul style="list-style-type: none"><li>• to a control (found in a window, in a page or in a frameset).</li><li>• to a window.</li></ul>
ShapePolygon	Modifies the shape currently displayed in a Shape control to get a custom shape (by specifying the coordinates of polygon points).
ShowToolTip	Enables or disables the mechanism for displaying the control tooltips.

TextEllipsis	Returns the text with ellipsis displayed in an Edit, Combo Box or Static control.
TextHeight	Calculates the height (in pixels) of the text displayed in an edit control, in a Static control or in a Combo Box control.
TextWidth	Calculates the width (in pixels) of the text displayed in an edit control, in a Combo Box control or in a Static control.
ToolTipColor	Modifies the colors used for displaying the tooltips.
ToolTipDelay	Used to modify: <ul style="list-style-type: none"><li>• the timeout before displaying the tooltips.</li><li>• the display duration of the tooltips.</li></ul>
ToolTipMargin	Modifies the margin of the text in the tooltips.
ToolTipMaxWidth	Modifies the maximum width of tooltips.

2.6.2 Window control search functions

The following functions are used to find a text in the different window controls:

ControlFindHighlight	Highlights, in a window or control, all the captions and texts corresponding to a given text.
ControlFindTextFirst	Finds and selects the first element corresponding to the sought text in a window or control.
ControlFindTextNext	Finds and selects the next element corresponding to the sought text in a window or in a control.
ControlFindTextPrevious	Finds and selects the previous element corresponding to the sought text in a window or control.

2.6.3 Assisted input management functions

The following functions are used to program assisted input in edit controls:

AssistedInputAdd	Adds a row to the list of values available in the autocomplete suggestions of an Edit control.
AssistedInputClose	Closes the list of values proposed by the assisted input for an edit control.
AssistedInputConfigure	Used to configure the filter and the opening of assisted input.

AssistedInputDeleteAll	Clears the list of values proposed for the assisted input of an edit control.
AssistedInputOpen	Opens the list of values proposed by the assisted input for an edit control.

2.6.4 RTF control functions

The following functions are used to manage the RTF edit controls:

RTFAdd	Adds a character string (in RTF or not) at the end of RTF edit control, RTF control in a report or character string containing a text in RTF.
RTFAddLink	Adds a link to a text section in an RTF Edit control.
RTFInsert	Inserts a character string in RTF into the content of an RTF control (found in a window or in a report) or into the content of a character string in RTF.
RTFLoad	Loads a file in RTF in an RTF control (found in a window or in a report).
RTFReplace	Replaces or inserts a character string in an RTF control (found in a window or in a report) or in a string in RTF (containing a text in RTF).
RTFSearch	Finds a character string in an RTF control (found in a window or in a report) or in a character string containing text in RTF format.
RTFSelection	Changes or lists the attributes of the text currently selected in an RTF control (found in a window or in a report).
RTFToHTML	Returns the HTML string corresponding to the specified RTF string.
RTFToText	Returns the text without the formatting characteristics specific to the RTF format.

2.6.5 Token management functions

These are the token management functions in the edit controls

EditAddToken	Adds a token to to a "Text token" Edit control.
EditDeleteAllToken	Deletes all the tokens from a "Text token" edit control.

EditDeleteToken	Deletes a token from a "Text token" edit control.
EditInsertToken	Inserts a token into a "Text token" edit control.
EditTokenCount	Returns the number of tokens present in a "Text token" edit control.

2.6.6 Animation management functions

These functions can be used to handle the animation of a control, a group of controls or a window:

AnimationEnabled	Enables or disables the animations in the controls.
AnimationGetParameter	Reads the value of one of the animation parameters defined for an Image or Static control.
AnimationMinFPS	Configures the minimum number of images per second in order for the animations (transition effects) to be enabled in the controls.
AnimationPlay	Plays an animation when a control, a group of controls or a window is modified.
AnimationPlayOn-ControlProperty	Animates the modification of an integer property between 2 values.
AnimationPrepare	Prepares the animation on a control, a group of controls or a window.
AnimationSetMode	Changes the type of animation: <ul style="list-style-type: none"><li>• for an image in an Image control.</li><li>• for a text in a static control.</li></ul>
AnimationSetParameter	Changes one of the animation parameters used for: <ul style="list-style-type: none"><li>• an animated image (in an Image control),</li><li>• an animated text (in a Static control).</li></ul>
ControlEnabled	Enables a control or a group of controls in a window. An animation can be set on the controls during this operation.
ControlGrayed	Grays a control or a group of controls. An animation can be set on the controls during this operation.
ControlInvisible	Makes a control (or a group of controls) invisible in a window. An animation can be set on the controls during this operation.

ControlVisible	Makes a control (or a group of controls) visible in a window. During this operation, an animation can be performed on the controls.
WinAnimationClosing	Modifies the type and duration of the animation used when closing the application windows.
WinAnimationOpening	Modifies the type and duration of animations used when opening the application windows.
WinAnimationNext	Modifies the type and duration of animations used when opening and closing the next window.

## 3. Basic functions

### 3.1 Character string management functions

The following functions handle character strings:

AnsiToOem	Converts an ANSI string (Windows) to an OEM string (DOS).
AnsiToUnicode	Converts: <ul style="list-style-type: none"><li>• an ANSI string (Windows) to a UNICODE string.</li><li>• a buffer containing an ANSI string (Windows) to a buffer containing a UNICODE string.</li></ul>
Asc	Returns the ASCII code of the specified character.
BufferToHexa	Converts a buffer to a hexadecimal string (e.g.: "4A 5B 00").
BufferToInteger	Extracts an integer from a binary buffer at a given position.
BufferToPhrase	Converts a buffer to a mnemonic phrase using the BIP39 algorithm.
BufferToReal	Extracts a real from a binary buffer at a given position.
CapitalizeEachWord	Capitalizes the first letter of all words in a string.
Charact	Returns the character corresponding to the specified ASCII code.
CharactType	Returns information about the type of character.
CharactTypeCount	Returns the number of characters matching a given type.
CharactUnicode	Returns the character corresponding to the specified Unicode code.
CommonLength	Returns the number of common characters between two strings (from the beginning or from the end of the string).
Complete	Returns a character string of a specified length.
CompleteDir	Adds a backslash to the end of a string, if necessary.
Contains	Allows you to check if a substring is within a string.
ExtractLine	Extracts a specific line from a character string.

ExtractString	Allows you to: <ul style="list-style-type: none"><li>extract a substring from a string based on a specified string separator.</li><li>search for substrings in a string based on a specified string separator.</li></ul>
ExtractStringBetween	Allows you to: <ul style="list-style-type: none"><li>extract a substring between two given separators from a character string.</li><li>search for substrings between two given separators in a character string.</li></ul>
HexaToBuffer	Converts a string with hexadecimal bytes to a binary buffer.
HexaToInt	Returns the numeric value of a hexadecimal string.
HTMLToRTF	Converts an HTML string or an HTML buffer to an RTF string.
HTMLToText	Converts an HTML string or buffer to text.
IntToHexa	Returns the hexadecimal string of a numeric value.
JSONToString	Converts the content of a JSON variable to a JSON string.
JSONToVariant	Converts a JSON string to a Variant variable.
JSONValid	Teste si une chaîne a un format JSON valide.
Left	Extracts the left part (i.e., the first characters) from a string or buffer.
Length	Returns: <ul style="list-style-type: none"><li>the length of a string, i.e., the number of characters in the string (including spaces and binary zeros).</li><li>the size of a buffer, i.e., the number of bytes in the buffer.</li></ul>
LengthToString	Converts a size in bytes to a string with human-readable units such as kilobytes, megabytes or terabytes.
LineToPosition	Returns the position of the first character of a line in a block of text.
Lower	Converts all characters in a string to lowercase according to the region settings defined in Windows.
MarkdownToDOCX	Converts a Markdown string to a DOCX file.
MarkdownToHTML	Converts a Markwdown string to an HTML buffer (encoded in UTF-8).

MarkdownToPDF	Converts a Markdown string (or buffer) to a buffer containing a PDF document.
MarkdownToText	Converts a Markdown string to text (without tags).
MatchRegularExpression	Checks whether a character string matches a specific format and retrieves the different substrings that match the format.
Middle	Extracts: <ul style="list-style-type: none"><li>a substring from a string starting at a specified position.</li><li>part of a buffer starting at a specified position.</li></ul>
NoAccent	Converts accented characters in a string to non-accented characters.
NoCharacter	Returns a string after removing the specified leading and trailing characters.
NoLeftCharacter	Returns a string after removing the specified characters from the left side of the initial string.
NoRightCharacter	Returns a string after removing the specified characters from the right side of the initial string.
NoSpace	Returns a string after removing the spaces: <ul style="list-style-type: none"><li>from the left and right side of the initial string.</li><li>within the string.</li></ul>
NumberInWords	Converts a number to words.
NumToFinancial-WritingChinese	Converts a numeric value (integer, real or currency) to a string in the Chinese financial writing format.
NumToString	Converts a numeric value (integer, real or currency) to a string according to a specified format.
OemToAnsi	Converts an OEM string (DOS) to an ANSI string (Windows).
Phonetic	Returns the phonetic transcription of the string passed as parameter.
PhraseToBuffer	Returns the buffer encoded in a BIP39 mnemonic phrase.
Position	Finds the position of a specified string within another string.
PositionOccurrence	Finds the Xth position of a string within another string.
PositionToLine	Returns the number of the line that contains a character identified by its position in a block of text.

RegexMatch	Checks whether a string matches the format defined by a regular expression.
RegexReplace	Replaces all the parts of a string that follow a specific format.
RegexSearch	Searches for all the parts of a string that follow a specific format.
RepeatString	Concatenates N number of copies of the same specified string or buffer.
Replace	Replaces all occurrences of a specified substring in a string with another specified substring.
Reverse	Returns the character that corresponds to the difference between the ASCII code of a specific character in a string and 255.
Right	Extracts the right part (i.e., the last characters) from a string or buffer.
StringBuild	Builds a character string from an initial string and parameters.
StringCompare	Compares two strings character by character: <ul style="list-style-type: none"><li>• according to the sequence of ASCII characters.</li><li>• according to the alphabetical order.</li></ul>
StringCount	Calculates: <ul style="list-style-type: none"><li>• the number of occurrences of a specific character string (by respecting the search criteria) in another character string.</li><li>• the number of occurrences of a set of strings in an array.</li></ul>
StringDelete	Deletes: <ul style="list-style-type: none"><li>• all the occurrences of a substring within a string.</li><li>• part of a string within another string. The part of the string to remove is identified by its position and length.</li></ul>
StringDeleteDuplicate	Removes duplicates among substrings of a string, based on one or more separators.
StringEndsWith	Checks whether a character string ends: <ul style="list-style-type: none"><li>• with a specific character string.</li><li>• with one of the character strings in an array.</li></ul>
StringFormat	Formats a character string according to the selected options.

StringIncrement	Increments a number at the beginning or at the end of a string.
StringInsert	Inserts a string into another character string at a given position.
StringRetrieve	Retrieves a character string in a format external to PC Soft (character string that is the result of a Windows API for example).
StringReverse	Reverses the characters of a string.
StringSplit	Extracts all the substrings from a character string based on one or more string separators.
StringStartsWith	Checks whether a character string starts: <ul style="list-style-type: none"><li>• with a specific character string.</li><li>• with one of the character strings in an array.</li></ul>
StringToJSON	Converts a JSON string into a JSON variable.
StringToNum	Returns the numeric value of a character string. This feature enables you to perform numeric calculations on the character strings.
StringToUTF8	Converts an ANSI or UNICODE string to UTF-8.
SystemIntegerToInteger	Converts a system integer to a 4-byte integer.
TextToHTML	Converts text to HTML (carriage returns are replaced by  , accented characters are replaced by their equivalent in HTML, etc.).
TextToRTF	Converts text to Rich Text Format.
Truncate	Truncates the right part of a string or buffer.
UncompleteDir	Removes the '/' or '\' characters from the end of a string, if necessary.
UnicodeToAnsi	Converts: <ul style="list-style-type: none"><li>• a UNICODE string to ANSI (Windows).</li><li>• a buffer containing a UNICODE string to a buffer containing an ANSI string (Windows).</li></ul>
Upper	Converts all characters in a string to uppercase according to the region settings defined in Windows.
URLDecode	Decodes an encoded URL (which means with characters in %xx format).



URLEncode	Encodes a URL with a sub-set of ASCII characters.
URLExtractPath	Returns the different elements of a URL: protocol, user, password, domain, port, resource path, resource name, resource extension, request and fragment.
UTF8ToAnsi	Converts a UTF-8 string to ANSI.
UTF8ToString	Converts a UTF-8 string to ANSI or UNICODE.
UTF8ToUnicode	Converts a UTF-8 string to Unicode
UTFConvert	Converts a string or buffer to the specified format: UTF, ANSI, Unicode.
Val	Returns the numeric value of a character string.
VariantToJSON	Converts the content of a Variant variable to JSON.
WordCount	Returns the number of words in a string.
WordToArray	Fills an array with the words from a string.

Other functions can also be used to handle character strings:

ArrayToString	Converts a one- or two-dimensional array to a character string.
Deserialize	Deserializes a buffer or a character string containing the data of a class, a structure or an array and their sub-elements.
StringToArray	Fills a one- or two-dimensional array with the content of a string.
StringToDate	Converts a date in character string format to a date in YYYYMMDD format.
TextToXML	Converts an ANSI string to an XML string.
XMLToText	Converts an XML string to an ANSI string.

### 3.2 Compression / encryption functions

The following functions are used to manage encryption and compression:

Compress	Compresses a string or a memory block (buffer) in binary format.
Decode	Decodes a character string that was encoded by <b>Encode</b> .

Decrypt	Decrypts a character string that was encrypted by <b>Encrypt</b> .
DecryptAsymmetric	Decrypts a buffer with a private/public key algorithm with RSA encryption.
DecryptAsymmetric-Enveloped	Performs the asymmetric encryption of an envelope that has been encrypted with EncryptAsymmetricEnveloped.
DecryptStandard	Decrypts a message that was encrypted with a symmetrical encryption algorithm (AES, DES, etc.).
Encode	Encodes a character string or a buffer. To decode the result, simply use <b>Decode</b> .
Encrypt	Encrypts a character string in binary format or in ASCII format.
EncryptAsymmetric	Encrypts a buffer using an asymmetric encryption algorithm (RSA) that requires a public key and a private key.
EncryptAsymmetric-Enveloped	Performs an asymmetric encryption of an envelope.
EncryptGenerateRSAKey	Generates an RSA private/public key pair.
EncryptStandard	Encrypts a character string or a binary buffer by using a symmetrical encryption algorithm (AES, DES, etc.)
LuhnCalculate	Calculates the next Luhn check digit for a given sequence.
LuhnCheck	Checks that the given sequence is valid according to the Luhn formula.
Uncompress	Decompresses a memory block (buffer) that was compressed by <b>Compress</b> .
UUDecode	Decodes a character string in UUEncode format.
UUEncode	Encodes a character string according to the UUEncode algorithm.

### 3.3 Hash functions

The hash functions are as follows:

HashCheckFile	Checks the Hash of a file for a specific type of algorithm.
HashCheckString	Checks the Hash of a character string for a specific type of algorithm.

HashFile	Calculates the Hash of a file according to a specific algorithm.
HashString	Calculates the Hash of a character string according to a specific algorithm.

These functions are used to manage simple hashes as well as hashes with message authentication.

### 3.4 Date and Time management functions

The following functions are used to manage dates and times:

Age	Returns the age according to the date of birth.
BankHoliday	Indicates whether a day is a public holiday or not.
BankHolidayAdd	Indicates that a day (or a list of days) is a public holiday.
BankHolidayDeleteAll	Clears the list of all public holidays.
BankHolidayList	Returns the list of the public holidays defined by <b>BankHolidayAdd</b> .
ChronoEnd	Stops a stopwatch and returns the time elapsed (in milliseconds) since the start of timing (call to <b>ChronoStart</b> ).
ChronoPause	Pauses a stopwatch in order to measure the time spent on a process.
ChronoReset	Stops and resets a running stopwatch.
ChronoResume	Restarts a stopwatch that was stopped by <b>ChronoEnd</b> or <b>ChronoPause</b> .
ChronoStart	Starts a stopwatch to measure the duration of a process (in milliseconds) and resets a running stopwatch.
ChronoValue	Indicates the time elapsed (in milliseconds) since the call to <b>ChronoStart</b> .
CurrentMonth	Returns the number of the current month.
CurrentYear	Returns the current year in integer format.
DateDifference	Calculates the number of days between two dates.
DatePicker	Opens the system date picker.

DateSys	Returns or modifies the system date (current date defined on the current computer).
DateTimeByDefault	Used to define the default value of Date, Time and DateTime variables.
DateTimeDifference	Calculates the difference between two dates and times.
DateTimeLocalToTimeZone	Converts a local date and time to the date and time in a particular time zone.
DateTimeLocalToUTC	Converts a local date and time (time zone, summer time, winter time) to UTC.
DateTimeTimeZoneToLocal	Converts a date and time from a particular time zone to a local date and time.
DateTimeToEpoch	Converts a local datetime to Epoch format (number of time units since 01/01/1970 at 00h00 UTC).
DateTimeToReal	Converts a date and time to an 8-byte real.
DateTimeToString	Converts a DateTime to string in the specified format.
DateTimeUTCToLocal	Converts a UTC date and time to local time (time zone, summer time, winter time).
DateTimeValid	Checks the validity of a date/time between January 01, 0001 at 0 hours 0 minutes 0 seconds 0 hundredths of a second and December 31, 9999 at 23 hours 59 minutes 59 seconds 99 hundredths of a second.
DateToDay	Returns the day that corresponds to a given date.
DateToDayInAlpha	Returns the name of the day that corresponds to a specified date.
DateToDayOfWeek	Returns the day of the week that corresponds to a given date.
DateToInteger	Converts a date to an integer.
DateToMonthInAlpha	Returns the name of the month that corresponds to a specified date.
DateToString	Converts: <ul style="list-style-type: none"><li>• a date to a string in the specified format.</li><li>• a date time to a string in the specified format.</li></ul>
DateToWeekNumber	Returns the week number within the year that corresponds to the specified date.

DateValid	Checks the validity of a date between January 01 0001 and December 31 9999.
DayNumberInAlpha	Returns the name of the day of the week according to its number.
DurationToString	Converts a duration to a character string in the specified format.
Easter	Returns the date of Easter for the specified year according to the western Christian calendar.
EpochToDateTime	Converts a date and a time in Epoch format (number of time units since 01/01/1970 at 00h00 UTC) to a local datetime.
FirstDayOfMonth	Returns the first day of the month.
FirstDayOfWeek	Returns the date of first day of the week (i.e., Monday).
FirstDayOfYear	Returns the date of the first day of the year.
IntegerToDate	Converts an integer into a date in YYYYMMDD format.
IntegerToDay	Returns the day of the week that corresponds to the specified date (this date is an integer).
IntegerToDayInAlpha	Returns the name of the day that corresponds to a specified date (this date is an integer).
IntegerToMonthInAlpha	Returns the name of the month that corresponds to the specified date (this date is an integer).
IntegerToTime	Converts an integer into a time in HHMMSSCC format.
IntegerToWeekNumber	Returns the week number within the year that corresponds to the specified date (this date is an integer).
LastDayOfMonth	Returns the date of the last day for the specified month.
LastDayOfWeek	Returns the date of the last day of the week corresponding to a given date.
LastDayOfYear	Returns the date of the last day of the year.
LeapYear	Indicates whether the specified year is a leap year or not.
MonthNumberInAlpha	Returns the name of the month according to its number.
Now	Returns or modifies the system time (current time defined for the current computer).

RealToDateTime	Converts an 8-byte real to a date and time.
StringToDate	Converts a date in character string format to a date in YYYYMMDD format.
StringToDuration	Converts a duration in character string format to a duration that can be used by a Duration variable.
StringToTime	Converts a time in character string format to a time in HHMMSSCC format that can be used by a <b>Time</b> variable].
SysDateTime	Returns or modifies the system date (current date and time defined on the current computer).
TimeDifference	Calculates the difference between two times in hundredths of a second.
TimePicker	Opens the system time picker.
TimeSys	Returns or modifies the system time (current time defined for the current computer).
TimeToInteger	Converts a time to an integer.
TimeToString	Converts a time to a string in the specified format.
TimeValid	Checks the time validity.
Today	Returns or modifies the system date (date of the day defined on the current computer).
Tomorrow	Returns the date of the day following the current day or a given date.
WeekNumber	Returns the week number within the year that corresponds to the specified date (this date is a character string).
WeekToDate	Returns the date of the Monday for the specified week and year.
Yesterday	Returns the date of the day preceding the current day or a given date.

### 3.5 Array management functions

These functions allow you to easily handle the Array, Associative Array and Dynamic Array variables:

Add	Adds an element in last position: <ul style="list-style-type: none"><li>• of a one-dimensional WLanguage array.</li><li>• of an advanced array property (array of events of gglCalendar, etc.).</li><li>• from a WLanguage list.</li></ul>
AddColumn	Adds a column at the end of a two-dimensional array property (xlsDocument variable for example]).
AddLine	Adds a row at the end: <ul style="list-style-type: none"><li>• of a one- or two-dimensional WLanguage array.</li><li>• of a two-dimensional array property of a variable type (xlsDocument variable for example).</li></ul>
AddReference	Adds the reference of an object at the last position in a one-dimensional array.
Apply	Executes a procedure for each element of a WLanguage array.
ArrayAdd	Adds an element in last position: <ul style="list-style-type: none"><li>• of a one-dimensional WLanguage array.</li><li>• of an advanced array property (array of events of gglCalendar, etc.).</li><li>• of a WLanguage list.</li></ul>
ArrayAddLine	Adds a row at the end: <ul style="list-style-type: none"><li>• of a one- or two-dimensional WLanguage array.</li><li>• of an advanced two-dimensional array property (xlsDocument, etc.).</li></ul>
ArrayAddReference	Adds the reference of an object at the last position in a one-dimensional array.
ArrayAddSorted	Adds an element to a one-dimensional WLanguage array taking into account: <ul style="list-style-type: none"><li>• the sort criteria defined by <b>ArraySort</b></li><li>• the comparison operator described in the class, if the elements of the array are class instances.</li></ul>
ArrayCopy	Copies the content of an array to another array.

ArrayCount	Returns: <ul style="list-style-type: none"><li>• the number of elements in a one-dimensional array.</li><li>• the number of rows in a two-dimensional array.</li><li>• the number of elements in the first dimension of a three-dimensional or N-dimensional array.</li></ul>
ArrayDelete	Deletes an element at a given position from a one-dimensional WLanguage array, from an advanced array property (array of gglCalendar events, etc.), from an associative array, from a WLanguage list.
ArrayDeleteAll	Deletes all elements from a one-dimensional or two-dimensional WLanguage array, from an associative array. <ul style="list-style-type: none"><li>• from an advanced array property (array of gglCalendar events, etc.), from a WLanguage queue, from a WLanguage stack, from a WLanguage list.</li></ul>
ArrayDeleteDuplicate	Deletes the duplicates from an array.
ArrayDeleteLine	Deletes a row at a given position from a two-dimensional array, from the advanced array property (array of gglCalendar events, etc.), The array size is automatically reduced.
ArrayDistinct	Returns an array without duplicates.
ArrayInfo	Returns the characteristics of an array: type of elements, number of dimensions, dimensions, etc.
ArrayInsert	Inserts an element at a given position: <ul style="list-style-type: none"><li>• into a one-dimensional array.</li><li>• into an associative array.</li><li>• into an advanced array property (array of gglCalendar events, etc.).</li><li>• into a WLanguage list.</li></ul>
ArrayInsertLine	Inserts a row at a given position: <ul style="list-style-type: none"><li>• into a two-dimensional WLanguage array.</li><li>• into an advanced two-dimensional array property (xlsDocument, etc.).</li></ul>
ArrayInsertReference	Adds the reference of an object at given position in a one-dimensional array.
ArrayIntersect	Returns an array corresponding to the intersection of two arrays.

ArrayMove	Allows you to: <ul style="list-style-type: none"><li>• move an element in a one-dimensional WLanguage array or in an advanced array property (array of gglCalendar events, etc.). The other array elements are moved accordingly.</li><li>• swap two elements in a one-dimensional WLanguage array or in an advanced array property (array of gglCalendar events, etc.).</li></ul>
ArrayMoveLine	Allows you to: <ul style="list-style-type: none"><li>• move a row in a two-dimensional array or in an advanced array property. The other array rows are moved accordingly.</li><li>• swap two rows in a two-dimensional array or in an advanced array property.</li></ul>
ArrayReverse	Reverses the content of a one-dimensional array.
ArraySeek	Searches for the index of an element in a WLanguage array.
ArraySeekByProcedure	Seeks an element in a one-dimensional WLanguage array by using a comparison procedure.
ArrayShuffle	Shuffles the elements of a WLanguage array.
ArraySort	Sorts a WLanguage array.
ArraySwapLine	Used to swap two rows in a two-dimensional array or in an advanced array property.
ArrayToCSV	Converts a one-dimensional array or a two-dimensional array into a character string in CSV format.
ArrayToString	Converts a one- or two-dimensional array to a character string.
ArrayUnion	Returns an array corresponding to the union of two arrays.
ArrayUnionAll	Returns an array corresponding to the union of two arrays (without deleting duplicates).
CheckAllFalse	Returns <u>True</u> if all the elements of an array are <u>False</u> .
CheckAllTrue	Returns <u>True</u> if all the elements of an array are <u>True</u> .
CheckAnyFalse	Returns <u>True</u> if at least one element of an array is <u>False</u> .
CheckAnyTrue	Returns <u>True</u> if at least one element of an array is <u>True</u> .

CSVToArray	Fills a one- or two-dimensional array of the specified type with the character string in CSV format passed in parameter.
Delete	Deletes an element at a given position from a one-dimensional WLanguage array, from an advanced array property (array of gglCalendar events, etc.), from an associative array, from a WLanguage list.
DeleteAll	Deletes all the elements from a one-dimensional or two-dimensional WLanguage array, from an associative array, from an advanced array property (array of gglCalendar events, etc.), from a WLanguage queue, from a WLanguage stack, from a WLanguage list.
DeleteColumn	Deletes a column at a given position from the advanced array property (array of gglCalendar events, etc.).
DeleteLine	Deletes a line at a given position: <ul style="list-style-type: none"><li>• from a two-dimensional array.</li><li>• from the advanced array property (array of gglCalendar events, etc.).</li></ul>
FileToArray	Fills an array of classes or structures with the content of an HFSQL data file or query.
Filter	Returns an array containing the elements of a source array for which a procedure returns <u>True</u> .
Insert	Inserts an element at a given position: <ul style="list-style-type: none"><li>• into a one-dimensional WLanguage array.</li><li>• into an associative array.</li><li>• into an advanced array property (array of gglCalendar events, etc.).</li><li>• into a WLanguage list.</li></ul>
InsertColumn	Inserts a column at a given position into an advanced two-dimensional array property (xlsDocument, ...).
InsertLine	Inserts a row at a given position: <ul style="list-style-type: none"><li>• into a two-dimensional WLanguage array.</li><li>• into an advanced two-dimensional array property (xlsDocument, etc.).</li></ul>
InsertReference	Adds the reference of an object at given position in a one-dimensional array.
Map	Executes a procedure on each element of a source array and returns the results in a WLanguage array.

Max	Returns the greatest value passed as parameter.
Mean	Calculates the mean of several elements: <ul style="list-style-type: none"><li>• elements found in an array,</li><li>• numeric values, ...</li></ul>
Min	Returns the lowest value.
Move	Allows you to: <ul style="list-style-type: none"><li>• move an element in a one-dimensional WLanguage array or in an advanced array property (array of gglCalendar events, etc.). The other array elements are moved accordingly.</li><li>• swap two elements in a one-dimensional WLanguage array or in an advanced array property (array of gglCalendar events, etc.).</li></ul>
MoveColumn	Moves or swaps a column in a two-dimensional array property contained in an advanced type (xlsDocument, etc.).
MoveLine	Allows you to: <ul style="list-style-type: none"><li>• move a row in a two-dimensional WLanguage array or in an advanced array property. The other array rows are moved accordingly.</li><li>• swap two rows in a two-dimensional WLanguage array or in an advanced array property.</li></ul>
Reduce	Applies a computation to each element of a WLanguage array.
Seek	Searches for the index of an element in a WLanguage array.
SeekByProcedure	Seeks an element in a one-dimensional array by using a comparison procedure.
Sort	Sorts a WLanguage array.
StringToArray	Fills a one- or two-dimensional array of the specified type with the character string passed in parameter.
Sum	Calculates the sum of the array elements.
SwapLine	Used to swap two rows in a two-dimensional array or in an advanced array property.

### 3.6 External file management functions

The following functions are used to manage external files:

fAddBuffer	Adds the content of a buffer at the end of a file.
fAddText	Adds the content of a character string at the end of a file.
fAttribute	Returns or modifies the attributes of a file.
fAttributeReadOnly	Gets and sets the read-only attribute of a file.
fBuildPath	Builds a complete file path from the directory and short name of a file. This function is mainly used to get a valid file name without having to worry about the presence of separators in the path ('/' or '\') or about their direction according to the platform.
fBuildRelativePath	Returns the path to a file relative to a reference directory.
fCacheDir	Returns the path of directory that will be used to store the data in cache on the internal storage space of application.
fChangeExtension	Changes the extension of a file on disk.
fChangeSize	Resizes an external file.
fClose	Closes an external file.
fCompare	Compares the content of two files bit by bit.
fCompress	Compresses a file.
fContentType	Returns the type of file according to the MIME standard (Multipurpose Internet Mail Extensions).
fCopyDir	Copies the contents of a directory and its subdirectories.
fCopyFile	Copies one or more files (regardless of their attributes).
fCopyFileWebFolder	Copies an image file from the data directory of application (or from one of its sub-directories) to the "_WEB" directory of application (or to one of its sub-directories).
fCreate	Creates an external file.
fCreateLink	Creates a link on a file at the specified location.
fCurrentDir	Identifies or modifies the current directory.
fCurrentDrive	Gets or sets the current drive.



fDataDir	Returns the full path of the directory of HFSQL data files specified by the user when installing the application.
fDataDirCommon	Returns a directory path for the shared data of the current application: <ul style="list-style-type: none"><li>• between all the users of the computer</li><li>• between all the TSE users</li></ul> This directory can be used without restriction in Windows XP, Vista, ... This directory is used to store: <ul style="list-style-type: none"><li>• the files for the general configuration of the application (.INI files, .XML files, etc.).</li><li>• the data files common to all the users (HFSQL .fic, .ndx and .mmo files).</li></ul>
fDataDirUser	Returns a directory path for the data of the current user of the current application.
fDate	Returns or modifies the different dates associated with a file or directory (date created, modified or accessed).
fDateTime	Returns or modifies the different dates and times associated with a file or directory (date created, modified or accessed).
fDecrypt	Decrypts a file that was encrypted by fCrypt.
fDelete	Deletes a file accessible from the current computer.
fDeleteFileWebFolder	Deletes an image file from the "_WEB" directory of the application (or from one of its sub-directories).
fDetectRemovableStorage	Detects whether a removable storage unit (CD, USB key, USB camera, ...) was added or removed.
fDir	Finds a file or directory.
fDirAttribute	Returns or modifies the attributes of a directory.
fDirectoryExist	Checks the existence of a directory.
fDirExist	Checks the existence of a directory.
fDirRename	Renames a directory.
fDirSize	Returns the size of a directory in bytes.
fDriveInfo	Returns information about a drive.

fDriveReady	Determines: <ul style="list-style-type: none"><li>• if a given drive (storage card, CD or DVD) is available,</li><li>• whether the drive exists.</li></ul>
fEncrypt	Encrypts a file in binary or ASCII format.
fExeDir	Returns the directory from which the execution is started.
fExtractPath	Returns the different elements of a path: disk, directories, name and extension of the file.
fExtractResource	Extracts a resource from the application into a physical device location.
fFileExist	Checks the existence of an external file.
fFileName	Returns the name of an external file that is currently opened.
fFind	Finds a character string or a buffer in a file opened by <b>fOpen</b> .
fGlobalDirCommon	Returns a directory path for the global data of current application (data shared between several applications), regardless of the current user.
fGlobalDirUser	Returns a directory path for the global data of current application (data shared between several applications), for the current user.
fGraphicFilter	Returns the list of image formats supported by the current platform, in the format expected by the filter of <b>fSelect</b> .
fImageSelect	Opens the image picker of Windows.
fInsertTextAtBeginning	Inserts text at the beginning of a file without overwriting the existing text.
fIsImage	Used to find out whether a file found on disk or a file contained in a buffer corresponds to an image format recognized by WINDEV, WEBDEV or WINDEV Mobile.
fListDirectory	Lists the subdirectories of a given directory and returns the full paths of listed subdirectories.
fListDisk	Returns the list of disks installed on the computer.
fListFile	Lists the files found in a directory and returns the list of files.
fListResource	Lists the resources integrated to the application.

fLoadBuffer	Loads the content of an external file (text, image, ...) in a buffer variable.
fLoadResource	Loads the content of a resource from a library.
fLoadText	Loads the content of a text file into any text control or text variable (string variable, edit control in a window, Static control in a report, etc.).
fLock	Entirely or partially locks an external file.
fLongName	Returns the long name of a file or directory.
fLongPath	Returns the full long path of a file or directory.
fMakeDir	Creates a directory (and the intermediate directories if necessary).
fMerge	Merges several files into a single one.
fMIMEType	Returns the type of file according to the MIME standard (Multipurpose Internet Mail Extensions).
fMoveDir	Moves a directory and its content.
fMoveFile	Moves a file from one directory to another.
fOpen	Opens an external file (ANSI or UNICODE) in order to handle it through programming.
fOpenTempFile	Creates and opens a temporary file.
fParentDir	Returns the path of the parent directory for the specified directory.
fRead	Reads: <ul style="list-style-type: none"><li>• a block of bytes (characters) in an external file (ANSI or Unicode),</li><li>• the content of an external file (ANSI or Unicode) and assigns it to a memory area.</li></ul>
fReadLine	Reads a line from an external file (ANSI or UNICODE).
fReadLineRegularExpression	Reads a line in an external file, gets sections of this line according to a regular expression and assigns them to a variable.
fRemoveDir	Deletes a directory from a disk.
fRename	Modifies the name of a file.

fReportsAndQueriesDir	Returns: <ul style="list-style-type: none"><li>• the full path of the directory for the custom reports and queries. This directory corresponds to the directory of the reports and queries visible by the user who created them.</li><li>• the full path of the directory for the shared reports and queries. This directory corresponds to the directory of the reports and queries visible by all the application users.</li></ul>
fResourceDir	Returns the path of the read-only resources of the application.
fSaveBuffer	Creates and fills a text file with the content of a string or buffer variable.
fSaveText	Creates and fills a text file with the content of a text control or text variable (string variable, edit control in a window, Static control in a report, ...).
fSeek	Returns and modifies the current position in an external file.
fSelect	Opens the Windows file picker.
fSelectDir	Opens a directory picker.
fSep	Returns the separator of directory according to the current platform ('\' or '/').
fSeparator	Returns a file path with normalized separators.
fShortName	Returns the short name of a file or directory.
fShortPath	Returns the full short path of a file or directory.
fSize	Returns the size of a file (in bytes).
fSizeUncompressed	Returns the size of a file before compression.
fSplit	Splits a file into several files.
fStopCompress	Stops the operation used to compress or decompress a file.
fTempDir	Returns the name of directory used to store the temporary files.
fTempFile	Returns the name of a unique temporary file.
fTempPath	Returns the name of directory used to store the temporary files.

fTime	Returns or modifies the different times associated with a file or directory (date created, modified or accessed).
fTrackDirectory	Detects the modifications performed on the content of a directory.
fTrackFile	Triggers the tracking of a file.
fTrackStop	Stops monitoring a file or directory.
fTrackStopAll	Stops all current monitoring tasks on files and directories.
fUncompress	Decompresses a file that was compressed by <b>fCompress</b> .
fUnlock	Entirely or partially unlocks an external file.
fWebDir	Returns the physical name of the directory containing the images, the JavaScript files and the Java files of the WEBDEV site.
fWrite	Writes a character string into an external file or a memory section.
fWriteLine	Writes a line into a text file (ANSI or UNICODE).

### 3.7 Shared memory area functions

The following functions are used to manage the shared memory areas:

fMemExist	Checks the existence of a shared memory zone.
fMemOpen	Opens a memory zone shared between several applications.
fMemWait	Waits for the end of process about the notifications of modification of a shared memory area.

### 3.8 Queue, stack and list management functions

The following functions are used to manage the Queue, Stack and List types:

Dequeue	Removes an element from the queue.
Enqueue	Adds an element to the queue.
ListInfo	Retrieves the characteristics of a WLanguage list: types of elements and number of elements.

Pop	Pops an element from a stack.
Push	Pushes an element onto the stack.
QueueInfo	Retrieves the characteristics of a queue: types of elements and number of elements.
StackInfo	Retrieves the characteristics of a stack: types of elements and number of elements.

### 3.9 Combination and Enumeration management functions

The following functions are used to manage the Combinations and Enumerations:

CombinationCheckName	Checks whether a combination of one of more options known by their names is valid.
CombinationFromName	Returns a combination of one or more options known by their names.
CombinationFromValue	Returns a combination of one or more options known by their associated values.
EnumerationCheckName	Checks whether an enumeration value known by its name is valid.
EnumerationCheckValue	Checks whether an enumeration value known by its associated value is valid.
EnumerationFromName	Returns an enumeration value known by its name.
EnumerationFromValue	Returns an enumeration value known by its associated value.

### 3.10 Java functions

These functions enable you to easily run some Java code:

JavaExecute	Starts a Java application or a Java applet by invoking the "Main" method of a given Java class.
JavaExecuteFunction	Starts a specific static function of a given Java class.
JavaLoad	Specifies the location of user classes required to run the Java application or the Java applet.

### 3.11 Python functions

The Python functions are as follows:

PythonAddPath	Adds the path to a directory or the path to the virtual environment to search for Python modules.
PythonExecute	Executes a function present in a Python module.
PythonInitialize	Specifies the location of the Python DLL to be loaded.
PythonTerminate	Unloads the Python DLL and frees memory.

## 4.Communication

### 4.1 Beacon functions

The following functions are used to manage Beacons:

BeaconDetectBackground	Notifies the application when the device enters or leaves the transmission range of a set of Beacons.
BeaconDetectPrecise	Used to find the Beacons near the device.
BeaconListBackground-Detection	Returns the list of active Beacon detections started in the background with <b>BeaconDetectBackground</b> .
BeaconStopBackground-Detection	Stops one or more Beacon detections in the background.
BeaconStopPrecise-Detection	Stops the Beacon detection started by <b>BeaconDetectPrecise</b> .

### 4.2 Bluetooth functions

The following functions are used to manage Bluetooth devices:

BTAcceptConnection	Used to find out whether a Bluetooth radio accepts (or not) the requests for connection coming from the devices.
BTActivate	Enables or disables the management of Bluetooth on the device.
BTChangeConnectivity	Used to configure a Bluetooth radio to accept (or not) the requests for connection coming from the devices.
BTChangeVisibility	Changes the visibility of a Bluetooth radio.
BTConnectDevice	Allows you to be authenticated toward a Bluetooth device.
BTDisableService	Disables a service on a Bluetooth device.
BTDisconnectDevice	Cancels the authentication beside a Bluetooth device.
BTEnableService	Enables a service provided by a Bluetooth device.
BTInfoDevice	Returns specific information about a Bluetooth device.

BTInfoRadio	Returns specific information about a Bluetooth radio connected to the computer.
BTIsVisible	Used to find out whether a Bluetooth radio is visible.
BTListDevice	Returns the list of accessible Bluetooth devices.
BTListRadio	Returns the list of Bluetooth radios connected to the computer.
BTListService	Returns the list of services provided by a Bluetooth device.
BTOpenDeviceProperties	Opens the window of properties for a Bluetooth device.
BTSelectDevice	Opens a system window to select a Bluetooth device.
BTStack	Returns the Bluetooth stack currently used.
BTStatus	Returns the current Bluetooth activation status on the device or asks to be notified when the activation status changes.

### 4.3 BTLE functions

The following functions are used to handle the Bluetooth Low Energy devices:

BTLECharacteristic-ChangeOfValue	Enables or disables the detection regarding the change of value for a Bluetooth Low Energy characteristic.
BTLECharacteristic-ReadValue	Reads the value of a characteristic on a Bluetooth Low Energy device.
BTLECharacteristic-WriteValue	Writes a value to a characteristic on a Bluetooth Low Energy device.
BTLEConnect	Used to connect to a Bluetooth Low Energy device.
BTLEConnectionStatus	Returns the current connection status to a Bluetooth Low Energy device or asks to be notified when the connection status changes.
BTLEDescriptorReadValue	Reads the value of a descriptor on a Bluetooth Low Energy device.
BTLEDescriptorWriteValue	Writes the value of a descriptor to a Bluetooth Low Energy device.
BTLEDisconnect	Used to disconnect from a Bluetooth Low Energy device.

BTLEGetACharacteristic	Retrieves the characteristic of a Bluetooth Low Energy service corresponding to the specified identifier (UUID).
BTLEGetADescriptor	Retrieves the descriptor of a Bluetooth Low Energy characteristic corresponding to the specified identifier (UUID).
BTLEGetAllCharacteristics	Retrieves all the characteristics (with the same identifier) associated with a BLE service.
BTLEGetAllServices	Retrieves all the services corresponding to a UUID.
BTLEGetAService	Retrieves the service of a Bluetooth Low Energy device corresponding to the specified identifier (UUID).
BTLEInitialize	Initializes Bluetooth Low Energy (required in Windows and iOS only)
BTLEInitializeDevice	Initializes a Bluetooth Low Energy device in Windows.
BTLEListDevice	Returns the list of accessible Bluetooth Low Energy devices.
BTLEStatus	Returns the current Bluetooth activation status on the device or asks to be notified when the activation status changes.

### 4.4 Wi-Fi functions

The following functions are used to manage Wi-Fi networks:

WiFiActivate	Enables or disables Wi-Fi on the device.
WiFiAddNetwork	Adds a new network to the list of Wi-Fi networks configured on the device.
WiFiConnect	Connects the device to the specified Wi-Fi network.
WiFiConnectionInfo	Returns the requested information about the current Wi-Fi connection of the device.
WiFiDeleteNetwork	Deletes a network from the list of Wi-Fi networks configured on the device.
WiFiDetectAccessPoint	Starts detecting the Wi-Fi access points currently accessible from the device.
WiFiListNetwork	Returns the list of Wi-Fi networks configured on the device.

WiFiListNetworkCard	Returns the list of Wi-Fi network cards currently available on the current computer.
WiFiSelectNetworkCard	Selects the Wi-Fi network card that will be used for the next calls to the WiFi functions.
WiFiStatus	Returns the current activation status of Wi-Fi on the device or asks to be notified when the activation status of Wi-Fi changes.

## 4.5 Email management functions

CloseSession	Closes a session with Lotus Notes or Outlook.
EmailAddFolder	Adds a folder into the messaging software (Outlook, messaging software using the IMAP protocol (Internet Message Access Protocol), ...).
EmailBuildSource	Generates the source code of the email message to be sent from the variables in the <b>Email</b> structure or in an <b>Email</b> variable.
EmailChangeFolder	Modifies the current folder in the messaging software (Outlook, messaging software using the IMAP protocol, ...).
EmailChangeStatus	Changes the status of an email on a messaging server.
EmailCheckAddress	Checks the validity of an email address.
EmailCloseSession	Closes an email management session depending on the selected mode (SMTP/POP3, IMAP, MS Exchange or Lotus Notes).
EmailConfigure	Used to configure the WLanguage functions for email management.
EmailCopy	Copies an email found in a directory to another directory of an IMAP server.
EmailCurrentFolder	Returns the name of current folder in the messaging software (Outlook, messaging software using the IMAP protocol (Internet Message Access Protocol), ...).
EmailDeleteMessage	Deletes the current email using the current protocol (POP3, IMAP, MS Exchange, Lotus Notes or Outlook).
EmailExportHTML	Retrieves the HTML code of an email to display it in an HTML control (HTML control, HTML Display control, etc.).

EmailExpunge	Destroys all the deleted emails marked as deleted ("Deleted" status) from the current folder of IMAP session.
EmailGetAll	Retrieves all the emails found on a server.
EmailGetIndicator	Retrieves the indicators defined on an email of IMAP session.
EmailGetTimeOut	Reads the value of the "timeout" for connecting to the SMTP, POP3 and IMAP messaging servers.
EmailImportHTML	Imports, into the Email structure or into an Email variable, the content of an HTML file in order to send it by email.
EmailImportSource	Populates an Email variable using the content of its <b>Source</b> property or the different variables of the email structure using the content of the <b>Email.Source</b> variable.
EmailListFolder	Returns the list of folders: <ul style="list-style-type: none"><li>• found in the Outlook messaging software.</li><li>• for an IMAP session.</li></ul>
EmailLoadAttachment	Adds an attached file to an email.
EmailMessageLength	Calculates the size of specified message (without loading the message).
EmailMsgError	Returns the message corresponding to the error identifier.
EmailNbMessage	Returns the number of incoming messages currently found: <ul style="list-style-type: none"><li>• on the email server (POP3 protocol only).</li><li>• in the Outlook messaging software.</li></ul>
EmailProgressBar	Manages a progress bar while sending and receiving emails.
EmailReadFirst	Reads the first incoming email according to the protocol used (POP3 or IMAP, MS Exchange, Lotus Notes or Outlook).
EmailReadFirstHeader	Reads the header of the first incoming email according to the protocol used (POP3 or IMAP, Lotus Notes or Outlook).
EmailReadLast	Reads the last incoming email according to the protocol used (POP3 or IMAP, MS Exchange, Lotus Notes or Outlook).
EmailReadLastHeader	Reads the header of last incoming email according to the protocol used (POP3 or IMAP, Lotus Notes or Outlook).



EmailReadMessage	Reads an incoming according to the protocol used (POP3 or IMAP, Lotus Notes or Outlook).
EmailReadMessageHeader	Reads the header of an incoming email according to the protocol used (POP3 or IMAP, Lotus Notes or Outlook).
EmailReadNext	Reads the incoming email found after the current email according to the protocol (POP3 or IMAP, MS Exchange, Lotus Notes or Outlook).
EmailReadNextHeader	Reads the header of the email found after the current email according to the protocol used (POP3 or IMAP, Lotus Notes or Outlook).
EmailReadPrevious	Reads the email found before the current email according to the protocol used (POP3 or IMAP, MS Exchange, Lotus Notes or Outlook).
EmailReadPreviousHeader	Reads the header of the email found before the current email according to the protocol used (POP3 or IMAP, Lotus Notes or Outlook).
EmailRemoveFolder	Deletes a folder from the Outlook messaging software or from an IMAP server.
EmailReset	Resets all the variables of the email structure or all the variables of an Email variable.
EmailRunApp	Starts the native application for sending emails found on the current computer: <ul style="list-style-type: none"><li>• Computer running Windows,</li><li>• Mobile device (Android, iPhone, iPad).</li></ul>
EmailSaveAttachment	Copies the attachment of current email onto the user computer (locally).
EmailSeekFirst	Finds one or more emails according to the criteria specified in the messaging software (Outlook, messaging software using the IMAP protocol, ...).
EmailSeekFirstNotes	Seeks one or more emails according to specified criteria, in a local or remote Lotus Notes or Domino database.
EmailSend	Sends an email via the SMTP protocol while controlling the "buffer" of the email.
EmailSendMessage	Sends an email using a given protocol (SMTP, MS Exchange, Lotus Notes, Outlook).
EmailSetTimeOut	Changes the value of the "timeout" for connecting to the SMTP, POP3 and IMAP messaging servers.

EmailStartIMAPSession	Starts a session for receiving and reading emails with the IMAP protocol.
EmailStartNotesSession	Allows you to access the data handled by the Lotus Notes messaging (emails, contacts, groups of contacts, tasks, appointments).
EmailStartOutlookSession	Allows you to access data handled by the Outlook messaging software (emails, contacts, groups of contacts, tasks, appointments and folders).
EmailStartPOP3Session	Starts a session for receiving and reading emails with the POP3 protocol.
EmailStartSession	Starts an email management session based on the selected management mode (POP3, SMTP, IMAP or MS Exchange).
EmailStartSMTPSession	Starts a session for sending emails with the SMTP protocol.
EmailStatus	Returns the status of an email sent via an SMTP session started in asynchronous mode.
EmailUpdate	Updates the messages of MS Exchange according to the email server.
OutlookDefaultProfile	Retrieves the default profile defined in Outlook.
OutlookListProfile	Lists the available Outlook profiles.
OutlookStartSession	Allows you to access data handled by the Outlook messaging software (emails, contacts, groups of contacts, tasks, appointments and folders).

## 4.6 HTTP functions

The following functions are used to handle the Internet pages, to send HTTP forms, ...:

HTTPAddFile	Adds a file to the HTTP form.
HTTPAddParameter	Adds a parameter to an HTTP form.
HTTPCancelForm	Cancels the declaration of an HTTP form and frees all its resources.
HTTPCertificate	Modifies the client certificate used by default to identify oneself on a server.
HTTPConfigure	Configures the functions that use the HTTP protocol.

HTTPCookieGet	Retrieves the cookies read by a call: <ul style="list-style-type: none"><li>• to <b>HTTPRequest</b> for an HTTP domain.</li><li>• to <b>HTTPSend</b> for an HTTP domain.</li></ul>
HTTPCookieManage	Enables or disables the management of cookies in an HTTP request run: <ul style="list-style-type: none"><li>• by <b>HTTPRequest</b>.</li><li>• by <b>HTTPSend</b>.</li></ul>
HTTPCookieRead	Returns the value of a cookie received further to an HTTP request.
HTTPCookieReplace	Replaces all the cookies stored for a domain: <ul style="list-style-type: none"><li>• by <b>HTTPRequest</b>.</li><li>• by <b>HTTPSend</b>.</li></ul>
HTTPCookieReset	Deletes all cookies (globally or for a domain) stored: <ul style="list-style-type: none"><li>• by the calls to <b>HTTPRequest</b>.</li><li>• by the calls to <b>HTTPSend</b>.</li></ul>
HTTPCookieWrite	Adds or modifies a cookie in an HTTP request.
HTTPCreateForm	Creates an HTTP form.
HTTPDestination	Indicates the backup file for the result of next HTTP request, run in the same thread.
HTTPGetResult	Retrieves the result or the header of last HTTP request run.
HTTPListCertificate	Lists the client certificates found on the computer to authenticate on a server.
HTTPProgressBar	Displays a progress bar when: <ul style="list-style-type: none"><li>• receiving the response to an HTTP request.</li><li>• sending an HTTP form (<b>HTTPSendForm</b>) or an HTTP query.</li></ul>
HTTPRequest	Starts an HTTP request on a server.
HTTPResetForm	Clears an HTTP form that is currently edited.
HTTPSend	Sends an HTTP request and waits for the response from the HTTP server.
HTTPSendForm	Sends an HTTP form.
HTTPSimulateNetwork	Simulates a slow network by limiting the download and upload speed.

HTTPTimeOut	Sets the timeout for WLanguage functions that use the HTTP protocol ( <b>HTTPRequest</b> , <b>SOAPEXecute</b> , <b>DotNetExecute</b> , <b>J2EEExecute</b> , etc.).
InternetConnected	Defines whether an Internet connection is enabled on the current computer. In mobile, you can also be notified when the status of the Internet connection changes.
OpenAPIToHTTP	Builds the HTTP request to call a function of an API. It can then be sent using <b>HTTPSend</b> .
Proxy	Specifies whether the communication functions that use HTTP or HTTPS must go through a proxy to execute their requests. A proxy is used to share the access to Internet between the different network computers.
RESTSend	Sends a REST request and waits for the server response.

## 4.7 Standard FTP functions

The following functions are used to manage FTP:

FTPAttribute	Identifies the attributes of a file found on an FTP server (File Transfer Protocol).
FTPCommand	Sends a specific FTP command to a server.
FTPConfigure	Modifies the behavior of standard FTP functions.
FTPConnect	Connects the current computer to an FTP server (File Transfer Protocol).
FTPCurrentDir	Identifies or modifies the current directory on an FTP server (File Transfer Protocol).
FTPDate	Returns the different dates of a file (date the file was created, modified or accessed) located on an FTP server (File Transfer Protocol).
FTPDateTime	Returns the different dates and times (creation, modification or access) associated with a file found on an FTP server (File Transfer Protocol).
FTPDeleteFile	Deletes a file from an FTP server (File Transfer Protocol).
FTPDisconnect	Closes the connection between the current computer and the FTP server (File Transfer Protocol).
FTPFileExist	Checks the existence of a file on an FTP server.

FTPGet	Transfers a file or directory from an FTP (File Transfer Protocol) server to the current computer.
FTPGetFile	Transfers a file from an FTP (File Transfer Protocol) server to the current computer.
FTPListFile	Lists the files (and/or the directories) found in a directory of an FTP server and returns the number of listed files (and/or directories).
FTPMakeDir	Creates a directory on an FTP server (File Transfer Protocol).
FTPName	Returns the name of last file accessed by an FTP function (File Transfer Protocol).
FTPProxy	Specifies whether the communication functions that use the FTP protocol must go through a proxy to run their requests.
FTPRemoveDir	Deletes a directory and its content (files and sub-directories) from an FTP server (File Transfer Protocol).
FTPRenameFile	Renames or moves a file found on an FTP server (File Transfer Protocol).
FTPSend	Transfers a file or directory to an FTP server (File Transfer Protocol).
FTPSize	Returns the size (in bytes) of a file found on an FTP server (File Transfer Protocol).
FTPTime	Returns the different times (creation, modification or access) associated with a file found on an FTP server (File Transfer Protocol).

## 4.8 SCP functions

The SCP functions are as follows:

SCPGetFile	Gets a file via the Secure Copy Protocol.
SCPSendFile	Transfers files via the Secure Copy Protocol.

## 4.9 J2EE functions

The J2EE functions are as follows:

J2EEAddHeader	Adds a header to the next procedure that will be run on the server.
J2EEError	Returns the error for the last J2EE procedure that failed.
J2EEGetResult	Returns the result of the last J2EE procedure that was successfully run.
J2EERun	Runs a procedure on a server of J2EE XML Web services.
J2EERunXML	Runs a procedure on a server of J2EE XML Web services.

## 4.10 OBEX functions

The OBEX functions are as follow:

OBEXConnect	Used to connect to a device that supports the OBEX protocol. Reminder: The OBEX protocol is a protocol for exchanging data via Infrared and Bluetooth.
OBEXDisconnect	Used to disconnect from a device that supports the OBEX protocol.
OBEXGetFile	Retrieves a file from a device that supports the OBEX protocol.
OBEXGetVCard	Retrieves a VCard from a device that supports the OBEX protocol.
OBEXListFile	Lists the files shared by a device that is using the OBEX protocol.
OBEXSendFile	Sends a file to a device that supports the OBEX protocol.
OBEXSendVCard	Sends a VCard file (virtual business card) to a device that supports the OBEX protocol.

### 4.11 .Net functions

DotNetAddHeader	Adds a header to the next procedure that will be run on the server.
DotNetDelegate	Initializes a.NET delegate.
DotNetError	Returns the error for the last .Net procedure that failed.
DotNetGetResult	Returns the result of the last .Net procedure that was successfully run.
DotNetRun	Runs a procedure on a .Net server.
DotNetRunXML	Runs a procedure on a server for .Net XML Web services.
DotNetVersion	Returns the version number of the .Net Framework available on the current computer.

### 4.12 SOAP functions

The following functions are used to manage the SOAP protocol:

SOAPAddAttribute	Adds an attribute to an automatic variable generated by the import of a Webservice.
SOAPAddHeader	Adds a header to the next SOAP function of a Webservice to run.
SOAPAddSAMLAssertion	Adds a SAML assertion to a Webservice request.
SOAPAddXMLSignature	Adds an XML signature to a Webservice request.
SOAPEncodeResult	Used to configure the format of the value returned by the WINDEV SOAP server to the SOAP client program.
SOAPError	Returns the error of the last SOAP procedure that failed.
SOAPGetHeader	Retrieves a header in the SOAP message received further to a call to a Webservice.
SOAPGetResult	Returns the result of the last SOAP procedure that was successfully run.
SOAPPrepare	Builds the SOAP request for calling a Webservice function.
SOAPRun	Runs a procedure on a SOAP server.
SOAPRunXML	Runs a procedure on a SOAP server.

SOAPTtoHTTP	Builds the HTTP request to call a function of a Webservice. This request can be modified to adapt it to the expectations of the Webservice, if necessary. It can then be sent using <b>HTTPSend</b> .
-------------	---

### 4.13 OAuth and OpenID protocol management functions

The following functions are used to manage OAuth and OpenID protocols:

AuthIdentify	Performs an authentication using the OAuth 2.0 protocol on any webservice.
AuthRefreshToken	Refreshes a user's OAuth2 token when it expires.
OpenIDReadIdentity	Retrieves the identity of a user contained in an <b>AuthToken</b> variable.

### 4.14 DLNA/UPnP functions

The following WLanguage functions are used to handle UPnP devices:

UPNPExecuteAction	Runs an action on the UPnP service.
UPNPFindDevice	Finds a device based on its unique name (UDN).
UPNPGetBrightness	Retrieves the brightness of a UPnP device.
UPNPGetContrast	Retrieves the contrast of a UPnP device.
UPNPGetMute	Retrieves the 'Mute' status of a UPnP device.
UPNPGetSharpness	Retrieves the sharpness of a UPnP device.
UPNPGetVolume	Retrieves the volume of a UPnP device.
UPNPListDevice	Lists the UPnP devices available on the network.
UPNPMediaOpen	Opens a media on the UPnP device.
UPNPMediaPause	Pauses the reading of the current media on the UPnP device.
UPNPMediaPlay	Starts reading the current media on the UPnP device.
UPNPMediaStop	Stops reading the current media on the UPnP device.

UPNPModifyBrightness	Modifies the brightness of a UPnP device.
UPNPModifyContrast	Modifies the contrast of a UPnP device.
UPNPModifyMute	Modifies the 'Mute' status of a UPnP device.
UPNPModifySharpness	Changes the sharpness of a UPnP device.
UPNPModifyVolume	Change the volume of a UPnP device.

### 4.15 Domotic management functions

The following functions are used to manage the X10 protocol:

DomoClose	Closes the communication opened by <b>DomoOpen</b> .
DomoOpen	Used to establish a connection with a X10 interface.
DomoSend	Sends a command to one or more electrical devices compatible with the X10 protocol.
DomoStatus	Used to find out whether an electrical device (compatible with the X10 protocol) is switched on or off.
DomoSwitchOff	Switches off one or more electrical devices compatible with the X10 protocol.
DomoSwitchOn	Switches on one or more electrical devices compatible with the X10. protocol.

### 4.16 OLE functions

GetActiveObject	Allows to use an Automation object already started on the current computer.
OLEActivate	Edits an OLE object contained in an OLE control.
OLECopy	Copies the object found in an OLE control into the Windows clipboard.
OLECreateCopy	Associates an OLE control with an OLE object while specifying a base file.
OLECreateEmpty	Associates an OLE control with an OLE object by invoking a specific OLE server.

OLELoad	Initializes an OLE control with an OLE object.
OLEPaste	Pastes the content of Windows clipboard into an OLE control.
OLEPlay	Used to: <ul style="list-style-type: none"><li>retrieve the list of available operations (other than edit) from a server,</li><li>perform an operation.</li></ul>
OLESave	Saves the object found in an OLE control in binary format.
OLEServer	Lists the OLE servers registered toward the current system.

### 4.17 Serial and parallel port management functions

sClose	Closes the specified serial port, parallel port or infrared port.
sEndEvent	Disables the detection of an event on a serial port.
sEscape	Runs different functions that directly affect the bits of serial port, parallel port or infrared port, independently of the communication protocol.
sEvent	Branches an event on a serial port.
sInEntryQueue	Retrieves the number of pending bytes in the input buffer of a specified serial port or infrared port.
sInExitQueue	Retrieves the number of bytes waiting to be transferred in the output buffer of the specified serial port or parallel port.
sListPort	Returns the list of serial and parallel ports on the computer.
sOpen	Opens and initializes a serial port, a parallel port or an infrared port.
sParameter	Defines or modifies the parameters for configuring the specified serial port, parallel port or infrared port.
sRead	Reads a character string in the entry buffer of the specified serial port.
sWrite	Writes a character string to the output buffer of the specified serial port, parallel port or infrared port.

### 4.18 USB functions

The following functions are used to manage USB devices:

USBDetectRemovable-Storage	Detects whether a removable storage unit (CD, USB key, USB camera, ...) was added or removed.
USBDisconnect	Disconnects or ejects a USB device.
USBFind	Finds a USB device according to keywords.
USBListDevice	Returns the list of USB devices currently plugged into the current computer.
USBProperty	Retrieves the value of a property for a USB device.

### 4.19 DNS functions

The following functions are used to manage DNS servers:

DNSAddAddress	Adds an IP/Name relation into the DNS server.
DNSDeleteAddress	Deletes a Name/IP Address relation from a DNS server.
DNSListAddress	Lists the IP addresses associated with a name in the DNS server.

### 4.20 Network management functions

The following functions are used to manage networks:

NetworkConnect	Associates a drive with a shared network directory.
NetworkDirName	Identifies the shared directory associated with a network drive.
NetworkDisconnect	Disconnects a network drive.
NetworkDomainName	Returns the name of the domain associated with the computer.
NetworkUser	Identifies the name of the connected user.

### 4.21 Socket management functions

The following functions are used to manage sockets:

SocketAccept	Run on the server, this function accepts a request for connecting to an existing socket found on the server.
SocketChange-TransmissionMode	Changes the transmission mode used on a socket.
SocketClientInfo	Allows the server to get information about the current client socket.
SocketClose	Closes a socket or a connection to a server socket.
SocketConfigure	Configures the use of Socket functions.
SocketConnect	Connects a client computer to a given socket.
SocketConnectBluetooth	Connects a client computer to a specific Bluetooth socket.
SocketConnectInfrared	Connects a client computer to a given socket via the infrared port.
SocketConnectSSL	Used to establish a secure connection to an SSL server.
SocketCreate	Creates a socket.
SocketCreateBluetooth	Creates a socket using a Bluetooth device.
SocketCreateInfrared	Creates a socket that is using the infrared port.
SocketCreateSSL	Creates a secure socket according to the SSL protocol.
SocketCreateUDP	Creates a socket that is using the UDP protocol.
SocketDeny	Run on the server, this function refuses a request for connection to an existing socket found on the server.
SocketExist	Checks the existence of socket.
SocketProxy	Specifies whether TCP sockets and WebSockets must go through a SOCKS5 proxy to execute requests.
SocketRead	Retrieves a message sent by another socket.
SocketWaitForConnection	Run on the server, this function is used to wait for a connection request coming from the client sockets.
SocketWrite	Writes a message intended to another socket.



WebSocketClientConnect	Establishes a connection between the client and a WebSocket server.
WebSocketClient-ConnectSSL	Establishes an SSL connection between the client and a WebSocket server.

## 4.22 WebSocket management functions

The following functions can be used in a "WebSocket server" project:

WebSocketDisconnect	Disconnects a client from a WebSocket server.
WebSocketExecute	Executes a procedure in the context of a client connected to the WebSocket server.
WebSocketExecute-GlobalService	Executes a global procedure in the context of the global service of the WebSocket server.
WebSocketList-ConnectedUser	Lists the clients connected to the WebSocket server.
WebSocketSend	Sends a message from the WebSocket server to the client.

## 4.23 WINDEV FTP functions

NetClientMessageBox	Displays a dialog box on each client computer connected to the server.
NetClientSendMessage	Sends a message from the server to all the connected client computers.
NetCloseRemoteAccess	Closes the line that was opened by <b>NetOpenRemoteAccess</b> .
NetConnect	Connects to an FTP server (File Transfer Protocol) or to an RPC server (Remote Procedure Call) powered by WINDEV.
NetDirList	Lists the directories and the files found: <ul style="list-style-type: none"><li>• on the FTP server (File Transfer Protocol) or on the RPC server (Remote Procedure Call) powered by WINDEV,</li><li>• on the client computer.</li></ul>
NetDisconnect	Disconnects from an FTP server (File Transfer Protocol) or from a RPC server (Remote Procedure Call) powered by WINDEV.

NetDiskList	Lists the available disks: <ul style="list-style-type: none"><li>• on the FTP server (File Transfer Protocol) or on the RPC server (Remote Procedure Call) powered by WINDEV,</li><li>• on the client computer.</li></ul>
NetEndServer	Stops an FTP (File Transfer Protocol) or RPC (Remote Procedure Call) server powered by WINDEV and started by <b>NetStartServer</b> .
NetEraseFile	Deletes a file from the FTP (File Transfer Protocol) or RPC (Remote Procedure Call) server powered by WINDEV.
NetExecute	Runs a specific program on an FTP (File Transfer Protocol) or RPC (Remote Procedure Call) server powered by WINDEV.
NetGetFile	Retrieves an existing file from an FTP server (File Transfer Protocol) powered by WINDEV.
NetInfoRemoteAccess	Returns information about the status of a connection as well as the statistics about this connection.
NetIPAddress	Returns the IP (Internet Protocol) address of a computer.
NetIPToMAC	Returns the MAC address corresponding to a given IP address.
NetListIPAddress	Returns the list of IP addresses (Internet Protocol) for a computer.
NetListRemoteAccess	Returns the list of available remote accesses.
NetMACAddress	Returns the MAC address (physical address) of one of the network cards found on the computer.
NetMachineName	Returns the name of the local computer or the name of a computer whose IP address is known (reverse DNS resolution).
NetMACToIP	Returns the IP address corresponding to a given MAC address.
NetMessageError	Returns the error message corresponding to the error number returned by <b>NetOpenRemoteAccess</b> and <b>NetCloseRemoteAccess</b> .
NetOpenRemoteAccess	Establishes a remote connection.
NetRemoteMessageBox	Displays a dialog box on the screen of the FTP (File Transfer Protocol) or RPC (Remote Procedure Call) server powered by WINDEV.

NetRenameFile	Renames (or moves) a file found on the FTP server (File Transfer Protocol) or on the RPC server (Remote Procedure Call) powered by WINDEV.
NetSendBuffer	Transmits a character string to an FTP server (File Transfer Protocol) or to a RPC server (Remote Procedure Call) powered by WINDEV.
NetSendFile	Transmits a file to an FTP server (File Transfer Protocol) powered by WINDEV.
NetSendMessage	Triggers a Windows event on the system of an FTP server (File Transfer Protocol) or RPC server (Remote Procedure Call) powered by WINDEV.
NetServerOption	Manages the rights of clients on the server (FTP server (File Transfer Protocol) or on the RPC server (Remote Procedure Call) powered by WINDEV).
NetStartServer	Starts an FTP (File Transfer Protocol) or RPC (Remote Procedure Call) server powered by WINDEV.

## 4.24 SNMP functions

The SNMP functions are as follows:

SNMPCloseSession	Closes an SNMP session.
SNMPGet	Reads one or more values of an agent with which an SNMP session was started.
SNMPGetNext	Reads the value found after the last value retrieved for an SNMP agent.
SNMPGetTable	Reads a table of SNMP values.
SNMPLoadMIB	Loads a MIB file in memory and analyzes it.
SNMPOIDAccess	Returns the authorized access modes of a specific OID.
SNMPOIDDescription	Returns the description of an OID.
SNMPOIDStatus	Returns the status of an OID.
SNMPOIDToString	Converts an OID from its numeric representation to its text representation.
SNMPOIDType	Returns the type of data in an OID.

SNMPSet	Writes one or more values onto an SNMP agent for which an SNMP session was started.
SNMPStartSession	Starts an SNMP session.
SNMPStringToOID	Converts an OID from its text representation to its numeric representation.
SNMPTrapDisable	Disables the interception of an SNMP trap.
SNMPTrapEnable	Enables the interception of an SNMP "trap" sent by an agent.

## 4.25 SSH functions

The following functions are used to connect and communicate with a SSH server:

SSHCommand	Sends a command to an SSH server. The session is automatically started and closed as soon as the command is sent.
SSHConnectShell	Starts a new SSH session of "Shell" type.
SSHDisconnectShell	Closes an SSH session that was opened by <b>SSHConnectShell</b> .
SSHRead	Reads the data found on the output buffer of the SSH session.
SSHWrite	Writes data into the specified SSH session.

## 4.26 Telephony functions

These functions allow you to easily handle all the telephony features of a modem from a WINDEV or WINDEV Mobile application:

iOSAppGroupDirectory	Returns the path of the container used to share files between several applications belonging to the same iOS App Group.
iOSExcludeFileFromiCloud	Excludes a file from the backups performed by the Apple cloud.
tapiAnswerCall	Answers an incoming call that was detected.
tapiBlindTransfer	Performs a "blind" transfer.

tapiCallDuring	Returns the duration of the call (difference between the start date and time of call and the end date and time of call).
tapiCalledID	Used to find out the phone number called.
tapiCallEnd	Returns the date and time of the end of call.
tapiCallerID	Used to find out the calling phone number (the one that calls).
tapiCallsBusy	Used to find out whether the number called is busy.
tapiCallsOver	Used to find out whether the incoming or outgoing call is ended.
tapiCallsWaiting	Used to find out whether the call is on hold.
tapiCallStart	Returns the date and time of the beginning of call (incoming or outgoing call).
tapiCapability	Returns the characteristics of a telephony device.
tapiCompleteTransfer	Performs a call transfer (with ability to retrieve the call).
tapiDevice	Selects the TAPI device that will be used during the following telephony operations: <b>tapiDial</b> , <b>tapiListen</b> .
tapiDeviceList	Lists the TAPI 2.0 and TAPI 3.1 compatible devices installed on the current computer.
tapiDial	Dials a phone number for a voice line.
tapiError	Indicates whether an error was returned by the last tapixxx function to the TAPI module.
tapiHangUp	Hangs up a phone line started with <b>tapiDial</b> or <b>tapiAnswerCall</b> .
tapiHold	Puts a call on hold.
tapiKeyPressed	Returns the history of the keys pressed on the phone keypad since the last call to <b>tapiKeyPressed</b> .
tapiLineDial	Dials a phone number for a voice line and chooses the device to use.
tapiListen	Starts the call detection (also called "call monitoring service").
tapiNoAnswer	Used to find out whether an answer was given to the call.

tapiOrigin	Allows you to determine the type of call (incoming or outgoing).
tapiPlay	Plays a sound file (.WAV) for the specified line.
tapiRecord	Records the current communication as a ".WAV" file.
tapiSendKey	Allows you to simulate the use of phone keys.
tapiStop	Stops reading a pre-recorded message ( <b>tapiPlay</b> ).
tapiStopCallDetection	Stops the specified call detection that was started by <b>tapiListen</b> .
tapiUnhold	Picks up a call on hold.

## 4.27 Fax functions

The following functions are used to send faxes:

FaxConnect	Establishes a connection to a fax server.
FaxDelete	Deletes a fax from the spooler and cancels it.
FaxDisconnect	Closes a connection to a fax server (established beforehand by <b>FaxConnect</b> ).
FaxInbox	Enumerates the pending faxes or the incoming faxes.
FaxOutbox	Enumerates the pending faxes or the outgoing faxes.
FaxPause	Pauses a fax in the spooler.
FaxRestart	Restarts a fax in the spooler.
FaxResume	Re-enables a fax paused in the spooler (pending fax).
FaxSend	Sends a fax according to the connection that was established beforehand ( <b>FaxConnect</b> ).
FaxStatus	Returns the status of a fax that was sent by <b>FaxSend</b> .

## 4.28 Stream management functions (video and sound)

The following functions are used to manage streams:

StreamAccept	Associates a stream connection request with a Conference control.
StreamChangeDriver	Specifies the camera that will be used as source for a stream and/or a Conference control.
StreamConfigure	Specifies the network parameters required to use the Conference controls on Internet.
StreamConnect	Connects a stream (and the associated control) to a specific address.
StreamDisconnect	Stops receiving or broadcasting a stream.
StreamInfo	Returns the name and/or number of the stream issuer.
StreamPlay	Resumes the reception or the emission of a data type in a stream.
StreamReject	Refuses a request for stream connection.
StreamStop	Stops receiving or broadcasting a type of data in a stream.

## 4.29 XML functions

JSONToXML	Converts a JSON string to XML.
TextToXML	Converts an ANSI string to an XML string.
XMLAddAttribute	Adds an attribute into an XML document.
XMLAddChild	Adds a child tag into an XML document.
XMLBuildString	Retrieves and formats the content of an XML document.
XMLCancelSearch	Cancels the search performed by <b>XMLFind</b> .
XMLChild	Positions on the first child element of current element.
XMLChildExist	Specifies whether the current tag has children corresponding to the sought type (tag, attribute, ...).
XMLClose	Closes an XML document created by <b>XMLDocument</b> .

XMLData	Returns the value of current element (tag or XML attribute) or the value of one of its attributes.
XMLDelete	Deletes the current element (as well as the entire branch) from an XML document.
XMLDocument	Creates a new XML document.
XMLElementName	Returns the name of current element (tag or attribute).
XMLElementType	Returns the type of current element (tag or attribute).
XMLExecuteXPath	Executes an XPATH query in an XML document or in an xmlDocument variable.
XMLExtractDocument	Creates a new XML document from an existing XML document.
XMLExtractString	Extracts the content of a tag found in an XML document.
XMLFind	Performs a search in an XML document.
XMLFindNamespace-ByName	Finds an XML namespace according to its XML node and in the parents of the node.
XMLFindNamespaceByURI	Finds an XML namespace from its URI (Universal Resource Identifier) in a node and in the parents of this node.
XMLFirst	Positions on the first element (tag or attribute) in the current tree level.
XMLFound	Checks whether the current element corresponds to the current browse or to the current search.
XMLInsertDocument	Inserts the current XML tag found in an XML document into another XML document.
XMLInsertElement	Inserts an XML element (tag or attribute) into an XML document.
XMLLast	Positions on the last element (tag or attribute) of the current tree level.
XMLModify	Modifies the content of the current element found in the XML document.
XMLNamespace	Returns the prefix of the namespace associated with the current tag in an XML document.
XMLNamespaceURI	Returns the URI of the namespace associated with the current tag in an XML document.

XMLNext	Positions on the next element (tag or attribute).
XMLOpen	Loads an XML document from a file, a URL or a string into an xmlDocument variable.
XMLOpenReader	Creates a reader used to read an XML document without loading it in memory.
XMLOut	Used to find out whether the element (tag or attribute) on which we want to be positioned is outside the XML document.
XMLParent	Positions on the parent element of current element (tag or attribute).
XMLParentName	Returns the name of parent tag for the current element (tag or attribute).
XMLPath	Returns the current position in the document in XPATH format.
XMLPrevious	Positions on the previous element (tag or attribute).
XMLRead	Returns the value of an XML element or the value of the attribute for an XML element.
XMLRename	Renames the current element (tag or attribute) in an XML document.
XMLRestorePosition	Restores the previously saved context of an XML document.
XMLResult	Returns the result of a calculation XPath query.
XMLRoot	Positions on the root tag of an XML document.
XMLSave	Saves an XML document in a file.
XMLSavePosition	Stores the current position in the XML document.
XMLSign	Signs an XML document.
XMLToJSON	Converts an XML string to JSON.
XMLToText	Converts a character string in XML format into a standard character string.
XMLValidDocument	Validates an XML document from an XSD schema.
XMLValidNode	Validates an XML node, its children and its attributes from its description in the XSD schema linked to the XML document.

XMLWrite	Creates or modifies the value of an XML element or the value of the attribute for an XML element.
----------	---

## 4.30 Lotus Notes, Outlook functions

### 4.30.1 Appointment management functions

The functions for managing the appointments are used to:

- manage the contacts found in the calendar of a Lotus Notes or Outlook messaging.
- manage the appointments found on a mobile device (Android/iOS).

AppointmentAdd	Adds an appointment in a Lotus Notes or Outlook calendar or in a calendar on a mobile device (Android or iOS).
AppointmentCreate	Displays the window for appointment creation of native application for managing appointments found on the Android device.
AppointmentDelete	Deletes the current appointment from the calendar of a Lotus Notes or Outlook messaging or an appointment from the calendar found on a mobile device (Android/iOS).
AppointmentDisplay	Displays an appointment in the native application for managing appointments found on the mobile device (Android or iOS).
AppointmentFirst	Reads the first appointment found in the calendar of a Lotus Notes or Outlook messaging.
AppointmentLast	Reads the last appointment in a Lotus Notes or Outlook calendar.
AppointmentList	Lists the appointments found on the mobile device (Android/iOS) and corresponding to the specified criteria.
AppointmentListCalendar	Lists the calendars available on the mobile device (Android/iOS).
AppointmentModify	Modifies the current appointment in a Lotus Notes or Outlook calendar or in the calendar found on a mobile device (Android/iOS).
AppointmentNext	Reads the appointment found after the current appointment in the calendar of a Lotus Notes or Outlook messaging.

AppointmentPrevious	Reads the appointment found before the current appointment in the calendar of a Lotus Notes or Outlook messaging.
AppointmentRead	Reads an appointment found in the calendar of a Lotus Notes or Outlook messaging.
AppointmentReset	Reinitializes the Appointment structure, the Appointment variable.
AppointmentToiCalendar	Exports an appointment or a list of appointments in iCalendar format.
iCalendarToAppointment	Lists the appointments described in a character string in iCalendar format.

4.30.2 Contact management functions

The functions for managing contacts are used to:

- manage the contacts found in the address book of a Lotus Notes or Outlook messaging.
- manage the contacts found on an Android phone.

ContactAdd	Adds a contact in a Lotus Notes or Outlook address book or into the list of contacts on the device (Android, iPhone or iPad, ...).
ContactCreate	Displays the window for contact creation of the native application for contact management found on the device (Android, iPhone or iPad, Universal Windows).
ContactDelete	Deletes the current contact from a Lotus Notes or Outlook address book or from the list of contacts on the phone (Android, iPhone or iPad, Universal Windows).
ContactDisplay	Opens the form of a contact in the device's native contacts application (Android, iPhone or iPad, Universal Windows).
ContactEdit	Opens the form of a contact in edit mode in the device's native contacts application (Android, iPhone, iPad, Universal Windows).
ContactFirst	Reads the first contact in a Lotus Notes or Outlook address book.
ContactLast	Reads the last contact in a Lotus Notes or Outlook address book.
ContactList	Lists the device contacts corresponding to the specified criteria.

ContactListSource	Lists the sources (or accounts) of the contacts saved on the device.
ContactModify	Modifies the current contact in a Lotus Notes or Outlook address book.
ContactNext	Reads the contact after the current contact in a Lotus Notes or Outlook address book.
ContactPrevious	Reads the contact before the current contact in a Lotus Notes or Outlook address book.
ContactRead	Reads a contact in a Lotus Notes or Outlook address book.
ContactReset	Reinitializes the Contact structure or a Contact variable.
ContactSearch	Searches for a contact in a Lotus Notes or Outlook address book.
ContactSelect	Displays the list of contacts found on the device (Android, iPhone or iPad, Universal Windows).

4.30.3 Contact groups management functions

The following functions used to manage the groups of contacts found in the address book of Lotus Notes, Domino or Outlook:

GroupAdd	Adds a group of contacts into the address book of a Lotus Notes or Outlook messaging.
GroupDelete	Deletes the current group of contacts from the address book of a Lotus Notes or Outlook messaging.
GroupFirst	Reads the first group of contacts found in the address book of a Lotus Notes or Outlook messaging.
GroupLast	Reads the last group of contacts in a Lotus Notes or Outlook address book.
GroupModify	Modifies the current group of contacts in the address book of a Lotus Notes or Outlook messaging.
GroupNext	Reads the group of contacts found after the current group in the address book of a Lotus Notes or Outlook messaging.
GroupPrevious	Reads the group of contacts before the current group in a Lotus Notes or Outlook address book.



GroupRead	Reads a group of contacts found in the address book of a Lotus Notes or Outlook messaging.
GroupReset	Reinitializes the <b>Group</b> structure.

### 4.31 Notes document management functions

The following functions are used to manage Notes documents found in Lotus Notes:

NotesActivateView	Indicates the view that must be handled in Lotus Notes.
NotesAddAttachment	Attaches a file to an item of the current document in Lotus Notes.
NotesCloseDatabase	Closes the local or remote database used (Lotus Notes or Domino).
NotesDeactivateView	Disables the view currently handled in Lotus Notes.
NotesDeleteAttachment	Deletes an attached file from an item of the current document in Lotus Notes.
NotesDeleteItem	Deletes the specified item from the current document in Lotus Notes.
NotesDocumentDelete	Deletes the current document from the Lotus Notes database browsed.
NotesDocumentFirst	Positions on the first document found in Lotus Notes.
NotesDocumentIdentifier	Returns the identifier of the current document in the Lotus Notes database browsed.
NotesDocumentLast	Positions on the last document found in Lotus Notes.
NotesDocumentNext	Positions on the document found after the current document in Lotus Notes.
NotesDocumentPrevious	Positions on the document found before the current document in Lotus Notes.
NotesDocumentSeek	Seeks a document in Lotus Notes.
NotesExtractAttachment	Extracts a file attached to an item of the current document in Lotus Notes.
NotesItem	Returns the value of items for the current Notes document in Lotus Notes.

NotesItemDimension	Returns the dimension of an item for the current document in Lotus Notes.
NotesListAttachment	Returns the list of files attached to an item of the current document in Lotus Notes.
NotesListItem	Returns the list of items for the current Notes document in Lotus Notes.
NotesListView	Returns the list of views found in the current Lotus Notes database.
NotesModifyAttachment	Modifies a file attached to an item of the current document in Lotus Notes.
NotesModifyItem	Modifies the specified item of the current document in Lotus Notes.
NotesNbAttachment	Returns the number of files attached to an item of the current document in Lotus Notes.
NotesOpenConnection	Establishes a connection to a Domino server.
NotesOpenDatabase	Gives access to the documents managed by Lotus Notes.
NotesOut	Used to find out whether a current document exists in Lotus Notes.
NotesReset	Creates an empty document in Lotus Notes.
NotesSave	Saves the current document in Lotus Notes.
NotesViewListColumn	Lists the columns found in the current view in Lotus Notes.
NotesViewSeekFirst	Seeks the first document found in the current view in Lotus Notes.
NotesViewSeekLast	Seeks the last document found in the current view in Lotus Notes.

### 4.32 Task management functions

The following functions are used to manage tasks found in the task list of a Lotus Notes or Outlook messaging:

TaskAdd	Adds a task to the task list of a Lotus Notes or Outlook messaging.
---------	---

TaskDelete	Deletes the current task from the task list of a Lotus Notes or Outlook messaging.
TaskFirst	Reads the first task found in the task list of a Lotus Notes or Outlook messaging.
TaskLast	Reads the last task in a Lotus Notes or Outlook task list.
TaskModify	Modifies the current task in a Lotus Notes or Outlook task list.
TaskNext	Reads the task found after the current task in the task list of a Lotus Notes or Outlook messaging.
TaskPrevious	Reads the task found before the current task in the task list of a Lotus Notes or Outlook messaging.
TaskRead	Reads a task in the task list of a Lotus Notes or Outlook messaging.
TaskReset	Reinitializes the Task structure.

### 4.33 LDAP functions

The LDAP functions are as follows:

LDAPAddAttribute	Adds an attribute or adds a new value to an existing attribute.
LDAPConnect	Used to connect to an LDAP server via the characteristics defined in the LDAPSession structure.
LDAPDelete	Deletes an object from the LDAP server.
LDAPDeleteAttribute	Deletes an attribute as well as its values.
LDAPDeleteAttributeValue	Deletes a value of a specified attribute.
LDAPDisconnect	Used to disconnect from a specific LDAP server.
LDAPFind	Performs a search in an LDAP server.
LDAPListAttribute	Lists the object attributes in an LDAP server.
LDAPListChildren	Lists the children of an object in an LDAP server.
LDAPMode	Modifies the operating mode of an LDAP session.
LDAPNbValue	Returns the number of values for an attribute.

LDAPRenameAttribute	Renames an attribute.
LDAPReplaceAttribute	Replaces all the values of an attribute by a specific value.
LDAPReplaceAttributeValue	Replaces a specific value of an attribute by a new value.
LDAPReset	Reinitializes the LDAPSession structure with the default values.
LDAPStartAdd	Initializes the addition of a new object into an LDAP server.
LDAPStartModify	Initializes the modification of an existing object in an LDAP server.
LDAPValidateAdd	Validates the addition of a new object into an LDAP server.
LDAPValidateModify	Validates the modification of an existing object in an LDAP server.
LDAPValue	Returns the value corresponding to the subscript passed as parameter for the specified attribute.

### 4.34 Google functions

#### 4.34.1 Google Contacts management functions

GglWrite	Creates and/or updates data on the Google server (addition or modification of data).
GglListContact	Retrieves the list of contacts associated with the specified Google account.
GglListContactByRequest	Retrieves a list of contacts from custom parameters.
GglRequest	Sends a communication request (HTTP request) to a Google service.
GglDelete	Deletes data from the Google server.

#### 4.34.2 Google document management functions

The following functions are used to manage documents on a Google server:

GglUploadDocument	Uploads a document to the Google server.
GglPrintDocument	Prints a document (text or PDF file, images, Word or Excel document, ...) by using the Cloud Print service of Google.

GglListDocument	Retrieves the list of documents available on the Google server for the current user.
GglGetDocument	Downloads a document from the Google Docs service.
GglRequest	Sends a communication request (HTTP request) to a Google service.
GglDelete	Deletes data from the Google server.

4.34.3 Google Calendar management functions

The following functions are used to manage calendars via the Google Calendar service:

GglWrite	Creates and/or updates data on the Google server (addition or modification of data).
GglListCalendar	Retrieves the list of Google calendars associated with the specified Google account.
GglGetCalendar	Retrieves a Google calendar and its events according to its title or identifier.
GglFillCalendar	Fills a calendar that was retrieved beforehand: the events corresponding to the calendar are retrieved (some criteria may be specified).
GglFillCalendarByRequest	Fills a calendar that was retrieved beforehand: the events corresponding to the calendar are retrieved from a custom request.
GglRequest	Sends a communication request (HTTP request) to a Google service.
GglDelete	Deletes data from the Google server.

4.34.4 Google Analytics functions

The following functions are used to send data to Google Analytics:

GglAnalyticsAddEvent	Adds an event beside Google Analytics
GglAnalyticsAddException	Adds an exception beside Google Analytics.
GglAnalyticsAddPage	Adds a page beside Google Analytics.
GglAnalyticsAddSocial-NetworkAction	Adds a social network action beside Google Analytics.
GglAnalyticsAddTiming	Adds a duration beside Google Analytics.

4.35 Google Maps management functions

The following functions are used to manage maps via the "Google Maps" service:

GglAddressToCoordinates	Retrieves the coordinates (latitude/longitude) of an address.
GglGetStaticMap	Retrieves the map of a specific location via the Google Maps service.
GglRequest	Sends a communication request (HTTP request) to a Google service.

4.36 RSS feed management functions

The following functions are used to manage RSS streams:

rssBuildString	Builds the RSS stream and returns the result (in XML format) in a character string.
rssDisplay	Builds a RSS stream and returns the content of the RSS stream to the client.
rssInitialize	Loads a RSS stream in memory.
rssSave	Builds the RSS stream and saves the RSS stream in an XML file.

4.37 Facebook functions

Caution: These functions are not available for WINDEV applications in Linux. The following functions are used to manage Facebook authentication.

FBGetUserInfo	Retrieves the Facebook information of connected user.
FBRequest	Performs a communication request (HTTP request) to the Facebook Graph API.
FBSessionStatus	Checks the status of a Facebook session.
FBStartSession	Used to authenticate and log in to Facebook.
FBStopSession	Stops the Facebook session that was started by <b>FBStartSession</b> .

### 4.38 MQTT functions

The following functions are used to manage MQTT clients:

MQTTConnect	Used to connect to a MQTT server (also called "Broker").
MQTTDisconnect	Allows you to disconnect from a MQTT server (also called "Broker").
MQTTPublish	Publishes a message on the MQTT broker.
MQTTSubscribe	Used to subscribe to a MQTT notification.
MQTTUnsubscribe	Used to unsubscribe from a MQTT notification.
MQTTUnsubscribeAll	Allows you to unsubscribe from all the existing subscriptions.

### 4.39 Geolocation/GPS functions

The following functions are used to handle geolocation data and to control GPS devices:

geoAzimuth	Returns the direction represented by a location in relation to another one.
geoDistance	Returns the distance between the two locations. This distance is expressed in meters by default.
geoGetAddress	Retrieves an address or a list of addresses corresponding to the description of a geographic location.
geoGetArea	Retrieves the coordinates of the geographic area that corresponds to a given description.
geoRunApp	Starts the device's native maps application to display: <ul style="list-style-type: none"><li>• a specific geographical location,</li><li>• a route between two locations.</li></ul>
geoTrackingDisable	Disables the tracking of user positions for a WINDEV Mobile application.
geoTrackingEnable	Enables the management of location tracking in a WINDEV Mobile application.
geoTrackingProcedure	Defines the WLanguage procedure that will be called when the location of the user changes.
geoTrackingStatus	Retrieves the status of the system for position tracking.

GPSDetectPosition	Asks to be notified when the device gets close to a given location.
GPSEnd	Indicates the end of the use of geolocation functions.
GPSFollowMovement	Requests to be notified periodically of the device's current location.
GPSGetPosition	Retrieves information about the current device position.
GPSInfo	Returns the information about the location provider used by the application for geolocation functions.
GPSInitParameter	Initializes the parameters of geolocation WLanguage functions and finds a location provider.
GPSLastPosition	Retrieves the information about the last known device position.
GPSStatus	Retrieves the activation status of the geolocation provider or asks to be notified when the status changes.
GPSStopDetection	Stops the location detection triggered by <b>GPSDetectPosition</b> .

### 4.40 In-App functions

The following WLanguage functions are used to manage in-App purchases:

inAppCheckSubscription	When starting the iOS application, checks whether the In-App subscriptions have been renewed.
inAppConsumeProduct	Used to consume, in an application, a product that was previously purchased by the user.
inAppListProductInfo	Retrieves, from the store, information about the "In-App" products associated with the application.
inAppListPurchase	Returns the list of application products purchased by the user and not consumed.
inAppPurchaseProduct	Sends a request for purchasing an "In-App" product associated with the application.
inAppRestorePurchases	Restores the purchases of non-consumable products previously performed by the user in an iOS application.

### 4.41 NFC functions

The following functions are used to manage the NFC:

NFCReadTag	Starts reading a NFC tag or enables the detection of NFC tags for the current window.
NFCSendTag	Sends a NFC tag to another device.
NFCStatus	Returns the activation status of the NFC sensor found on the device.
NFCWriteTag	Starts writing data to an NFC tag.

### 4.42 Mobile phone management functions

The following functions are used to manage the phone:

AndroidActivity-ResultProcedure	Allows you to specify the procedure to be called to get the result returned by a third-party Activity.
AppleAdvertisingIdentifier	Returns the device's advertising identifier (iOS only).
AppleRequestTracking-AuthorizationAsynchronous	Displays the permission request for user activity tracking.
AppleSignIn	Used to implement authentication via "Sign In with Apple" on devices running on iOS 13 and higher.
AppleTracking-AuthorizationStatus	Determines if the app has been allowed to track activity based on user data (iOS only).
BrightnessAuto	Returns or modifies the activation status of the automatic adjustment of the brightness on the device.
BrightnessGet	Returns the current setting of the brightness for the screen of the device.
BrightnessModify	Modifies the setting of the brightness for the device screen.
ChangeTheme	Is used to change, through programming, the theme (dark or light) used by the mobile application.
CheckUserIdentity	Allows you to check the real identity of the current user via biometrics check.
DeepLinkProcedure	Defines the WLanguage procedure executed when the user clicks on a deep link.

GetTheme	Is used to find out the current theme of a mobile application.
iOSApplicationOpen-URLProcedure	Allows you to specify the procedure to be called when the application needs to open a resource identified by a URL.
LEDSwitchOff	Switches the LED (Light-Emitting Diode) of the device off.
LEDSwitchOn	Switches the LED (Light-Emitting Diode) of the device on.
MobileNetwork-ConnectionInfo	Returns information regarding the current connection to data on the mobile network.
MobileNetworkStatus	Returns the current status of the connection to data on the mobile network or asks to be notified when the status of the connection to data changes.
NotifAdd	Adds a notification into the system bar of device (Android) or into the notification center (iOS).
NotifDelete	Deletes a notification displayed in the system bar of the device.
NotifDeleteCategory	Deletes a notification category from the application.
NotifListCategory	Lists the notification categories of the application.
NotifModify	Modifies a notification currently displayed in the system bar of the device.
NotifPushDisable	Disables the management of push notifications for a WINDEV Mobile application (Android or iOS).
NotifPushEnable	Enables the management of push notifications in a WINDEV Mobile application (Android or iOS).
NotifPushListInvalid-Identifiers	Returns the list of invalid identifiers on the APN service of Apple.
NotifPushProcedure	Specifies the WLanguage procedure called when a push notification is received by a WINDEV Mobile application (Android or iOS).
NotifPushSend	Sends a push notification to a mobile device (iOS or Android).
PowerStatus	Used to get various information about the battery (main or secondary) of the device on which the application is running.

RecorderAction	Runs an action on the recorder or dictaphone found on the current device.
ScreenSize	Returns the size (in inches) of the diagonal of the screen on the device where the application is run.
ScreenType	Returns the type of screen on which the application is run: phone, tablet, computer or watch.
SIPVisible	Displays the keyboard.
tapiDialerCall	Opens the default telephony application (dialer) found on the phone and dials the specified number.
tapiDialerDisplay	Opens the default telephony application (dialer) and displays the specified number. No call is made.
VibrationStop	Stops the vibrations of the device (even if they have been triggered by another application).
VibrationTrigger	Triggers the vibrations of the device.
VolumeGet	Retrieves the current setting of the volume for a sound source.
VolumeModify	Modifies the setting of the volume for a sound source.
VolumePhysicalButton	Identifies or modifies the sound source associated with the physical buttons used to set the volume of the device.

Remark: To display a notification on a browser from a WEBDEV site, use **NotificationDisplay**.

### 4.43 SMS functions

These functions are used to easily send and read SMSs:

SMSDelete	Deletes an SMS message from the memory of the Android phone. This function should no longer be used.
SMSFirst	Reads the first SMS message stored on the Android phone.
SMSNbMessage	Returns the number of SMS messages stored on an Android phone.
SMSNext	Reads the following SMS message stored on the Android phone.
SMSReset	Resets all variables of SMS structure.

SMSRunApp	Starts the native application for sending SMSs found on the device (Android or iPhone/iPad).
SMSSend	Sends an SMS.

### 4.44 iCloud functions

The functions to handle keys in an Apple iCloud-based container of "key-value" pairs are as follows:

iCloudCléValeurListeClé	Lists the keys contained in an Apple iCloud-based container of "key-value" pairs.
iCloudKeyValueExist	Checks if there is a key in an Apple iCloud-based container of "key-value" pairs.
iCloudKeyValueRead	Reads the value of a key in an Apple iCloud-based container of "key-value" pairs.
iCloudKeyValueRemove	Removes a key from an Apple iCloud-based container of "key-value" pairs.
iCloudKeyValueWatch	Defines the procedure called when the value of a key is changed in an Apple iCloud-based container of "key-value" pairs.
iCloudKeyValueWrite	Stores the value of a key in an Apple iCloud-based container of "key-value" pairs.

### 4.45 Functions specific to iOS - Apple Watch communication

The WLanguage functions specific to the communication between the iOS applications and the associated Apple Watch applications are as follows:

awProcedureGet-AndSendInfo	Specifies the WLanguage procedure that will be called when receiving information coming from <b>awSendAndGetInfo</b> .
awProcedureGetInfo	Specifies the WLanguage procedure that will be called when receiving information coming from <b>awSendInfo</b> .
awSendAndGetInfo	Sends data from an iOS phone to an Apple Watch (or conversely) while expecting a response.
awSendInfo	Sends an information from an iOS phone to an Apple Watch (or conversely).



### 4.46 Sensor functions

The following functions are used to manage the compass and the accelerometer of a mobile device (phone or tablet):

SensorDetectBeginShake	Notifies when the device is shaken in any direction.
SensorDetectChange-Acceleration	Notifies when acceleration forces are applied to the device along any of the three axes.
SensorDetectChange-Orientation	Allows you to be notified when the orientation of the device changes on any one of its 3 axes.
SensorDetectEndShake	Notifies when the device has been shaken in any direction.
SensorEnd	Indicates the end of the use of the sensor functions.
SensorGetOrientation	Retrieves the current orientation of the device on one of its 3 axes.

### 4.47 Download management functions on an Android device

The following functions are used to download files from an Android application:

DownloadAdd	Adds a new download in the device's download manager.
DownloadGetInfo	Retrieves the details of a download present in the download manager.
DownloadList	Lists the downloads from the current application in the download manager.
DownloadRemove	Cancels a download by removing it from the download manager, and deletes any existing files.

## 5.Multimedia management

### 5.1 Image and drawing management functions

The drawing functions are as follows:

ArrowEnd	Sets the position of the target end of an arrow in a "Shape" control.
ArrowStart	Sets the position of the source end of an arrow in a "Shape" control.
BitmapInfo	Retrieves the characteristics of an image file or the characteristics of a PDF file : <ul style="list-style-type: none"><li>• format,</li><li>• width and height,</li><li>• number of bits per pixel.</li></ul>
ColorHue	Calculates the hue of a color.
ColorLightness	Calculates the lightness of a color.
ColorSaturation	Calculates the saturation of a color.
dAlphaBlend	Blends two images together. This allows you to create fade-in effects between 2 images.
dArc	Draws an arc of circle or an arc of ellipse: <ul style="list-style-type: none"><li>• in an Image control,</li><li>• in an Image variable,</li><li>• in a WDPic variable (on the background layer),</li><li>• in a picLayer variable.</li></ul>
dBackground	Declares the default background color for the rectangles, circles, etc.: <ul style="list-style-type: none"><li>• in an Image control,</li><li>• in an Image variable,</li><li>• in a WDPic variable (on the background layer),</li><li>• in a picLayer variable.</li></ul>
dBlur	Blurs the entire given area: <ul style="list-style-type: none"><li>• in an Image control,</li><li>• in an Image variable,</li><li>• in a WDPic variable (on the background layer),</li><li>• in a picLayer variable.</li></ul>

dBorder	Draws the borders.
dChangeMode	Changes the drawing mode used by the drawing functions ( <b>dLine</b> , <b>dRectangle</b> , ...).
dChord	Draws the chord of a circle (intersection between an ellipse and a straight line): <ul style="list-style-type: none"><li>• in an Image control,</li><li>• in an Image variable,</li><li>• in a WDPic variable (on the background layer),</li><li>• in a picLayer variable.</li></ul>
dCircle	Draws a circle or an ellipse: <ul style="list-style-type: none"><li>• in an Image control,</li><li>• in an Image variable,</li><li>• in a WDPic variable (on the background layer),</li><li>• in a picLayer variable.</li></ul>
dClear	Clears the content of an Image variable or Image control and fills it with the specified color.
dCopy	Copies an image to another image.
dCopyBit	Copies an image: <ul style="list-style-type: none"><li>• from an Image control to another Image control.</li><li>• from an Image variable to another Image variable.</li></ul>
dCopyScreenImage	Copies the screen content to an Image variable or to an Image control.
dCopyWindowImage	Copies the image from a WINDEV window to an Image variable or to an Image control.
dCrop	Returns the cropped part of an image. This image is found in an Image variable.
dDisplay	Displays the drawing in the Image control.
dDominantImageColor	Returns the dominant (i.e. most frequent) color of an image in an Image variable or in an Image control.
dEncrustColor	Encrusts a color in an image found in an Image variable].
dEndDrawing	Deletes all the drawings made since the last call to <b>dStartDrawing</b> .

dFill	Colors an area: <ul style="list-style-type: none"><li>• in an Image control,</li><li>• in an Image variable,</li><li>• in a WDPic variable (on the background layer),</li><li>• in a picLayer variable.</li></ul>
dFont	Declares the font that will be used by dText: <ul style="list-style-type: none"><li>• in an Image control,</li><li>• in an Image variable,</li><li>• in a WDPic variable (on the background layer),</li><li>• in a picLayer variable.</li></ul>
dHalo	Adds a halo effect on an image found in an Image variable].
dHorizontalSymmetry	Flips an image with respect to a horizontal axis (symmetry with respect to a horizontal axis).
dImageToArray	Copies the RGB pixels of the image into a one-dimensional or two-dimensional array. The array of colors is cleared then it is filled with the integers representing the colors (24 bits) of the image.
dInvertColor	Inverts the colors: <ul style="list-style-type: none"><li>• in an Image control,</li><li>• in an Image variable,</li><li>• in a WDPic variable (on the background layer),</li><li>• in a picLayer variable.</li></ul>
dLine	Draws a line: <ul style="list-style-type: none"><li>• in an Image control,</li><li>• in an Image variable,</li><li>• in a WDPic variable (on the background layer),</li><li>• in a picLayer variable.</li></ul>
dLoadImage	Loads an image in an Image variable from a memo, a file or a binary buffer.
dModifyContrast	Changes the contrast of an image contained: <ul style="list-style-type: none"><li>• in an Image control,</li><li>• in an Image variable.</li></ul>
dModifyHSL	Modifies the hue, the saturation and the lightness of an image found: <ul style="list-style-type: none"><li>• in an Image control,</li><li>• in an Image variable.</li></ul>

dModifyHue	Changes the hue of an image: <ul style="list-style-type: none"><li>• in an Image control,</li><li>• in an Image variable.</li></ul>
dModifyLightness	Changes the lightness of an image found: <ul style="list-style-type: none"><li>• in an Image control,</li><li>• in an Image variable.</li></ul>
dModifyOpacity	Modifies the opacity of an image (which means the alpha channel of the image).
dModifySaturation	Modifies the saturation of an image found: <ul style="list-style-type: none"><li>• in an Image control,</li><li>• in an Image variable.</li></ul>
dPen	Declares the default color and style for the lines, rectangles, circles, ... found: <ul style="list-style-type: none"><li>• in an Image control,</li><li>• in an Image variable,</li><li>• in a WDPic variable (on the background layer),</li><li>• in a picLayer variable.</li></ul>
dPixelColor	Identifies the color of a pixel: <ul style="list-style-type: none"><li>• in an Image control,</li><li>• in an Image variable.</li><li>• in a picLayer variable.</li><li>• in a WDPic variable.</li></ul>
dPixelOpacity	Identifies the opacity of a pixel in an image that has an Alpha channel. This image can be: <ul style="list-style-type: none"><li>• an Image control,</li><li>• an Image variable,</li><li>• a WDPic variable (on the background layer),</li><li>• a picLayer variable.</li></ul>
dPoint	Draws a point: <ul style="list-style-type: none"><li>• in an Image control,</li><li>• in an Image variable,</li><li>• in a WDPic variable (on the background layer),</li><li>• in a picLayer variable.</li></ul>

dPolygon	Draws a polygon: <ul style="list-style-type: none"><li>• in an Image control,</li><li>• in an Image variable,</li><li>• in a WDPic variable (on the background layer),</li><li>• in a picLayer variable.</li></ul>
dPolyline	Draws a line composed of multiple segments.
DrawingAvailable	Indicates whether the drawing features are supported by the browser.
dRectangle	Draws a rectangle: <ul style="list-style-type: none"><li>• in an Image control,</li><li>• in an Image variable,</li><li>• in a WDPic variable (on the background layer),</li><li>• in a picLayer variable.</li></ul>
dRectangleGradient	Draws a rectangle: <ul style="list-style-type: none"><li>• in an Image control,</li><li>• in an Image variable,</li><li>• in a WDPic variable (on the background layer),</li><li>• in a picLayer variable.</li></ul>
dResize	Resizes an image: <ul style="list-style-type: none"><li>• in an Image control,</li><li>• in an Image variable.</li><li>• in a picLayer variable].</li></ul>
dRotation	Performs a rotation: <ul style="list-style-type: none"><li>• of an Image control,</li><li>• of an Image variable,</li><li>• of a picLayer variable.</li></ul>
dRoundedRectangle	Draws a rounded rectangle: <ul style="list-style-type: none"><li>• in an Image control,</li><li>• in an Image variable,</li><li>• in a WDPic variable (on the background layer),</li><li>• in a picLayer variable.</li></ul>
dSaveImageBMP	Saves an image: <ul style="list-style-type: none"><li>• in a file in BMP format.</li><li>• in memory.</li></ul>

dSaveImageGIF	Saves an image: <ul style="list-style-type: none"><li>• in a file in GIF format.</li><li>• in memory.</li></ul>
dSaveImageICO	Saves an image in icon format either in a file, or in memory.
dSaveImageJPEG	Saves an image: <ul style="list-style-type: none"><li>• in a JPEG file.</li><li>• in memory.</li></ul>
dSaveImagePNG	Saves an image: <ul style="list-style-type: none"><li>• in a PNG file.</li><li>• in memory.</li></ul>
dSaveImageTIFF	Saves an image in TIFF format into a file or memory.
dShadow	Adds a shadow effect on an image found in an Image variable].
dSlice	Draws a circle section or an ellipse section: <ul style="list-style-type: none"><li>• in an Image control,</li><li>• in an Image variable,</li><li>• in a WDPic variable (on the background layer),</li><li>• in a picLayer variable.</li></ul>
dStartDrawing	Indicates that the drawing functions that will be used are intended for: <ul style="list-style-type: none"><li>• the specified Image control,</li><li>• the specified Image variable,</li><li>• the specified WDPic variable (on the background layer),</li><li>• the specified picLayer variable.</li></ul>
dStraighten	Transforms an image by applying perspective correction.
dText	Draws a text: <ul style="list-style-type: none"><li>• in an Image control,</li><li>• in an Image variable,</li><li>• in a WDPic variable (on the background layer),</li><li>• in a picLayer variable.</li></ul>
dTextMarkdown	Draws Markdown text: <ul style="list-style-type: none"><li>• in an Image control,</li><li>• in an Image variable,</li><li>• in a WDPic variable (on the background layer),</li><li>• in a picLayer variable].</li></ul>

dTextRTF	Draws RTF text: <ul style="list-style-type: none"><li>• in an Image control,</li><li>• in an Image variable.</li></ul>
dTransferToImage	Creates an image from a buffer of pixels encoded in a specific format.  <b>This function is an advanced function.</b> This function is mainly used to handle the images returned by external APIs via Image variables.
dVerticalSymmetry	Flips an image with respect to a vertical axis (symmetry with respect to a vertical axis).
HSL	Creates a color from its hue, saturation and lightness.
HTMLToRGB	Converts the HTML color into RGB color.
RGB	Defines a color from its Red, Green and Blue components.
RGBBlue	Returns the blue component of an RGB color.
RGBGreen	Returns the green component of an RGB color.
RGBRed	Returns the red component of an RGB color.
RGBToHTML	Returns the HTML color corresponding to the RGB color.

## 5.2 Exif functions

The following functions are used to handle the Exif tags:

ImageInfoClose	Closes the image file (in JPEG or TIFF format) used by <b>ImageInfoValue</b> and/or <b>ImageInfoList</b> and frees the image from memory.
ImageInfoList	Returns the list of Exif tags available for the image used.
ImageInfoModify	Modifies the value of an Exif tag for the image used.
ImageInfoSave	Saves the modifications performed on the Exif tags of an image by <b>ImageInfoModify</b> .
ImageInfoValue	Returns the value of an Exif tag for the image used.

### 5.3 Image scrolling management functions

Several functions can be used to manage images:

- Web-specific image management functions (especially those that manage the scrolling of images).
- Functions for managing images in the Image controls.

ImageArea	Returns the number of the image area clicked by the Web user.
ImageFirst	Displays the first image for the scroll.
ImageLast	Displays the last image for the scroll.
ImageLayerVisible	Used to make a layer found in a WDPIC image visible or invisible.
ImageNext	Displays the next image for the scroll.
ImageOccurrence	Returns the total number of images taken into account by the scroll.
ImagePrevious	Displays the previous image for the scroll.
ImageScrollingPosition	Returns the displayed image.
ImageStartScrolling	Starts the automatic scroll of images.
ImageStopScrolling	Stops the automatic scroll of images.
ImageXPos	Returns the horizontal position (X) of the mouse cursor in relation to the specified Image control.
ImageYPos	Returns the vertical position (Y) of the mouse cursor in relation to the specified Clickable Image control.

### 5.4 TWAIN functions

The following functions are used to manage Twain devices:

TwainCurrentSource	Allows you to find out and change the Twain device used by default.
TwainListSource	Returns the list of drivers for the Twain devices connected to the current computer.
TwainProperty	Used to find out or modify the properties of the current Twain device.

TwainScanArea	Used to define or re-initialize the area to scan.
TwainSelectSource	Displays a standard dialog box used to list the different Twain devices available on the current computer.
TwainSourceHeight	Returns the height (in millimeters) of the document captured by a Twain device.
TwainSourceWidth	Returns the width (in millimeters) of the document captured by a Twain device.
TwainState	Returns the status of the current source.
TwainToBMP	Allows you to perform one of the following actions: <ul style="list-style-type: none"><li>• save the document coming from the Twain device in a Bitmap file (".BMP" extension).</li><li>• view the document coming from the Twain device in the user interface of the device.</li><li>• save the document coming from the Twain device directly in memory in Bitmap format (in a buffer).</li></ul>
TwainToClipboard	Allows you to: <ul style="list-style-type: none"><li>• copy the document coming from the Twain device into the clipboard.</li><li>• view the document coming from the Twain device in the user interface of device.</li></ul>
TwainToControl	Allows you to : <ul style="list-style-type: none"><li>• view the document coming from the Twain device in an Image control.</li><li>• view the document coming from the Twain device in the user interface of device.</li></ul>
TwainToGIF	Allows you to: <ul style="list-style-type: none"><li>• save the document coming from the Twain device in a GIF file.</li><li>• view the document coming from the Twain device in the user interface of device.</li><li>• save the document coming from the Twain device directly in memory in GIF format (in a buffer).</li></ul>
TwainToJPEG	Allows you to: <ul style="list-style-type: none"><li>• save the document coming from the Twain device in a JPEG file.</li><li>• view the document coming from the Twain device in the user interface of device.</li><li>• save the document coming from the Twain device directly in memory in JPEG format (in a buffer).</li></ul>

TwainToPDF	Allows you to: <ul style="list-style-type: none"><li>• save the document coming from the Twain device in a PDF file.</li><li>• view the document coming from the Twain device in the user interface of device.</li><li>• save the document coming from the Twain device directly in memory in PDF format (in a buffer).</li></ul>
TwainToTIFF	Allows you to: <ul style="list-style-type: none"><li>• save the document coming from the Twain device in a TIFF file.</li><li>• view the document coming from the Twain device in the user interface of device.</li><li>• save the document coming from the Twain device directly in memory in TIFF format (in a buffer).</li></ul>

## 5.5 Scanner functions

The scanner functions are as follows:

ScanCapability	Used to determine if a scanner has a given capability.
ScanGet	Retrieves the characteristics of a scanner in a variable of type Scanner.
ScanList	Lists the scanners connected to the current computer.
ScanProperty	Used to get and change the properties of a scanner.
ScanSelect	Opens a standard dialog box with the available scanners.
ScanToImage	Retrieves a document from a scanner in a variable of type <b>Image</b> .
ScanToPDF	Saves a document from a scanner to a PDF file.
ScanWithUserInterface	Saves a document from a scanner to one or more files, using the scanner's user interface.

## 5.6 Graphic string management functions

The functions for managing graphic strings are mainly used to display the graphic strings in:

- the rows found in the List Box or Combo Box controls,
- Static controls.
- the title bar or the status bar of a window.

The following WLanguage functions are used to create and handle graphic strings:

gAlignment	Builds a sequence of characters containing an aligned text or an aligned image.
gBackground	Builds a sequence of characters used to define the display color (background color of a text, circle, rectangle, ...).
gBackgroundGradient	Describes a display color in gradient mode.
gCircle	Builds a sequence of characters containing a circle or an ellipse.
gClickableLink	Adds a clickable link in a graphic string.
gCoord	Builds a sequence of characters used to define the position of the input cursor for a text.
gFont	Builds a sequence of characters describing the font that will be used for the text.
gFontBold	Builds a sequence of characters used to specify whether the text that follows the call to this function is in bold or not.
gFontItalic	Builds a sequence of characters used to specify whether or not the text that follows the call to this function is in italic type.
gFontSize	Builds a sequence of characters used to specify the size of the font for a text found in a graphic string.
gFontStrikeout	Builds a sequence of characters used to specify whether the text that follows the call to this function is crossed out or not.
gFontUnderlined	Builds a sequence of characters to specify whether or not the text that follows the call to this function is underlined.
gImage	Builds a sequence of characters containing an image.
gImageExists	Checks whether an image was loaded in memory.
gLine	Builds a sequence of characters containing a line.



gLink	Adds an "invisible" value into an element found in a List Box or Combo Box control.  Function kept for compatibility. It is recommended to use <b>gStoredValue</b> .
gLinkActivate	Allows (or not) retrieving the value defined by <b>gLink</b> in a List Box or a Combo Box control.  Function kept for compatibility. It is recommended to use <b>gStoredValueEnable</b> .
gPen	Builds a sequence of characters used to describe the color of the text and/or the color of the shape outline (rectangle or circle).
gRectangle	Builds a sequence of characters used to draw a rectangle.
gStoredValue	Adds an "invisible" value to an element in a List Box, Combo Box or Scheduler control.
gStoredValueEnable	Allows (or not) retrieving the value defined by <b>gStoredValue</b> in a List Box or a Combo Box control.
gToRetrieve	Describes the text value to display and to return when retrieving the value of an element found in a List Box control or in a Combo Box control.
gToText	Deletes the graphic elements from a character string that uses <b>gStoredValue</b> , <b>gImage</b> , ...
LoadImage	Loads an image in memory so that it can be used later by <b>gImage</b> .
LoadImageSheet	Loads an image set in memory so that it can be used later by <b>gImage</b> .

## 5.7 CD/DVD functions

The following functions are used to manage CD/DVD players:

CDEject	Opens or closes the door of CD/DVD player selected on the current computer.
CDList	Returns the list of CD/DVD players installed on the current computer.

## 5.8 Burn functions

The burn functions are as follows:

BurnerAddDirectory	Adds all the files found in a directory to the compilation.
BurnerAddFile	Adds a file to the compilation.
BurnerCancel	Cancels the current burn process.
BurnerCompilationInfo	Retrieves the characteristics of current compilation.
BurnerEject	Opens or closes the door of current burner.
BurnerErase	Erases the files found on a rewritable CD or DVD.
BurnerList	Returns the list of burners installed on the current computer.
BurnerMediaFound	Used to find out whether the burner is empty or not.
BurnerMediaInfo	Retrieves the characteristics of the CD/DVD found in the current burner.
BurnerMediaType	Used to find out and modify the format of the CD or DVD to burn.
BurnerPath	Returns the path of the current burner.
BurnerProperty	Used to find out and modify the properties of the current burner.
BurnerSave	Burns the files found in the compilation onto the CD or DVD.
BurnerSelect	Used to select the default burner.
BurnerStatus	Returns the status of the current burner.
BurnerVolumeName	Allows you to find out or modify the name of the CD or DVD to burn.

## 5.9 MCI functions

The MCI functions are as follows:

MCIBeginning	Positions at the beginning of a media file that was opened beforehand <b>MCIOpen</b> .
MCIClose	Closes a media file.

MCIEnd	Positions at the end of a media file that was opened beforehand <b>MCIOpen</b> .
MCIErr	Returns the number of the last error caused by a MCI function (Media Control Interface).
MCIExecute	Runs an MCI command (Media Control Interface).
MCIHeight	Returns the height (in pixels) of a media file used to display images.
MCIMsgErr	Returns the caption of the last error that occurred on a MCI function (Media Control Interface).
MCIOpen	Opens a media file.
MCIPause	Pauses playback of the media file previously opened ( <b>MCIOpen</b> ).
MCIPlay	Plays a media file.
MCIPosition	Returns or changes the current position in a media file previously opened ( <b>MCIOpen</b> ).
MCIPositionFormat	Fixes the format used to define the position in a media file (opened by <b>MCIOpen</b> ).
MCIRetrieve	Returns the last result of the MCI command run by <b>MCIExecute</b> .
MCISize	Returns or changes the size of a media file previously opened ( <b>MCIOpen</b> ).
MCIStatus	Returns the status of a media file that was opened beforehand <b>MCIOpen</b> .
MCIStopPlaying	Stops the execution of a media file that was opened beforehand ( <b>MCIOpen</b> ).
MCITrack	Returns the current track of a media file previously opened ( <b>MCIOpen</b> ).
MCITrackCount	Returns the number of tracks found in a media file that was opened beforehand ( <b>MCIOpen</b> ).
MCIVolume	Modifies or returns the sound volume.
MCIWidth	Returns the width (in pixels) of a media file used to display images.
MCIWindowSize	Changes the size and position of the display window of a media file previously opened ( <b>MCIOpen</b> ).

## 5.10 OCR (Optical Character Recognition) management functions

The following functions are used to manage the OCR technology integrated to WINDEV, WEBDEV and WINDEV Mobile:

OCRAddLanguage	Adds a language to the native OCR system.
OCRDeleteAllLanguages	Deletes all languages from the native OCR system.
OCRDetectTextArea	Returns all text areas of an image without reading the text.
OCRExtractText	Reads the text contained in an image.
OCRExtractTextBlock	Returns all the text blocks of an image.
OCRLoadLanguage	Adds a language to the native OCR system.
OCRModelPathDirectory	Used to set the directory that contains the language models for the native OCR system.
OCRParameter	Initializes the general options of the native OCR (Optical Character Recognition) system.

## 5.11 Image object detection functions

The following functions are used to detect objects in images:

AIDetect	Detects different elements in an image: object, faces, etc, ... This detection is achieved through artificial intelligence (AI).
AIDetectContour	Detects the contours of the shapes in an image.
AIDetectDrawing	Classifies freehand drawings, including circles, squares and triangles.
AIDetectModel	Runs a trained neural network model on an image. This function allows you to use artificial intelligence to detect objects in an image.

## 5.12 Video functions

The following functions are used to manage videos:

VideoCapture	Captures the images received from the camera: <ul style="list-style-type: none"><li>• as an image: takes a photo.</li><li>• as a video: records a video.</li></ul>
VideoConfigure	Allows you to configure the video capture driver.
VideoDisplay	Displays in a Camera control the video sequence currently transmitted by a "web camera" or by the camera found on the mobile device.
VideoGenerateThumbnail	Generates a thumbnail of the specified photo or video.
VideoListDriver	Lists the video capture drivers installed on the current computer.
VideoListParameter	Lists the values supported by the device camera for a given parameter.
VideoParameter	Returns or modifies the value of a parameter for the device camera.
VideoRequestPermission	Requests permission to use the device's camera.
VideoSetDriver	Connects the "Camera" control to a specific driver for video capture.
VideoStartApp	Starts the device's native camera application to record a video or to take a photo.
VideoStartDecoding	Starts decoding bar codes in a Camera control.
VideoStatus	Returns the status of a Camera control.
VideoStop	Stops playing or recording a video in a Camera control.
VideoStopDecoding	Stops decoding a bar code in a Camera control.

## 6.Databases

### 6.1 HFSQL management functions

The following functions are used to manage HFSQL data files:

Convert	Converts a numeric value into a binary string in order to perform a search on a numeric key.
HAccelerateSpeed	Reorganizes the internal structure of the indexes to optimize the speed for accessing the data.
HActivateAutoFilter	Enables an automatic filter on the linked files when browsing an XML file.
HActivateFilter	Enables the filter that was previously created for the specified data file (view or query).
HActivateTrigger	Re-enables a trigger that was disabled by <b>HDeactivateTrigger</b> .
HAdd	Adds: <ul style="list-style-type: none"><li>• the record found in memory into the data file (query or view).</li><li>• the record found in a Record variable into the data file (query or view).</li></ul>
HAlias	Creates a logical alias of a data file (or query) or cancels all existing aliases.
HBackward	Moves backward several records from the current position in the data file, according to a specified item.
HBuildKeyValue	Builds the value of a composite key to implement a filter ( <b>HFilter</b> ) or to perform a search ( <b>HReadSeekFirst</b> , <b>HReadSeekLast</b> , etc.).
HBuildKeyValueANSI	Regardless of the platform used, the data stored in the HFSQL files is in ANSI format.
HCancelAlias	Cancels an alias that was declared beforehand by <b>HAlias</b> .

HCancelDeclaration	Used to: <ul style="list-style-type: none"><li>• Delete a declaration made previously using <b>HDeclare</b>, <b>HDeclareExternal</b>, HDescribeFile. After this operation, the data file is no longer recognized by the HFSQL engine.</li><li>• Free the resources of a query (after calling <b>HExecuteQuery</b> or <b>HExecuteSQLQuery</b>).</li></ul>
HCancelSeek	Cancels the current search criterion.
HChangeConnection	Dynamically changes the connection associated with a data file.
HChangeDir	Changes the path to a data file (i.e. directory in which the file will be handled).
HChangeKey	Changes the browse item.
HChangeLocation	Modifies the search mode of physical data files (.FIC, .NDX, ...).
HChangeLogDir	Modifies the location of log files corresponding to an HFSQL data file.
HChangeName	Modifies the physical name of a data file.
HChangeRplDir	Modifies the location for the description of subscriber replica (RPL file).
HCheckIndex	Checks whether the data found in the index file (.NDX file) properly refers the data found in the data file (.FIC file).
HCheckMemo	Checks the structural integrity of memos in an HFSQL data file.
HCheckStructure	Defines the mode for comparing data files.
HCheckUnaltered	Checks whether one or all records in an unalterable data file have not been modified (with an hexadecimal editor for example).
HClose	Closes a data file or all the data files opened by the current user: all the corresponding physical data files are closed for the current user.
HCloseAnalysis	Closes the current analysis.
HCloseConnection	Closes a connection to a database.

HCompareItem	Compares two values by sorting them according to the specified HFSQL index item: all the sort options specified for the item are taken into account (sensitivity to the case, to the punctuation, sort direction, Unicode language, ...).
HComputer	Stores a unique computer number or identifier in order to use the log and transactions in network.
HConnect	Redefines one or more connection parameters via Native Connector (Access, SQL Server, MySQL or Oracle) or via an OLE DB access on a specific table or set of tables.
HConvertX	Converts a numeric value into a binary string in order to perform a search on a numeric key.
HCopyRecord	Copies: <ul style="list-style-type: none"><li>• the content of the current record (loaded in memory) to the current data file record.</li><li>• the content of the Record variable to the current data file record.</li></ul>
HCreateMasterReplica	Creates the description file of a Master Replica (logged replication or universal replication).
HCreateMovableReplica	Creates a movable replication.
HCreateSubscriberReplica	Creates the description file of a subscriber replica.
HCreateView	Creates an HFSQL view. This function uses the former view mechanism.
HCreation	Creates an empty data file (".FIC" extension) with the index file and the memo file if necessary.
HCreationIfNotFound	Performs the following operations: <ul style="list-style-type: none"><li>• If the file does not exist, creates an empty data file (".FIC" file) with the index file and the memo file if necessary. The function is equivalent to <b>HCreation</b>.</li><li>• If the file exists, opens the file.</li></ul>
HCross	Crosses a record in a data file.
HDBCcreation	Ends the description of the structure of an xBase data file through programming.
HDBDescribeField	Describes through programming each item found in the structure of an xBase file described by <b>HDBDescribeFile</b> .
HDBDescribeFile	Describes a file in dBase3 format (most common format) through programming.

HDBDescribeIndex	Describes through programming the different index files that will be created.
HDBIndex	Opens an xBase index file.
HDBOpen	Opens the xBase data file and the "memo" file if it exists.
HDBOpenNoLock	In single-user mode, opens an xBase data file without locking it.
HDBSortType	Returns or modifies the sequence of text items in the xBase files.
HDeactivateAutoFilter	Disables an automatic filter on the linked files when browsing an XML file.
HDeactivateFilter	Temporarily disables the filter on a data file (view or query).
HDeactivateTrigger	Disables a trigger.
HDeclare	Declares a description of data file (found in an analysis) in the current project.
HDeclareExternal	Temporarily imports into the current analysis the description of a data file from an existing HFSQL data file.
HDelete	Deletes a record from a data file (query or view).
HDeleteAll	Deletes all records from a data file, an HFSQL view or a query.
HDeleteTrigger	Deletes a trigger.
HDeleteView	Deletes a previously created view. This function uses the former view mechanism.
HDescribeConnection	Describes a new connection to an external database.
HDescribeFile	Describes a data file through programming.
HDescribeFullTextIndex	Describes a full-text index of data file created through programming.
HDescribeItem	Describes a file item through programming.
HDescribeLink	Describes a link between two data files through programming.
HDescribeTrigger	Adds or modifies a trigger on an HFSQL data file.
HDuplicateRecord	Duplicates the record read in a data file: the record found in memory is added into the data file (query or view).

HEndNoModif	Unlocks a file that was locked by the same program with <b>HNoModif</b> .
HError	Returns the number of the last error triggered by the HFSQL engine.
HErrorDuplicates	Used to check whether a duplicate error occurred.
HErrorInfo	Returns a detailed information about the last error triggered by the HFSQL engine.
HErrorIntegrity	Used to check whether an integrity error occurred.
HErrorLock	Used to check whether a lock error occurred.
HErrorModification	Used to: <ul style="list-style-type: none"><li>• return the value of a record item if a modification conflict occurs.</li><li>• check if a modification conflict occurred.</li></ul>
HErrorPassword	After the execution of a read or write function in a data file, used to find out whether an error caused by a wrong password occurred on this data file.
HErrorStatusModification	Returns the status of a record during a modification conflict.
HExecuteQuery	Declares a query created in the query editor to the HFSQL engine and runs this query.
HExecuteQuery-Asynchronous	Executes a SELECT query asynchronously.
HExecuteSQLQuery	Initializes a query written in SQL language and declares this query to the HFSQL engine.
HExecuteSQLQuery-Asynchronous	Executes an SQL query asynchronously.
HExecuteView	Runs a view that was created beforehand (can be used to refresh the view data for example). This function uses the former view mechanism.
HExportCSV	Exports the records found in a data file (HFSQL or OLE DB), a view or a query to a CSV file.
HExportJSON	Exports the records found in a data file (HFSQL or OLE DB), view or query to a JSON file.
HExportJSONString	Exports the records from a data file (HFSQL or OLE DB), view or query to a JSON string.

HExportXLS	Exports records from a data file (HFSQL or OLE DB), view or query to an Excel file (XLSX).
HExportXML	Exports the records found in a data file (HFSQL or OLE DB), view or query to an XML file.
HExtractMemo	Extracts the content of a binary memo item from an HFSQL file, a query or an HFSQL view to a physical file (on disk).
HFileExist	Used to find out: <ul style="list-style-type: none"><li>• whether a file exists, which means whether it was physically created (<b>HCreation</b> or <b>HCreationIfNotFound</b>).</li><li>• whether a view or a query was defined.</li></ul>
HFilter	Defines and enables a filter on a data file, view or query.
HFilterContains	Defines and enables a "Contains" filter on a data file, view or query.
HFilterIdentical	Defines and enables a filter used to find the exact value of a string item.
HFilterIncludedBetween	Defines and enables an "Included between" filter on a file, view or query.
HFilterStartsWith	Defines and enables a "Start with" filter on a file, view or query.
HFirst	Sets the position on the first record of a data file according to the specified browse item.
HFlush	Forces the operating system of the computer where the data files are found to write data onto the disk.
HForward	Moves several records forward from the current position in the data file, according to a specified item.
HFound	Checks whether the current record corresponds to the current filter or search.
HFree	Transforms the crossed records ( <b>HCross</b> ) into deleted records.
HFreePosition	Deletes a position saved by <b>HSavePosition</b> .
HFreeQuery	Free the resources of a query (after the call to <b>HExecuteQuery</b> or <b>HExecuteSQLQuery</b> ).
HFTListWord	Lists the words of the current record that will be taken into account by the full-text indexing.

HGetCurrentPosition	Returns the approximate position of current record in the data file.
HHistoryModification	Returns the modifications performed on one or more items of a given record. The result can be displayed in a list box or in a table to allow the user to view the modifications made to the specified file.
HImportHF55	Imports a Hyper File 5.5 file into an HFSQL Classic file.
HImportJSON	Imports a JSON file into a data file in HFSQL Classic format (data file described in the data model editor or declared by <b>HDeclare</b> , <b>HDeclareExternal</b> or <b>HDescribeFile</b> ).
HImportText	Imports a Text file into a data file in HFSQL Classic or Client/Server format (data file described in the data model editor, or declared by <b>HDeclare</b> , <b>HDeclareExternal</b> or <b>HDescribeFile</b> ).
HImportXLS	Imports an XLS file into a data file in HFSQL Classic or Client/Server format (data file described in the data model editor, or declared by <b>HDeclare</b> , <b>HDeclareExternal</b> or <b>HDescribeFile</b> ).
HImportXML	Imports an XML file into an HFSQL Classic or Client/Server data file (data file described in the data model editor or declared by <b>HDeclare</b> , <b>HDeclareExternal</b> or <b>HDescribeFile</b> ).
HIndex	Rebuilds the file index (".NDX" file).
HIndexingInProgress	Indicates that a reindex operation is currently performed on an HFSQL data file and returns the percentage of reindexing already performed.
HInfoAnalysis	Returns information about an analysis (WDD file).
HInfoFile	Returns the characteristics of an HFSQL Classic or Client/Server data file.
HInfoMemo	Returns the characteristics of binary and text memos.
HInfoReplica	Returns information about the specified replica.
HInitHF55	Used to access specific functions for managing xBase (or dBase) data if Native xBase Access provided with WINDEV or WEBDEV 14 is not available.
HInitSubscriber	Initializes the range of automatic identifiers for the description file of subscriber replica (".RPL" extension).



HJSONToRecord	Imports JSON data into the current record.
HLast	Sets the position on the last record of a data file according to a browse item.
HLinkMemo	Used to associate a file with a binary memo item or to cancel the existing link between a file and a binary item.
HListAnalysis	Returns the list of WINDEV or WEBDEV analyses (in HFSQL Classic format) available in a given directory.
HListConnection	Returns the list of connections currently described in the application.
HListCustomFolder	Returns the list of Custom-Folders (also called groups) defined in the analysis.
HListFile	Returns the list of files: <ul style="list-style-type: none"><li>• found in the current analysis or in a specific analysis recognized by the HFSQL engine. The files defined by <b>HDeclare</b>, <b>HDeclareExternal</b> and <b>HDescribeFile</b> are taken into account.</li><li>• available on a connection via a native access or via an OLE DB driver.</li><li>• for a group of files defined in the data model editor.</li></ul>
HListFullTextIndex	Returns the list of full-text indexes of a file (a query or a view) recognized by the HFSQL engine.
HListItem	Returns the list of items: <ul style="list-style-type: none"><li>• found in a data file ( a query or a view) known to the HFSQL engine. The data files defined by <b>HDeclare</b>, <b>HDeclareExternal</b> and <b>HDescribeFile</b> are taken into account.</li><li>• found in a Record variable.</li></ul>
HListKey	Returns the list of keys found in a file (query or view) recognized by the HFSQL engine.
HListLink	Returns the list of links (Merise) in the current analysis or in a specific analysis
HListProvider	Returns the list of OLE DB providers and/or Native Connectors installed on the current computer.
HListQueryParameter	Returns the list of parameters for a query created in the query editor.

HListREP	Returns the list of assignments for the data files handled by the current application, which means the list of physical files corresponding to the files described in the analysis.
HListStopWord	Returns the list of stop words used by a full-text index.
HListSynonym	Returns the list of synonyms used by a full-text index.
HListTrigger	Returns the list of triggers applied to one or more HFSQL data files.
HLockFile	Locks a data file and restricts the access to this data file for all other sites or applications.
HLockRecNum	Locks a record and restricts the access to this record for all the other applications.
HLogInfo	Adds comments to the log when saving the operation.
HLogRecreate	Used to re-create an empty log.
HLogRestart	Restarts the logging process on a file.
HLogStop	Stops the log process of a file.
HMergeView	Creates an HFSQL view from two previously created views ( <b>HCreateView</b> ). This function uses the former view mechanism.
HMigrateLinked-CompositeKey	Migrates the values of the linked composite keys coming from a file in Hyper File 5.5 format to the HFSQL Classic format.
HMode	Changes the mode and the method for locking data files.
HModify	Modifies the specified record or the record found in memory in the data file (query or view).
HModifyStructure	Used to update the structure of an HFSQL data file by performing an automatic data modification (also called data synchronization).
HNbRec	Returns the number of records in a file, query or HFSQL view: active records, crossed records, deleted records, etc.
HNext	Sets the position on the next data file record according to a browse item.
HNoModif	Forbids all the modifications on a data file (for all the programs, including the one that requested the no-modification policy).

HOnError	Customizes the management of HFSQL errors.
HOpen	Opens a data file.
HOpenAnalysis	Opens an analysis in HFSQL Classic format.
HOpenConnection	Opens a connection to a specific database.
HOptimize	Used to handle the idle periods of a software (period without processes) to optimize the queries and the browses that will be run thereafter.
HOptimizeQuery	Optimizes the Select queries by using idle times when handling an application (period without processes).
HOut	Used to find out whether the record on which you want to be positioned is located outside the data file, filter, view or query.
HPass	Defines the password used to create or open a data file.
HPrepareQuery	Initializes a query and declares this query to the database server in order to optimize the next executions of this query.
HPrepareSQLQuery	Initializes a query written in SQL language and declares this query to the database server in order to optimize the next executions of this query.
HPrevious	Positions on the previous file record according to a browse item.
HRead	Reads a record in a file according to a given record number.
HReadFirst	Sets the position on the first record according to a browse item.
HReadLast	Sets the position on the last record of a data file according to a browse item.
HReadNext	Sets the position on the next data file record according to a browse item.
HReadPrevious	Sets the position on the previous record of a file according to a browse item.
HReadSeek	Sets the position on the first record whose value for a specific item is greater than or equal to a sought value (generic search).

HReadSeekFirst	Sets the position on the first record whose value for a specific item is strictly equal to a sought value (exact-match search).
HReadSeekLast	Sets the position on the last record whose value for a specific item is less than or equal to a sought value (exact-match search).
HRecNum	Returns: <ul style="list-style-type: none"><li>• the number of the current record in the HFSQL data file.</li><li>• the number of the current record in the HFSQL view.</li></ul>
HRecordDate	Returns the date and time of the last write operation performed on a record in an HFSQL file (Classic or Client/Server).
HRecordToJSON	Retrieves the structure and value of the current record and exports them into a string in JASON format.
HRecordToString	Converts the content of the current record to string (in a data file, HFSQL view, query, etc.).
HRecordToXML	Retrieves the structure and the value of the current record and exports them into a character string in XML format.
HRecreateSubscriberReplica	Re-creates the description file of a subscriber replica (universal replication only) according to the information of the master replica (.RPM and .SYN files).
HRefreshView	Asks to recalculate the content of a materialized view.
HRegenerateFile	Regenerates a data file from its log.
HReset	Initializes: <ul style="list-style-type: none"><li>• one or all variables of file items with their default values.</li><li>• one or all items of a Record variable with their default values.</li></ul>
HRestorePosition	Restores the context of a previously saved file ( <b>HSavePosition</b> ): current record, filter, pointers.
HRetrieveItem	Returns the content of an item found in the current record (in the data file, view, query, ...).
HRetrieveRecord	Returns the content of the current record (in a file, a view or a query, ...).
HRplDeclareLink	Declares a (1, 1) (0, n) link between two tables.

HRplFilterProcedure	Defines the WLanguage procedure that will be called whenever a replication operation is performed on a specific file.
HRplManageFile	Defines the options used for the universal replication of a file: <ul style="list-style-type: none"><li>the replication direction.</li><li>the management mode of conflicts.</li></ul>
HRplManagetItem	Specifies the replication options for an item: the item can be replicated or not.
HRplPass	Sets the passwords used to protect the universal replication files.
HSave	Modifies the current record if it exists, otherwise adds a new record.
HSavePosition	Stores the current file context: current record, filter, pointers.
HSecurity	Enables or disables the automatic security mechanism on one or more data files.
HSeek	Positions on the first record of the data file whose value for a specific item is greater than or equal to a sought value (generic search by default).
HSeekFirst	Positions on the first file record whose value for a specific item is greater than or equal to a sought value.
HSeekLast	Sets the position on the last file record whose value for a specific item is less than or equal to a sought value.
HSetDuplicates	Enables or disables the management of duplicates on a unique key.
HSetIntegrity	Enables or disables the management of an integrity constraint on a file link.
HSetLog	Enables or disables log management on a file.
HSetMemo	Used to modify the management mode of memo items.
HSetPosition	Sets the position on a record, based on the approximate position of one of its items.
HSetREP	Enables or disables the management of .REP file.
HSetReplication	Temporarily disables (or re-enables) the replication.

HSetTransaction	Enables or disables the management of transactions for one or more files.
HSetTrigger	Enables or disables the management of triggers.
HSortView	Sorts an HFSQL view by creating an index on a view item. This function uses the former view mechanism.
HStatCalculate	Performs various statistical calculations on the file keys.
HStatDate	Returns the date of the last update for the index statistics.
HState	Used to find out the state of a record.
HStatNbDuplicates	Returns the number of duplicates for a given key item.
HStatNbRec	Returns the number of entries for a given key item.
HStatNbRecRange	Returns an estimate regarding the number of entries for a given key item in a given interval of values.
HStatTime	Returns the time of the last update for the index statistics.
HSubstDir	Associates the data directory specified in the analysis with a directory found on disk.
HSynchronizeReplica	Synchronizes the master replica and the subscriber replica: the operations performed on one of the replicas are transferred into the other replica.
HToFile	Copies a data source (file, query, view, ...) to a physical HFSQL file with the same description. This file is neither encrypted nor password protected.
HToltem	Assigns the specified value to an item of the current record.
HTransaction	Starts a transaction on the data files (HFSQL or accessed via a Native Connector) and creates the transaction file.
HTransactionCancel	If a transaction is in progress, cancels all the operations performed on the data files in transaction since the start of transaction.
HTransactionEnd	Validates the current transaction: <ul style="list-style-type: none"><li>the modifications performed on the data file since the start of transaction (<b>HTransactionStart</b>) are validated.</li><li>the transaction file is deleted (if the transaction is the last transaction in progress for a network application)</li><li>the records locked in read-only by the transaction are unlocked.</li></ul>

HTransactionFree	Transforms all the records "in transaction" into "Normal" records if these records do not belong to a transaction currently in progress.
HTransactionInProgress	Used to find out whether a transaction is in progress.
HTransactionInterrupted	Used to find out whether a transaction was interrupted (the transaction was neither validated nor canceled).
HTransactionStart	Starts a transaction on the data files (HFSQL or accessed via a Native Connector) and creates the transaction file.
HTriggerRecordBefore	Retrieves the value of the current record before the trigger(s) are executed.
HUnlockFile	Unlocks the records of a data file: <ul style="list-style-type: none"><li>locked by <b>HLockFile</b>.</li><li>individually locked by a locking read function.</li></ul>
HUnlockRecNum	Unlocks a record locked by: <ul style="list-style-type: none"><li><b>HLockRecNum</b>,</li><li>a read function used with a locking parameter (<b>HRead</b> associated with the hLockWrite or hLockReadWrite constant for example).</li></ul>
HVersion	Used to find out: <ul style="list-style-type: none"><li>whether the content of a file was modified.</li><li>whether the content of a file used by a query was modified.</li></ul>
HViewToFile	Saves the modifications performed in an HFSQL view (by <b>HModify</b> , <b>HDelete</b> or <b>HCross</b> ) in the corresponding data file. This function uses the former view mechanism.
HWrite	Writes a record into a data file without updating the indexes corresponding to all keys used in the file.
WithSpace	Adds or deletes the spaces found on the right of a text item when reading it.

## 6.2 Link functions

Link functions are used to link the interface with the data files (and vice versa). These functions are:

ControlToSource	Assigns the control value to the source of a control link.
-----------------	--

FileToPage	Automatically updates the controls of a page with: <ul style="list-style-type: none"><li>the values of the bound items in the current record (loaded in memory) of the data file described in the data model editor. This operation is performed regardless of the state of the controls (grayed, inactive or invisible).</li><li>the values of the bound WLanguage variables. This operation is performed regardless of the state of the controls (grayed, inactive or invisible).</li></ul>
FileToScreen	Automatically updates the controls in a window or page with: <ul style="list-style-type: none"><li>the values of the bound items in the current record (loaded in memory) of the data file described in the data model editor. This operation is performed regardless of the state of the controls (grayed, inactive or invisible).</li><li>the values of the bound WLanguage variables. This operation is performed regardless of the state of the controls (grayed, inactive or invisible).</li></ul>
PageToFile	Automatically updates: <ul style="list-style-type: none"><li>the value of the items in a data file with the value of the controls in the page. The data file items bound to the window controls are automatically updated with the values of these controls (even if they are grayed, inactive or invisible).</li><li>the value of the WLanguage variables with the value of the window controls. The WLanguage variables bound to the page controls are automatically updated with the values of these controls (even if they are grayed, inactive or invisible).</li></ul>
PageToSource	Automatically updates: <ul style="list-style-type: none"><li>the value of the items in a data file with the value of the controls in the page.</li><li>the value of the WLanguage variables with the value of the page controls.</li></ul>
ScreenToFile	Automatically updates: <ul style="list-style-type: none"><li>the memory value of the items in a data file with the value from the controls in the window or page. The data file items bound to the window controls are automatically updated with the values of these controls (even if they are grayed, inactive or invisible).</li><li>the value of the WLanguage variables with the value from the controls in the window or page. The WLanguage variables bound to the window controls are automatically updated with the values of these controls (even if they are grayed, inactive or invisible).</li></ul>

ScreenToSource	Automatically initializes: <ul style="list-style-type: none"><li>the memory value of the items in a data file with the value from the controls in the window or page.</li><li>the value of the WLanguage variables with the value of the controls found in the window or in the page.</li></ul>
SourceToPage	Automatically updates the controls of a page with: <ul style="list-style-type: none"><li>the values of the bound items in the current record (loaded in memory) of the HFSL data file described in the data model editor.</li><li>the values of the bound WLanguage variables.</li></ul>
SourceToScreen	Automatically updates the controls in a window or page with: <ul style="list-style-type: none"><li>the values of the bound items in the current record (loaded in memory) of the HFSL data file described in the data model editor.</li><li>the values of the bound WLanguage variables.</li></ul>

### 6.3 HFSQL Cluster management functions

The following functions are used to manage an HFSQL cluster:

HClusterAddNode	Enables a node in an HFSQL cluster.
HClusterDeleteNode	Disables a node in a HFSQL cluster.
HClusterIgnoreSynchro	Defines a node of the HFSQL cluster as data source to perform the cluster synchronization.
HClusterMasterNode	Returns the name of the master node in an HFSQL cluster.
HClusterNodeInfo	Returns the status of each cluster node by interrogating the coordinator.
HClusterParameter	Reads and modifies the parameters of an HFSQL cluster.
HClusterStart	Starts an HFSQL cluster. This cluster was stopped beforehand (by <b>HClusterStop</b> for example).
HClusterState	Returns the status of an HFSQL cluster by interrogating its coordinator.
HClusterStop	Suspends the execution of an HFSQL cluster. The nodes of the cluster will no longer accept any connection from the client applications.

HClusterSynchronization-ErrorInfo	Returns the error that occurred during the previous synchronization step of HFSQL cluster.
HClusterSynchronizationInfo	Returns the progress of synchronization for a node of an HFSQL cluster.

You also have the ability to use the following functions:

HLoadParameter	Reads a parameter that was saved from a stored procedure by <b>HSaveParameter</b> .
HSaveParameter	Saves a persistent value on the HFSQL server.
HDeleteParameter	Deletes a parameter previously saved by <b>HSaveParameter</b> .

### 6.4 HFSQL Client/Server management functions

The following functions are used to manage the HFSQL Client/Server engine:

HActivateServerTrigger	Re-enables a server trigger that was disabled by <b>HDeactivateServerTrigger</b> .
HAddGroup	Adds a group of users.
HAddLink	Adds an integrity rule between two data files on the server.
HAddScheduledBackup	Adds a scheduling for full backup (with or without differential backup) on the server defined by the connection.
HAddScheduledOptimization	Adds an optimization task of HFSQL Client/Server data files.
HAddScheduling	Creates a scheduled task on an HFSQL server: <ul style="list-style-type: none"><li>stored procedure,</li><li>backup,</li><li>optimization,</li><li>refresh of materialized view.</li></ul>
HAddTask	Adds a scheduled task on the server defined by the connection.
HAddUser	Adds a user to a database.
HBackup	Makes a backup copy of the content of an HFSQL server: all databases on the server, one or more databases, one or more data files.

HCancelBackup	Cancels a current backup.
HChangePassword	Changes the password of an HFSQL Classic or HFSQL Client/Server data file.
HCheckServerUpdate	Checks for the latest version of the HFSQL server that can be automatically updated.
HClearWorkingDir	Clears and destroys the temporary directory previously created when <b>HServerWorkingDir</b> was executed.
HConnectionQuality	Returns the quality level of connection: the higher the level is, the faster the connection will be.
HCopyFile	<p>Copies an HFSQL data file (.fic, .ndx and .mmo files if they exist):</p> <ul style="list-style-type: none"><li>• from the HFSQL server to the server (to perform a backup for example).</li><li>• from the HFSQL server to the client (to perform a local backup for example).</li><li>• from the client to the HFSQL server (to update the data files for example).</li><li>• from an HFSQL server to another HFSQL server.</li></ul>
HCreateServerTrigger	Adds or modifies a server trigger on the HFSQL server.
HDeactivateServerTrigger	Disables a HFSQL Client/Server server trigger on a server.
HDeleteBackup	Deletes a backup previously created with <b>HBackup</b> .
HDeleteDatabase	Deletes a database located on an HFSQL server.
HDeleteDirectory	Deletes a directory from an HFSQL Client/Server database.
HDeleteFile	Deletes the HFSQL data files (.fic, .ndx, .ftx and .mmo files if they exist) from the server.
HDeleteGroup	Deletes (from the server) a group of users associated with a connection.
HDeleteLink	Deletes an integrity rule between two data files on the server.
HDeleteParameter	Deletes a parameter previously saved by <b>HSaveParameter</b> .
HDeleteQuery	Deletes a query (used by stored procedures) from an HFSQL server.
HDeleteScheduledBackup	Deletes a scheduled backup from an HFSQL Client/Server server.

HDeleteScheduled-Optimization	Deletes a scheduled optimization task from HFSQL Client/Server data files.
HDeleteScheduling	<p>Deletes a scheduled task found on an HFSQL server:</p> <ul style="list-style-type: none"><li>• stored procedure,</li><li>• backup,</li><li>• optimization,</li><li>• refresh of materialized view.</li></ul>
HDeleteServerTrigger	Deletes a server trigger.
HDeleteSet	Deletes a set of stored procedures from an HFSQL server.
HDeleteTask	Deletes a scheduled task from an HFSQL Client/Server server.
HDeleteUser	Deletes a user associated with a connection to the HFSQL server.
HDescribeServerTrigger	Adds or modifies a server trigger.
HDisconnectClient	Displays a message on the Client computer(s) and disconnects the application.
HDropView	Removes a materialized view from the HFSQL server.
HDuplicateDatabase	Duplicates the database associated with a connection on an HFSQL server.
HEndNoDatabaseAccess	Re-allows the access (which means the connection) to one or more databases accessible via a connection.
HExecuteProcedure	Runs a stored procedure.
HExecuteScheduledBackup	Forces the execution of a scheduled backup.
HExecuteScheduling	<p>Immediately runs a scheduled task regardless of its schedule:</p> <ul style="list-style-type: none"><li>• stored procedure,</li><li>• backup,</li><li>• optimization,</li><li>• refresh of materialized view.</li></ul>
HInfoBackup	Returns information about one or more backups performed on an HFSQL Client/Server server.
HInfoDatabaseProperty	Used to find out the properties of a database on an HFSQL server.



HInfoDatabaseRights	Allows you to find out the rights granted to a user or to a group on a database.
HInfoFileProperty	Allows you to find out the properties of a data file located in an HFSQL server.
HInfoFileRights	Allows you to find out the rights granted to a user or to a group on an HFSQL Client/Server data file.
HInfoGroup	Returns information about the specified group of users.
HInfoLock	Returns information about the lock set on a data file, on one or all the records in a data file.
HInfoLog	Returns information about the server logs.
HInfoServer	Returns the specified information about the server.
HInfoServerProperty	Allows you to find out the properties of an HFSQL server.
HInfoServerRights	Allows you to find out the rights granted to a user or group of users on a server.
HInfoTask	Returns the characteristics of a scheduled task in a hScheduledTask variable].
HInfoUser	Updates the variables for user management with the information regarding the specified user.
HInfoView	Returns information about a materialized view.
HInfoViewRights	Allows you to find out the rights granted to a user or to a group on an SQL view or on a materialized view.
HListConnectedUser	Lists the users currently connected to one or more files handled by a Client/Server connection.
HListDatabase	Lists the Client/Server databases associated with a connection.
HListGroup	Returns the list of groups of users defined for a connection.
HListParameter	Returns the list of parameters saved from the stored procedures on the server.
HListScheduledBackup	Lists the full and differential backups that have been scheduled on HFSQL Client/Server.
HListScheduledOptimization	Lists the scheduled optimization tasks of HFSQL Client/Server data files for a connection.

HListScheduling	Lists the scheduled tasks defined on an HFSQL server: <ul style="list-style-type: none"><li>• stored procedure,</li><li>• backup,</li><li>• optimization,</li><li>• refresh of materialized view.</li></ul>
HListServer	Lists the HFSQL servers installed on a computer.
HListServerTrigger	Lists the different triggers available on a connection or on one of the connection files.
HListStoredElement	Returns the list of elements stored on an HFSQL server (sets of procedures, stored procedures or queries).
HListTask	Lists the scheduled tasks of an HFSQL Client/Server server for a given connection.
HListUser	Returns the list of users defined for a connection.
HLoadParameter	Reads a parameter that was saved from a stored procedure by <b>HSaveParameter</b> .
HManageServerUpdate	Used to manage the automatic update of HFSQL Client/Server servers by specifying the check frequency regarding the availability of updates.
HManageTask	Enables or disables a scheduled task on an HFSQL Client/Server server.
HModifyDatabaseProperty	Modifies the properties of a database located on an HFSQL server.
HModifyDatabaseRights	Modifies the rights granted to a user or to a group for a HFSQL Client/Server database.
HModifyFileProperty	Modifies the properties of an HFSQL file located on a server.
HModifyFileRights	Modifies the rights granted to a user or group on an HFSQL Client/Server data file.
HModifyGroup	Modifies the group information according to the elements found in the corresponding variables for group management.
HModifyScheduledBackup	Modifies a backup scheduling.
HModifyScheduled-Optimization	Modifies a scheduled optimization task on the HFSQL server defined by the connection.

HModifyScheduling	Modifies a scheduled task on an HFSQL server: <ul style="list-style-type: none"><li>• stored procedure,</li><li>• backup,</li><li>• optimization,</li><li>• refresh of materialized view.</li></ul>
HModifyServerProperty	Modifies the properties of an HFSQL server.
HModifyServerRights	Modifies the rights granted to a user or to a group on an HFSQL server.
HModifyTask	Modifies a scheduled task on the HFSQL server defined by the connection.
HModifyUser	Modifies the information about a user according to the elements found in the corresponding variables for user management.
HModifyViewRights	Modifies the rights granted to a user or to a group on a SQL view or on a materialized view.
HNoDatabaseAccess	Denies all new connections to a database or to all databases.
HNotifAddCCRecipient	Adds recipients for the notifications sent via the Control Centers (WDBal messaging tool).
HNotifAddEmailRecipient	Adds recipients for the notifications sent by email.
HNotifConfigure	Specifies and configures the server used to send notifications by the HFSQL server.
HNotifDeleteCCRecipient	Deletes the recipients of a notification sent via the Control Centers (WDBal messaging tool).
HNotifDeleteEmailRecipient	Deletes the recipients of an email notification.
HNotifListCCRecipient	Returns the list of recipients of a notification sent via the Control Centers (WDBal messaging tool).
HNotifListEmailRecipient	Returns the list of recipients for a notification by email.
HOnServerCall	Customizes the management of message display on the client computer and the management of disconnection from a client computer.
HOperationInProgress	Returns the list of automatic operations in progress on an HFSQL server.
HPriority	Finds out and modifies the priority of the calling application.

HPriorityClient	Modifies the priority of a client application.
HReconnect	Performs a reconnection to the server for all the interrupted connections. The uninterrupted connections are not modified.
HResetClient	Initializes the structure for managing the Client computers (HClient structure).
HResetGroup	Initializes the structure for group management (HGroup) with the default values.
HResetUser	Initializes the structure for user management (HUser) with the default values.
HRestoreBackup	Used to restore: <ul style="list-style-type: none"><li>• an immediate backup (performed by <b>HBackup</b> or via the HFSQL Control Center),</li><li>• a scheduled backup (described by <b>HAddScheduledBackup</b> or via the HFSQL Control Center.</li></ul>
HRetrieveBackup	Retrieves locally the content of a compressed backup (in ZIP format) performed on an HFSQL server.
HRetrieveLog	Creates a text file containing the server logs between two given dates.
HSaveParameter	Saves a persistent value on the HFSQL server.
HSendMessageToClient	Displays a message on the client computers.
HServerMaintenance	Checks and optimizes the system tables of an HFSQL server.
HServerStatus	Allows you to find out the status of a HFSQL server (this function uses the MantaManager service).
HServerWorkingDir	Returns the path of a temporary directory on the HFSQL server.
HSetCache	Used to configure the management of caches in the HFSQL Client/Server engine.
HSetServer	Used to find out and modify some settings of HFSQL Client/Server server.
HSimulateNetwork	Simulates the operating mode of HFSQL Client/Server on an ADSL or 3G network.

HStartServer	Allows you to start an HFSQL server (this function is using the MantaManager service).
HStartServerUpdate	Used to immediately start the automatic server update.
HStopServer	Stops an HFSQL server.
HSuggestKey	Returns a list of keys to add to data files to improve query performance.
HSuggestKeyEnd	Finishes collecting information on executed queries.
HSuggestKeyStart	Starts collecting information on executed queries.
HTrack	Starts tracking write actions (addition, modification or deletion) performed on a file found in an HFSQL server.
HTrackStop	Stops tracking a data file on an HFSQL server.
HTransactionIsolation	Configures the isolation mode of transactions for a connection to an HFSQL server.
HTransactionList	Returns the list of current or interrupted transactions on the server for the specified connection.
HTransactionRollback-ToSavepoint	Rolls the transaction back to the savepoint.
HTransactionSavepoint	Declares a transaction savepoint.
HUpdateLink	Updates on the server all the links of an HFSQL Client/Server data file.
HUpdateQuery	Creates or updates a query on an HFSQL server.
HUpdateSet	Creates or updates a set of procedures on an HFSQL server.
HUpdateView	Creates or updates the description of an SQL view or materialized view from the analysis to an HFSQL server.

## 6.5 SQL database access functions

The list of SQL functions available in WLanguage is as follows:

SQLAssociate	Automatically associates each column of the query result with a control or with a variable of the application.
SQLChangeConnection	Modifies the current connection.
SQLClose	Declares the end of the query execution and frees the memory resources allocated during the execution of the query.
SQLCol	Retrieves the content of the specified column from the query result, for the current line.
SQLColumn	Returns the characteristics of all the columns (or items): <ul style="list-style-type: none"><li>• for a given table.</li><li>• for a given query.</li></ul>
SQLColumnFormat	Returns the different types of columns supported: <ul style="list-style-type: none"><li>• by the data source identified by SQLConnect</li><li>• by the database accessed by a specific connection (connection described in the analysis or described by HDescribeConnection or HOpenConnection).</li></ul>
SQLConnect	Connects the current application to a database that must be interrogated via SQL.
SQLConnectWS	Connects the current application to a database that will be queried by SQL through a Webservice proxy (SOAP).
SQLDisconnect	Closes the current connection and frees the memory used by the connection.
SQLExec	Names and runs an SQL query.
SQLExecWDR	Runs the SQL code of a query created in the query editor.
SQLFetch	Goes to the next row (i.e next record) of the query result.
SQLFirst	Positions on the first row of the query result.
SQLGetCol	Retrieves the content of the specified column from the query result, for the current line.
SQLGetMemo	Retrieves the content of a Memo column from the query result, for the current row.
SQLGetTextMemo	Retrieves the content of a Text Memo column found in the query result, for the current line.

SQLInfo	Initializes the different SQL variables with information relative to the last query run.
SQLLast	Sets the position on the last row of the query result.
SQLListSource	Lists the data sources declared on a local site.
SQLListTable	List all the tables that can be accessed by the connected database system.
SQLLock	Used to lock: <ul style="list-style-type: none"><li>• The entire data table, in read and write mode. Its data cannot be accessed by other computers.</li><li>• The records selected by the query. This data cannot be accessed by the other computers. This method can be used to update records for example.</li></ul>
SQLModify	Modifies the content of a row in the query result.
SQLNext	Positions on the next row of the query result.
SQLPrevious	Positions on the previous row of the query result.
SQLReqExists	Checks the existence of a query.
SQLSeek	Positions on the specified result row of the query.
SQLTable	Transfers the result of a query to a Table control populated programmatically (a List Box or Combo Box control), with the possibility of Partial Fetch (the result is retrieved by blocks of rows).
SQLTransaction	Begins, commits or rolls back a transaction: <ul style="list-style-type: none"><li>• on the files of a database opened by SQLConnect,</li><li>• on the files of a database opened by a connection (OLE DB or Native Access/Native Connector) described in the data model editor or by HOpenConnection.</li></ul>

## 6.6 HBase functions

The following functions are used to manage the HBase databases:

hbCreateTable	Creates an HBase table or modifies the description of an HBase table.
hbDelete	Deletes: <ul style="list-style-type: none"><li>• one or more cells from an HBase table for one or more columns.</li><li>• a row from an HBase table.</li></ul>
hbDeleteTable	Deletes an HBase table.
hbGetTableDescription	Retrieves the description of an HBase table.
hbListTable	Retrieves the list of tables found on an HBase server.
hbRead	Reads cells (values of items) of a row (record) in an HBase table.
hbWrite	Writes cells (values of items) to one or more rows (records) in an HBase table.

## 6.7 HDFS functions

The following functions are used to manage files on a HDFS system:

hdfsCloseConnection	Closes a connection to a Hadoop server.
hdfsCopyFrom	Copies a file found on a Hadoop server to the local computer.
hdfsCopyTo	Copies a file located on the local machine to a Hadoop server.
hdfsCreateDirectory	Creates a directory on a Hadoop server.
hdfsDelete	Deletes a file or directory from a Hadoop server.
hdfsList	Lists the content of a directory found on a Hadoop server.
hdfsOpenConnection	Opens a connection to a Hadoop server.
hdfsRename	Renames a file or a directory found on a Hadoop server.

## 6.8 Replication management functions

The following functions are used to manage the different types of replication:

HChangeRplDir	Modifies the location for the description of subscriber replica (RPL file).
HCreateSubscriberReplica	Creates the description file of a subscriber replica.
HCreateMasterReplica	Creates the description file of a Master Replica (logged replication or universal replication).
HCreateMovableReplica	Creates a movable replication.
HSetReplication	Temporarily disables (or re-enables) the replication.
HInfoReplica	Returns information about the specified replica.
HInitSubscriber	Initializes the range of automatic identifiers for the description file of subscriber replica (".RPL" extension).
HRecreateSubscriberReplica	Re-creates the description file of a subscriber replica (universal replication only) according to the information of the master replica (.RPM and.SYN files).
HRplDeclareLink	Declares a (1, 1) (0, n) link between two tables.
HRplManageFile	Defines the options used for the universal replication of a file: <ul style="list-style-type: none"><li>• the replication direction.</li><li>• the management mode of conflicts.</li></ul>
HRplManageItem	Specifies the replication options for an item: the item can be replicated or not.
HRplPass	Sets the passwords used to protect the universal replication files.
HRplFilterProcedure	Defines the WLanguage procedure that will be called whenever a replication operation is performed on a specific file.
HRSAddConfig	Adds a replication between two HFSQL server onto the master server.
HRSWaitForDataProcess	Waits for: <ul style="list-style-type: none"><li>• the replication data on the master server to be sent to the subscriber server.</li><li>• the replication data received on the subscriber server to be entirely applied.</li></ul>

HRSExecute	Immediately runs a recurring replication between HFSQL servers: the replication is triggered before the scheduling.
HRSInfo	Reads the replication configuration for an HFSQL server that takes part in one or more replications.
HRSInit	Configures an HFSQL server in order for this server to be a master server or a subscriber server for a replication between HFSQL servers.
HRSListConfig	Lists the replications available on a master HFSQL server.
HRSModifyConfig	Modifies some parameters of an existing replication between two HFSQL servers.
HRSDeleteConfig	Deletes a replication between two HFSQL servers.
HSynchronizeReplica	Synchronizes the master replica and the subscriber replica: the operations performed on one of the replicas are transferred into the other replica.
ReplicInfo	Returns information about a replication.
ReplicOpen	Opens a replication in direct mode or with a replication server.
ReplicDeleteSubscription	Deletes the subscription of a replication on the replication server.
ReplicSynchronize	Synchronizes the data corresponding to a replication.

## 6.9 Memcached functions

The following functions are used to manage Memcached databases:

MemcachedAdd	Adds a new key/value couple into the Memcached cache.
MemcachedAddKeyGroup	Adds a new key/value couple into the Memcached cache on a group of servers.
MemcachedAddPrefix	Adds a value at the start of existing value in the cache for the given key.
MemcachedAddPrefix-KeyGroup	Adds a value at the start of existing value in the cache for the given key on a group of servers.
MemcachedAddSuffix	Adds a value at the end of existing value in the cache for the given key.

MemcachedAddSuffix-KeyGroup	Adds a value at the end of existing value in the cache for the given key on a group of servers.
MemcachedAssign	Modifies a key/value couple in the Memcached cache: <ul style="list-style-type: none"><li>• If the key exists, the value is modified.</li><li>• If the key does not exist, the key/value couple is added.</li></ul>
MemcachedAssignKeyGroup	Modifies a key/value couple in the Memcached cache on a group of servers: <ul style="list-style-type: none"><li>• If the key exists, the value is modified.</li><li>• If the key does not exist, the key/value couple is added.</li></ul>
MemcachedCAS	Modifies the value of a key if this one was not modified.
MemcachedCASKeyGroup	Modifies the value of a key on a group of servers if this one was not modified.
MemcachedDecrement	Decrements the value associated with a key. The key is created if it does not exist.
MemcachedDecrement-KeyGroup	Decrements the value associated with a key on a group of servers.
MemcachedDelete	Deletes a key from the Memcached cache.
MemcachedDeleteKeyGroup	Deletes a key from the Memcached cache on a group of servers.
MemcachedGetKeyGroup	Retrieves the values of keys stored on a group of Memcached servers.
MemcachedIncrement	Increments the value associated with a key.
MemcachedIncrement-KeyGroup	Increments the value associated with a key on a group of servers.
MemcachedModify	Modifies a key/value couple in the Memcached cache.
MemcachedModifyKeyGroup	Modifies a key/value couple in the Memcached cache on a group of servers.
MemcacheGet	Retrieves the values of keys stored on a Memcached server.

## 6.10 MongoDB functions

The following functions are used to manage MongoDB databases and GridFS:

MongoAdd	Adds one or more documents into a collection of MongoDB database.
MongoAggregationPipeline	Used to retrieve documents that enter a multi-stage pipeline and are transformed into aggregated results.
MongoCreate	Creates a connection to a MongoDB server or cluster.
MongoCreateCollection	Creates a collection in a MongoDB database while specifying the options of this collection.
MongoDeleteAll	Deletes all the documents (corresponding to the specified filter) from a collection of a MongoDB database.
MongoDeleteCollection	Deletes a collection from a MongoDB database.
MongoDeleteDatabase	Deletes a MongoDB database as well as its collections.
MongoDeleteOne	Deletes a document from a collection in a MongoDB database.
MongoExecuteCommand	Runs a generic command on a MongoDB database or on a MongoDB collection.
MongoFind	Finds documents in a MongoDB collection.
MongoGridFSDeleteFile	Deletes a file identified by its name from a MongoDB file system.
MongoGridFSDeleteFileByID	Deletes a file identified by its identifier from a MongoDB file system.
MongoGridFSGetFile	Retrieves (on the local computer) a file identified by its name and found in a MongoDB file system.
MongoGridFSGetFileByID	Retrieves (on the local computer) a file identified by its identifier and found in a MongoDB file system.
MongoGridFSGetFileInfo	Allows you to get information about a MongoDB file (gridFS).
MongoGridFSListFile	Retrieves the list of MongoDB files found in the MongoDB file system (gridFS).
MongoGridFSSendFile	Copies a file from the local computer to a MongoDB file system (gridFS).
MongoInfo	Returns information about a MongoDB database.



MongoModifyAll	Modifies documents in a collection of a MongoDB database.
MongoModifyOne	Modifies a document in a collection of MongoDB database.

### 6.11 Redis management functions

The following functions are used to manage a Redis database:

RedisConnect	Opens a connection to a Redis server (or cluster).
RedisExecuteCommand	Executes a command on a Redis server and retrieves the result.
RedisGet	Gets the value of a key from a Redis server as a string.
RedisPipelineAdd	Adds a Redis command in a pipeline.
RedisPipelineGet	Retrieves the result of the execution of a Redis command that was added by <b>RedisPipelineAdd</b> .
RedisPipelineReset	Ignores the results of the execution of a Redis command that was added by <b>RedisPipelineAdd</b> .
RedisPublish	Publishes a message on a Redis broadcast channel.
RedisSet	Changes the value of a key on a Redis server.
RedisSubscribe	Subscribes to a Redis broadcast channel.
RedisUnsubscribe	Cancels a subscription to a Redis broadcast channel.

### 6.12 3-Tier functions

The following WLanguage functions support 3-tier architecture:

Deserialize	Deserializes a buffer or a character string containing the data from a class, structure, array (including an associative array), queue, stack, list or advanced variable, as well as their subelements.
FileToMemory	Automatically initializes the values of the members found in a class or in a structure with the values of the items of the current file record.

MemoryToFile	Automatically initializes the memory value of the items for the current file record with the values of the members of a class or structure.
Serialize	Transforms the following elements into a specific format: <ul style="list-style-type: none"><li>• a structure (and its subelements),</li><li>• a class (and its subelements),</li><li>• an array (including the associative arrays),</li><li>• a queue,</li><li>• a stack,</li><li>• a list.</li></ul>

### 6.13 SAP functions

These functions allow you to use the BAPI functions of a SAP server:

SAPConnect	Connects a SAP R/3 system from the information given in the SAP structure.
SAPDisconnect	Disconnects a SAP R/3 system.
SAPExecute	Calls a synchronous function or BAPI on a SAP system.

### 6.14 Salesforce functions

The following functions are used to manage Salesforce databases:

SFAdd	Creates a record in a file of a Salesforce database.
SFConnect	Allows you to authenticate yourself beside a Salesforce platform.
SFConvertLead	Converts the leads.
SFDelete	Deletes records from a Salesforce database.
SFDisconnect	Disconnects: <ul style="list-style-type: none"><li>• the Salesforce platform.</li><li>• the current Salesforce session.</li></ul>
SFEmptyRecycleBin	Clears the specified records from the Salesforce recycle bin.
SFExecuteProcess	Requests the execution of a specific process.

SFExecuteQuery	Runs an SOQL query (Salesforce Object Query Language) on a Salesforce database.
SFForwardQuery	Continues to run a SOQL query (Salesforce Object Query Language) on the records of a Salesforce database.
SFMerge	Merges the Salesforce records (also called Salesforce objects).
SFModify	Modifies the records of a file defined in Salesforce.
SFRead	Retrieves the Salesforce records from their identifiers.
SFSearch	Performs a search among the records found in a Salesforce database.
SFSeekDeleted	Seeks and retrieves the records (Salesforce objects) deleted during the specified period.
SFSeekModified	Seeks and retrieves the records (Salesforce objects) modified during the specified period.
SFUndelete	Restores previously deleted Salesforce records.

### 6.15 Local storage management functions

The following functions are used to perform a local storage:

LocalStorageAdd	Adds a value to the local storage.
LocalStorageAvailable	Indicates whether the local storage is available or not for the current browser.
LocalStorageDelete	Deletes a value from the local storage.
LocalStorageDeleteAll	Deletes all the values from the local storage.
LocalStorageGet	Retrieves a value from the local storage.
LocalStorageOccurrence	Returns the number of values for the local storage.
LocalStorageValueName	Returns the name of a value for the local storage.

## 7. Print functions

The print functions are as follows:

iAddAttachment	Adds an attachment in a PDF file generated by a PDF print export.
iAddBookmark	Adds a bookmark in the report viewer or when exporting to PDF.
iAreaHeight	Calculates the height of a fixed-width area required to print the entire text that uses the current font.
iBorder	Prints a border at specified coordinates.
iChangeSubReportSource	Modifies through programming the report associated with a sub-report in a composite report.
iCloseReport	Stops printing the current report immediately.
iColumnEnd	Forces a column break in a multi-column report.
iColumnNum	Returns the number of the current column in a multi-column report.
iConfigure	Configures the printer for a print performed in WLanguage: <ul style="list-style-type: none"><li>• by opening the configuration window.</li><li>• by modifying the printer used for the current print.</li></ul>
iConfigureReport	Configures the printer by opening the configuration window (WINDEV and WINDEV Mobile).
iCreateFont	Creates a new print font.
iCurrentFile	Returns the name of the file currently generated.
iDestination	Configures the print destination.
iDidotFont	Selects the unit that will be used for the height of the fonts (created by <b>iCreateFont</b> ): DIDOT point or PICA point.
iDirImageHTML	Used to select the directory in which the images must be generated during a print in HTML format.
iDocumentCanceled	Used to find out whether the current print was canceled by the user.
iDocumentPrinted	Determines whether the user printed the document from the report viewer.

iEllipse	Prints an ellipse (in a rectangle).
iEndPrinting	Signals the end of the document to print and actually starts printing the data stored in the printer spooler via the following functions: <b>iPrint</b> , <b>iPrintImage</b> , <b>iNewLine</b> , etc.
iEndReport	Forces the report print to stop.
iEscape	Sends an ESCAPE command or a data command to a printer.
iFont	Selects the default font.
iForceComplement	Forces the printing of a Body Complement block.
iGroupAdd	Adds a report to a group of reports.
iGroupConfigure	Modifies the parameters of a report found in a group of reports.
iGroupPrint	Starts printing a report found in a group of reports.
iHLine	Prints a horizontal line at specified coordinates.
ilmageHeight	Calculates the height of the image to print (in millimeters).
ilmageWidth	Calculates the width of the image to print (in millimeters).
iInfoPrinter	Retrieves the characteristics of the current or default printer.
ilnitInternalReport	Initializes the parameters of an internal report used by a report.
ilnitReportQuery	Initializes the query linked to the report (for a query with parameters).
ilnitReportQueryConnection	Initializes the query linked to the report.
ilnitSubReport	Initializes the parameters of a subreport of a composite report.
iLastFile	Returns: <ul style="list-style-type: none"><li>• the full name of the last file generated during the print (PDF, RTF, text, ...).</li><li>• the list of created duplicate files.</li></ul>
iLine	Prints a line at specified coordinates.
iLink	Adds a link to a URL address into the HTML page currently printed.

iListNestedReports	Returns: <ul style="list-style-type: none"><li>• the list of nested reports currently printed (reports run by <b>iPrintReport</b> in one of the processes of the report currently printed).</li><li>• the list of sequenced reports currently printed (iSequenceAdd).</li></ul>
iListPrinter	Returns the list of printers installed on the current computer.
iMargin	Defines the "logical" print margins.
iMMToPica	Converts the coordinates of a point (expressed in millimeters) into system coordinates.
iNestedHeaderFooter	Prints (or not) the page headers and footers of nested report.
iNewLine	Prints a line (in Windows standard, with a rounded end) at specified coordinates.
iPageEnd	Forces the move to the next page during the report print.
iPageHeight	Calculates the height of printable page (in millimeters) while taking the top or bottom margins into account.
iPageHeightLeft	Calculates the available height (in millimeters) left on the current page while taking into account the defined margins (top or bottom).
iPageNum	Returns or initializes the number of the page currently printed.
iPageWidth	Calculates the width (in millimeters) of the printable page while taking the left or right margins into account.
iParameter	Retrieves the printer configuration and configures the print.
iParameterDuplicate	Configures the automatic creation of the duplicate copy generated during the next print.
iParameterExport	Used to configure the exports performed from the report viewer: <ul style="list-style-type: none"><li>• parameters for sending emails.</li><li>• parameters for creating the different files.</li></ul>
iParameterPDF	Defines the options for the generated PDF file.

iParameterPreview	Configures the print preview: <ul style="list-style-type: none"><li>• display a message regarding the resources of current computer.</li><li>• display or not the ribbon panes (as well as the buttons found).</li><li>• the export options available in the mini preview window.</li></ul>
iParameterReport	Configures the printer by opening the configuration window.
iParameterViewer	Configures the report viewer.
iParameterWatermark	Adds a watermark into a report (report created in the report editor or printed report).
iParameterXLS	Defines the options for the XLS file generated during the print.
iPicaToMM	Converts the coordinates of a point (expressed in system coordinates) into millimeters.
iPreview	Configures the print destination.
iPrint	Sends the character string passed as parameter to the print buffer.
iPrintArea	Prints a text in a rectangular area.
iPrintAreaHTML	Prints a text in HTML format in a rectangular area.
iPrintAreaMarkdown	Prints Markdown text in a rectangular area.
iPrintAreaRTF	Prints rich text format (RTF) in a rectangular area.
iPrintBarCode	Prints a bar code (inside a rectangle).
iPrintBlock	Forces a report block to print.
iPrintBodyComplement	Forces a body complement block to print while specifying the height of complement block.
iPrintDiagram	Prints a diagram from: <ul style="list-style-type: none"><li>• a Diagram Editor control,</li><li>• a "wddiag" file,</li><li>• a Diagram variable.</li></ul>
iPrintDoc	Prints a document in "docx" format.
iPrintDuplicate	Prints a duplicate copy of a report.
iPrintImage	Sends the image file to print to the print buffer.

iPrintingNation	Modifies the current language for the next print (programmed print or report print).
iPrintPDF	Prints the content: <ul style="list-style-type: none"><li>• of a PDF file.</li><li>• of a buffer containing a PDF.</li><li>• of a PDF Reader control.</li></ul>
iPrintReport	Prints a report created with the report editor.
iPrintSignature	Prints an image of the signature in a rectangular area.
iPrintWord	Sends the character string passed in parameter to the print "buffer".
iPrintXLS	Prints the current worksheet of a document in "xlsx" or "xls" format.
iReportPrintingStatus	Returns the status of the current print.
iReset	Reinitializes the print settings (margins, font, paper format, orientation, ...) stored in the print library.
iResetNbPages	Reinitializes the current number of pages, used in the preset controls such as "Number of pages" and "Page number and number of pages".
iRoundBorder	Prints a border with rounded outlines at specified coordinates.
iSelectFont	Creates a new print font from the standard window for font selection...
iSequencingAdd	Adds a report into a sequence of report prints.
iSequencingAddDiagram	Adds a Diagram document into a sequence of report printouts.
iSequencingAddDoc	Adds a print of Word Processing document into a sequence of report prints.
iSequencingAddPDF	Adds a print of PDF document into a sequence of report prints.
iSequencingAddXLS	Adds an XLS or XLSX document into a sequence of report print jobs.
iSequencingPrint	Prints a sequence of reports.
iSetAdditionalXMP	Adds XMP tags to the metadata of the generated PDF file.

iSkipLine	Forces one or more line break.
iSkipPage	Generates a page break: the corresponding code is sent to the printer queue.
iSubstBlock	Substitutes one block of the report with another block while the report is printed.
iTextHeight	Calculates the height of the font for the text to print (in millimeters).
iTextWidth	Calculates the width (in millimeters) of the text to print, according to the specified font.
iTransparentMagenta	Modifies the management mode of Magenta color in order for this color to be considered as being transparent (or not) in all the forthcoming prints.
iVLine	Prints a vertical line at specified coordinates.
iWindowCancel	Configures the print cancel window..
iXPos	Used to manage the horizontal position (X-coordinate) of print cursor in the page.
iYPos	Used to manage the vertical position (Y-coordinate) of print cursor in the page.

## 8.Maths

### 8.1 Financial functions

The financial functions are as follows:

FinCurrentVal	Returns the current value of investment.
FinDecreasingRedemption	Calculates the amortization value of a good over a specific period of time, via the formula for fixed-rate decreasing amortization.
FinError	Returns the number of the last error caused by a Finxxx function (financial functions).
FinFutureVal	Returns the future value of an investment.
FinInterestRate	Calculates the interest rate for a loan over a specific period of time with fixed payments.
FinLinearRedemption	Calculates the value of linear amortization of a product over a specific period of time.
FinNetCurrentVal	Returns the net current value of investment.
FinPaymentNb	Returns the number of payments required to pay off a capital according to a given rate.
FinPeriodInterest	Calculates for a given period the amount of interests due for a loan paid off by fixed periodic payments with a fixed interest rate.
FinRedemption	Creates an amortization matrix for a fixed-rate loan over a defined period of time.
FinRepayVal	Returns the amount of each regular payment for an investment with fixed interest rate and fixed payment.

## 8.2 Chart management functions

The following functions are used to manage charts

gr3DSPParameter	Retrieves or modifies a parameter for drawing a chart in 3D Spatial mode.
grAddData	Adds a data into a chart.
grAddTimeData	Adds data at a precise moment in a time chart.
grAutoRefresh-CategoryLabel	Identifies or defines whether the category labels are refreshed whenever the chart is drawn.
grAutoRefreshSeries	Identifies or defines whether a series is refreshed whenever the chart is drawn.
grAxisPosition	Indicates the position where one of the chart axes (X or Y-coordinates) will be drawn.
grAxisTitle	Defines the title of X-axis and the title of Y-axis in a chart.
grAxisTitleFont	Modifies the font used for the titles of the chart axes.
grAxisVisible	Configures how the axes are displayed in Line or Column charts.
grBarColor	Initializes the color of a specific bar in a Column or Stacked Column chart.
grCategoryColor	Initializes the category color for a chart whose type is: <ul style="list-style-type: none"><li>• Pie,</li><li>• Column,</li><li>• Bubble chart,</li><li>• Waterfall.</li></ul>
grCategoryLabel	Initializes the label of a data category (or section) in a chart.
grClearPict	Erases the chart drawing (in an Image control or in a window).
grColor	Returns and initializes the color of different chart elements.
grColumnAggregation	Defines the algorithm for aggregating data in a grouped or stacked Column chart.
grColumnSpacing	Indicates the spacing between each data category in a column chart or in a Minimum/Maximum stock chart.
grComparativeSeriesPosition	Fixes the position (left or right) of a series in a comparative Histogram chart.

grCreate	Creates a specific type of chart.
grDeleteAll	Erases a chart and destroys this chart.
grDeleteSeries	Deletes a data series from a chart.
grDestinationBMP	Defines a BitMap file as destination of a chart (BMP means BitMap).
grDestinationControl	Defines an Image control as destination of a chart.
grDestinationEMF	Defines a meta file as chart destination (EMF means Enhanced Meta File).
grDestinationImage	Defines an Image variable as destination of a chart.
grDestinationWMF	Defines a meta file as chart destination.
grDestinationWnd	Defines a window as chart destination.
grDonutHolePercentage	Modifies the percentage corresponding to the radius of the hole in a Donut chart.
grDraw	Draws a chart according to the specified parameters.
grExist	Checks the existence of a chart in memory.
grGetData	Retrieves a data from a chart.
grGradient	Enables or disables the display of color gradients in a chart.
grGraduate	Indicates the step for the marks on the vertical/horizontal axis of a chart (which means the interval between two marks).
grGridlines	Displays or hides the chart gridlines.
grHMAddData	Adds a data into a Heatmap chart.
grHMCaptionX	Modifies the caption of a category on the X-axis of a Heatmap chart.
grHMCaptionY	Modifies the caption of a category on the Y-axis of a Heatmap chart.
grHMColor	Specifies the colors of different values in a Heatmap chart.
grHMDeleteColor	Deletes the colors specified by <b>grHMColor</b> from a Heatmap chart.
grHMGetData	Retrieves data from a Heatmap chart.



grHolePercentage	Modifies the percentage that corresponds the hole radius in the following charts: <ul style="list-style-type: none"><li>• Donut,</li><li>• Radial Bar chart.</li></ul>
grImageSize	Defines the size of the image containing the chart.
grIncreaseData	Adds a value to a data in a chart.
grInfoXY	Returns information about the series found at a specific chart point.
grLabel	Indicates various parameters of additional legend in a chart.
grLabelFont	Modifies the font used for the category labels of a chart.
grLegend	Indicates the presence and position of legend in a chart.
grLegendFont	Modifies the font used for the chart legend.
grLineThickness	Retrieves and/or modifies the value of the line thickness in a line chart.
grLoadParameter	Restores the parameters used to draw a chart.
grMask	Formats the data displayed by <b>grLabel</b> or by <b>grTooltip</b> .
grMenu	Defines the type of context menu displayed by a chart.
grMenuOption	Modifies the text of an option in the context menu of a chart.
grOrientation	Modifies the orientation of chart axes.
grOrigin	Modifies the start and end marks on one of the chart axes.
grOverlayChart	Used to display two different types of charts in the same chart.
grParameter	Retrieves or modifies a chart parameter.
grPiePullOut	Pulls out a section (or all sections) in a chart.
grPieStartAngle	Indicates the start angle of first section in a Pie chart, Semi-circular chart or Donut chart.

grPointColor	Initializes the color of a specific point in one of the following chart types: <ul style="list-style-type: none"><li>• Line,</li><li>• Scatter,</li><li>• Bubble chart.</li></ul>
grPointInfo	Returns the screen coordinates of a point or the screen coordinates of a value found in a chart.
grPointSize	Configures the size of a specific point in a Line or Scatter chart.
grPointType	Modifies the shape of point for a specific category or value found in a Line, Area, Scatter and Radar chart.
grPrint	Prints a chart.
grRaised	Used to display a chart with raised effect.
grRBAddData	Adds data to a Radial Bar chart.
grRBMaximumValue	Sets the maximum value in a Radial Bar chart.
grRBRotationDirection	Sets the rotation direction in a Radial Bar chart.
grRealTimeInterval	Sets the time interval visible in a time chart with real-time display.
grRotation	Used to: <ul style="list-style-type: none"><li>• Turn a chart around one of its axes in order to better view the data.</li><li>• Find out the rotation angle of a chart on one of its axes.</li></ul>
grSaveBMP	Saves a chart that was previously drawn in a file in BMP format.
grSaveEMF	Saves a chart that was drawn beforehand in a file in EMF format.
grSaveImage	Saves the image of a chart in an Image variable.
grSaveParameter	Saves the parameters of a chart in the format of a compressed string.
grSaveWMF	Saves a chart that was drawn beforehand in WMF format.
grScatter3DAddData	Adds a new point to display in a 3D Scatter chart.
grScatter3DAltitudeColor	Defines the different colors used according to the altitude levels of a series found in a 3D Scatter chart.

grScatter3DDelete-AltitudeColor	Deletes the altitude colors specified by <b>grScatter3DAltitudeColor</b> .
grScatter3DDelete-MarkCaption	Deletes the captions of marks found in a 3D Scatter chart.
grScatter3DGridlines	Displays the gridlines for the different planes of a 3D Scatter chart.
grScatter3DLinkPoint	Links (or not) the points found in a "3D Scatter" chart.
grScatter3DMarkCaption	Used to specify the mark caption for a 3D Scatter chart.
grScatter3DProjection	Indicates whether the points found in a series of a 3D Scatter chart must be be projected on a given plane.
grScatterAddDataXY	Adds a data into a "Scatter" chart.
grScatterGetDataXY	Retrieves data from a "Scatter" chart.
grScatterLinkPoint	Links (or not) the points found in a "Scatter" chart.
grScatterPointSize	Defines the size of points in a "Scatter" chart.
grScatterSeriesColor	Initializes the color of a series in a "Scatter" chart.
grScatterSeriesLabel	Initializes the label of data series in a "Scatter" chart.
grSeriesColor	Initializes the color of a series or the color of a section.
grSeriesLabel	Initializes the label of a data series in a chart.
grSeriesLineType	Modifies the type of line for a series found in a Line, Area, Scatter and Radar chart.
grSeriesOpacity	Modifies the opacity factor of a series in a chart.
grSeriesPointType	Modifies the type of point for a series found in a Line, Area, Scatter and Radar chart.
grSeriesSecondaryAxis	Defines whether a series will be drawn on the secondary axis.
grSeriesType	Defines the type of series used in a composite chart (chart containing several types of charts).
grSmoothing	Smoothes a Line chart or a Scatter chart by using the "cubic splines" algorithm.
grSourceCategoryLabel	Defines the source of the category labels.
grSourceSeries	Defines the source of a series.

grStartAngle	Sets the starting angle of the first section in the following types of chart: <ul style="list-style-type: none"><li>• Pie, Semi-circular or Donut.</li><li>• Sunburst.</li><li>• Radial Bar chart.</li></ul>
grStockAddData	Adds data into a stock chart (candlestick, barcharts or minmax).
grStockGetData	Retrieves a data from a stock chart (candlestick, barcharts or minmax).
grSunburstAddData	Adds a data to a Sunburst chart.
grSunburstGetData	Retrieves a numeric value from a Sunburst chart.
grSunburstPullOut	Pulls out a section (or all the sections) in a Sunburst chart.
grSunburstRootSection	Changes or specifies the path of the root in a Sunburst chart.
grSunburstSectionColor	Changes the color of a section in a Sunburst chart.
grSunburstStartAngle	Specifies the starting angle of the first section in a Sunburst chart.
grSurfaceAddData	Adds a data into a Surface chart.
grSurfaceAltitudeColor	Defines the color used for the altitude (Z axis) of a surface chart.
grSurfaceDelete-AltitudeColor	Deletes the altitude colors specified by <b>grSurfaceAltitudeColor</b> .
grSurfaceDelete-MarkCaption	Deletes the captions of marks found in a Surface chart.
grSurfaceGetData	Retrieves a data from a Surface chart.
grSurfaceGridlines	Displays the gridlines for the different planes of a Surface chart.
grSurfaceMarksCaption	Allows you to specify the mark caption for a Surface chart.
grSurfaceMesh	Enables or disables the mesh on a Surface chart.
grTimeLevelMask	Defines the display mask for a level of time scale in a chart.
grTimeLevelMax	Defines the number of levels to display in the time scale of a chart as well as the maximum level.

grTitle	Modifies the caption and/or position of the chart title.
grTitleFont	Modifies the font used for the chart title.
grTooltip	Displays and formats: <ul style="list-style-type: none"><li>the tooltip associated with each section of a chart (section of a pie chart, column of a column chart, etc.). Also allows you to find out the characteristics of the tooltip associated with each chart section.</li><li>a custom tooltip, associated with a point.</li></ul>
grType	Identifies or modifies the type of chart.
grWaterfallCategoryType	Indicates the category type of a Waterfall chart.
grWaterfallColor	Indicates the color for a type of category in a Waterfall chart.
grWndSize	Defines the size of the window containing the chart.
grZoom	Defines the zoom of a chart.

### 8.3 Math functions

Abs	Returns the absolute value of a numeric value or numeric expression (integer or real).
ArcCos	Returns the arc cosine of a numeric value (integer or real).
ArcSin	Returns the arc sine of a numeric value (integer or real).
ArcTan	Returns the arc tangent of a numeric value (integer or real).
ArcTan2	Returns the arc tangent 2 of the values passed in parameter.
Conversion	Converts a value from a unit to another one.
Cos	Calculates the cosine of an angle.
CoTan	Calculates the cotangent of an angle.
DecimalPart	Returns the decimal part of a number.
DecimalToSexagesimal	Returns the sexagesimal angle (in base 60) corresponding to a decimal angle.
Exp	Calculates the exponential of a numeric value (reverse of Napierian logarithm).

Factorial	Returns the factorial of an integer number.
IntegerPart	Returns the integer part of a number.
IsEven	Identifies an even number.
IsOdd	Identifies an odd number.
Ln	Calculates the Napierian logarithm (reverse of exponential) of a numeric value.
Log	Calculates the decimal logarithm of a numeric value.
Power	Raise a number to a power.
Root	Returns the root of a number.
Round	Returns the rounded value of a numeric value according to the specified number of decimal places.
RoundCeil	Returns the value of a numeric value rounded to the nearest greater integer.
RoundFloor	Returns the value of a numeric value rounded to the nearest lower integer.
RoundToMultiple	Returns a numeric value rounded to the nearest multiple of another numeric value.
SexagesimalToDecimal	Returns the decimal angle corresponding to a sexagesimal angle.
Sin	Returns the sine of an angle.
Tangent	Calculates the tangent of an angle.

Other functions that can be used:

Random	Returns a random number.
InitRandom	Initializes the generator of random numbers.
Max	Returns the greatest value passed as parameter.
Min	Returns the lowest value.
Mean	Calculates the mean of several elements: <ul style="list-style-type: none"><li>elements found in an array,</li><li>numeric values, ...</li></ul>

NumToString	Converts a numeric value (integer, real or currency) to a string according to a specified format.
Sum	Calculates the sum of the array elements.

## 8.4 Binary values management functions

The following functions are used to manage binary values:

BinaryAND	Returns the result of a logical AND (bit by bit) between two values.
BinaryNOT	Returns the result of a logical NOT (bit by bit) performed on a value.
BinaryOR	Returns the result of the (bitwise) logical OR between two values.
BinaryXOR	Returns the result of the (bitwise) logical exclusive OR between two values.

## 8.5 Matrix management functions

The following functions are used to manage the matrices:

MatAdd	Adds two matrices of same dimensions (same number of rows and same number of columns).
MatCopy	Creates the copy of a matrix.
MatCreate	Creates a matrix containing 0 row and 0 column.
MatDelete	Deletes an existing matrix.
MatDeterminant	Calculates the determinant of a square matrix (the number of rows is equal to the number of columns).
MatError	Identifies the type of the last error caused by a Matxxx function (functions for matrix management).
MatExist	Checks the whether the matrix exists in memory.
MatFill	Initializes all the elements found in a matrix of a given size.
MatFloatAdd	Adds a value to each matrix element.
MatFloatMultiply	Multiplies each matrix element by a value.
MatInvert	Calculates the inverse of a square matrix.

MatMultiply	Multiplies two matrices.
MatNbColumn	Returns the number of columns found in a matrix.
MatNbLine	Returns the number of rows found in a matrix.
MatRead	Reads the value of a matrix element.
MatReadColumn	Reads the value of all the elements found in a matrix column.
MatReadLine	Reads the value of all elements found in a matrix row.
MatStack	Compresses the memory footprint occupied by a matrix.
MatTranspose	Calculates the transposed matrix.
MatWrite	Writes an element into a matrix.

## 8.6 Statistical functions

The statistical functions are as follows:

StatAverage	Calculates the arithmetic, geometric or harmonic mean for a series of values.
StatAverageDeviation	Calculates the average deviation of values in relation to their arithmetic mean.
StatCorrelation	Calculates the coefficient of correlation between two series of values.
StatCovariance	Calculates the covariance between two series of values.
StatError	Returns the error number for the last error caused by a Statxxx function (statistical functions).
StatMax	Calculates the maximum value for a series of values.
StatMin	Calculates the minimum value for a series of values.
StatStandardDeviation	Calculates the standard deviation for a series of values.
StatStandardDeviationP	Calculates the standard deviation for a full series of values.
StatSum	Calculates the sum for a series of values.
StatVariance	Calculates the variance for a series of values.
StatVarianceP	Calculates the variance for a full series of values.

## 8.7 MATLAB functions

The following functions are used to access MATLAB:

mlDeleteVariable	Deletes a variable from the MATLAB session.
mlEnd	Ends a MATLAB session. This session was initialized by <b>mlInitialize</b> .
mlExecute	Evaluates and runs a command on a MATLAB session.
mlGetVariable	Retrieves the value of a variable for the MATLAB session.
mlInitialize	Initializes a new independent MATLAB session with an empty workspace in order to run functions.
mlListVariable	Lists the variables found in the MATLAB session.
mlLoadWorkspace	Loads all the variables of a MATLAB file in the session.
mlResetWorkSpace	Deletes all the variables from the MATLAB session.
mlSaveWorkspace	Saves the session workspace in a MATLAB file.
mlSetVariable	Writes the value of a variable for the MATLAB session.

**Caution:** MatLab must be installed on the Web server and must be configured so that Internet users can access it.

## 8.8 Memory area management functions

The following functions are used to manage the memory areas:

fMemExist	Checks the existence of a shared memory zone.
fMemOpen	Opens a memory zone shared between several applications.
fMemWait	Waits for the end of process about the notifications of modification of a shared memory area.

# 9.Other functions

## 9.1 Certificate management functions

The following functions are used to handle the Windows certificates:

CertificateCheckDuplicate	Checks the signature of a duplicate.
CertificateCheckFile-SignatureBuffer	Checks the correspondence between a signature and a file. The signature was stored as a Buffer variable.
CertificateCheckFile-SignatureFile	Checks the correspondence between a signature and a file. The signature was kept as an external file.
CertificateCheckString	Checks the correspondence between a signature and a string.
CertificateExtract	Extracts a certificate from a signature buffer or from a print duplicate.
CertificateExtract-ExtensionValue	Returns the value of a certificate extension.
CertificateList	Returns the list of certificates available on the computer.
CertificateLoad	Loads a certificate from a file or a buffer.
CertificateOpenProperties	Opens the Windows window used to find out the properties of a certificate.
CertificateSelect	Opens a window for selecting the certificates. The listed certificates are the certificates installed in the personal store.
CertificateSignExecutable	Signs an executable file with a certificate.
CertificateSignFile	Creates the signature of a file. This signature can be stored in a Buffer variable or in a text file.
CertificateSignString	Creates the signature of a character string.
TrustedCertificateAdd	Adds a certificate to the list of trusted root authorities used to validate certification strings.
TrustedCertificateReset	Removes all the certificates added by <b>TrustedCertificateAdd</b> .

## 9.2 Registry management functions

The following functions are used to manage the Windows registry:

RegistryCopyKey	Copies a registry key with all its subkeys and values.
RegistryCreateKey	Creates a key in the Windows registry.
RegistryDeleteKey	Deletes a subkey from the Windows registry.
RegistryDeleteValue	Deletes a value from the Windows registry.
RegistryExist	Checks the existence of a key in the Windows registry.
RegistryFirstSubKey	Identifies the path of first subkey of specified key in the Windows registry.
RegistryListKey	Lists the sub-keys of a registry key.
RegistryListValue	Returns the name (and possibly the type) of values for a key found in the registry.
RegistryNextKey	Identifies the key found after the specified key in the Windows registry.
RegistryQueryValue	Reads a value in the Windows registry.
RegistryRename	Renames a key in the registry.
RegistrySeek	Seeks a character string in the registry.
RegistrySetValue	Writes a value to a key in the Windows registry.
RegistryValueType	Returns the type of value found in the registry.

## 9.3 Archiving functions

The following WLanguage functions are used to create and manage archives:

zipAddDirectory	Adds all the files found in a directory and in its subdirectories into an archive and compresses them.
zipAddFile	Adds a file (of any type) into an archive in CAB, ZIP, WDZ and 7z, TAR or TGZ (TAR.GZ) format and compresses it.
zipAddFileList	Adds a list of files (of any type) into an archive in ZIP, CAB, WDZ or 7z format and compresses it.
zipChangePath	Modifies the stored path of a file in an archive (in ZIP, WDZ or 7z format).

zipClose	Closes an archive.
zipCompressionLevel	Changes the compression level used when creating an archive in ZIP format or in 7z format.
zipCreate	Creates an archive.
zipCreateExe	Creates a self-extracting archive as a 32-bit executable (.EXE).
zipCurrentFile	Returns the name of the archived file currently handled by the functions for adding and extracting files.
zipDeleteAll	Deletes all files from an archive.
zipDeleteFile	Deletes a file from an archive.
zipDeleteFileList	Deletes a list of files from an archive.
zipExist	Used to find out whether an archive exists.
zipExtractAll	Extracts all files from an archive and decompresses them.
zipExtractDirectory	Automatically extracts and decompresses files from a directory in the file tree of an archive to a physical location.
zipExtractFile	Extracts a file from an archive and automatically decompresses it to a physical location or in memory.
zipExtractFileList	Extracts and decompresses a list of files found in an archive to a physical location.
zipExtractPath	Returns the path of a file found in the archive.
zipFileSize	Returns the size of one of the files found in the archive (in CAB, RAR, WDZ, ZIP, 7z, TAR or TGZ (TAR.GZ) format): <ul style="list-style-type: none"><li>the compressed size of the file.</li><li>the size of file before compression.</li></ul>
zipFilter	Returns the list of archive formats supported by the current platform, in the format expected by the filter of <b>fSelect</b> .
zipFindFile	Finds a file in an archive (in CAB, ZIP, RAR, WDZ or 7z format).



zipInfoFile	Returns the characteristics of a file found an archive: <ul style="list-style-type: none"><li>• The name of the file and its stored path.</li><li>• The initial size of the file.</li><li>• The compressed size of file.</li><li>• The date and time of file creation.</li><li>• The date and time of file modification.</li><li>• The date and time of file access.</li><li>• The file encryption mode.</li></ul>
zipIsMulti	Returns the type of archive: single-part archive or multi-part archive.
zipListFile	Returns the list of files found in an archive.
zipMerge	Merges the different sub-archives found in a multi-part archive to create a single-part archive.
zipMsgError	Returns the message associated with an error code on an archive.
zipNbFile	Returns the number of files found in an archive (in CAB, RAR, ZIP, WDZ or 7z format).
zipNbPart	Returns the number of sub-archives (parts) found in an archive in CAB, RAR, ZIP, WDZ, 7z, TAR or TGZ (TAR.GZ) format.
zipNbPartNeeded	Returns the number of parts of a given size required to split a single-part archive.
zipOpen	Opens an existing archive.
zipOpenCAB	Opens an existing archive in CAB format.
zipOpenRAR	Opens an existing archive in RAR format.
zipPassword	Defines the password used to add files into the specified archive and to extract files from the specified archive (RAR, ZIP and WDZ format).
zipSelectFile	Selects files from an archive.
zipSize	Returns the size of archive (in ZIP, RAR, CAB, WDZ or 7z format): <ul style="list-style-type: none"><li>• the size of all compressed files found in the archive.</li><li>• the size of all files found in the archive before compression.</li></ul>
zipSplit	Splits an archive into several sub-archives of a given size.

**Remark:** The archiving functions can be used in a thread without locking the other threads during their execution.

## 9.4 XLS file handling functions

The following functions are used to manage XLS files:

xlsAddWorksheet	Adds or inserts a new worksheet into an Excel document.
xlsClose	Closes an XLS file.
xlsColumnName	Retrieves the title of a column found in an XLS file.
xlsColumnType	Returns the type of data entered in a column of an XLS file.
xlsCoordinatesToName	Retrieves the name of a cell, XLS worksheet or Spreadsheet control according to its coordinates.
xlsCurrentWorksheet	Used to find out and modify the current worksheet in an XLS file.
xlsData	Retrieves the content of a cell found in an XLS file.
xlsDeleteWorksheet	Deletes a worksheet from the Excel document.
xlsMsgError	Returns the caption of the last error caused by an XLS function.
xlsNameToCoordinates	Returns the coordinates of a cell, XLS file or spreadsheet control according to its name.
xlsNbColumn	Returns the number of columns found in a worksheet of an XLS file.
xlsNbRow	Returns the number of rows found in an XLS file.
xlsNbWorksheet	Returns the number of worksheets found in an XLS file.
xlsOpen	Opens an Excel file (xls orxlsx files).
xlsSave	Saves an XLS document.
xlsVersion	Returns the Excel version that was used to save the file.

### 9.5 PDF file management functions

The following functions are used to manage PDF files:

PDFAddImage	Adds an image to a page in an existing PDF file.
PDFAddWatermark	Adds a watermark to all or some pages of an existing PDF file.
PDFClose	Closes the PDF file loaded in memory in a pdfDocument variable.
PDFExtractPage	Returns an Image type variable containing a page from a PDF file.
PDFGetSignature	Gets the signature of a PDF document as a buffer.
PDFGetSignatureDate	Gets the date of one of the signatures of a PDF document.
PDFIsProtected	Checks whether the PDF file requires a password in order to be read.
PDFMerge	Merges several existing PDF files into a single PDF file. The files are merged one after another. The paths of PDF files can be found in an array.
PDFMergeWithPassword	Merges several existing and password-protected PDF files into a single PDF file.
PDFNbSignatures	Gets the number of signatures in a PDF document.
PDFNumberOfPages	Returns the total number of pages found in a PDF file.
PDFOpen	Opens a PDF document and loads it in a pdfDocument variable.
PDFPassword	Protects a PDF document with a password.
PDFSave	Saves the content of a pdfDocument variable in a ".PDF" file.
PDFSign	Applies an approval signature to an existing PDF.
PDFToText	Extracts the text found in a PDF file.

### 9.6 HTML file management functions

The following functions handle HTML documents:

HTMLBuildString	Generates a UTF-8 string containing an HTML document.
HTMLEnableDebugger	Enables the HTML/JavaScript debugger on CEF-based HTML controls: HTML Display or HTML Editor control.
HTMLFindElementByClass	Searches for elements whose "class" attribute matches a specific value in an HTML document (or in an HTML node).
HTMLFindElementByID	Searches for the element whose "ID" attribute matches a specific value in an HTML document (or in an HTML node).
HTMLFindElementByName	Searches for elements whose "name" attribute matches a specific value in an HTML document (or in an HTML node).
HTMLFindElementByTag	Searches for elements that correspond to a tag in an HTML document (or an HTML node).
HTMLInsertAfter	Adds a node after the specified node in an HTML document.
HTMLInsertBefore	Adds a node before the specified node in an HTML document.
HTMLInsertFirstChild	Adds a child at the start of the children of the current node in an HTML document.
HTMLInsertLastChild	Adds a child after the child of the current node in an HTML document.
HTMLModifyAttribute	Adds or changes an attribute of the current node of the HTML document.
HTMLOpen	Creates an htmlDocument variable from an HTML document.
HTMLRemove	Removes the specified node and its descendants in an HTML document.
HTMLRemoveAttribute	Removes a given attribute
HTMLRemoveChild	Removes a given node in the current node of an HTML document.
HTMLSave	Saves the content of an htmlDocument variable to an HTML file.
HTMLToPDF	Converts an HTML string or buffer to a buffer containing a PDF document.

HTMLUnwrapAllChildren	Deletes the specified node in an HTML document without deleting its children, which take its place.
HTMLWrap	Wraps the specified node in a given node of an HTML document.
HTMLWrapAllChildren	Wraps all children of the given node in a new node in an HTML document.

## 9.7 Functions for managing multilingual applications

The following functions are used to manage multilingual applications:

ChangeCharset	Retrieves and/or changes the character set used: <ul style="list-style-type: none"><li>by all the screen fonts.</li><li>by all the operations performed on the character strings (functions and/or operators).</li></ul>
ChangeKeyboard	Gets or sets the language associated with the keyboard.
ChangeSeparator	Changes the value of the separators (decimal separator or thousand separator) used in the numeric masks for internationalization.
LanguageToCharset	Returns the character set corresponding to a language.
LanguageToName	Returns the name of a language.
LanguageToNation	Returns the nation corresponding to a language.
LoadError	Loads the file containing the error messages in the desired language into memory.
MessTranslate	Retrieves a message found in the code of a program in the current language.
Nation	Defines or identifies the display language (captions, messages, ...) used for the windows, pages, reports and controls.
NationFramework	Defines or identifies the language of the framework.
NationRightToLeft	Allows you to find out whether the current language is written from right to left.
NationToLanguage	Returns the language corresponding to a nation.
NationToName	Returns the name of the language corresponding to a nation.

NationToSubLanguage	Returns the default sub-language corresponding to a nation.
---------------------	---

## 9.8 Clipboard management functions

The following functions are used to manage the clipboard:

Clipboard	Retrieves the text or image found in the system clipboard.
ClipboardFormat	Identifies the format of information found in the clipboard.
ClipboardHeight	Calculates the height of an image (in bitmap format) found in the clipboard.
ClipboardRTF	Retrieves an RTF string from the clipboard (the content of the clipboard is in RTF format).
ClipboardUnicode	Returns the text found in the clipboard in Unicode format.
ClipboardWidth	Calculates the width of an image (in bitmap format) found in the clipboard.
ToClipboard	Writes text or image information into the system clipboard.
ToClipboardRTF	Writes a character string in RTF format into the clipboard.

## 9.9 Mouse management functions

The following functions are used to manage the mouse:

ControlOver	Identifies the control hovered over with the mouse cursor.
CursorDisplay	Displays or hides the mouse cursor.
CursorCapture	Directs all the events associated with the mouse toward a specific window or control, regardless of the screen object where the mouse event occurs.
CursorPos	Returns and modifies the position of the mouse cursor.
HourglassNetwork	Displays or hides an endless progress bar: <ul style="list-style-type: none"><li>in the status bar of an iOS application.</li><li>in the title bar of windows found in an Android application.</li></ul>

HourGlass	Transforms the mouse cursor into hourglass (and conversely). In Android and iOS, displays an endless progress bar in a window.
MouseXPos	Returns the horizontal position (X) of the mouse cursor relative to the specified window or control. Used in a mouse or stylus process (click, hover, left or right button down, etc.).
MouseYPos	Returns the vertical position (Y) of the mouse cursor relative to the specified window or control. Used in a mouse or stylus process (click, hover, left or right button down, etc.).

## 9.10 Drag and Drop management functions

The following functions are used to manage Drag and Drop:

DnDAccept	Indicates the action accepted by the target of Drag/Drop and manages the mouse cursor during Drag/Drop.
DnDCacheDashElement	Adds a widget into a Dashboard control during Drag and Drop.
DnDCacheData	Indicates the type of data and the data to copy/move during Drag/Drop.
DnDCursor	Manages the mouse cursor and specifies the action allowed during Drag and Drop.
DnDEvent	Indicates the procedure to run during an event of Drag/Drop.
DnDGetData	Retrieves a specific type of data during Drag/Drop.
DnDIsDataAvailable	Checks whether a specific type of data is available during Drag/Drop.
ExplorerAccept	Enables or disables the ability to perform "Drag and Drop" from the Windows explorer to a WINDEV window or control.
ExplorerRetrieve	Retrieves the number and the name of the files "dropped" from the explorer.

## 9.11 System functions

### 9.11.1 Recycle bin management functions

The following functions are used to manage the recycle bin:

RecycleBinClear	Clears the recycle bin of the system.
RecycleBinDelete	Displays a confirmation message for deleting a file from the recycle bin.
RecycleBinListFile	Lists the files found in the recycle bin of the system.
RecycleBinRestore	Restores a file found in the recycle bin of the system.

### 9.11.2 Windows Scheduler management functions

The following functions are used to manage the Windows scheduler:

SchedulerAddTask	Creates a task in the Windows scheduler (if this task does not already exist).
SchedulerAddTrigger	Creates a trigger for a scheduled task.
SchedulerDeleteTask	Deletes a scheduled task.
SchedulerDeleteTrigger	Deletes a trigger from a scheduled task.
SchedulerModifyTask	Modifies the parameters of a scheduled task.
SchedulerModifyTrigger	Modifies the parameters of a trigger of a scheduled task, i.e. the parameters that will define the execution of a task.
SchedulerOpen-TaskProperties	Opens the property page of a scheduled task.
SchedulerReset	Populates the ScheduledTask and/or TriggerScheduledTask structures with the default values.
SchedulerTaskList	Retrieves the list of scheduled tasks.
SchedulerTaskProperties	Reads the properties of a scheduled task and updates the ScheduledTask structure.
SchedulerTaskStatus	Gets or sets the status of a task created in the Windows scheduler.
SchedulerTriggerProperties	Reads the properties of trigger of a scheduled task and updates the TriggerScheduledTask structure.

9.11.3 Windows event management functions

The following functions can be used to manage events:

EndEvent	Cancels the interception of a Windows event on a WINDEV control or window (implemented by <b>Event</b> ).
EndTimer	Ends the execution of a timer triggered by <b>Timer</b> .
EndTimerSys	Ends the execution of a timer triggered by <b>TimerSys</b> .
Event	Intercepts a Windows event on a control, a group of controls or a WINDEV window.
EventExist	Indicates whether a specific procedure is run when a Windows event is sent to a control or window.
Handle	Returns the system "Handle" (HWND) of a WINDEV control or window.
HandleParentNext	Forces the parent of the next window to open.
Multitask	MultiTask is used to: <ul style="list-style-type: none"><li>• define a timeout,</li><li>• give control back to Windows,</li><li>• give control back to Windows and to WLanguage.</li></ul>
PostMessage	Sends a Windows message to a control or to a window.
SendMessage	Sends a Windows message to a control or to a window.
Timer	Periodically and automatically calls a WLanguage procedure.
TimerSys	Periodically and automatically calls a WLanguage procedure.
Wait	Temporarily stops the program execution.

9.11.4 Windows interactive notification management functions

The following functions are used to handle interactive notifications:

SysNotificationClear	Removes all interactive notifications previously sent by the current application.
SysNotificationCompatible	Checks if the operating system supports interactive notifications.
SysNotificationRemove	Removes a previously sent interactive notification.

SysNotificationRemoveID	Removes all previously sent interactive notifications that match the specified identifier (ID).
SysNotificationSend	Sends an interactive notification to the system in order to be displayed.

9.11.5 Miscellaneous Windows functions

The miscellaneous Windows functions are as follows:

API	Runs a function found in an external DLL.
APIParameter	Modifies the options for calling <b>API</b> and <b>CallDLL32</b> .
BitcoinValidAddress	Checks whether a Bitcoin wallet address is valid.
CallDLL32	Runs a function found in an external DLL.
CallInterface	Runs a method of an interface of an object implemented in a DLL external to the WINDEV framework.
COMCallMethod	Runs a method of an interface of a COM object instantiated by <b>COMCreateInstance</b> .
COMCreateInstance	Instantiates a COM object (Component Object Model).
COMQueryInterface	Calls the QueryInterface method of the specified COM object to get a specific interface of this object so that it can be used with <b>COMCallMethod</b> .
CreateShortcut	Creates a shortcut on the Windows desktop, in the "Start" menu of Windows or in a specific directory.
DeleteShortcut	Deletes a shortcut previously created by CreateShortcut from the Windows desktop, from the "Start" menu of Windows or from a specific directory.
FreeDLL	Frees the 32-bit library (DLL) that was loaded in memory by <b>LoadDLL</b> .
HiWord	Returns the two high bytes of an integer.
IBANValid	Checks the validity of an International Bank Account Number (IBAN).
IconInstall	Creates an icon (associated with a program) in a group of Windows programs, with the associated command line.
Instance	Returns the "system handle" (HINSTANCE) of the current program.

KeyPressed	Checks which key is pressed.
LoadDLL	Loads the specified library (DLL) in memory.
LoWord	Returns the two low bytes of an integer.
MakeInteger	Builds a 4-byte integer from two 2-byte integers.
OpenSSLCheck	Checks, via a public key, that the signature of data performed with the OpenSSL library is correct.
Ping	Checks whether a network address can be reached by using the ICMP protocol (similar to the PING network utility).
sComputeCrc16	Checks a character string before and after transmission between WINDEV applications and WEBDEV sites.
sComputeCrc32	Calculates the Cyclical Redundancy Check (CRC) of buffer.
SendKey	Simulates the keystrokes and the mouse actions.
Transfer	Copies a block of bytes or a fixed-length string from a memory address into another one.
WindowsVersion	Returns the runtime mode of Windows.

9.11.6 System-specific functions

The system functions are as follows:

SysAndroidVersion	Returns information about the Android version used by the application.
SysBatteryOptimBackground	Used to manage battery optimization: <ul style="list-style-type: none"><li>Identifies the battery optimization mode for the current application.</li><li>Prompts the user to disable battery optimization for a given application on the device.</li></ul> This function can be used to tell the operating system to stop preventing the application from running in the background.
SysCacheExternalStorage	Returns the path of directory that will be used to store the data in cache on the external storage space of application.
SysChangeEnvironment	Modifies the environment variables of the operating system found on the current computer.

SysChangeScreenResolution	Modifies the resolution of a screen.
SysColor	Identifies or modifies the color of a Windows element.
SysColorRes	Returns the resolution of the screen in number of colors.
SysDetectLockedSession	Detects the change of lock for the Windows session.
SysDeviceInfo	Retrieves information from the mobile device where the application is run
SysDir	Returns the path of a system directory.
SysDirExternalStorage	Returns the path of the directory that will be used to store files in the external storage space of the Android device.
SysDirStorageCard	Returns the path of the root directory of the primary external storage space.
SysEnvironment	Returns the environment variables of the operating system for the current computer.
SysErrorMode	Modifies the display mode of errors.
SysGetDC	Retrieves the DC (Device Context) of a window, control or screen.
SysGetExtLink	Returns the command linked to an extension.
SysIconAdd	Adds an icon into the taskbar (lower-right corner of the screen).
SysIconAddImage	Adds an icon that overlays the application icon in the taskbar.
SysIconBadge	Returns or modifies the value displayed in the badge associated with the icon of an iOS or Universal Windows 10 App application.
SysIconDelete	Deletes an icon from the taskbar (next to the clock).
SysIconDeleteImage	Deletes the icon that overlays the application icon in the taskbar.
SysIconDisplayTooltip	Displays a tooltip for an icon found in the taskbar.
SysIconize	Used to find out whether a window is minimized.
SysIconModify	Modifies the icon file and/or the hover text of the icon in the taskbar (lower-right corner of the screen).



SysIdentifier	Returns the unique identifier of the current mobile device.
SysIMEI	Returns the IMEI number of an Android phone.
SysInstance	Returns the instance of a module.
SysIOSVersion	Returns information about the iOS version used by the application.
SysLinkExtension	Associates an extension (.txt, .avi, etc.) to a default execution program.
SysListScreen	Returns the list of screens connected to one or more video cards.
SysListScreenResolution	Returns the possible resolutions for a screen.
SysListVideoCard	Returns the name of the video cards connected to the current computer.
SysMemoryExternalStorage	Returns the amount of available memory on the external storage area of the device.
SysMetric	Returns the resolution of a specific element.
SysNameExe	Returns the name and full path of an application.
SysNameMainScreen	Returns the name of the main screen for the computer.
SysNation	Returns the nation currently used by the system.
SysNbExternalStorage	Returns the number of external storage spaces found on an Android device.
SysNumberScreen	Returns the number of screens currently connected to the current computer.
SysPowerStatus	Used to get various information about the battery (main or secondary) of the device on which the application is running.
SysRecentDocAdd	Adds a file into the list of recently opened documents.
SysRecentDocList	Returns the list of recently opened documents.
SysReleaseDC	Frees the current DC (Device Context) retrieved by <b>SysGetDC</b> .
SysRemovableExternal-Storage	Used to find out whether the external storage space corresponds to the movable memory (SD Card).

SysScreenOrientation	Allows you to: <ul style="list-style-type: none"><li>• get the screen orientation of the device (Android, iOS or Universal Windows phone).</li><li>• set the screen orientation of the device (Android phone).</li></ul>
SysScreenRectangle	Returns the coordinates of the display rectangle corresponding to all the screens.
SysScreenResolution	Returns the resolution of a screen.
SysSerialNum	Returns the IMEI number of the current device.
SysSetFocus	Sets focus to a window or control.
SysShutdown	Stops the system or closes the current Windows session.
SysSpace	Returns the physical memory space for the current computer.
SysStandby	Allows you to: <ul style="list-style-type: none"><li>• enable and disable the standby mode (Android)</li><li>• display the system screen used to configure the standby mode (Universal Windows Apps)</li></ul>
SysStatusExternalStorage	Returns the availability status of the external storage space of the device.
SysStatusStorageCard	Returns the availability status of the storage card on the device.
SysTheme	Returns the current theme used on the mobile device.
SysThumbnailAddButton	Adds a button into the thumbnail of the application in the taskbar.
SysThumbnailDeleteAll	Deletes all the buttons associated with the thumbnail of the application from the taskbar.
SysThumbnailDeleteButton	Deletes one of the buttons found in the thumbnail of the application from the taskbar.
SysThumbnailModifyButton	Modifies one of the buttons found on the application thumbnail in the taskbar.
SysUnlinkExtension	Deletes the association between an extension and a command that was created by <b>SysLinkExtension</b> .
SysValidHandle	Checks the validity of a window handle.

SysVersion	Returns information about the PHP version used on the current server.
SysWinActive	Returns the handle of the active window or activates a window.
SysWindowsVersion	Returns information about the Windows version used on the current computer.
SysWinHandle	Returns the handle of a window identified by its title.
SysWinShow	Modifies the display status of a window.
SysWinTitle	Returns the title of the window.
SysXRes	Returns the horizontal screen resolution.
SysYRes	Returns the vertical screen resolution.

9.11.7 Thread management functions

The following functions are used to manage threads:

CriticalSection	Enables a name or on-variable critical section in the statement USING ... IN.
CriticalSectionEnd	Marks the end of critical section: the code will be run by another thread.
CriticalSectionStart	Marks the start of a critical section: no other thread will be able to run the code as long as the current thread does not exit from the critical section.
EventChange	Modifies the status of an event.
EventClose	Closes a synchronization event between several threads.
EventCreate	Creates an event.
EventDestroy	Explicitly destroys an event.
EventOpen	Opens a synchronization event between several threads.
EventWait	Locks the current thread while waiting for the specified event to be opened.
ExecuteMainThread	Triggers the execution of the procedure specified in the main application thread.
ExecuteMainThread-Asynchronous	Executes a procedure in the main thread of the application without waiting for the end of its execution.

MutexCreate	Explicitly creates a mutex.
MutexDestroy	Explicitly destroys a mutex.
MutexEnd	Signals that the thread frees the mutex.
MutexStart	Locks the current thread while waiting for the mutex to be freed.
SemaphoreCreate	Creates a semaphore.
SemaphoreDestroy	Explicitly destroys a semaphore.
SemaphoreEnd	Allows one or more threads to exit from the area protected by the semaphore.
SemaphoreStart	Locks the current thread until the semaphore is opened (which means until a "free" spot becomes available in the protected section).
ThreadCurrent	Returns the name of the thread currently run.
ThreadEnd	Ends the execution of the current thread.
ThreadExecute	Starts the execution of a secondary thread.
ThreadMode	Changes the management mode of threads.
ThreadPause	Pauses the current thread during the specified duration.
ThreadPersistent	Makes a thread persistent.
ThreadPriority	Returns or modifies the priority level of a thread.
ThreadRequestStop	Sends a stop request to a thread.
ThreadSendSignal	The current thread sends a signal to the specified thread in order to unlock it.
ThreadState	Returns the current status of a thread.
ThreadStopRequested	Checks if a stop request has been sent to the running thread.
ThreadWait	Waits for the end of the execution of the specified thread.
ThreadWaitSignal	Locks the current thread until it receives a signal from another thread.

9.11.8 Parallel task management functions

The following functions are used to manage parallel tasks:

ParallelTaskCancel	Immediately ends the execution of the current task: the task is flagged as <i>ptsCanceled</i> .
ParallelTaskConfigure	Modifies the parameters of parallel tasks.
ParallelTaskExecute	Triggers the execution of a parallel task.
ParallelTaskExecuteAfter	Indicates a continuation parallel task that will be run when one of the specified parallel task is ended.
ParallelTaskExecuteAfterAll	Indicates a continuation parallel task that will be run when all the tasks found in an array of parallel tasks are ended.
ParallelTaskExecute-AfterOne	Indicates a continuation parallel task that will be run once the first task is ended in an array of parallel tasks.
ParallelTaskRequest-Cancelation	Requests the cancelation of a parallel task.
Cancelation	Waits for the execution of a parallel task.
ParallelTaskWaitAll	Waits for the end of execution of all parallel tasks found in an array.

9.12 Font management functions

The following functions are used to manage fonts (drawing fonts, print fonts, ...):

dFont	Declares the font that will be used by <b>dText</b> : <ul style="list-style-type: none"><li>• in an Image control,</li><li>• in an <b>Image</b> variable,</li><li>• in a WDPic variable (on the background layer),</li><li>• in a picLayer variable.</li></ul>
gFont	Builds a sequence of characters describing the font that will be used for the text.
gFontBold	Builds a sequence of characters used to specify whether the text that follows the call to this function is in bold or not.
gFontItalic	Builds a sequence of characters used to specify whether or not the text that follows the call to this function is in italic type.

gFontUnderlined	Builds a sequence of characters to specify whether or not the text that follows the call to this function is underlined.
gFontSize	Builds a sequence of characters used to specify the size of the font for a text found in a graphic string.
grLabelFont	Modifies the font used for the category labels of a chart.
grLegendFont	Modifies the font used for the chart legend.
grTitleFont	Modifies the font used for the chart title.
grAxisTitleFont	Modifies the font used for the titles of the chart axes.
iCreateFont	Creates a new print font.
iFont	Selects the default font.
iDidotFont	Selects the unit that will be used for the height of the fonts (created by <b>iCreateFont</b> ): DIDOT point or PICA point.
iSelectFont	Creates a new print font from the standard window for font selection...
FontCreate	Creates a font definition.
FontList	Returns the list of valid fonts installed on the current computer (TrueType, Postscript, ...).
FontSelect	Opens the standard window for font selection and selects the specified font.

9.13 User Groupware management functions

The following functions are used to customize the management of the User Groupware.

gpwActivateUser	Validates a new user in the User Groupware database by activating the user.
gpwAddAuthLogin	Adds a login button (Facebook, Google, Microsoft, etc.) in the login window or page.
gpwAddUser	Adds a user to the User Groupware database.
gpwAddUserToGroup	Adds a user to a group of the User Groupware database.
gpwAuthLogin	Allows you to connect to the User Groupware using authentication with an account such as Facebook, Google, Microsoft, etc.

gpwBiometricAuthentication	Enables biometric authentication for the connected user or retrieves the current status for the connected user.
gpwCheckPassword	Checks a password (encrypted or not).
gpwCheckUser	Checks the existence and validity of the password for a user of the user groupware.
gpwConnectUser	Connects a user to the User Groupware.
gpwDefault	Returns the value that corresponds to "DEFAULT" according to the framework being used.
gpwDisconnectUser	Disconnects the current user from the User Groupware.
gpwEncryptedPasswords	Used to find out whether the passwords of the user groupware are encrypted.
gpwEncryptPassword	Returns the encrypted password.
gpwEnumControl	Enumerates the controls found in a window or page.
gpwFindUser	Retrieves the characteristics of a user of the user groupware.
gpwForgottenPassword	Runs an action to retrieve or change the password of a user.
gpwGetFileInfo	Retrieves the directory of the data files found in the User Groupware.
gpwGetLDAPInfo	Retrieves the configuration of the LDAP server used for the User Groupware.
gpwGetRights	Used to get the rights granted to a control, a window, a page or a report.
gpwGetUserInfo	Retrieves information about the current user (application that is using the User Groupware).
gpwImportLDAPUsers	Imports the users stored in a LDAP directory into the user file of user groupware.
gpwInitAnalysis	Initialize the analysis of user groupware.
gpwInLDAPMode	Allows you to find out whether identification of users in the user groupware will be done via a LDAP directory.
gpwListOAuthParameters	Retrieves the list of OAuth parameters defined for the User Groupware, i.e. the list of providers (Google, Facebook, etc.) to which it is possible to connect via the User Groupware.

gpwLogin	Connects a user to an application that is using the User Groupware.
gpwMode	Returns the runtime mode of the User Groupware.
gpwModifyUser	Modifies a user in the database of user groupware.
gpwNone	Returns the value of the word "NONE" according to the framework being used.
gpwOpen	Opens the login window or page of user groupware.
gpwOpenConfiguration	Opens the User Groupware configuration window or page.
gpwOpenConnection	Opens a connection to a specific Client/Server database to manage the files of user groupware in Client/Server mode.
gpwOpenLDAPSession	Informs the User Groupware that the authentication of users will be done via an LDAP directory.
gpwOpenSite	Allows you to start the site when using the user groupware.
gpwRegisteredUsers	Returns the value that corresponds to "Registered users" according to the framework being used.
gpwRemoteConnect	Allows connecting to the application via the User Groupware Webservice.
gpwResetPassword	Resets the password of a registered user.
gpwSupervisor	Returns the username of the "Supervisor" account according to the framework being used.
gpwVisitor	Returns the value that corresponds to the word "Visitor" according to the framework used.

## 9.14 Service management functions

The following functions are used to manage services:

EndService	Ends the execution of the current service.
ServiceContinue	Restarts a paused service.
ServiceControl	Sends a control message to a service.
ServiceExist	Checks whether a service is installed.
ServiceInfo	Returns information regarding the configuration of a service.

ServiceInstall	Installs a Windows service or a Linux daemon according to the information given in the members of the Service variable.
ServiceList	Returns the list of services found on the current computer or on the specified remote computer.
ServiceModify	Modifies the configuration of a service according to the information given in the members of the Service variable.
ServicePause	Pauses a service.
ServiceRefresh	Asks a service to re-read its configuration information.
ServiceRestart	Restarts a service (already started).
ServiceStart	Starts a service.
ServiceStatus	Returns the current status of a service.
ServiceStop	Stops a service.
ServiceUninstall	Uninstalls a service.
ServiceWait	Pauses the current service during the specified duration.
ServiceWaitStatus	Waits for a service to be in a specific status.
ServiceWriteEventLog	Writes an event into the log of Windows events.

### 9.15 AAF (Automatic Application Features) management functions

The following functions are used to manage AAFs (Automatic Application Features):

AAFChangeCaption	Customizes the text of a menu option for context menus corresponding to an AAF (Automatic Application Feature).
AAFConfigure	Allows you to specify the behavior of some Automatic Application Features (AAF). You can for example configure the option for exporting to Excel used by the automatic menu of tables.
AAFDisable	Disables an Automatic Application Feature (AAF) on a control, window or on the current application.
AAFEEnable	Re-enables an Automatic Application Feature (AAF) previously disabled by <b>AAFDisable</b> .

AAFExecute	Runs an Automatic Application Feature (AAF) on a control or window. This function can be used to start the spelling checker on an edit control for example.
AAFExportParameter	Saves all customizations made via AAFs (Automatic Application Features) in a file.
AAFExportSecurity	Secures the export of confidential data in the AAFs (Automatic Application Features) that are used to export, print or save data.
AAFImportParameter	Loads all customizations made via AAFs (Automatic Applications Features) that have previously been saved to a file.
AAFPopupMenuToString	Returns the list of options in the context menu actually displayed by the control.
AAFOpenHelp	Opens the help window of an AAF
AAFPossible	Indicates whether an AAF (Automatic Application Feature) can be run on a specific control.

### 9.16 Miscellaneous WEBDEV functions

The various WEBDEV functions are as follows:

ASPDisplay	Calls an external ASP script and returns the result page in the current browser window.
ASPEXecute	Calls an external .asp script and returns the result in a string.
CapsLockVerify	Checks whether the CapsLock key is pressed.
CertificateClientInfo	Returns information about the certificate used by the client computer.
ColorPalette	Returns a color found in the current palette.
ConnectionCount	Returns the number of instances of the WEBDEV site currently run on the server.
DynamicServingConfigure	Indicates the environment that will be used by the "Dynamic Serving" to choose the set of pages to display.
ExecuteDelayedProcedure	Runs a delayed procedure (in the WEBDEV application server).

FileDisplay	Displays a file in the browser of Web user.
FolderWeb	Returns the path of the directory containing: <ul style="list-style-type: none"><li>• the images,</li><li>• the JavaScript files,</li><li>• the Java applet files,</li><li>• the other files accessible from the browser.</li></ul>
FolderWeb_15	Caution: when migrating a project from WEBDEV 1.5 to WEBDEV 17: <ul style="list-style-type: none"><li>• FolderWeb was replaced with FolderWeb_15.</li><li>• a warning is automatically generated in your project.</li></ul>
HTMLClassAdd	Adds a class into the HTML classes of a control.
HTMLClassDelete	Deletes a class from the HTML classes of a control.
HTMLClassToggle	Toggles a class in the HTML classes of a control: if the class does not exist, it is added; if the class already exists, it is deleted.
jQuery	Runs a JavaScript method (or several chained methods) of the jQuery library on a page element.
jQueryExecute	Runs a JavaScript method of the jQuery library on a page element.
JSEndEvent	Deletes the association between a WLanguage browser function and an event (implemented by <b>JSEvent</b> ).
JSEvent	Associates a browser procedure with an event on an object in browser code.
JSInfoEvent	Used to handle the JavaScript properties of the browser event that triggered the code execution.
JSInterruptEvent	Interrupts the process of current event.
JSMethod	Used to run a JavaScript method on an element found in the current page.
JSONExecute	Calls a server URL that returns data in JSON format (JavaScript Object Notation).
JSONExecuteExternal	Calls an external server URL that returns data in JSON format (JavaScript Object Notation).
JSProperty	Used to handle specific features on the objects found in the current page.

PHPDisplay	Calls an external PHP script and returns the result page in the current browser window.
PHPExecute	Calls an external .php script and returns the result in a string.
ScriptDisplay	Calls an external script or page (.php, .asp, .mhtml or.mht) and returns the result page in the current browser window.
ScriptExecute	Calls an external script (.asp or .php) and returns the result in a string.
SessionHeartBeatEnable	Enables or disables the "heartbeat" of the current session.
SessionPrologue	Enables or disables a WLanguage procedure called on each request in a session.
SSLActive	Used to enable or disable the secure SSL mode.
StringDisplay	Returns a specific string (or a buffer) to the client browser in response to a request.

### 9.17 Miscellaneous WINDEV functions

AJAXCallInProgress	Used to determine if the request being processed is an AJAX request.
AutomationEvent	Branches a procedure to the event of an automation object.
AutomationParameter	Configures the management of accesses to the ActiveX objects and to the Automation objects.
Beep	Emits a beep.
BuildAutomationVariant	Builds an Automation variant parameter.
Clone	Constructs a copy of a class instance: <ul style="list-style-type: none"><li>• An object of the actual class of the class instance is allocated.</li><li>• The members of the class instance are copied to the members of the new object.</li></ul>
CommandLine	Identifies and retrieves the different elements of the command line passed as a parameter to the current program.



Compile	Dynamically compiles a procedure whose source code is supplied.
ComponentInfo	Returns information about a component.
ConfigureAAF	Used to specify the behavior of some Automatic Application Features (AAF).
DataType	Identifying the type of a WLanguage variable.
diffApply	Applies a difference buffer to a specific buffer.
diffCreate	Compares two buffers and creates a buffer containing the differences between these two buffers.
Dimension	Returns the size of a variable (string, numeric, array, ...) or resizes a dynamic array.
DisableAAF	Disables an Automatic Application Feature (AAF) on a control, window or on the current application.
DrawingStyle	Allows you to find out and modify: <ul style="list-style-type: none"><li>• the display mode of grayed buttons found in the windows of your application.</li><li>• the display mode of translucent border for the windows of your application.</li><li>• the mode for loading large images.</li></ul>
EndAutomatedProcedure	Stops the current or forthcoming execution of an automatic procedure (defined by the properties of the procedure in the code editor).
EndProgram	Ends the execution of the current program.
EvaluateExpression	Evaluates the value of an expression built in a character string.
Execute	Starts the execution of a process through programming.
ExecuteAAF	Runs an Automatic Application Feature (AAF) on a control or window.
ExecuteAncestor	When the code is overridden, used to run the process of the template corresponding to the process of the current control.
ExecuteCode	Runs the WLanguage code found in a character string.
ExecuteFunction	Executes a WLanguage function identified by its name.

ExecuteJS	Runs JavaScript code in the context of the Web page displayed by an HTML Display control.
ExecutePresetAction	Runs a preset WINDEV action.
ExecuteProcess	Starts the execution of a process programmatically, i.e. without the user having to perform any particular action in the interface.
ExecutionMode	Allows you to: <ul style="list-style-type: none"><li>• find out and/or modify the behavior of the application or site when some errors are generated.</li><li>• optimize the calls to the functions declared externally.</li><li>• find out and/or change how the application uses JITc.</li><li>• find out and/or modify the management mode of low references for all the instantiated objects.</li></ul>
GenerateGeometricImage	Randomly generates an image with geometric shapes. This image can be used to temporarily fill an Image control, for example.
GenerateLorem	Generates a random "Lorem ipsum" text to temporarily populate a control with text, for example.
GeneratePassword	Automatically generates a password with specific characteristics (size, characters used, ...).
GetColor	Allows you to get a set of harmonious colors without having to use a table of colors.
GetGUID	Calculates a globally unique identifier (named "GUID"), whose size is: <ul style="list-style-type: none"><li>• equal to 16 bytes, which means 32 hexadecimal characters.</li><li>• equal to 32 bytes, which means 64 hexadecimal characters.</li></ul>
GetIdentifier	Calculates a unique identifier (integer) for an executable or for a session.
GetUUID	Calculates a 128-bit Universal Unique Identifier ("UUID").
GetUUID256	Calculates a 256-bit Universal unique identifier ("UUID").
HelpFile	Returns or modifies the name of the help file used by the context-sensitive help of a window.
In64bitMode	Indicates whether the code is run in 64-bit mode or not.

InAJAXMode	Specifies whether the code is run from: <ul style="list-style-type: none"><li>• a process in AJAX mode.</li><li>• a procedure called by <b>AJAXExecute</b> or <b>AJAXExecuteAsynchronous</b>.</li></ul>
InAndroidEmulatorMode	Indicates whether the code is run in the Android emulator.
InAndroidMode	Indicates whether the code is run in Android mode.
InAndroidSimulatorMode	Indicates whether the code is run in the Android simulator.
InAppleWatchEmulatorMode	Indicates whether the code is run in the Apple Watch emulator.
InAppleWatchMode	Indicates whether the code currently run is in Apple Watch mode.
InAppleWatch-SimulatorMode	Indicates whether the code is run in the Apple Watch simulator.
InAutomaticTestMode	Used to find out whether an automatic test is currently run or not.
InAWPMode	Used to determine if the page currently run is an Active WEBDEV Page.
InBackgroundMode	Indicates whether the code is run when the application is in background.
InComponentMode	Indicates whether the code is run: <ul style="list-style-type: none"><li>• from a WINDEV application, a WEBDEV site or a PHP page.</li><li>• from a component.</li></ul>
InDelayedProcedureMode	Indicates whether the code is run from a delayed procedure or from a scheduled task on the WEBDEV Application Server.
InDeviceOwnerMode	Indicates whether the Android application is in "Device owner" mode.
InFactoryMode	Indicates whether the code is run by an action plan of the software factory.
IniOSEmulatorMode	Indicates whether the code is run in the iOS emulator (iPhone, iPad).
IniOSMode	Indicates whether the code currently run is in iOS mode (iPhone, iPad).

IniOSSimulatorMode	Indicates whether the code is run in the iOS simulator (iPhone, iPad).
INIRead	Reads the content of an INI file (or the content of a file structured like .INI files).
InitRandom	Initializes the generator of random numbers.
INIWrite	Writes a specific value into an INI file (or into a file structured like an.INI file).
InJavaMode	Indicates whether the code is run in Java.
InKioskMode	Indicates whether the Android application is in kiosk mode.
InLinuxMode	Indicates whether the code is run in Linux mode or not.
InMacCatalystMode	Indicates whether the application is running in Mac Catalyst mode (running an iPad application on a Mac).
InPageWithinWindowMode	Indicates if the page is displayed in the WEBDEV Page control of a window.
InPHPMode	Indicates whether the code is run in PHP.
InPHPSimulatorMode	Indicates whether the code is run in PHP simulator mode.
InPrelaunchedSessionMode	Used to find out whether the site is started in "pre-launched session" mode of WEBDEV.
InReportsAndQueriesMode	Indicates whether the code is run during the test of a report or query in Reports & Queries.
InServiceMode	Indicates whether the code is run in a service.
InSimulatorMode	Used to find out whether the application is started in "simulator test mode" from WINDEV Mobile.
InStoredProcedureMode	Indicates whether the code is run on an HFSQL server.
InSysNotificationMode	Used to determine if the application is launched from an interactive notification in "Windows 10".
InTestMode	Indicates whether the application or the site is started: <ul style="list-style-type: none"><li>• from the WINDEV or WEBDEV editor,</li><li>• from the executable or browser.</li><li>• from the WEBDEV administrator.</li></ul>
InTSEMode	Indicates whether the code is run: <ul style="list-style-type: none"><li>• from a computer in TSE mode or from a remote desktop.</li><li>• from a computer directly.</li></ul>

InUMCMode	Indicates whether the code is run from a code written in a user macro-code.
InUniversalWindows-AppMode	Indicates whether the code is run in Universal Windows 10 App mode.
InUniversalWindowsApp-SimulatorMode	Indicates whether the code is run in the simulator for the applications in Universal Windows 10 App mode.
InVGAMode	Indicates whether the application is run in VGA mode on a mobile device.
InWebMode	Indicates whether the current process is run: <ul style="list-style-type: none"><li>• from a WINDEV executable.</li><li>• from the WEBDEV runtime engine (AWP).</li></ul>
InWebserviceMode	Used to find out whether the execution is in progress in the AWWWS engine.
InWebSocketServer-GlobalServiceMode	Indicates if the current code is being executed on the global service of the WebSocket server.
InWebSocketServerMode	Indicates if the current code is being executed on the WebSocket server.
InWidgetMode	Indicates whether the code is run in Android Widget mode.
InWindowsMode	Indicates whether the code is run in Windows mode or not.
IsNumeric	Enables you to find out whether a variable or a control is: <ul style="list-style-type: none"><li>• a numeric (integer, real, currency or numeric).</li><li>• a string that can be converted into numeric.</li></ul>
LargeFontFactor	Returns the current enlargement ratio of a WINDEV window (only if Windows is in "large fonts" mode).
LoadWDL	Loads a library of WINDEV objects in memory (WDL).
NumHelp	Returns the number of the help context associated with the specified control.
OpenDocument	Opens a document in a new (modeless) window using the appropriate WINDEV editor (word processor, spreadsheet, PDF reader or image editor).
OpenDocument-Asynchronous	Opens a document using the appropriate editor
ProfilerEnd	Stops "collecting data" for the performance profiler.

ProfilerStart	Starts "collecting information" for the performance profiler.
Random	Returns a random number.
RandomCryptoBuffer	Returns a buffer to the specified size filled with random data, that can be used with the cryptographic technologies.
RandomCryptoInteger	Returns a random number that can be used with the cryptographic technologies.
RunReportsAndQueries	Starts Reports and Queries and opens the WINDEV project corresponding to the current application.
SelectColor	Opens the standard window for color selection and selects the specified color.
Sound	Plays a sound in WAV format.
StyleDrawCaptionGrayed	Customizes how text is drawn in grayed out controls.
ToolTipMargin	Modifies the margin of the text in the tooltips.
Trace	Displays the requested information (content of a control, for example) in a window opened next to the current window.
TraceBuild	Displays a customizable message in the trace window opened in parallel with the current window.
TraceDeleteAll	Clears the content of trace window opened in parallel with the current window.
TraceEnd	Closes the trace window if this window was opened or stops saving the trace information in a text file.
TraceStart	Opens the trace window or saves the trace information in a text file.
TrialVersion	Used to find out whether the current execution of the application is in "trial version" mode.
TypeVar	Identifies the type of an expression, a variable (during a call to a procedure for example) or a control.
UnloadWDL	Frees the library that was loaded in memory by <b>LoadWDL</b> .
VariableReset	Resets the variable to its initial value.
VariantConvert	Converts the value stored in a Variant variable].
WHelp	Displays a file or a help page in HLP or CHM format.

WINDEVVersion	Returns the version number of the wdxxxvm DLL.
---------------	--

### 9.18 Project management functions

These functions are used to easily handle the elements found in a WINDEV project:

ComponentExecute	Executes a global procedure present in an external component. This external component is loaded with <b>ComponentLoad</b> .
ComponentList	Returns the list of external components (".WDK" files) used by the current WINDEV or WINDEV Mobile application or by the current WEBDEV site.
ComponentLoad	Loads an external component. This component can be: <ul style="list-style-type: none"><li>• an external component present in the current project.</li><li>• an external component independent of the current project.</li></ul>
ComponentLoadAll	Recursively loads in memory all the external components used in the project.
ComponentLocate	Specifies the access path to an external component of project.
EnumElement	Used to enumerate the project elements: windows, pages, reports, queries, ...
EnumSubElement	Used to enumerate the sub-elements of a project element.
ListDLL	Returns: <ul style="list-style-type: none"><li>• <b>at runtime</b>: the list of libraries (".DLL" files) used by the current WINDEV or WINDEV Mobile application or by the current WEBDEV site. Only the libraries loaded in memory are listed.</li><li>• <b>in test mode</b>: all the libraries installed on the current computer.</li></ul>
ProjectInfo	Returns specific information about the project currently run.
WindowCount	Calculates the number of windows currently opened in the current application.

### 9.19 Debugging functions

The debugging functions are as follows:

dbgAssert	Used to find out and signal the possible errors via an assertive programming.
dbgAuditOption	Retrieves or modifies the options of the dynamic audit.
dbgAuditStatus	Retrieves and changes the status of the dynamic audit.
dbgCheckDifferent	Checks whether the specified expressions are different in assertive programming and in automated test mode.
dbgCheckEqual	Checks whether the expressions are equal in assertive programming and in automated test mode.
dbgCheckFalse	Checks whether an expression is false in assertive programming and in automated test mode.
dbgCheckNotNull	Checks whether an expression is not Null in assertive programming and in automated test mode.
dbgCheckNull	Checks whether an expression is Null in assertive programming and in automated test mode.
dbgCheckTrue	Checks whether an expression is set to True in assertive programming and in automated test mode.
dbgConnect	Used to connect a remote debugger to an application, service, Webservice, etc.
dbgEnableAssert	Enables or disables the assertive programming.
dbgEnableAudit	Enables the dynamic audit through programming.
dbgEnableLog	Enables the runtime log through programming.
dbgEndMemoryDiff	Saves a memory dump containing the memory resources that have been allocated and that have not been freed since the previous call to <b>dbgStartMemoryDiff</b> .
dbgEndRecording	Stops recording the scenario.
dbgError	Provokes: <ul style="list-style-type: none"><li>• an assertion during the test of element or project (Go mode).</li><li>• an error in automated test mode.</li></ul>
dbgInfo	Returns a debug information.

dbgLogOption	Retrieves or modifies the options of the runtime log.
dbgLogStatus	Returns and modifies the status of runtime log.
dbgOnError	Defines the WLanguage procedure that will be called whenever an error is detected by a dbgCheckXXX function.
dbgSaveDebugDump	Saves a dump file of the application.
dbgSaveMemoryDump	Saves a dump of memory used by the application.
dbgStandardOutput	Writes an information into the standard output stream "stdout" (also called "console").
dbgStartMemoryDiff	Starts a difference operation of the memory used by the application.
dbgStartRecording	Starts recording a test scenario (also called reproduction scenario).
dbgWriteCommentAudit	Writes a comment into the current dynamic audit.
dbgWriteWarningAudit	Writes a custom runtime warning into the dynamic audit.

## 9.20 Functions for managing Executables in Console mode

The following functions are used to manage different streams of executables in "console" mode:

ConsoleError	Writes a character string in the standard error output stream "stderr".
ConsoleRead	Reads a string in the standard input stream "stdin".
ConsoleWrite	Writes a character string in the standard output stream "stdout" (also called "console").

## 9.21 Activation key management functions

The following functions are used to manage the activation keys:

KeyCalcActivationKey	Calculates the activation key of application from the initial key.
KeyCompareKey	Compares an initial key with an activation key.

KeyGenerateInitialKey	Generates the initial key (unique license number) for the activation system of the application.
KeyGetIdentifier	Retrieves the identifier that was used to generate an initial key.

## 9.22 WEBDEV scheduled task management functions

The following functions are used to manage scheduled WEBDEV tasks:

AppServerAddScheduled-Task	Adds a new scheduled task to the application (site or webservice).
AppServerDelete-ScheduledTask	Deletes a scheduled task associated with the application (site or Webservice).
AppServerInfoScheduled-Task	Reads the description of a scheduled task.
AppServerListScheduled-Task	Returns the list of scheduled tasks for the application (site or Webservice).
AppServerModify-ScheduledTask	Modifies an existing scheduled task.
AppServerRun-ScheduledTask	Immediately starts the execution of a scheduled task on a WEBDEV Application Server.

## 9.23 Action plan management functions (software factory)

The following functions are used to manage the actions plans of the software factory:

ActionPlanConfigure	Configures an action plan in the software factory.
ActionPlanExecute	Runs an instance of the action plan in the software factory.
ActionPlanInfo	Retrieves information regarding the execution of an instance of action plan.
apDisableAction	Disables the execution of an action found in an action plan.
apExecute	Runs an action plan.

apFullProjectName	Retrieves the full physical name of a project opened during the execution of an action plan.
apLog	Saves an information in a "log" file during the execution of an action plan.

### 9.24 Hasp functions

The Hasp functions are as follows:

Hasp	Interrogates an electronic HASP key (HASP-3, HASP-4, MemoHasp or NetHasp).
HaspHLEncrypt	Encrypts a character string by using the algorithms of the specified Hasp key.
HaspHLLDecrypt	Decrypts a character string by using the algorithms of the Hasp key.
HaspHLInfo	Returns information about the electronic Hasp key.
HaspHLLogin	Connects the application with a Hasp key.
HaspHLLogout	Closes the connection of the application to a Hasp key.
HaspHLRead	Reads data (character string or buffer) in the internal memory of the Hasp key.
HaspHLWrite	Writes data (character string or buffer) into the internal memory of the Hasp key.

### 9.25 Background task management functions

The following functions are used to manage background tasks:

BackgroundTaskAdd	Adds a new background task.
BackgroundTaskDelete	Deletes a background task (that was added by <b>BackgroundTaskAdd</b> ).
BackgroundTaskList	Returns the list of background tasks for the application.

### 9.26 Telemetry management functions

The following functions are used to manage the telemetry:

TelemetryAddMeasure	Adds a measure for a value or time sensor.
TelemetryAddPassage	Signals the code execution for a passage sensor.
TelemetryConfigure	Used to find out and modify the telemetry parameters.
TelemetryDisable	Disables the telemetry. The telemetry data is not sent to the server anymore.
TelemetryEnable	Enables the telemetry manually.
TelemetryIdentifyUser	Identifies the current user for whom the telemetry data is saved.
TelemetryIsEnabled	Used to find out whether the telemetry is enabled.
TelemetrySendData	Sends telemetry data immediately.

### 9.27 Action functions

The following functions are used to manage the actions with Undo/Redo:

ActionDeclareUndo	Declares how to cancel an action that was added by <b>ActionDo</b> . This function <b>MUST</b> be called when running the procedure supplied to <b>ActionDo</b> .
ActionDo	Creates and runs a custom action (defined through programming) with support of "Undo/Redo".
ActionRedo	Runs the last action canceled.
ActionUndo	Cancels the last action performed by the end user.

### 9.28 AJAX management functions

The AJAX functions are as follows:

AJAXAsynchronous-CallPending	Used to find out whether a server procedure called by <b>AJAXExecuteAsynchronous</b> is currently run.
AJAXAvailable	Used to find out whether the AJAX technology is supported by the current browser.



AJAXCancelAsynchronous-Call	Cancels the automatic execution of the browser procedure called by <b>AJAXExecuteAsynchronous</b> .
AJAXExecute	Runs a server procedure without refreshing the page.
AJAXExecuteAsynchronous	Runs a server procedure without refreshing the page.

### 9.29 Photo album management functions

The following functions are used to manage a photo album:

AlbumPicker	Retrieves a photo, an image or a video stored in the photo album of the mobile device.
AlbumSave	Saves an image, a photo or a video in the photo album of the mobile device.
PhotoRunApp	Starts the native camera application of the device in order to take a photo.

### 9.30 Application management functions with "Live update"

The following functions are used to manage "live updates":

AppActivateVersion	Activates one of the application versions available in the history of versions on the reference setup server.
AppChangeParameter	Changes an information in the mechanism for automatic application update.
AppControl	Enables (or disables) the remote control for the current multi-user WINDEV application. This is an advanced function.
AppDeleteVersion	Deletes a version from the history of versions found on the reference setup server.
AppDetectUPDAvailable	Disables or customizes the automatic check for available updates.
AppInstallUPD	Triggers the automatic update of current application.
AppInstallVersion	Installs a specific application version.
AppListVersion	Lists the versions available on the reference setup server.

AppParameter	Returns information about the mechanism for automatic application update.
AppUPDAvailable	Allows you to find out whether an update is available for the specified application.
AppVersionInfo	Returns information about a version available on the setup server.

### 9.31 Bar Code management functions

The following functions are used to manage the bar codes:

BCCalculateControl-Character	Returns the check code of a bar code.
BCCapture	Decodes the information stored in a bar code by using the device camera (Android, iPhone, iPad).
BCDecode	Decodes the information stored in a QR Code bar code.
BCToImage	Generates the image of a bar code.

### 9.32 DDE functions

The following functions are used to manage the DDE:

DDEConnect	Establishes a DDE connection between the current program and the recipient according to a given topic.
DDEDisconnect	Deletes a connection between the current program and a recipient.
DDEError	Returns the runtime status report of a DDE function.
DDEEvent	Associates a WLanguage procedure with a DDE event: when the DDE event occurs, the associated WLanguage procedure is automatically run.
DDEExecute	Sends a command that will be run on the server.
DDEItem	Identifies the item affected by a DDE event.
DDELink	Creates a hot link with a data.
DDERecipient	Identifies the recipient of a DDE connection.

DDERetrieve	Retrieves a data sent by a program (the recipient of the connection for the specified object).
DDESend	Sends a data to the program connected via DDE.
DDEStart	Starts the execution of a program (an executable for example) from the current application.
DDEString	Returns the information retrieved by <b>DDERetrieve</b> .
DDETopic	Identifies the subject of conversation associated with a DDE connection.
DDEUnLink	Interrupts a link between an item and a data.
DDEUpdate	Modifies a linked data.
DDEWarmLink	Creates a warm link with a data.

### 9.33 Continent and country management functions

The following functions are used to manage continents and countries:

ContinentGet	Returns the Continent variable that corresponds to one of the following values: <ul style="list-style-type: none"><li>• a continent constant (2-character continent code),</li><li>• a continent caption.</li></ul>
ContinentList	Returns the list of continents.
CountryGet	Returns the Country variable corresponding to an ISO code or to the country caption according to ISO 3166-1.
CountryList	Returns the list of all countries or countries of a given continent according to ISO 3166-1.

### 9.34 Multi-touch management functions

The following functions are used to manage gestures on a phone (iOS/Android) or in a site (multi-touch):

GestureCurrentPointer	Returns the pointer index (finger or stylus) that triggered the call to a process specific to the multi-touch feature in an Image control.
GestureNbPointer	Returns the number of pointers in contact with the screen.

GesturePointerType	Returns the type of pointer used on the device: finger, mouse or stylus.
GesturePosX	Returns the horizontal position of the pointer (finger or stylus).
GesturePosY	Returns the vertical position of the pointer (finger or stylus).

### 9.35 Runtime error management functions

The following functions are used to manage the errors and the exceptions:

ErrorChangeParameter	Configures the automatic behavior that will be implemented if an error occurs in the current process.
ErrorInfo	Retrieves information about the last error that occurred in a function of a WINDEV, WINDEV Mobile or WEBDEV component.
ErrorPropagate	Propagates a non-fatal error that can be processed by the calling code if the management of non-fatal errors was enabled for this code (in the code editor or by <b>ErrorChangeParameter</b> ).
ErrorReset	Resets the detection of non-fatal errors: <ul style="list-style-type: none"><li>• The result of <b>ErrorInfo</b> is reset.</li><li>• The <b>ErrorOccurred</b> variable returns to <u>False</u>.</li></ul>
ErrorRestoreParameter	Restores the error process of a process.
ErrorThrow	Throw a non-fatal error.
ExceptionChangeParameter	Configures the automatic behavior that will be implemented if an exception occurs in the current process.
ExceptionDisplay	Displays the standard window of exceptions with the content of the current exception.
ExceptionEnable	Re-enables the current exception process if the exception was corrected.
ExceptionInfo	Retrieves information about the current exception.
ExceptionPropagate	Propagates an exception.
ExceptionRestoreParameter	Restores the exception process of a process.
ExceptionThrow	Artificially triggers the security mechanism of WLanguage by throwing an exception.

### 9.36 Euro functions

The following functions are used to manage the Euro:

eChangeCurrency	Converts a currency into another currency.
eCurrencyName	Returns the caption of a currency.
eDefaultMemCurrency	Identifies or modifies the stored currency used by default.
eDefaultViewCurrency	Identifies or modifies the display currency used by default.
eExchangeRate	Returns the value of the exchange rate for a currency.
eInitCurrency	Adds or changes the characteristics of a currency in the currency exchange table.
eRoundDifference	Returns the rounding difference between: <ul style="list-style-type: none"><li>• the value displayed in a "Currency + Euro" control,</li><li>• the associated control used for dual display.</li></ul>
eToCurrency	Converts a value expressed in Euro into another currency.
eToEuro	Converts to Euro a value expressed in a specific currency.

### 9.37 End-user UI management functions

The following WLanguage functions allow the end user to manipulate the UI:

WinEdActive	Used to: <ul style="list-style-type: none"><li>• find out whether the mode "UI modification by end user" is enabled in a window (or internal window).</li><li>• enable (or disable) the mode "UI modification by end user" in a window (or internal window).</li></ul>
WinEdCreateNote	Creates a Repositionable Note control.
WinEdDeclareUndo	Declares how to cancel an action that was added by <b>WinEdDo</b> .
WinEdDeselectAll	Deselects all the controls when the interface is modified by the end user (Mode "UI modification by end user" enabled).
WinEdDo	Runs an action (defined through programming) with support for Undo/Redo.
WinEdLoad	Loads the modifications performed by the end user in a window.

WinEdRedo	Runs the last action canceled by the end user when modifying the interface (Mode "UI modification by end user" enabled).
WinEdRestore	Cancels all the modifications made by the end user (Mode "UI modification by end user" enabled).
WinEdSave	Saves the modifications made in the mode "Modification of UI by the end user".
WinEdSaveNote	Exports the repositionable notes created by the user into a file.
WinEdSelect	Returns one of the controls selected when the interface is modified by the end user (Mode "UI modification by end user" enabled).
WinEdSelectMinus	Deselects a control when the interface is modified by the end user ("UI modification by end user" mode enabled).
WinEdSelectOccurrence	Indicates the number of controls selected when the interface is modified by the end user (Mode "UI modification by end user" enabled).
WinEdSelectPlus	Selects a control when the interface is modified by the end user ("UI modification by end user" mode enabled).
WinEdUndo	Cancels the last action performed by the end user when modifying the interface (Mode "UI modification by end user" enabled).

### 9.38 Browser management functions

The following functions are used to manage browsers:

AddFavorite	Adds an Internet address into the list of favorites for the Web user.
BrowserClose	Closes the current browser window and stops the execution of the current code.
BrowserHeight	Returns the height (in pixels) of the client area where the page is displayed.
BrowserHistoryAdd	Adds an entry into the history of navigation by associating data. This data will be transmitted when going back to this entry.

BrowserHistoryModify	Modifies the data of current entry in the navigation history. This data will be transmitted when going back to the current entry.
BrowserIPAddress	Returns the IP address of the client computer connected to the WEBDEV site.
BrowserIsConnected	Indicates whether the browser is connected to the network.
BrowserMobile	Allows you to find out whether the browser used by the Web user is started on a mobile device.
BrowserName	Returns the name of the browser of the Web user ("user agent").
BrowserOpen	Opens a new browser window.
BrowserOS	Indicates the operating system installed on the browser of the Web user.
BrowserPlatform	Returns the user's browser platform.
BrowserPrint	Prints the page currently displayed in the browser.
BrowserRunApp	Opens the default Web browser of the current device.
BrowserSlice	Returns the number of the slice displayed in a page in Responsive Web Design mode.
BrowserType	Returns the type of browser used by the Web user.
BrowserVersion	Returns the version of the browser used by the Web user.
BrowserWidth	Returns the width (in pixels) of the client area where the page is displayed.
CookieDelete	Deletes a cookie on the user's computer.
CookieRead	Retrieves the value of a cookie saved on the computer of Web user.
CookieWrite	Writes a cookie onto the computer of the Web user.
EmailOpenMail	Opens the default messaging software: <ul style="list-style-type: none"><li>• of Web user on the browser computer.</li><li>• of the user on the current Windows computer.</li><li>• on the phone.</li></ul>
FullScreenDisable	Disables the "Full screen" mode of browser.

FullScreenEnable	Switches the browser to "Full screen" mode. The browser occupies the entire screen.
NotificationDisplay	Displays a message outside the browser.

### 9.39 Speech recognition management functions

The following functions are used to manage speech recognition:

SpeechRecognition-AddCommand	Adds a voice command into the current window.
SpeechRecognition-DeleteCommand	Deletes a voice command previously added with <b>SpeechRecognitionAddCommand</b> .
SpeechRecognitionTrigger	Triggers the device's speech recognition service.

### 9.40 Speech synthesis management functions

The following functions are used to manage speech synthesis:

SpeechSynthesisInitialize	Initializes the parameters of speech synthesis for the current application.
SpeechSynthesisInProgress	Used to find out whether a read operation is in progress on the engine for speech synthesis.
SpeechSynthesisListVoices	Returns the list of voices available on the device for the speech synthesis engine.
SpeechSynthesisReadFile	Reads the content of the specified file by using the engine for speech synthesis found on the device.
SpeechSynthesisReadText	Reads the specified text using the device's text-to-speech engine.
SpeechSynthesisStop	Stops all the current read operations on the engine for speech synthesis.

### 9.41 Spelling check functions

The following functions are used to manage the spelling checker:

SpellAddDictionary	Adds a word to the dictionary of OpenOffice: <ul style="list-style-type: none"><li>• for the instance of the current application.</li><li>• for all the applications.</li></ul>
SpellAvailable	Indicates whether a dictionary is available for the spelling checker (check performed in the controls that use the integrated correction or check performed by WLanguage functions).
SpellCheck	Indicates whether the spelling of a word is correct (which means whether it is found in the OpenOffice dictionary installed on the computer).
SpellSuggest	Returns the list of possible words found in the dictionary of OpenOffice that can correct a word that is misspelled.

### 9.42 Upload functions

The following functions are used to manage the file upload:

UploadCopyFile	Saves on the server a file "uploaded" by the Web user. This upload may have been performed: <ul style="list-style-type: none"><li>• via an "Upload" edit control.</li><li>• via a single-file or multi-file Upload control.</li></ul>
UploadCurrentFile	Indicates the file currently uploaded via the Upload control.
UploadCurrentFileSize	Returns the total size (in bytes) of the file currently uploaded via an Upload control.
UploadCurrentFileSizeSent	Returns the size (in bytes) already sent for the file currently uploaded via an Upload control.
UploadDelete	Deletes a file from the list of files to upload: the file will not be uploaded on the server.
UploadDeleteAll	Clears the list of files to upload: no file will be uploaded on the server.
UploadFileName	Returns the name of a file "uploaded" by the Web user. This upload may have been performed: <ul style="list-style-type: none"><li>• via an "Upload" edit control.</li><li>• via a single-file or multi-file Upload control.</li></ul>

UploadFileSize	Returns the total size (in bytes) of a file found in an Upload control.
UploadSize	Returns the total size (in bytes) of the file currently uploaded via an Upload control.
UploadSizeSent	Returns the total size (in bytes) of files already sent by the current upload via an Upload control.
UploadStart	Starts sending the selected files into an Upload control.

### 9.43 URI functions

The following functions are used to manage URIs:

BufferToURI	Saves the contents of a buffer to a resource identified by its URI.
FileToURI	Saves the contents of a file to a resource identified by its URI.
URIGetInfo	Retrieves information of a URI resource.
URISelect	Opens a system window to select a resource from: <ul style="list-style-type: none"><li>• from the device file system,</li><li>• from the Cloud (Google Drive, Microsoft OneDrive, etc.)</li><li>• or from one of the content managers installed on the device.</li></ul>
URIToBuffer	Loads in the background the content of a resource (image, text, etc.) identified by its URI in a buffer variable.
URIToFile	Saves, in the background, the content of a resource (image, text, etc.) identified by its URI in an external file.

### 9.44 Lock functions

The following functions are used to lock an application:

LockAppOff	Unlocks an application that was locked automatically or by <b>LockAppPassword</b> .
LockAppOn	Locks a WINDEV application.

LockAppPassword	Specifies a specific password for unlocking the current application (application locked automatically or by <b>LockAppOn</b> ).
LockAppTimeout	Returns or modifies the idle time before a WINDEV application is locked.
ValidatePassword	Indicates whether the specified password is valid for the current user.

9.45 SaaS functions

The SaaS functions are as follows:

SaaSAdminAddClient	Adds a SaaS client account.
SaaSAdminAddPricing	Adds a SaaS pricing.
SaaSAdminAddService	Adds a SaaS service into the description of a site.
SaaSAdminAddSite	Adds a SaaS site.
SaaSAdminAddSubscription	Adds a subscription to a pricing of a SaaS site for a client account.
SaaSAdminAddUser	Adds a SaaS user.
SaaSAdminClientConnection	Returns the connection to the database of a client account for a SaaS site.
SaaSAdminConfigure-Currency	Configures the currency used in the connection reports of the SaaS system.
SaaSAdminConfigure-SiteData	Configures the management of data for a SaaS site.
SaaSAdminConfigureSMTP	Configures the SMTP session associated with the administrator account of a SaaS session.
SaaSAdminConnect	Connects and authenticates the administrator of the SaaS Webservice.
SaaSAdminCreateClient-Database	Creates the database of a client account for a SaaS site.
SaaSAdminDeleteClient	Deletes a SaaS client account.
SaaSAdminDeletePricing	Deletes a SaaS pricing.

SaaSAdminDeleteService	Deletes a SaaS service.
SaaSAdminDeleteSite	Deletes a SaaS site.
SaaSAdminDelete-Subscription	Deletes a SaaS subscription.
SaaSAdminDeleteUser	Deletes a SaaS user.
SaaSAdminDisconnect	Ends the management session of a SaaS database.
SaaSAdminFindClient	Finds a SaaS client account.
SaaSAdminFindPricing	Finds a SaaS pricing.
SaaSAdminFindService	Finds the description of a SaaS service.
SaaSAdminFindSite	Find a site in the database of SaaS Webservice.
SaaSAdminFindSubscription	Finds a SaaS subscription.
SaaSAdminFindUser	Finds a user of a SaaS client account.
SaaSAdminIsConnected	Indicates whether a SaaS management session was opened.
SaaSAdminListClient	Lists the client accounts of SaaS Webservice database.
SaaSAdminListClient-Subscription	Lists the SaaS subscriptions of a client account.
SaaSAdminListPricing	Lists the pricing of a SaaS site.
SaaSAdminListService	Lists the SaaS services associated with a SaaS site, a pricing or a user.
SaaSAdminListSite	Lists the SaaS sites declared in the database of a WEBDEV SaaS Webservice.
SaaSAdminListSubscription	Lists the SaaS subscriptions associated with a site or pricing.
SaaSAdminListUser	Lists the users of the SaaS Webservice database.
SaaSAdminListVisit	Returns the list of visits to a SaaS site.
SaaSAdminModifyClient	Modifies a SaaS client account.
SaaSAdminModify-ClientConnection	Modifies the connection to the client database for a SaaS site.
SaaSAdminModifyPricing	Modifies a SaaS pricing.



SaaSAdminModifyService	Modifies a SaaS service.
SaaSAdminModifySite	Modifies a SaaS site.
SaaSAdminModify-Subscription	Modifies a SaaS subscription.
SaaSAdminModifyUser	Modifies a SaaS user.
SaaSAdminPricing-DisableService	Disables a service in a SaaS pricing.
SaaSAdminPricing-EnableService	Enables a service in a SaaS pricing.
SaaSAdminReadSite-Parameter	Reads a site parameter that was saved by <b>SaaSAdminWriteSiteParameter</b> .
SaaSAdminSiteConnection	Returns the connection to the common database or to the reference database of the SaaS site.
SaaSAdminSubscription-AllocateLicense	Assigns a subscriber license to a SaaS user.
SaaSAdminSubscription-DeallocateLicense	Cancels a subscription license for a SaaS user.
SaaSAdminWriteSite-Parameter	Saves an information specific to the client.
SaaSChangePassword	Changes the password of the connected user.
SaaSCheckService	Checks the access rights of the user to a service of the SaaS site.
SaaSClientConnexion	Returns the characteristics of the connection to the client database.
SaaSConnect	Authenticates a user of a SaaS site beside the SaaS webservice that manages the site.
SaaSConnectedUser	Returns the user currently connected via <b>SaaSConnect</b> .
SaaSDisconnect	Disconnect the user of a SaaS site beside the SaaS webservice that manages the site.
SaaSIsConnected	Defines whether the connection to the SaaS webservice is still established.
SaaSReadSiteParameter	Reads an information specific to the client for the current SaaS site.

SaaSSendSecurityCode	Sends the security code to the user so that he can change his password.
SaaSWriteSiteParameter	Saves a specific information for a SaaS site in the configuration of a client account.

### 9.46 UMC management functions

The following functions are used to manage UMC:

InUMCMode	Indicates whether the code is run from a code written in a user macro-code.
UMCActive	Used to find out and modify the activation of UMC management (User Macro-Code).
UMCOpen	Opens the edit window of User Macro-Code for the current window.

### 9.47 Test-specific functions

The following functions can be used in the code of an automated test:

TestAddIteration	Adds an iteration into the current test and gives the values of test parameters for this iteration.
TestCheck	Writes a result into the status report of a test according to the result of a boolean expression.
TestCheckDifferent	Checks whether the specified expressions are different in assertive programming and in automated test mode.
TestCheckEqual	Checks whether the expressions are equal in assertive programming and in automated test mode.
TestCheckFalse	Checks whether an expression is false in assertive programming and in automated test mode.
TestCheckNotNull	Checks whether an expression is not Null in assertive programming and in automated test mode.
TestCheckNull	Checks whether an expression is Null in assertive programming and in automated test mode.
TestCheckTrue	Checks whether an expression is set to True in assertive programming and in automated test mode.

TestError	Provokes: <ul style="list-style-type: none"><li>an assertion during the test of element or project (Go mode).</li><li>an error in automated test mode.</li></ul>
TestOnError	Provokes: <ul style="list-style-type: none"><li>an assertion during the test of element or project (Go mode).</li><li>an error in automated test mode.</li></ul>
TestWriteResult	Writes information or an error into the status report of a unit test, in the test editor.

### 9.48 Emulation functions

The emulation functions are used to emulate the different types of windows that may appear during an automated test. These functions, used in the code of the test, are used to perform an action. They prevent the test from being locked.

The emulation functions are as follows:

EmulateActionBar	Emulates an Action Bar during the automated tests run on a mobile device (Android or iOS).
EmulateCheckBox	Emulates the selection of an option in a Check Box control.
EmulateCombo	Emulates one of the following actions on a Combo Box control: <ul style="list-style-type: none"><li>click on the button to expand the list of the Combo Box control.</li><li>select an element in a Combo Box control.</li></ul>
EmulateConfirm	Emulates the next call to <b>Confirm</b> and automatically clicks the button passed in parameter.
EmulateDialog	Emulates the next call to <b>Dialog</b> and automatically clicks the button passed in parameter.
EmulateDnD	Emulates a "Drag & Drop" gesture in automated tests.
EmulateError	Emulates the next call to <b>Error</b> and automatically clicks the "OK" button.
EmulateInfo	Emulates the next call to <b>Info</b> and automatically clicks the "OK" button.
EmulateInput	Performs an input by emulation in a control while checking whether this control is an editable control.

EmulateMenu	Emulates the next menu that will be opened and automatically runs the menu option passed as parameter.
EmulateMouse	Simulates a mouse event (left click, right click, double click, ...).
EmulateOKCancel	Emulates the next call to <b>OKCancel</b> and automatically clicks the button passed in parameter.
EmulatePane	Emulates the selection of a pane in a Tab, Ribbon or Sidebar control.
EmulatePaneDock	Emulates the integration of an undocked pane in the associated dynamic Tab control.
EmulatePaneUndock	Emulates the undocking of a pane in a dynamic Tab control.
EmulateRadioButton	Emulates the selection of an option in a Radio Button control.
EmulateSimulator	Emulates the window of the mobile (Android/iOS) simulator.
EmulateSpeed	Defines the runtime speed for the emulating functions.
EmulateWarning	Emulates the next call to <b>Warning</b> and automatically clicks the "OK" button.
EmulateWindow	Indicates the test that will be run during the next window opening.
EmulateYesNo	Emulates the next call to <b>YesNo</b> and automatically clicks the button specified in parameter.

### 9.49 Webservice management functions

The following functions are used to manage Webservices:

CancelWebserviceContext	Deletes from the webservice context a variable added by <b>DeclareWebserviceContext</b> .
DeclareWebserviceContext	Used to declare a list of variables whose value will be persistent between the successive calls to a Webservice.
WebserviceClientIPAddress	Returns the IP address of the computer using and calling the Webservice.
WebserviceParameter	Gets the value of a parameter in a call to a SOAP or REST Webservice request.

WebserviceReadHTTP-Header	Reads the HTTP header received in a call to a SOAP or REST Webservice.
WebserviceWriteCustom-HTTPCode	Specifies the custom HTTP code that will be returned when the execution of the REST Webservice function is completed.
WebserviceWriteHTTPCode	Specifies the HTTP code that will be returned at the end of the execution of the REST Webservice function.
WebserviceWriteHTTP-Header	Adds an additional HTTP header to a request of a REST Webservice.
WebserviceWriteMIMEType	Indicates the MIME type of the response returned by the Webservice (case of a call to a REST Webservice).

A

AAFCChangeCaption	228	AnsiToOem	71
AAFCConfigure	228	AnsiToUnicode	71
AAFDDisable	228	apDisableAction	241
AAFEEnable	228	apExecute	241
AAFEExecute	229	apFullProjectName	242
AAFEExportParameter	229	API	217
AAFEExportSecurity	229	APIParameter	217
AAFIImportParameter	229	apLog	242
AAFOpenHelp	229	AppActivateVersion	244
AAFPopupMenuToString	229	AppChangeParameter	244
AAFPossible	229	AppControl	244
Abandon	11	AppDeleteVersion	244
Abs	200	AppDetectUPDAvailable	244
ActionBarSearchVisible	61	AppInstallUPD	244
ActionDeclareUndo	243	AppInstallVersion	244
ActionDo	243	AppleAdvertisingIdentifier	132
ActionPlanConfigure	241	AppleSignIn	132
ActionPlanExecute	241	AppListVersion	244
ActionPlanInfo	241	Apply	82
ActionRedo	243	AppointmentAdd	121
ActionUndo	243	AppointmentCreate	121
Add	82	AppointmentDelete	121
AddColumn	82	AppointmentDisplay	121
AddFavorite	249	AppointmentFirst	121
AddLine	82	AppointmentLast	121
AddReference	82	AppointmentList	121
Age	78	AppointmentListCalendar	121
AIDetect	151	AppointmentModify	121
AIDetectContour	151	AppointmentNext	121
AIDetectDrawing	151	AppointmentPrevious	122
AIDetectModel	151	AppointmentRead	122
AJAXAvailable	243	AppointmentReset	122
AJAXCallInProgress	231	AppointmentToiCalendar	122
AJAXExecute	244	AppParameter	245
AJAXExecuteAsynchronous	244	AppUPDAvailable	245
AlbumPicker	244	AppVersionInfo	245
AlbumSave	244	ArcCos	200
AnimationEnabled	69	ArcSin	200
AnimationGetParameter	69	ArcTan	200
AnimationMinFPS	69	ArcTan2	200
AnimationPlay	69	ArrayAdd	82
AnimationPrepare	69	ArrayAddLine	82
AnimationSetMode	69	ArrayAddReference	82
AnimationSetParameter	69	ArrayAddSorted	82
		ArrayCopy	82
		ArrayCount	83

INDEX

ArrayDelete	83	BannerLast	61
ArrayDeleteAll	83	BannerNext	61
ArrayDeleteDuplicate	83	BannerPrevious	61
ArrayDeleteLine	83	BannerStartSliding	61
ArrayDistinct	83	BannerStopSliding	61
ArrayInfo	83	BCCapture	245
ArrayInsert	83	BCDecode	245
ArrayInsertLine	83	BCToImage	245
ArrayInsertReference	83	BeaconDetectBackground	95
ArrayIntersect	83	BeaconDetectPrecise	95
ArrayMove	84	Beep	231
ArrayMoveLine	84	BinaryAND	202
ArrayReverse	84	BinaryNOT	202
ArraySeek	84	BinaryOR	202
ArraySeekByProcedure	84	BinaryXOR	202
ArrayShuffle	84	BitcoinValidAddress	217
ArraySort	84	BitmapInfo	137
ArraySwapLine	84	BreadcrumbAddLink	61
ArrayToCSV	84	BreadcrumbDeleteAll	61
ArrayToString	84	BreadcrumbDeleteLink	62
ArrayUnion	84	BreadcrumbInsertLink	62
ArrayUnionAll	84	BreadcrumbModifyLink	62
ArrowEnd	137	BreakAdd	34
ArrowStart	137	BreakDelete	34
Asc	71	BreakDeleteAll	34
ASPDDisplay	229	BrightnessAuto	132
ASPExecute	229	BrightnessGet	132
AssistedInputAdd	67	BrightnessModify	132
AssistedInputClose	67	BrowserClose	249
AssistedInputConfigure	67	BrowserHeight	249
AssistedInputDeleteAll	68	BrowserHistoryAdd	249
AssistedInputOpen	68	BrowserHistoryModify	250
AuthIdentify	107	BrowserIPAddress	250
AuthRefreshToken	107	BrowserIsConnected	250
AutomationEvent	231	BrowserMobile	250
AutomationParameter	231	BrowserName	250
awProcedureGetInfo	135	BrowserOpen	250
awSendAndGetInfo	135	BrowserOS	250
awSendInfo	135	BrowserPlatform	250
		BrowserPrint	250
		BrowserRunApp	250
		BrowserSlice	250
		BrowserType	250
		BrowserVersion	250
		BrowserWidth	250
		BTAcceptConnection	95
		BTActivate	95
		BTChangeConnectivity	95
		BTChangeVisibility	95
		BTConnectDevice	95

B

BackgroundTaskAdd	242
BackgroundTaskDelete	242
BackgroundTaskList	242
BankHoliday	78
BankHolidayAdd	78
BankHolidayDeleteAll	78
BankHolidayList	78
BannerFirst	61

BTDisableService	95
BTDisconnectDevice	95
BTEnableService	95
BTInfoDevice	95
BTInfoRadio	96
BTIsVisible	96
BTLEConnect	96
BTLEConnectionStatus	96
BTLEDescriptorReadValue	96
BTLEDescriptorWriteValue	96
BTLEDisconnect	96
BTLEGetACharacteristic	97
BTLEGetADescriptor	97
BTLEGetAllCharacteristics	97
BTLEGetAllServices	97
BTLEGetAService	97
BTLEInitialize	97
BTLEInitializeDevice	97
BTLEListDevice	97
BTLEStatus	97
BTListDevice	96
BTListRadio	96
BTListService	96
BTOpenDeviceProperties	96
BTSelectDevice	96
BTStack	96
BTStatus	96
BufferToHexa	71
BufferToInteger	71
BufferToPhrase	71
BufferToReal	71
BufferToURI	253
BuildAutomationVariant	231
BuildBrowsingTable	24
BurnerAddDirectory	149
BurnerAddFile	149
BurnerCancel	149
BurnerCompilationInfo	149
BurnerEject	149
BurnerErase	149
BurnerList	149
BurnerMediaFound	149
BurnerMediaInfo	149
BurnerMediaType	149
BurnerPath	149
BurnerProperty	149
BurnerSave	149
BurnerSelect	149
BurnerStatus	149
BurnerVolumeName	149

C

CalendarOpen	41
CalendarPicker	41
CalendarPosition	41
CalendarSelect	41
CalendarSelectCount	41
CalendarSelectMinus	41
CalendarSelectPlus	41
CallDLL32	217
CallInterface	217
CameraCapture	54
CameraChangeDriver	54
CameraConfigure	54
CameraDisplay	54
CameraGenerateThumbnail	54
CameraListDriver	54
CameraListParameter	54
CameraParameter	54
CameraPhoto	54
CameraRequestPermission	54
CameraRunApp	54
CameraStartDecoding	54
CameraStatus	54
CameraStop	54
CameraStopDecoding	54
CameraVideoStart	54
CameraVideoStop	54
CancelAWPContext	15
CancelWebserviceContext	259
CapitalizeEachWord	71
CapsLockVerify	229
CaptchaDisplay	61
CaptchaVerify	61
CDEject	148
CDList	148
CellCloseDialog	15
CellDisplayDialog	15
CertificateCheckDuplicate	205
CertificateCheckString	205
CertificateClientInfo	229
CertificateExtract	205
CertificateList	205
CertificateLoad	205
CertificateOpenProperties	205
CertificateSelect	205
CertificateSignExecutable	205
CertificateSignFile	205
CertificateSignString	205
ChangeAction	15

ChangeCharset	212	Complete	71
ChangeKeyboard	212	CompleteDir	71
ChangeSeparator	212	ComponentExecute	238
ChangeSkinTemplate	11	ComponentInfo	232
ChangeSourcePage	15	ComponentList	238
ChangeSourceWindow	40	ComponentLoad	238
ChangeTarget	15	ComponentLoadAll	238
ChangeTheme	132	ComponentLocate	238
Charact	71	Compress	76
CharactType	71	COMQueryInterface	217
CharactTypeCount	71	ConfigureAAF	232
CharactUnicode	71	ConfigureAWPContext	15
CheckAllFalse	84	Confirm	19
CheckAllTrue	84	ConfirmAsynchronous	19
CheckAnyFalse	84	ConnectionCount	229
CheckAnyTrue	84	ConsoleError	240
CheckBoxAdd	22	ConsoleRead	240
CheckBoxCount	22	ConsoleWrite	240
CheckBoxDelete	22	ContactAdd	122
CheckBoxInsert	22	ContactCreate	122
CheckUserIdentity	132	ContactDelete	122
ChronoEnd	78	ContactDisplay	122
ChronoPause	78	ContactEdit	122
ChronoReset	78	ContactFirst	122
ChronoResume	78	ContactLast	122
ChronoStart	78	ContactList	122
ChronoValue	78	ContactListSource	123
Clipboard	213	ContactModify	123
ClipboardFormat	213	ContactNext	123
ClipboardHeight	213	ContactPrevious	123
ClipboardRTF	213	ContactRead	123
ClipboardUnicode	213	ContactReset	123
ClipboardWidth	213	ContactSearch	123
Clone	231	ContactSelect	123
Close	11	Contains	71
CloseSession	98	ContextClose	15
ColorHue	137	ContextExist	15
ColorLightness	137	ContextOpen	15
ColorPalette	229	ContinentGet	246
ColorSaturation	137	ContinentList	246
CombinationCheckName	93	ControlEnabled	69
CombinationFromName	93	ControlFindHighlight	67
CombinationFromValue	93	ControlFindTextFirst	67
ComboClose	23	ControlFindTextNext	67
ComboOpen	23	ControlFindTextPrevious	67
COMCallMethod	217	ControlGrayed	69
COMCreateInstance	217	ControlInvisible	69
CommandLine	231	ControlOver	213
CommonLength	71	ControlPane	38
Compile	232	ControlTab	38

ControlToSource	166	DateTimeTimeZoneToLocal	79
ControlVisible	70	DateTimeToEpoch	79
Conversion	200	DateTimeToReal	79
Convert	153	DateTimeToString	79
CookieDelete	250	DateTimeUTCToLocal	79
CookieRead	250	DateTimeValid	79
CookieWrite	250	DateToDay	79
Cos	200	DateToDayInAlpha	79
CoTan	200	DateToDayOfWeek	79
CountryGet	246	DateToInteger	79
CountryList	246	DateToMonthInAlpha	79
CreateShortcut	217	DateToString	79
CriticalSection	222	DateToWeekNumber	79
CriticalSectionEnd	222	DateValid	80
CriticalSectionStart	222	DayNumberInAlpha	80
CSVToArray	85	dBackground	137
CurrentMonth	78	dbgAssert	239
CurrentPage	15	dbgAuditOption	239
CurrentTitle	11	dbgAuditStatus	239
CurrentWin	11	dbgCheckDifferent	239
CurrentYear	78	dbgCheckEqual	239
CursorCapture	213	dbgCheckFalse	239
CursorDisplay	213	dbgCheckNotNull	239
CursorPos	213	dbgCheckNull	239
		dbgCheckTrue	239
		dbgConnect	239
		dbgEnableAssert	239
dAlphaBlend	137	dbgEnableAudit	239
dArc	137	dbgEnableLog	239
DashAddWidget	39	dbgEndMemoryDiff	239
DashCount	39	dbgEndRecording	239
DashDelete	39	dbgError	239
DashDeleteAll	39	dbgInfo	239
DashDisplay	39	dbgLogOption	240
DashGetWidgetPosition	39	dbgLogStatus	240
DashInfoXY	39	dbgOnError	240
DashInitialConfiguration	39	dbgSaveDebugDump	240
DashLoadConfiguration	39	dbgSaveMemoryDump	240
DashMode	39	dbgStandardOutput	240
DashMoveWidget	39	dbgStartMemoryDiff	240
DashResizeWidget	39	dbgStartRecording	240
DashSaveConfiguration	39	dbgWriteCommentAudit	240
DataType	232	dbgWriteWarningAudit	240
DateDifference	78	dBlur	137
DatePicker	78	dBorder	138
DateSys	79	dChangeMode	138
DateTimeByDefault	79	dChord	138
DateTimeDifference	79	dCircle	138
DateTimeLocalToTimeZone	79	dClear	138
DateTimeLocalToUTC	79	dCopy	138



dCopyBlt	138	DiagramSave	58	DocPosition	47	dSaveImageICO	142
dCopyScreenImage	138	DiagramSearch	58	DocPositionTopPage	47	dSaveImageJPEG	142
dCopyWindowImage	138	DiagramSelectMinus	58	DocPrint	47	dSaveImagePNG	142
dCrop	138	DiagramSelectPlus	58	DocRedo	47	dSaveImageTIFF	142
DDEConnect	245	DiagramShapeToImage	58	DocRedoCount	47	dShadow	142
DDEDisconnect	245	DiagramStartModification	58	DocReplace	48	dSlice	142
DDEError	245	DiagramToImage	58	DocSave	48	dStartDrawing	142
DDEEvent	245	DiagramToSVG	58	DocSelect	48	dStraighten	142
DDEExecute	245	DiagramUngroup	58	DocStartModification	48	dText	142
DDEItem	245	Dialog	19	DocToHTML	48	dTextMarkdown	142
DDELink	245	DialogAsynchronous	19	DocToImage	48	dTextRTF	143
DDERecipient	245	diffApply	232	DocToPDF	48	dTransferToImage	143
DDERetrieve	246	diffCreate	232	DocToText	48	DurationToString	80
DDESend	246	dImageToArray	139	DocUndo	48	dVerticalSymmetry	143
DDEStart	246	Dimension	232	DocUndoCount	48	DynamicServingConfigure	229
DDEString	246	dInvertColor	139	DocUndoRedoDeleteAll	48	DynamicSiteDisplay	15
DDETopic	246	DisableAAF	232	DocUpdateFormula	48	<b>E</b>	
DDEUnLink	246	dLine	139	DocUpdateTableOfContents	48		
DDEUpdate	246	dLoadImage	139	DomoClose	108		
DDEWarmLink	246	dModifyContrast	139	DomoOpen	108		
dDisplay	138	dModifyHSL	139	DomoSend	108	Easter	80
dDominantImageColor	138	dModifyHue	140	DomoStatus	108	eChangeCurrency	248
DecimalPart	200	dModifyLightness	140	DomoSwitchOff	108	eCurrencyName	248
DecimalToSexagesimal	200	dModifyOpacity	140	DomoSwitchOn	108	eDefaultMemCurrency	248
DeclareAWPContext	15	dModifySaturation	140	DotNetAddHeader	106	eDefaultViewCurrency	248
DeclareWebserviceContext	259	DnDAccept	214	DotNetDelegate	106	EditAddToken	68
Decode	76	DnDCacheDashElement	214	DotNetError	106	EditDeleteAllToken	68
Decrypt	77	DnDCacheData	214	DotNetGetResult	106	EditDeleteToken	69
DecryptAsymmetric	77	DnDCursor	214	DotNetRun	106	EditInsertToken	69
DecryptStandard	77	DnDEvent	214	DotNetRunXML	106	EditTokenCount	69
DeepLinkProcedure	132	DnDGetData	214	DotNetVersion	106	eExchangeRate	248
DelayBeforeClosing	11	DnDIsDataAvailable	214	DownloadAdd	136	eInitCurrency	248
Delete	85	DNSAddAddress	110	DownloadGetInfo	136	EmailAddFolder	98
DeleteAll	85	DNSDeleteAddress	110	DownloadList	136	EmailBuildSource	98
DeleteColumn	85	DNSListAddress	110	DownloadRemove	136	EmailChangeFolder	98
DeleteLine	85	DocAdd	46	dPen	140	EmailChangeStatus	98
DeleteShortcut	217	DocChangeZOrder	46	dPixelColor	140	EmailCheckAddress	98
dEncrustColor	138	DocClose	46	dPixelOpacity	140	EmailCloseSession	98
dEndDrawing	138	DocDelete	46	dPoint	140	EmailConfigure	98
Dequeue	92	DocDeleteTableOfContents	47	dPolygon	141	EmailCopy	98
Deserialize	184	DocEndModification	47	dPolyline	141	EmailCurrentFolder	98
dFill	139	DocExitEdit	47	DrawingAvailable	141	EmailDeleteMessage	98
dFont	139, 224	DocFind	47	DrawingStyle	232	EmailExportHTML	98
dHalo	139	DocFindElementByName	47	dRectangle	141	EmailExpunge	99
dHorizontalSymmetry	139	DocGoIntoEdit	47	dRectangleGradient	141	EmailGetAll	99
DiagramEndModification	58	DocInfoXY	47	dResize	141	EmailGetIndicator	99
DiagramGroup	58	DocInsert	47	dRotation	141	EmailGetTimeOut	99
DiagramInfoXY	58	DocInsertTable	47	dRoundedRectangle	141	EmailImportHTML	99
DiagramLoad	58	DocInsertTableOfContents	47	dSaveImageBMP	141	EmailImportSource	99
DiagramLoadLibrary	58	DocOpen	47	dSaveImageGIF	142	EmailListFolder	99
						EmailLoadAttachment	99



EmailMessageLength	99	EmulateWarning	259	ExceptionRestoreParameter	247	fClose	87
EmailMsgError	99	EmulateWindow	259	ExceptionThrow	247	fCompare	87
EmailNbMessage	99	EmulateYesNo	259	Execute	232	fCompress	87
EmailOpenMail	250	Encode	77	ExecuteAAF	232	fContentType	87
EmailProgressBar	99	Encrypt	77	ExecuteAncestor	232	fCopyDir	87
EmailReadFirst	99	EncryptAsymmetric	77	ExecuteCode	232	fCopyFile	87
EmailReadFirstHeader	99	EncryptGenerateRSAKey	77	ExecuteDelayedProcedure	229	fCopyFileWebFolder	87
EmailReadLast	99	EncryptStandard	77	ExecuteFunction	232	fCreate	87
EmailReadLastHeader	99	EndAutomatedProcedure	232	ExecuteJS	233	fCreateLink	87
EmailReadMessage	100	EndEvent	216	ExecuteMainThread	222	fCurrentDir	87
EmailReadMessageHeader	100	EndProgram	232	ExecutePresetAction	233	fCurrentDrive	87
EmailReadNext	100	EndService	227	ExecuteProcess	233	fDataDir	88
EmailReadNextHeader	100	EndTimer	216	ExecuteRefreshUI	11	fDataDirCommon	88
EmailReadPrevious	100	EndTimerSys	216	ExecuteRefreshUIElement	11	fDataDirUser	88
EmailReadPreviousHeader	100	Enqueue	92	ExecuteRefreshUIParent	11	fDate	88
EmailRemoveFolder	100	EnumElement	238	ExecutionMode	233	fDateTime	88
EmailReset	100	EnumerationCheckName	93	Exp	200	fDecrypt	88
EmailRunApp	100	EnumerationCheckValue	93	ExplorerAccept	214	fDelete	88
EmailSaveAttachment	100	EnumerationFromName	93	ExplorerRetrieve	214	fDeleteFileWebFolder	88
EmailSeekFirst	100	EnumerationFromValue	93	ExtractLine	71	fDetectRemovableStorage	88
EmailSeekFirstNotes	100	EnumMenu	11	ExtractString	72	fDir	88
EmailSend	100	EnumSubElement	238	ExtractStringBetween	72	fDirAttribute	88
EmailSendMessage	100	EpochToDateTime	80			fDirectoryExist	88
EmailSetTimeout	100	eRoundDifference	248			fDirExist	88
EmailStartIMAPSession	101	Error	19			fDirRename	88
EmailStartNotesSession	101	ErrorAsynchronous	19			fDirSize	88
EmailStartOutlookSession	101	ErrorBuild	20	Factorial	201	fDriveInfo	88
EmailStartPOP3Session	101	ErrorChangeParameter	247	fAddBuffer	87	fDriveReady	89
EmailStartSession	101	ErrorInfo	247	fAddText	87	fEncrypt	89
EmailStartSMTPSession	101	ErrorPropagate	247	fAttribute	87	fExeDir	89
EmailStatus	101	ErrorReset	247	fAttributeReadOnly	87	fExtractPath	89
EmailUpdate	101	ErrorRestoreParameter	247	FaxConnect	117	fExtractResource	89
EmulateActionBar	258	ErrorThrow	247	FaxDelete	117	fFileExist	89
EmulateCheckBox	258	ErrorWithTimeout	20	FaxDisconnect	117	fFileName	89
EmulateCombo	258	eToCurrency	248	FaxInbox	117	fFind	89
EmulateConfirm	258	eToEuro	248	FaxOutbox	117	fGlobalDirCommon	89
EmulateDialog	258	EvaluateExpression	232	FaxPause	117	fGlobalDirUser	89
EmulateDnD	258	Event	216	FaxRestart	117	fGraphicFilter	89
EmulateError	258	EventChange	222	FaxResume	117	FileDisplay	230
EmulateInfo	258	EventClose	222	FaxSend	117	FileToArray	85
EmulateInput	258	EventCreate	222	FaxStatus	117	FileToMemory	184
EmulateMenu	259	EventDestroy	222	FBGetUserInformation	129	FileToMemoryList	22
EmulateMouse	259	EventExist	216	FBRequest	129	FileToMemoryTable	24
EmulateOKCancel	259	EventOpen	222	FBSessionStatus	129	FileToPage	167
EmulatePane	259	EventWait	222	FBStartSession	129	FileToScreen	167
EmulatePaneDock	259	ExceptionChangeParameter	247	FBStopSession	129	FileToURI	253
EmulatePaneUndock	259	ExceptionDisplay	247	fBuildPath	87	FileToURI	253
EmulateRadioButton	259	ExceptionEnable	247	fBuildRelativePath	87	Filter	85
EmulateSimulator	259	ExceptionInfo	247	fCacheDir	87	fImageSelect	89
EmulateSpeed	259	ExceptionPropagate	247	fChangeExtension	87	FinCurrentVal	193
				fChangeSize	87	FinDecreasingRedemption	193

F

FinError	193	fReportsAndQueriesDir	91
FinFutureVal	193	fResourceDir	91
FinInterestRate	193	fSaveBuffer	91
FinLinearRedemption	193	fSaveText	91
FinNetCurrentVal	193	fSeek	91
FinPaymentNb	193	fSelect	91
FinPeriodInterest	193	fSelectDir	91
FinRedemption	193	fSep	91
FinRepayVal	193	fSeparator	91
fInsertTextAtBeginning	89	fShortName	91
FirstDayOfMonth	80	fShortPath	91
FirstDayOfWeek	80	fSize	91
FirstDayOfYear	80	fSizeUncompressed	91
fIsImage	89	fSplit	91
fListDirectory	89	fStopCompress	91
fListDisk	89	fTempDir	91
fListFile	89	fTempFile	91
fListResource	89	fTempPath	91
fLoadBuffer	90	fTime	92
fLoadResource	90	FTPAttribute	103
fLoadText	90	FTPCommand	103
fLock	90	FTPConfigure	103
fLongName	90	FTPConnect	103
fLongPath	90	FTPCurrentDir	103
fMakeDir	90	FTPDate	103
fMemExist	92, 204	FTPDateTime	103
fMemOpen	92, 204	FTPDeleteFile	103
fMemWait	92, 204	FTPDisconnect	103
fMerge	90	FTPFileExist	103
fMIMEType	90	FTPGet	104
fMoveDir	90	FTPGetFile	104
fMoveFile	90	FTPListFile	104
FolderWeb	230	FTPMakeDir	104
FolderWeb_15	230	FTPName	104
FontCreate	225	FTPProxy	104
FontList	225	FTPRemoveDir	104
FontSelect	225	FTPRenameFile	104
fOpen	90	FTPSend	104
fOpenTempFile	90	FTPSize	104
fParentDir	90	FTPTime	104
FramesetDisplay	15	fTrackDirectory	92
FramesetRefresh	16	fTrackFile	92
FramesetUse	16	fTrackStop	92
fRead	90	fTrackStopAll	92
fReadLine	90	FullScreen	12
fReadLineRegularExpression	90	FullScreenDisable	250
FreeAWPContext	16	FullScreenEnable	251
FreeDLL	217	fUncompress	92
fRemoveDir	90	fUnlock	92
fRename	90	fWebDir	92

fWrite	92	GetTheme	133
fWriteLine	92	GetUUID	233
		GetUUID256	233
		gFont	147, 224
		gFontBold	147, 224
		gFontItalic	147, 224
gAlignment	147	gFontSize	147, 225
GalleryClosePopup	32	gFontStrikeout	147
GalleryDisplayPopup	32	gFontUnderlined	147, 225
GalleryPopupDisplayed	32	GglAddressToCoordinates	129
GanttAddHoliday	45	GglAnalyticsAddEvent	128
GanttAddLink	45	GglAnalyticsAddException	128
GanttAddTask	45	GglAnalyticsAddPage	128
GanttChangeMode	45	GglAnalyticsAddTiming	128
GanttDateTimePosition	45	GglDelete	127, 128
GanttDeleteAll	46	GglFillCalendar	128
GanttDeleteLink	46	GglFillCalendarByRequest	128
GanttDeleteTask	46	GglGetCalendar	128
GanttGetTask	46	GglGetDocument	128
GanttListLink	46	GglGetStaticMap	129
GanttListTask	46	GglListCalendar	128
GanttOrganizeTask	46	GglListContact	127
GanttTimeDisplayed	46	GglListContactByRequest	127
GanttWorkingHour	46	GglListDocument	128
GanttWorkingWeekDay	46	GglPrintDocument	127
gBackground	147	GglRequest	127, 128, 129
gBackgroundGradient	147	GglUploadDocument	127
gCircle	147	GglWrite	127, 128
gClickableLink	147	gImage	147
gCoord	147	gImageExists	147
GenerateGeometricImage	233	gLine	147
GenerateLorem	233	gLink	148
GeneratePassword	233	gLinkActivate	148
geoAzimuth	130	gPen	148
geoDistance	130	GPSTetectPosition	131
geoGetAddress	130	GPSEnd	131
geoGetArea	130	GPSFollowMovement	131
geoRunApp	130	GPSGetPosition	131
geoTrackingDisable	130	GPSInfo	131
geoTrackingEnable	130	GPSInitParameter	131
geoTrackingProcedure	130	GPSLastPosition	131
geoTrackingStatus	130	GPSStatus	131
GestureCurrentPointer	246	GPSStopDetection	131
GestureNbPointer	246	gpwActivateUser	225
GesturePointerType	247	gpwAddAuthLogin	225
GesturePosX	247	gpwAddUser	225
GesturePosY	247	gpwAddUserToGroup	225
GetActiveObject	108	gpwAuthLogin	225
GetColor	233	gpwBiometricAuthentication	226
GetGUID	233	gpwCheckPassword	226
GetIdentifier	233		

gpwCheckUser	226	grDestinationBMP	195	grPointSize	197	grSurfaceMesh	199
gpwConnectUser	226	grDestinationControl	195	grPointType	197	grTimeLevelMask	199
gpwDefault	226	grDestinationEMF	195	grPrint	197	grTimeLevelMax	199
gpwDisconnectUser	226	grDestinationImage	195	grRaised	197	grTitle	200
gpwEncryptedPasswords	226	grDestinationWMF	195	grRBAAddData	197	grTitleFont	200, 225
gpwEncryptPassword	226	grDestinationWnd	195	grRBMaximumValue	197	grTooltip	200
gpwEnumControl	226	grDonutHolePercentage	195	grRBRotationDirection	197	grType	200
gpwFindUser	226	grDraw	195	grRealTimeInterval	197	grWaterfallCategoryType	200
gpwForgottenPassword	226	gRectangle	148	grRotation	197	grWaterfallColor	200
gpwGetFileInfo	226	grExist	195	grSaveBMP	197	grWndSize	200
gpwGetLDAPInfo	226	grGetData	195	grSaveEMF	197	grZoom	200
gpwGetRights	226	grGradient	195	grSaveImage	197	gStoredValue	148
gpwGetUserInfo	226	grGraduate	195	grSaveParameter	197	gStoredValueEnable	148
gpwImportLDAPUsers	226	grGridlines	195	grSaveWMF	197	gToRetrieve	148
gpwInitAnalysis	226	grHMAddData	195	grScatter3DAddData	197	gToText	148
gpwInLDAPMode	226	grHMCaptionX	195	grScatter3DAltitudeColor	197		
gpwListOAuthParameters	226	grHMCaptionY	195	grScatter3DGridlines	198		
gpwLogin	227	grHMCColor	195	grScatter3DLinkPoint	198		
gpwMode	227	grHMDeleteColor	195	grScatter3DMarkCaption	198		
gpwModifyUser	227	grHMGetData	195	grScatter3DProjection	198		
gpwNone	227	grHolePercentage	196	grScatterAddDataXY	198		
gpwOpen	227	grImageSize	196	grScatterGetDataXY	198		
gpwOpenConfiguration	227	grIncreaseData	196	grScatterLinkPoint	198		
gpwOpenConnection	227	grInfoXY	196	grScatterPointSize	198		
gpwOpenLDAPSession	227	grLabel	196	grScatterSeriesColor	198		
gpwOpenSite	227	grLabelFont	196, 225	grScatterSeriesLabel	198		
gpwRegisteredUsers	227	grLegend	196	grSeriesColor	198		
gpwRemoteConnect	227	grLegendFont	196, 225	grSeriesLabel	198		
gpwResetPassword	227	grLineThickness	196	grSeriesLineType	198		
gpwSupervisor	227	grLoadParameter	196	grSeriesOpacity	198		
gpwVisitor	227	grMask	196	grSeriesPointType	198		
gr3DSPParameter	194	grMenu	196	grSeriesSecondaryAxis	198		
grAddData	194	grMenuOption	196	grSeriesType	198		
grAddTimeData	194	grOrientation	196	grSmoothing	198		
grAutoRefreshSeries	194	grOrigin	196	grSourceCategoryLabel	198		
grAxisPosition	194	GroupAdd	123	grSourceSeries	198		
grAxisTitle	194	GroupDelete	123	grStartAngle	199		
grAxisTitleFont	194, 225	GroupFirst	123	grStockAddData	199		
grAxisVisible	194	GroupLast	123	grStockGetData	199		
grBarColor	194	GroupModify	123	grSunburstAddData	199		
grCategoryColor	194	GroupNext	123	grSunburstGetData	199		
grCategoryLabel	194	GroupPrevious	123	grSunburstPullOut	199		
grClearPict	194	GroupRead	124	grSunburstRootSection	199		
grColor	194	GroupReset	124	grSunburstSectionColor	199		
grColumnAggregation	194	grOverlayChart	196	grSunburstStartAngle	199		
grColumnSpacing	194	grParameter	196	grSurfaceAddData	199		
grComparativeSeriesPosition	194	grPiePullOut	196	grSurfaceAltitudeColor	199		
grCreate	195	grPieStartAngle	196	grSurfaceGetData	199		
grDeleteAll	195	grPointColor	197	grSurfaceGridlines	199		
grDeleteSeries	195	grPointInfo	197	grSurfaceMarksCaption	199		

H

hbDeleteTable	179	HCreation	155	HDisconnectClient	171	HIndex	159
hbGetTableDescription	179	HCreationIfNotFound	155	HDropView	171	HIndexingInProgress	159
hbListTable	179	HCross	155	HDuplicateDatabase	171	HInfoAnalysis	159
hbRead	179	HDBCCreation	155	HDuplicateRecord	156	HInfoBackup	171
HBuildKeyValue	153	HDBDescribeField	155	HelpFile	233	HInfoDatabaseProperty	171
HBuildKeyValueANSI	153	HDBDescribeFile	155	HEndNoDatabaseAccess	171	HInfoDatabaseRights	172
hbWrite	179	HDBDescribeIndex	156	HEndNoModif	157	HInfoFile	159
HCancelAlias	153	HDBIndex	156	HError	157	HInfoFileProperty	172
HCancelBackup	170	HDBOpen	156	HErrorDuplicates	157	HInfoFileRights	172
HCancelDeclaration	154	HDBOpenNoLock	156	HErrorInfo	157	HInfoGroup	172
HCancelSeek	154	HDBSortType	156	HErrorIntegrity	157	HInfoLock	172
HChangeConnection	154	HDeactivateAutoFilter	156	HErrorLock	157	HInfoLog	172
HChangeDir	154	HDeactivateFilter	156	HErrorModification	157	HInfoMemo	159
HChangeKey	154	HDeactivateServerTrigger	170	HErrorPassword	157	HInfoReplica	159, 180
HChangeLocation	154	HDeactivateTrigger	156	HErrorStatusModification	157	HInfoServer	172
HChangeLogDir	154	HDeclare	156	HexaToBuffer	72	HInfoServerProperty	172
HChangeName	154	HDeclareExternal	156	HexaToInt	72	HInfoServerRights	172
HChangePassword	170	HDelete	156	HExecuteProcedure	171	HInfoTask	172
HChangeRplDir	154, 180	HDeleteAll	156	HExecuteQuery	157	HInfoUser	172
HCheckIndex	154	HDeleteBackup	170	HExecuteScheduledBackup	171	HInfoView	172
HCheckMemo	154	HDeleteDatabase	170	HExecuteScheduling	171	HInfoViewRights	172
HCheckServerUpdate	170	HDeleteDirectory	170	HExecuteSQLQuery	157	HInitHF55	159
HCheckStructure	154	HDeleteFile	170	HExecuteView	157	HInitSubscriber	159, 180
HCheckUnaltered	154	HDeleteGroup	170	HExportCSV	157	HiWord	217
HClearWorkingDir	170	HDeleteLink	170	HExportJSON	157	HJSONToRecord	160
HClose	154	HDeleteLink	170	HExportJSONString	157	HLast	160
HCloseAnalysis	154	HDeleteParameter	170	HExportXLS	158	HLinkMemo	160
HCloseConnection	154	HDeleteQuery	170	HExportXML	158	HListAnalysis	160
HClusterAddNode	168	HDeleteScheduledBackup	170	HExtractMemo	158	HListConnectedUser	172
HClusterDeleteNode	168	HDeleteScheduling	171	HFileExist	158	HListConnection	160
HClusterIgnoreSynchro	168	HDeleteServerTrigger	171	HFilter	158	HListCustomFolder	160
HClusterMasterNode	168	HDeleteSet	171	HFilterContains	158	HListDatabase	172
HClusterNodeInfo	168	HDeleteTask	171	HFilterIdentical	158	HListFile	160
HClusterParameter	168	HDeleteTrigger	156	HFilterIncludedBetween	158	HListFullTextIndex	160
HClusterStart	168	HDeleteUser	171	HFilterStartsWith	158	HListGroup	172
HClusterState	168	HDeleteView	156	HFirst	158	HListItem	160
HClusterStop	168	HDescribeConnection	156	HFlush	158	HListKey	160
HClusterSynchronizationInfo	169	HDescribeFile	156	HForward	158	HListLink	160
HCompareItem	155	HDescribeFullTextIndex	156	HFound	158	HListParameter	172
HComputer	155	HDescribeItem	156	HFree	158	HListProvider	160
HConnect	155	HDescribeLink	156	HFreePosition	158	HListQueryParameter	160
HConnectionQuality	170	HDescribeServerTrigger	171	HFreeQuery	158	HListREP	161
HConvertX	155	HDescribeTrigger	156	HFTListWord	158	HListScheduledBackup	172
HCopyFile	170	hdfsCloseConnection	179	HGetCurrentPosition	159	HListScheduledOptimization	172
HCopyRecord	155	hdfsCopyFrom	179	HHistoryModification	159	HListScheduling	173
HCreateMasterReplica	155, 180	hdfsCopyTo	179	HImportHF55	159	HListServer	173
HCreateMovableReplica	155, 180	hdfsCreateDirectory	179	HImportJSON	159	HListServerTrigger	173
HCreateServerTrigger	170	hdfsDelete	179	HImportText	159	HListStopWord	161
HCreateSubscriberReplica	155, 180	hdfsList	179	HImportXLS	159	HListStoredElement	173
HCreateView	155	hdfsOpenConnection	179	HImportXML	159	HListSynonym	161
		hdfsRename	179				



HListTask	173	HPass	162	HSeek	164	HTMLEditorSearch	60
HListTrigger	161	HPrepareQuery	162	HSeekFirst	164	HTMLEditorSelect	60
HListUser	173	HPrepareSQLQuery	162	HSeekLast	164	HTMLEditorToEmail	60
HLoadParameter	173	HPrevious	162	HSendMessageToClient	175	HTMLEditorToPDF	60
HLockFile	161	HPriority	174	HServerMaintenance	175	HTMLEnableDebugger	211
HLockRecNum	161	HPriorityClient	175	HServerStatus	175	HTMLFindElementByClass	211
HLogInfo	161	HRead	162	HServerWorkingDir	175	HTMLFindElementByID	211
HLogRecreate	161	HReadFirst	162	HSetCache	175	HTMLFindElementByName	211
HLogRestart	161	HReadLast	162	HSetDuplicates	164	HTMLFindElementByTag	211
HLogStop	161	HReadNext	162	HSetIntegrity	164	HTMLFromEmail	59
HManageServerUpdate	173	HReadPrevious	162	HSetLog	164	HTMLInsertAfter	211
HManageTask	173	HReadSeek	162	HSetMemo	164	HTMLInsertBefore	211
HMergeView	161	HReadSeekFirst	163	HSetPosition	164	HTMLInsertFirstChild	211
HMode	161	HReadSeekLast	163	HSetREP	164	HTMLInsertLastChild	211
HModify	161	HRecNum	163	HSetReplication	164, 180	HTMLModifyAttribute	211
HModifyDatabaseProperty	173	HReconnect	175	HSetServer	175	HTMLNavigate	59
HModifyDatabaseRights	173	HRecordDate	163	HSetTransaction	165	HTMLOnErrorJS	59
HModifyFileProperty	173	HRecordToJSON	163	HSetTrigger	165	HTMLOpen	211
HModifyFileRights	173	HRecordToString	163	HSimulateNetwork	175	HTMLRemove	211
HModifyGroup	173	HRecordToXML	163	HSL	143	HTMLRemoveAttribute	211
HModifyScheduledBackup	173	HRecreateSubscriberReplica	163, 180	HSortView	165	HTMLRemoveChild	211
HModifyScheduling	174	HRefreshView	163	HStartServer	176	HTMLSave	211
HModifyServerProperty	174	HRegenerateFile	163	HStartServerUpdate	176	HTMLToPDF	211
HModifyServerRights	174	HReset	163	HStatCalculate	165	HTMLToRGB	143
HModifyStructure	161	HResetClient	175	HStatDate	165	HTMLToRTF	72
HModifyTask	174	HResetGroup	175	HState	165	HTMLToText	72
HModifyUser	174	HResetUser	175	HStatNbDuplicates	165	HTMLUnwrapAllChildren	212
HModifyViewRights	174	HRestoreBackup	175	HStatNbRec	165	HTMLValidateForm	59
HNbRec	161	HRestorePosition	163	HStatNbRecRange	165	HTMLWrap	212
HNext	161	HRetrieveBackup	175	HStatTime	165	HTMLWrapAllChildren	212
HNoDatabaseAccess	174	HRetrieveItem	163	HStopServer	176	HToFile	165
HNoModif	161	HRetrieveLog	175	HSubstDir	165	HToItem	165
HNotifAddCCRecipient	174	HRetrieveRecord	163	HSuggestKey	176	HTrack	176
HNotifAddEmailRecipient	174	HRplDeclareLink	163, 180	HSuggestKeyEnd	176	HTrackStop	176
HNotifConfigure	174	HRplFilterProcedure	164, 180	HSuggestKeyStart	176	HTransaction	165
HNotifDeleteCCRecipient	174	HRplManageFile	164, 180	HSynchronizeReplica	165, 181	HTransactionCancel	165
HNotifDeleteEmailRecipient	174	HRplManageItem	164, 180	HTMLBuildString	211	HTransactionEnd	165
HNotifListCCRecipient	174	HRplPass	164, 180	HTMLClassAdd	230	HTransactionFree	166
HNotifListEmailRecipient	174	HRSAddConfig	180	HTMLClassDelete	230	HTransactionInProgress	166
HOnError	162	HRSDeleteConfig	181	HTMLClassToggle	230	HTransactionInterrupted	166
HOnServerCall	174	HRSExecute	181	HTMLControlValue	59	HTransactionIsolation	176
HOpen	162	HRSInfo	181	HTMLEditorExecuteJS	60	HTransactionList	176
HOpenAnalysis	162	HRSInit	181	HTMLEditorFormatSelection	60	HTransactionSavepoint	176
HOpenConnection	162	HRSListConfig	181	HTMLEditorFromEmail	60	HTransactionStart	166
HOperationInProgress	174	HRSModifyConfig	181	HTMLEditorGet	60	HTriggerRecordBefore	166
HOptimize	162	HRSWaitForDataProcess	180	HTMLEditorGetDocument	60	HTTPAddFile	101
HOptimizeQuery	162	HSave	164	HTMLEditorGetSelection	60	HTTPAddParameter	101
HourGlass	214	HSaveParameter	175	HTMLEditorInsert	60	HTTPCancelForm	101
HourglassNetwork	213	HSavePosition	164	HTMLEditorInsertTable	60	HTTPCertificate	101
HOut	162	HSecurity	164	HTMLEditorReplace	60	HTTPConfigure	101

HTTPCookieGet	102	Iconize	12	In64bitMode	233	InStoredProcedureMode	235
HTTPCookieManage	102	iCreateFont	187, 225	InAJAXMode	234	InSysNotificationMode	235
HTTPCookieRead	102	iCurrentFile	187	InAndroidEmulatorMode	234	IntegerPart	201
HTTPCookieReplace	102	IdentifierAWPContext	16	InAndroidMode	234	IntegerToDate	80
HTTPCookieReset	102	iDestination	187	InAndroidSimulatorMode	234	IntegerToDay	80
HTTPCookieWrite	102	iDidotFont	187, 225	inAppCheckSubscription	131	IntegerToDayInAlpha	80
HTTPCreateForm	102	iDirImageHTML	187	inAppConsumeProduct	131	IntegerToMonthInAlpha	80
HTTPDestination	102	iDocumentCanceled	187	InAppleWatchEmulatorMode	234	IntegerToTime	80
HTTPGetResult	102	iDocumentPrinted	187	InAppleWatchMode	234	IntegerToWeekNumber	80
HTTPListCertificate	102	iEllipse	188	inAppListProductInfo	131	InternetConnected	103
HTTPProgressBar	102	iEndPrinting	188	inAppListPurchase	131	InTestMode	235
HTTPRequest	102	iEndReport	188	inAppPurchaseProduct	131	InTSEMode	235
HTTPResetForm	102	iEscape	188	inAppRestorePurchases	131	IntToHexa	72
HTTPSend	102	iFont	188, 225	InAutomaticTestMode	234	InUMCMode	236, 257
HTTPSendForm	102	iForceComplement	188	InAWPMode	234	InVGAMode	236
HTTPSimulateNetwork	102	iGroupAdd	188	InBackgroundMode	234	InWebMode	236
HTTPTimeOut	103	iGroupConfigure	188	InComponentMode	234	InWebserviceMode	236
HUnlockFile	166	iGroupPrint	188	InDelayedProcedureMode	234	InWebSocketServerMode	236
HUnlockRecNum	166	iHLine	188	InDeviceOwnerMode	234	InWidgetMode	236
HUpdateLink	176	iImageHeight	188	iNestedHeaderFooter	189	InWindowsMode	236
HUpdateQuery	176	iImageWidth	188	iNewLine	189	iOSAppGroupDirectory	115
HUpdateSet	176	iInfoPrinter	188	InFactoryMode	234	iOSExcludeFileFromiCloud	115
HUpdateView	176	iInitInternalReport	188	Info	20	iPageEnd	189
HVersion	166	iInitReportQuery	188	InfoAsynchronous	20	iPageHeight	189
HViewToFile	166	iInitReportQueryConnection	188	InfoBuild	20	iPageHeightLeft	189
HWrite	166	iInitSubReport	188	InfoWithTimeout	20	iPageNum	189
		iLastFile	188	IniOSEmulatorMode	234	iPageWidth	189
		iLine	188	IniOSMode	234	iParameter	189
		iLink	188	IniOSSimulatorMode	235	iParameterDuplicate	189
iAddAttachment	187	iListNestedReports	189	INIRead	235	iParameterExport	189
iAddBookmark	187	iListPrinter	189	InitRandom	235	iParameterPDF	189
iAreaHeight	187	ImageArea	144	InitWindow	12	iParameterPreview	190
IBANValid	217	ImageFirst	144	INIWrite	235	iParameterReport	190
iBorder	187	ImageInfoClose	143	InJavaMode	235	iParameterViewer	190
iCalendarToAppointment	122	ImageInfoList	143	InKioskMode	235	iParameterWatermark	190
iCalendarToOrganizer	42	ImageInfoModify	143	InLinuxMode	235	iParameterXLS	190
iCalendarToScheduler	43	ImageInfoSave	143	InMacCatalystMode	235	iPicaToMM	190
iChangeSubReportSource	187	ImageInfoValue	143	InPageWithinWindowMode	235	iPreview	190
iCloseReport	187	ImageLast	144	InPHPMode	235	iPrint	190
iCloudCléValeurListeClé	135	ImageLayerVisible	144	InPHPSimulatorMode	235	iPrintArea	190
iCloudKeyValueExist	135	ImageNext	144	InPrelaunchedSessionMode	235	iPrintAreaHTML	190
iCloudKeyValueRead	135	ImageOccurrence	144	Input	20	iPrintAreaMarkdown	190
iCloudKeyValueRemove	135	ImagePrevious	144	InReportsAndQueriesMode	235	iPrintAreaRTF	190
iCloudKeyValueWatch	135	ImageScrollingPosition	144	Insert	85	iPrintBarcode	190
iCloudKeyValueWrite	135	ImageStartScrolling	144	InsertColumn	85	iPrintBlock	190
iColumnEnd	187	ImageStopScrolling	144	InsertLine	85	iPrintBodyComplement	190
iColumnNum	187	ImageXPos	144	InsertReference	85	iPrintDiagram	190
iConfigure	187	ImageYPos	144	InServiceMode	235	iPrintDoc	190
iConfigureReport	187	iMargin	189	InSimulatorMode	235	iPrintDuplicate	190
IconInstall	217	iMMToPica	189	Instance	217	iPrintImage	190



iPrintingNation	191	JavaLoad	93
iPrintPDF	191	jQuery	230
iPrintReport	191	jQueryExecute	230
iPrintSignature	191	JSEndEvent	230
iPrintWord	191	JSEvent	230
iPrintXLS	191	JSInfoEvent	230
iReportPrintingStatus	191	JSInterruptEvent	230
iReset	191	JSMethod	230
iResetNbPages	191	JSONExecute	230
iRoundBorder	191	JSONExecuteExternal	230
iSelectFont	191, 225	JSONToString	72
iSequencingAdd	191	JSONToVariant	72
iSequencingAddDiagram	191	JSONToXML	118
iSequencingAddDoc	191	JSONValid	72
iSequencingAddPDF	191	JSPROPERTY	230
iSequencingAddXLS	191		
iSequencingPrint	191		
iSetAdditionalXMP	191		
IsEven	201	KanbanAddCard	53
iSkipLine	192	KanbanAddList	53
iSkipPage	192	KanbanCount	53
IsNumeric	236	KanbanDeleteAllCard	53
IsOdd	201	KanbanDeleteCard	53
iSubstBlock	192	KanbanDisplay	53
iTextHeight	192	KanbanFilter	53
iTextWidth	192	KanbanInfoXY	53
iTransparentMagenta	192	KanbanInsertCard	53
iVLine	192	KanbanListIndex	53
iWindowCancel	192	KanbanMoveCard	53
IWListAdd	40	KanbanMoveList	53
IWListCount	40	KanbanPositionList	53
IWListDelete	40	KeyCalcActivationKey	240
IWListDeleteAll	40	KeyCompareKey	240
IWListDisplay	40	KeyGenerateInitialKey	241
IWListInsert	41	KeyGetIdentifier	241
IWListNext	41	KeyPressed	218
IWListPosition	41		
IWListPrevious	41		
iXPos	192		
iYPos	192	LanguageToCharset	212
		LanguageToName	212
		LanguageToNation	212
		LargeFontFactor	236
J2EEAddHeader	105	LastDayOfMonth	80
J2EEError	105	LastDayOfWeek	80
J2EEGetResult	105	LastDayOfYear	80
J2EERun	105	LDAPAddAttribute	126
J2EERunXML	105	LDAPConnect	126
JavaExecute	93	LDAPDelete	126
JavaExecuteFunction	93	LDAPDeleteAttribute	126

K

L

LDAPDeleteAttributeValue	126	LocalStorageDeleteAll	186
LDAPDisconnect	126	LocalStorageGet	186
LDAPFind	126	LocalStorageOccurrence	186
LDAPListAttribute	126	LocalStorageValueName	186
LDAPListChildren	126	LockAppOff	253
LDAPMode	126	LockAppOn	253
LDAPNbValue	126	LockAppPassword	254
LDAPRenameAttribute	127	LockAppTimeout	254
LDAPReplaceAttribute	127	Log	201
LDAPReplaceAttributeValue	127	LooperAdd	32
LDAPReset	127	LooperAddInProgress	32
LDAPStartAdd	127	LooperAddIP	32
LDAPStartModify	127	LooperAddIW	32
LDAPValidateAdd	127	LooperAddLine	32
LDAPValidateModify	127	LooperBreakSubscript	32
LDAPValue	127	LooperCancelSwipe	32
LeapYear	80	LooperCollapse	32
LEDSwitchOff	133	LooperCollapseAll	32
LEDSwitchOn	133	LooperCollapseExpand	32
Left	72	LooperCount	32
Length	72	LooperDelete	32
LengthToString	72	LooperDeleteAll	32
LineToPosition	72	LooperDeleteSelect	32
ListAdd	22	LooperDisplay	32
ListCount	22	LooperEndFilling	33
ListDelete	23	LooperExpand	33
ListDeleteAll	23	LooperExpandAll	33
ListDisplay	23	LooperInfoXY	33
ListDLL	238	LooperInsert	33
ListInfo	92	LooperInsertIP	33
ListInfoXY	23	LooperInsertIW	33
ListInsert	23	LooperInsertLine	33
ListModify	23	LooperModify	33
ListMoveLine	23	LooperModifyIP	33
ListPosition	23	LooperModifyIW	33
ListSeek	23	LooperModifyLine	33
ListSelect	23	LooperMoveLine	33
ListSelectCount	23	LooperPosition	33
ListSelectMinus	23	LooperRecNum	33
ListSelectPlus	23	LooperRefreshVisible	33
ListSort	23	LooperSave	33
Ln	201	LooperSearch	34
LoadDLL	218	LooperSearchEverywhere	34
LoadError	212	LooperSelect	34
LoadImage	148	LooperSelectCount	34
LoadImageSheet	148	LooperSelectMinus	34
LoadWDL	236	LooperSelectPlus	34
LocalStorageAdd	186	LooperSort	34
LocalStorageAvailable	186	LooperStartFilling	34
LocalStorageDelete	186	LooperToExcel	34

LooperToWorld	34	MatCreate	202	MemcachedAddSuffix	181	mlzDeleteAll	36
LooperToXML	34	MatDelete	202	MemcachedAssign	182	mlzInsertLine	36
Lower	72	MatDeterminant	202	MemcachedAssignKeyGroup	182	MobileNetworkStatus	133
LoWord	218	MatError	202	MemcachedCAS	182	MongoAdd	183
LuhnCalculate	77	MatExist	202	MemcachedCASKeyGroup	182	MongoAggregationPipeline	183
LuhnCheck	77	MatFill	202	MemcachedDecrement	182	MongoCreate	183
M		MatFloatAdd	202	MemcachedDelete	182	MongoCreateCollection	183
		MatFloatMultiply	202	MemcachedDeleteKeyGroup	182	MongoDeleteAll	183
		MatInvert	202	MemcachedGetKeyGroup	182	MongoDeleteCollection	183
		MatMultiply	203	MemcachedIncrement	182	MongoDeleteDatabase	183
		MatNbColumn	203	MemcachedModify	182	MongoDeleteOne	183
MakeInteger	218	MatNbLine	203	MemcachedModifyKeyGroup	182	MongoExecuteCommand	183
Map	85	MatRead	203	MemcacheGet	182	MongoFind	183
MapAddImage	56	MatReadColumn	203	MemoryToFile	185	MongoGridFSDeleteFile	183
MapAddItinerary	56	MatReadLine	203	MenuAdd	18	MongoGridFSDeleteFileByID	183
MapAddMarker	56	MatStack	203	MenuAddMenu	18	MongoGridFSGetFile	183
MapAddShape	56	MatTranspose	203	MenuAddOption	18	MongoGridFSGetFileByID	183
MapChangeStyle	56	MatWrite	203	MenuAddPopup	18	MongoGridFSGetFileInfo	183
MapClosePopup	56	Max	86	MenuAddSeparator	18	MongoGridFSListFile	183
MapConfigureGglAPI	56	Maximize	12	MenuAddURLOption	18	MongoGridFSSendFile	183
MapDeleteAll	56	MCIBeginning	149	MenuClone	18	MongoInfo	183
MapDeleteImage	56	MCIClose	149	MenuDelete	18	MongoModifyAll	184
MapDeleteItinerary	56	MCIEnd	150	MenuExist	18	MongoModifyOne	184
MapDeleteMarker	56	MCIErr	150	MenuInsertMenu	18	MonthNumberInAlpha	80
MapDeleteShape	56	MCIExecute	150	MenuInsertOption	19	MouseXPos	214
MapDisplayArea	56	MCIHeight	150	MenuInsertSeparator	19	MouseYPos	214
MapDisplayPopup	57	MCIMsgErr	150	MenuInvisible	19	Move	86
MapDisplayPosition	57	MCIOpen	150	MenuIsMarked	19	MoveColumn	86
MapEndOfMove	57	MCIPause	150	MenuLabel	19	MoveLine	86
MapExecuteJS	57	MCIPlay	150	MenuMark	19	MQTTConnect	130
MapFollowMovement	57	MCIPosition	150	MenuSelectMinus	19	MQTTDisconnect	130
MapGetImage	57	MCIPositionFormat	150	MenuSelectPlus	19	MQTTPublish	130
MapGetJSObject	57	MCIRetrieve	150	MenuState	19	MQTTSubscribe	130
MapGetPosition	57	MCISize	150	MenuUnMark	19	MQTTUnsubscribe	130
MapGetStyle	57	MCIStatus	150	Message	20	MQTTUnsubscribeAll	130
MapInfoPosition	57	MCIStopPlaying	150	MessTranslate	212	MultimediaBeginning	55
MapInfoXY	57	MCITrack	150	Middle	73	MultimediaClose	55
MapItineraryDistance	57	MCITrackCount	150	Min	86	MultimediaControlVisible	55
MapLicenseBing	57	MCIVolume	150	mlDeleteVariable	204	MultimediaDuration	55
MapLicenseGgl	57	MCIWidth	150	mlEnd	204	MultimediaEnd	55
MapLimitArea	57	MCIWindowSize	150	mlExecute	204	MultimediaHeight	55
MapModifyImage	57	MDIActive	18	mlGetVariable	204	MultimediaOpen	55
MapModifyMarker	57	MDIEnumChild	18	mlInitialize	204	MultimediaPause	55
MapModifyShape	57	MDIMother	18	mlListVariable	204	MultimediaPlay	55
MapPopupDisplayed	57	MDIOpen	18	mlLoadWorkspace	204	MultimediaPosition	55
MarkdownToDOCX	72	MDIWindowMenu	18	mlResetWorkSpace	204	MultimediaStatus	55
MarkdownToHTML	72	Mean	86	mlSaveWorkspace	204	MultimediaStop	55
MarkdownToPDF	73	MemcachedAdd	181	mlSetVariable	204	MultimediaVolume	55
MarkdownToText	73	MemcachedAddKeyGroup	181	mlzAddLine	36	MultimediaWidth	55
MatAdd	202	MemcachedAddPrefix	181	mlzDelete	36	Multitask	216
MatchRegularExpression	73						
MatCopy	202						

NotifPushSend	133	OrganizerAddAppointment	42
Now	80	OrganizerBackgroundColor	42
NumberInWords	73	OrganizerChangeMode	42
NumHelp	236	OrganizerDateVisible	42
NumToString	73	OrganizerDeleteAll	42
		OrganizerDisplay	42
		OrganizerDisplayNextPeriod	42
		OrganizerInfoXY	42
OBEXConnect	105	OrganizerListAppointment	42
OBEXDisconnect	105	OrganizerMode	42
OBEXGetFile	105	OrganizerPosition	42
OBEXGetVCard	105	OrganizerSelectedRange	42
OBEXListFile	105	OrganizerSelectMinus	43
OBEXSendFile	105	OrganizerSelectPlus	43
OBEXSendVCard	105	OrganizerTimeDisplayed	43
OCRAddLanguage	151	OrganizerToiCalendar	43
OCRDeleteAllLanguages	151	OrganizerWorkingHour	43
OCRDetectTextArea	151	OrganizerZoom	43
OCRExtractText	151	OrgDelete	58
OCRExtractTextBlock	151	OrgDeleteAll	59
OCRLoadLanguage	151	OrgDisplay	59
OCRModelPathDirectory	151	OrgInfoXY	59
OCRParameter	151	OrgModify	59
OemToAnsi	73	OrgOccurrence	59
OKCancel	20	OrgSaveImage	59
OKCancelAsynchronous	20	OrgSelect	59
OLEActivate	108	OrgSelectMinus	59
OLECopy	108	OrgSelectPlus	59
OLECreateCopy	108	OutlookDefaultProfile	101
OLECreateEmpty	108	OutlookListProfile	101
OLELoad	109	OutlookStartSession	101
OLEPaste	109		
OLEPlay	109		
OLESave	109		
OLEServer	109		
Open	12		
OpenAPIToHTTP	103		
OpenAsynchronous	12		
OpenChild	12		
OpenDocument	236		
OpenIDReadIdentity	107		
OpenMainMenu	12		
OpenMobileWindow	12		
OpenPopup	12		
OpenPopupAsynchronous	12		
OpenPopupPosition	12		
OpenSister	12		
OpenSSLCheck	218		
OrgAdd	58		
OrgAddChild	58		

PageToEmail	17	PHPExecute	231
PageToFile	167	PhraseToBuffer	73
PageToJSP	17	PicClose	52
PageToPHP	17	PicCreate	52
PageToSource	167	PicMergeLayer	52
PageUse	17	PicOpen	52
PageVisible	17	PicRectangularSnip	53
PageWriteHeaderHTTP	17	PicRectangularSnipOCR	53
PanelActivate	40	PicResize	53
PanelAttach	40	PicRotation	53
PanelCreate	40	PicSave	53
PanelDetach	40	Ping	218
PanelLoadConfiguration	40	Pop	93
PanelPosition	40	PopupAnimate	17
PanelSaveConfiguration	40	PopupClose	17
ParallelTaskCancel	224	PopupClosePage	17
ParallelTaskConfigure	224	PopupDisplay	17
ParallelTaskExecute	224	PopupDisplayPage	17
ParallelTaskExecuteAfter	224	Position	73
ParallelTaskExecuteAfterAll	224	PositionOccurrence	73
ParallelTaskWaitAll	224	PositionToLine	73
PDFAddImage	210	PostMessage	216
PDFAddWatermark	210	Power	201
PDFClose	210	PowerStatus	133
PDFExtractPage	210	PreviousPage	17
PDFGetSignature	210	PreviousWin	12
PDFGetSignatureDate	210	ProfilerEnd	236
PDFIsProtected	210	ProfilerStart	237
PDFMerge	210	ProgressBar	20
PDFMergeWithPassword	210	ProgressBarDisable	62
PDFNbSignatures	210	ProgressBarEnable	62
PDFNumberOfPages	210	ProgressBarExecute	62
PDFOpen	210	ProjectInfo	238
PDFPassword	210	Proxy	103
PDFReaderClose	52	Push	93
PDFReaderGoToBookmark	52	PVTAxisXY	35
PDFReaderHighlight	52	PVTCalculateAll	35
PDFReaderInfoXY	52	PVTCalculateUpdate	35
PDFReaderOpen	52	PVTCollapse	35
PDFReaderRotatePage	52	PVTCollapseAll	35
PDFReaderSave	52	PVTCompareDateRange	35
PDFReaderSeekFirst	52	PVTEExpand	35
PDFReaderSeekNext	52	PVTEExpandAll	35
PDFReaderSeekPrevious	52	PVTFilter	35
PDFSave	210	PVTHeaderValue	35
PDFSign	210	PVTInfoXY	35
PDFToText	210	PVTListPositionHeader	35
Phonetic	73	PVTLoad	35
PhotoRunApp	244	PVTSave	35
PHPDisplay	231	PVTSelect	35

PVTSelectCount	36
PVTToExcel	36
PythonAddPath	94
PythonExecute	94
PythonInitialize	94
PythonTerminate	94

Q

QueueInfo	93
-----------	----

R

RadioButtonAdd	22
RadioButtonCount	22
RadioButtonDelete	22
RadioButtonInsert	22
Random	237
RandomCryptoBuffer	237
RandomCryptoInteger	237
RealToDateTime	81
RecorderAction	134
RecycleBinClear	215
RecycleBinDelete	215
RecycleBinListFile	215
RecycleBinRestore	215
RedisConnect	184
RedisExecuteCommand	184
RedisGet	184
RedisPipelineAdd	184
RedisPipelineGet	184
RedisPipelineReset	184
RedisPublish	184
RedisSet	184
RedisSubscribe	184
RedisUnsubscribe	184
Reduce	86
RegexMatch	74
RegexReplace	74
RegexSearch	74
RegistryCopyKey	206
RegistryCreateKey	206
RegistryDeleteKey	206
RegistryDeleteValue	206
RegistryExist	206
RegistryFirstSubKey	206
RegistryListKey	206
RegistryListValue	206
RegistryNextKey	206
RegistryQueryValue	206

RegistryRename	206
RegistrySeek	206
RegistrySetValue	206
RegistryValueType	206
RepeatString	74
Replace	74
ReplicDeleteSubscription	181
ReplicInfo	181
ReplicOpen	181
ReplicSynchronize	181
RequestRefreshUI	12
RequestRefreshUIElement	12
RequestRefreshUIParent	12
Restore	12
RESTSend	103
Reverse	74
RGB	143
RGBBlue	143
RGBGreen	143
RGBRed	143
RGBToHTML	143
RibbonAdaptSize	38
RibbonCloseGrouping	38
RibbonEndExpandedMode	38
RibbonHighlightPane	38
Right	74
Root	201
Round	201
RoundCeil	201
RoundFloor	201
RoundToMultiple	201
rssBuildString	129
rssDisplay	129
rssInitialize	129
rssSave	129
RTFAdd	68
RTFAddLink	68
RTFInsert	68
RTFLoad	68
RTFReplace	68
RTFSearch	68
RTFSelection	68
RTFToHTML	68
RTFToText	68
RunReportsAndQueries	237

S

SaaSAdminAddClient	254
SaaSAdminAddPricing	254



SaaSAdminAddService	254	ScanSelect	146	SemaphoreCreate	223	SFUndelete	186
SaaSAdminAddSite	254	ScanToImage	146	SemaphoreDestroy	223	Sin	201
SaaSAdminAddSubscription	254	ScanToPDF	146	SemaphoreEnd	223	sInEntryQueue	109
SaaSAdminAddUser	254	ScanWithUserInterface	146	SemaphoreStart	223	sInExitQueue	109
SaaSAdminClientConnection	254	SchedulerAddAppointment	43	SemiDynamicPageDisplay	17	SIPVisible	134
SaaSAdminConfigureSMTP	254	SchedulerAddResource	43	sEndEvent	109	SiteAddress	18
SaaSAdminConnect	254	SchedulerAddTask	215	SendKey	218	sListPort	109
SaaSAdminDeleteClient	254	SchedulerAddTrigger	215	SendMessage	216	SMSDelete	134
SaaSAdminDeletePricing	254	SchedulerBackgroundColor	43	SensorDetectBeginShake	136	SMSFirst	134
SaaSAdminDeleteService	255	SchedulerChangeMode	43	SensorDetectEndShake	136	SMSNbMessage	134
SaaSAdminDeleteSite	255	SchedulerDateVisible	43	SensorEnd	136	SMSNext	134
SaaSAdminDeleteUser	255	SchedulerDeleteAll	43	SensorGetOrientation	136	SMSReset	134
SaaSAdminDisconnect	255	SchedulerDeleteResource	43	Serialize	185	SMSRunApp	135
SaaSAdminFindClient	255	SchedulerDeleteTask	215	ServiceContinue	227	SMSSend	135
SaaSAdminFindPricing	255	SchedulerDeleteTrigger	215	ServiceControl	227	SNMPCloseSession	114
SaaSAdminFindService	255	SchedulerDisplay	43	ServiceExist	227	SNMPGet	114
SaaSAdminFindSite	255	SchedulerDisplayNextPeriod	43	ServiceInfo	227	SNMPGetNext	114
SaaSAdminFindSubscription	255	SchedulerInfoXY	44	ServiceInstall	228	SNMPGetTable	114
SaaSAdminFindUser	255	SchedulerListAppointment	44	ServiceList	228	SNMPLoadMIB	114
SaaSAdminIsConnected	255	SchedulerMode	44	ServiceModify	228	SNMPOIDAccess	114
SaaSAdminListClient	255	SchedulerModifyTask	215	ServicePause	228	SNMPOIDDescription	114
SaaSAdminListPricing	255	SchedulerModifyTrigger	215	ServiceRefresh	228	SNMPOIDStatus	114
SaaSAdminListService	255	SchedulerPosition	44	ServiceRestart	228	SNMPOIDToString	114
SaaSAdminListSite	255	SchedulerPositionDateTime	44	ServiceStart	228	SNMPOIDType	114
SaaSAdminListSubscription	255	SchedulerPositionResource	44	ServiceStatus	228	SNMPSet	115
SaaSAdminListUser	255	SchedulerReset	215	ServiceStop	228	SNMPStartSession	115
SaaSAdminListVisit	255	SchedulerSelectedRange	44	ServiceUninstall	228	SNMPStringToOID	115
SaaSAdminModifyClient	255	SchedulerSelectMinus	44	ServiceWait	228	SNMPTrapDisable	115
SaaSAdminModifyPricing	255	SchedulerSelectPlus	44	ServiceWaitStatus	228	SNMPTrapEnable	115
SaaSAdminModifyService	256	SchedulerTaskList	215	ServiceWriteEventLog	228	SOAPAddAttribute	106
SaaSAdminModifySite	256	SchedulerTaskProperties	215	sEscape	109	SOAPAddHeader	106
SaaSAdminModifyUser	256	SchedulerTaskStatus	215	SessionHeartBeatEnable	231	SOAPAddSAMLAssertion	106
SaaSAdminSiteConnection	256	SchedulerTimeDisplayed	44	SessionPrologue	231	SOAPAddXMLSignature	106
SaaSChangePassword	256	SchedulerToiCalendar	44	sEvent	109	SOAPEncodeResult	106
SaaSCheckService	256	SchedulerTriggerProperties	215	SexagesimalToDecimal	201	SOAPError	106
SaaSClientConnexion	256	SchedulerWorkingHour	44	SFAdd	185	SOAPGetHeader	106
SaaSConnect	256	sClose	109	SFConnect	185	SOAPGetResult	106
SaaSConnectedUser	256	sComputeCrc16	218	SFConvertLead	185	SOAPPPrepare	106
SaaSDisconnect	256	sComputeCrc32	218	SFDelete	185	SOAPRun	106
SaaSIsConnected	256	SCPGetFile	104	SFDisconnect	185	SOAPRunXML	106
SaaSReadSiteParameter	256	SCPSendFile	104	SFEmptyRecycleBin	185	SOAPTtoHTTP	107
SaaSSendSecurityCode	257	ScreenSize	134	SFExecuteProcess	185	SocketAccept	111
SaaSWriteSiteParameter	257	ScreenToFile	167	SFExecuteQuery	186	SocketClientInfo	111
SAPConnect	185	ScreenToSource	168	SFForwardQuery	186	SocketClose	111
SAPDisconnect	185	ScreenType	134	SFMerge	186	SocketConfigure	111
SAPExecute	185	ScriptDisplay	231	SFModify	186	SocketConnect	111
ScanCapability	146	ScriptExecute	231	SFRead	186	SocketConnectBluetooth	111
ScanGet	146	Seek	86	SFSearch	186	SocketConnectInfrared	111
ScanList	146	SeekByProcedure	86	SFSeekDeleted	186	SocketConnectSSL	111
ScanProperty	146	SelectColor	237	SFSeekModified	186	SocketCreate	111

SocketCreateBluetooth	111	SpreadsheetListComment	50	SSHDisconnectShell	115	Sum	86
SocketCreateInfrared	111	SpreadsheetListImage	50	SSHRead	115	SwapLine	86
SocketCreateSSL	111	SpreadsheetListName	50	SSHWrite	115	sWrite	109
SocketCreateUDP	111	SpreadsheetListWorksheet	50	SSLActive	231	SysAndroidVersion	218
SocketDeny	111	SpreadsheetLoad	50	StackInfo	93	SysBatteryOptimBackground	218
SocketExist	111	SpreadsheetMergeCell	50	StatAverage	203	SysCacheExternalStorage	218
SocketProxy	111	SpreadsheetModifyName	50	StatAverageDeviation	203	SysChangeEnvironment	218
SocketRead	111	SpreadsheetMoveWorksheet	50	StatCorrelation	203	SysChangeScreenResolution	219
SocketWaitForConnection	111	SpreadsheetPosition	50	StatCovariance	203	SysColor	219
SocketWrite	111	SpreadsheetRecalculateAll	50	StatError	203	SysColorRes	219
sOpen	109	SpreadsheetReplace	50	StatMax	203	SysDateTime	81
Sort	86	SpreadsheetSave	51	StatMin	203	SysDetectLockedSession	219
Sound	237	SpreadsheetSeek	51	StatStandardDeviation	203	SysDeviceInfo	219
SourceToPage	168	SpreadsheetSelectComment	51	StatStandardDeviationP	203	SysDir	219
SourceToScreen	168	SpreadsheetSelectMinus	51	StatSum	203	SysDirExternalStorage	219
sParameter	109	SpreadsheetSelectPlus	51	StatusBarAddCell	20	SysDirStorageCard	219
SpeechRecognitionTrigger	251	SpreadsheetSortSelection	51	StatusBarDeleteCell	20	SysEnvironment	219
SpeechSynthesisInitialize	251	SpreadsheetStyleSelection	51	StatVariance	203	SysErrorMode	219
SpeechSynthesisInProgress	251	SpreadsheetUnmergeCell	51	StatVarianceP	203	SysGetDC	219
SpeechSynthesisListVoices	251	SQLAssociate	177	StreamAccept	118	SysGetExtLink	219
SpeechSynthesisReadFile	251	SQLChangeConnection	177	StreamChangeDriver	118	SysIconAdd	219
SpeechSynthesisReadText	251	SQLClose	177	StreamConfigure	118	SysIconAddImage	219
SpeechSynthesisStop	251	SQLCol	177	StreamConnect	118	SysIconBadge	219
SpellAddDictionary	252	SQLColumn	177	StreamDisconnect	118	SysIconDelete	219
SpellAvailable	252	SQLColumnFormat	177	StreamInfo	118	SysIconDeleteImage	219
SpellCheck	252	SQLConnect	177	StreamPlay	118	SysIconDisplayTooltip	219
SpellSuggest	252	SQLConnectWS	177	StreamReject	118	SysIconize	219
SpreadsheetAddData	48	SQLDisconnect	177	StreamStop	118	SysIconModify	219
SpreadsheetAddFormula	48	SQLExec	177	StringBuild	74	SysIdentifier	220
SpreadsheetAddImage	48	SQLExecWDR	177	StringCompare	74	SysIMEI	220
SpreadsheetAddName	49	SQLFetch	177	StringCount	74	SysInstance	220
SpreadsheetAddWorksheet	49	SQLFirst	177	StringDelete	74	SysIOSVersion	220
SpreadsheetAllowProcedure	49	SQLGetCol	177	StringDeleteDuplicate	74	SysLinkExtension	220
SpreadsheetBorderSelection	49	SQLGetMemo	177	StringDisplay	231	SysListScreen	220
SpreadsheetClose	49	SQLGetTextMemo	177	StringEndsWith	74	SysListScreenResolution	220
SpreadsheetControlFormula	49	SQLInfo	178	StringFormat	74	SysListVideoCard	220
SpreadsheetDeleteAll	49	SQLLast	178	StringIncrement	75	SysMemoryExternalStorage	220
SpreadsheetDeleteColumn	49	SQLListSource	178	StringInsert	75	SysMetric	220
SpreadsheetDeleteImage	49	SQLListTable	178	StringRetrieve	75	SysNameExe	220
SpreadsheetDeleteName	49	SQLLock	178	StringReverse	75	SysNameMainScreen	220
SpreadsheetDeleteRow	49	SQLModify	178	StringSplit	75	SysNation	220
SpreadsheetEndModification	49	SQLNext	178	StringStartsWith	75	SysNbExternalStorage	220
SpreadsheetFreeze	49	SQLPrevious	178	StringToArray	86	SysNotificationClear	216
SpreadsheetGetFormula	49	SQLReqExists	178	StringToDate	81	SysNotificationCompatible	216
SpreadsheetGetMerge	49	SQLSeek	178	StringToDuration	81	SysNotificationRemove	216
SpreadsheetGetSelection	49	SQLTable	178	StringToJSON	75	SysNotificationRemoveID	217
SpreadsheetGetSortInfo	49	SQLTransaction	178	StringToNum	75	SysNotificationSend	217
SpreadsheetInputFunction	50	sRead	109	StringToTime	81	SysNumberScreen	220
SpreadsheetInsertColumn	50	SSHCommand	115	StringToUTF8	75	SysPowerStatus	220
SpreadsheetInsertRow	50	SSHConnectShell	115	StyleDrawCaptionGrayed	237	SysRecentDocAdd	220



SysRecentDocList	220	TableDeleteChild	31	TableSwapBranch	29	TaskPrevious	126
SysReleaseDC	220	TableDeleteSelect	25, 31	TableSwapLine	28, 29	TaskRead	126
SysScreenOrientation	221	TableDeleteWithAnimation	25	TableToClipboard	28, 31	TaskReset	126
SysScreenRectangle	221	TableDisableFilter	25	TableToExcel	28, 31	TelemetryAddMeasure	243
SysScreenResolution	221	TableDisplay	25, 28	TableToText	28, 31	TelemetryAddPassage	243
SysSerialNum	221	TableEnableFilter	25	TableToWord	28, 31	TelemetryConfigure	243
SysSetFocus	221	TableEnd	25	TableToXML	28, 31	TelemetryDisable	243
SysShutdown	221	TableEndFilling	25	TableTypeItem	31	TelemetryEnable	243
SysSpace	221	TableEnumBreak	25	TabLoadConfiguration	38	TelemetryIdentifyUser	243
SysStandby	221	TableEnumColumn	25, 29	TabOpen	38	TelemetryIsEnabled	243
SysStatusExternalStorage	221	TableExpandAll	29	TabSaveConfiguration	38	TelemetrySendData	243
SysStatusStorageCard	221	TableFilteredColumn	25, 29	TabStatus	38	TestAddIteration	257
SystemIntegerToInteger	75	TableFixColumn	25, 29	Tangent	201	TestCheck	257
SysTheme	221	TableFormulaAdd	25	tapiAnswerCall	115	TestCheckDifferent	257
SysThumbnailAddButton	221	TableFormulaDeleteAll	25	tapiBlindTransfer	115	TestCheckEqual	257
SysThumbnailDeleteAll	221	TableGiveChild	30	tapiCallDuring	116	TestCheckFalse	257
SysThumbnailDeleteButton	221	TableGiveParent	30	tapiCalledID	116	TestCheckNotNull	257
SysThumbnailModifyButton	221	TableInfoXY	26, 30	tapiCallEnd	116	TestCheckNull	257
SysUnlinkExtension	221	TableInputInProgress	26, 30	tapiCallerID	116	TestCheckTrue	257
SysValidHandle	221	TableInputSearch	26	tapiCallsBusy	116	TestError	258
SysVersion	222	TableInsert	26, 30	tapiCallsOver	116	TestOnError	258
SysWinActive	222	TableInsertChild	30	tapiCallsWaiting	116	TestWriteResult	258
SysWindowsVersion	222	TableInsertColumn	26	tapiCallStart	116	TextToHTML	75
SysWinHandle	222	TableInsertLine	26, 30	tapiCapability	116	TextToRTF	75
SysWinShow	222	TableItemStatus	29	tapiCompleteTransfer	116	TextToXML	118
SysWinTitle	222	TableListChild	30	tapiDevice	116	ThreadCurrent	223
SysXRes	222	TableModify	26, 30	tapiDeviceList	116	ThreadEnd	223
SysYRes	222	TableModifyColumn	26	tapiDial	116	ThreadExecute	223
<b>T</b>		TableModifyLine	26, 30	tapiDialerCall	134	ThreadMode	223
		TableMoveBranch	29	tapiDialerDisplay	134	ThreadPause	223
TabClose	38	TableMoveColumn	26, 29	tapiError	116	ThreadPersistent	223
TabCloseAll	38	TableMoveLine	26	tapiHangUp	116	ThreadPriority	223
TableAdd	24, 28	TablePosition	26, 30	tapiHold	116	ThreadRequestStop	223
TableAddChild	28	TablePrint	26	tapiKeyPressed	116	ThreadSendSignal	223
TableAddColumn	24	TableRecNum	27	tapiLineDial	116	ThreadState	223
TableAddInProgress	24	TableRefreshVisible	27, 30	tapiListen	116	ThreadStopRequested	223
TableAddLine	24, 28	TableSave	27	tapiNoAnswer	116	ThreadWait	223
TableAddLineWithAnimation	24	TableSearch	27	tapiOrigin	117	ThreadWaitSignal	223
TableAddTitleHeader	24	TableSearchChild	28	tapiPlay	117	TimeDifference	81
TableAdjust	24, 28	TableSearchEverywhere	27	tapiRecord	117	TimelineAddEvent	45
TableBreakSubscript	24	TableSelect	27, 31	tapiSendKey	117	TimelineAddTrack	45
TableChildCount	29	TableSelectCount	27, 31	tapiStop	117	TimelineChangeMode	45
TableCollapse	29	TableSelectMinus	27, 31	tapiStopCallDetection	117	TimelineDeleteAll	45
TableCollapseAll	29	TableSelectPlus	27, 31	tapiUnhold	117	TimelineDeleteEvent	45
TableCollapseExpand	24	TableSelectToFile	27	TaskAdd	125	TimelineDeleteTrack	45
TableColumnSubscript	24, 29	TableSetFocus	27	TaskDelete	126	TimelineDisplay	45
TableCount	24, 30	TableSort	27	TaskFirst	126	TimelineListEvent	45
TableDelete	25, 31	TableSortChild	31	TaskLast	126	TimePicker	81
TableDeleteAll	25, 31	TableSortedColumn	27	TaskModify	126	Timer	216
		TableStartFilling	28	TaskNext	126	TimerSys	216

TimeSys	81	TreeMoveItem	37
TimeToInteger	81	TreePosition	37
TimeToString	81	TreeSelect	37
TimeValid	81	TreeSelectCount	37
ToastDeleteAll	21	TreeSelectMinus	37
ToastDisplay	21	TreeSelectPlus	37
ToastDisplayIW	21	TreeSort	37
ToastDisplayPopup	21	TreeStatus	37
ToClipboard	213	TreeTypeItem	37
ToClipboardRTF	213	TrialVersion	237
Today	81	Truncate	75
Tomorrow	81	TrustedCertificateAdd	205
ToolbarDock	38	TrustedCertificateReset	205
ToolbarPosition	38	TwainCurrentSource	144
ToolbarUndock	39	TwainListSource	144
ToolTipMargin	237	TwainProperty	144
Trace	237	TwainScanArea	145
TraceBuild	237	TwainSelectSource	145
TraceDeleteAll	237	TwainSourceHeight	145
TraceEnd	237	TwainSourceWidth	145
TraceStart	237	TwainState	145
Transfer	218	TwainToBMP	145
TreeAdd	36	TwainToClipboard	145
TreeCollapse	36	TwainToControl	145
TreeCollapseAll	36	TwainToGIF	145
TreeCopyItem	36	TwainToJPEG	145
TreeCount	36	TwainToPDF	146
TreeDelete	36	TwainToTIFF	146
TreeDeleteAll	36	TypeVar	237
TreeDeleteChild	36		
TreeDisplay	36		
TreeExpand	36		
TreeExpandAll	36	UMCActive	257
TreeFind	37	UMCOpen	257
TreeGetSubscript	37	UncompleteDir	75
TreeGiveChild	37	Uncompress	77
TreeID	37	UnicodeToAnsi	75
TreeIdentifier	37	UnloadWDL	237
TreeInfoXY	37	UploadCopyFile	252
TreeInsert	37	UploadCurrentFile	252
TreeListItem	37	UploadCurrentFileSize	252
TreeMapAdd	55	UploadCurrentFileSizeSent	252
TreeMapCount	55	UploadDelete	252
TreeMapDelete	56	UploadDeleteAll	252
TreeMapDeleteAll	56	UploadFileName	252
TreeMapInfoXY	56	UploadFileSize	253
TreeMapSearch	56	UploadSize	253
TreeMapSeek	56	UploadSizeSent	253
TreeMapSelectGroup	56	UploadStart	253
TreeModify	37	UPNPEecuteAction	107

## U

UPNPFindDevice	107	VideoListParameter	152
UPNPGetBrightness	107	VideoParameter	152
UPNPGetContrast	107	VideoRequestPermission	152
UPNPGetMute	107	VideoSetDriver	152
UPNPGetSharpness	107	VideoStartApp	152
UPNPGetVolume	107	VideoStartDecoding	152
UPNPListDevice	107	VideoStatus	152
UPNPMediaOpen	107	VideoStop	152
UPNPMediaPause	107	VideoStopDecoding	152
UPNPMediaPlay	107	VolumeGet	134
UPNPMediaStop	107	VolumeModify	134
UPNPModifyBrightness	108	VolumePhysicalButton	134
UPNPModifyContrast	108		
UPNPModifyMute	108		
UPNPModifySharpness	108		
UPNPModifyVolume	108		
Upper	75	Wait	216
URIGetInfo	253	WaitWindowEnd	13
URISelect	253	WaitWindowStart	13
URIToBuffer	253	Warning	20
URIToFile	253	WarningAsynchronous	20
URLDecode	75	WebserviceClientIPAddress	259
URLEncode	76	WebserviceParameter	259
URLExtractPath	76	WebserviceWriteHTTPCode	260
USBDDisconnect	110	WebserviceWriteMIMETYPE	260
USBFind	110	WebSocketClientConnect	112
USBListDevice	110	WebSocketDisconnect	112
USBProperty	110	WebSocketExecute	112
Use	13	WebSocketSend	112
UTF8ToAnsi	76	WeekNumber	81
UTF8ToString	76	WeekToDate	81
UTF8ToUnicode	76	WHelp	237
UTFConvert	76	WiFiActivate	97
UUDecode	77	WiFiAddNetwork	97
UUEncode	77	WiFiConnect	97
		WiFiConnectionInfo	97
		WiFiDeleteNetwork	97
		WiFiDetectAccessPoint	97
		WiFiListNetwork	97
		WiFiListNetworkCard	98
Val	76	WiFiSelectNetworkCard	98
ValidatePassword	254	WiFiStatus	98
VariableReset	237	WinActivateDDW	13
VariantConvert	237	WinAdaptHeight	13
VariantToJSON	76	WinAdaptSize	13
VibrationStop	134	WinAdaptWidth	13
VibrationTrigger	134	WinAnimationClosing	70
VideoCapture	152	WinAnimationNext	70
VideoConfigure	152	WinAnimationOpening	70
VideoDisplay	152	WinBackgroundImage	13
VideoGenerateThumbnail	152	WinChangeAlias	13
VideoListDriver	152		

**W**

**V**

WinChangeLayout	13	WordToArray	76
WinCopyForm	13	<b>X</b>	
WinCurrentLayout	13		
WINDEVVersion	238		
WinDisableEffect	13		
WindowBitmap	13	XamlDisable	62
WindowCount	238	XamlEvent	62
WindowsVersion	218	XamlProperty	62
WinEdActive	248	xlsAddWorksheet	209
WinEdCreateNote	248	xlsClose	209
WinEdDeclareUndo	248	xlsColumnName	209
WinEdDeselectAll	248	xlsColumnType	209
WinEdDo	248	xlsCoordinatesToName	209
WinEdLoad	248	xlsCurrentWorksheet	209
WinEdRedo	249	xlsData	209
WinEdRestore	249	xlsDeleteWorksheet	209
WinEdSave	249	xlsMsgError	209
WinEdSaveNote	249	xlsNameToCoordinates	209
WinEdSelect	249	xlsNbColumn	209
WinEdSelectMinus	249	xlsNbRow	209
WinEdSelectOccurrence	249	xlsNbWorksheet	209
WinEdSelectPlus	249	xlsOpen	209
WinEdUndo	249	xlsSave	209
WinForceDDW	13	xlsVersion	209
WinGiveSuitableHeight	13	XMLAddAttribute	118
WinGiveSuitableWidth	13	XMLAddChild	118
WinIconBarHeight	13	XMLBuildString	118
WinInactiveEffect	13	XMLCancelSearch	118
WinInHeight	14	XMLChild	118
WinInWidth	14	XMLChildExist	118
WinInitialized	14	XMLClose	118
WinInput	14	XMLData	119
WinInWidth	14	XMLDelete	119
WinInXPos	14	XMLDocument	119
WinInYPos	14	XMLElementName	119
WinOrientation	14	XMLElementType	119
WinOutHeight	14	XMLExecuteXPath	119
WinOutWidth	14	XMLExtractDocument	119
WinOutXPos	14	XMLExtractString	119
WinOutYPos	14	XMLFind	119
WinPasteForm	14	XMLFindNamespaceByURI	119
WinRateDDW	14	XMLFirst	119
WinRedraw	14	XMLFound	119
WinRefreshVisible	14	XMLInsertDocument	119
WinScreenRectangle	14	XMLInsertElement	119
WinSize	14	XMLLast	119
WinSlidingVisible	15	XMLModify	119
WinStatus	15	XMLNamespace	119
WinUsefulSize	41	XMLNamespaceURI	119
WithSpace	166	XMLNext	120
WordCount	76	XMLOpen	120

XMLOpenReader	120	zipIsMulti	208
XMLOut	120	zipListFile	208
XMLParent	120	zipMerge	208
XMLParentName	120	zipMsgError	208
XMLPath	120	zipNbFile	208
XMLPrevious	120	zipNbPart	208
XMLRead	120	zipNbPartNeeded	208
XMLRename	120	zipOpen	208
XMLRestorePosition	120	zipOpenCAB	208
XMLResult	120	zipOpenRAR	208
XMLRoot	120	zipPassword	208
XMLSave	120	zipSelectFile	208
XMLSavePosition	120	zipSize	208
XMLSign	120	zipSplit	208
XMLToJSON	120		
XMLToText	120		
XMLValidDocument	120		
XMLValidNode	120		
XMLWrite	121		

Y

YesNo	20
YesNoAsynchronous	21
Yesterday	81

Z

zipAddDirectory	206
zipAddFile	206
zipAddFileList	206
zipChangePath	206
zipClose	207
zipCompressionLevel	207
zipCreate	207
zipCreateExe	207
zipCurrentFile	207
zipDeleteAll	207
zipDeleteFile	207
zipDeleteFileList	207
zipExist	207
zipExtractAll	207
zipExtractDirectory	207
zipExtractFile	207
zipExtractFileList	207
zipExtractPath	207
zipFileSize	207
zipFilter	207
zipFindFile	207
zipInfoFile	208

