



zmw[Self]Control Version 1.1

Helps you to build Microsoft Windows impressive features in your application.

Let's install zmw[Self]Control Plugin.

Step 1- Open VisualNEO-Win after placing zmwSelfControl into your plugins folder.

Step 2- Click on Options Menu and select Install-Plugins.

Step 3- In the Plug-Ins panel, click on install button and select zmw[Self]Control.nbp

Step 4- If zmw[Self]Control can be seen in installed Plug-Ins list it means zmw[Self]Control has been installed

Registering a zmw[Self]Control Plugin.

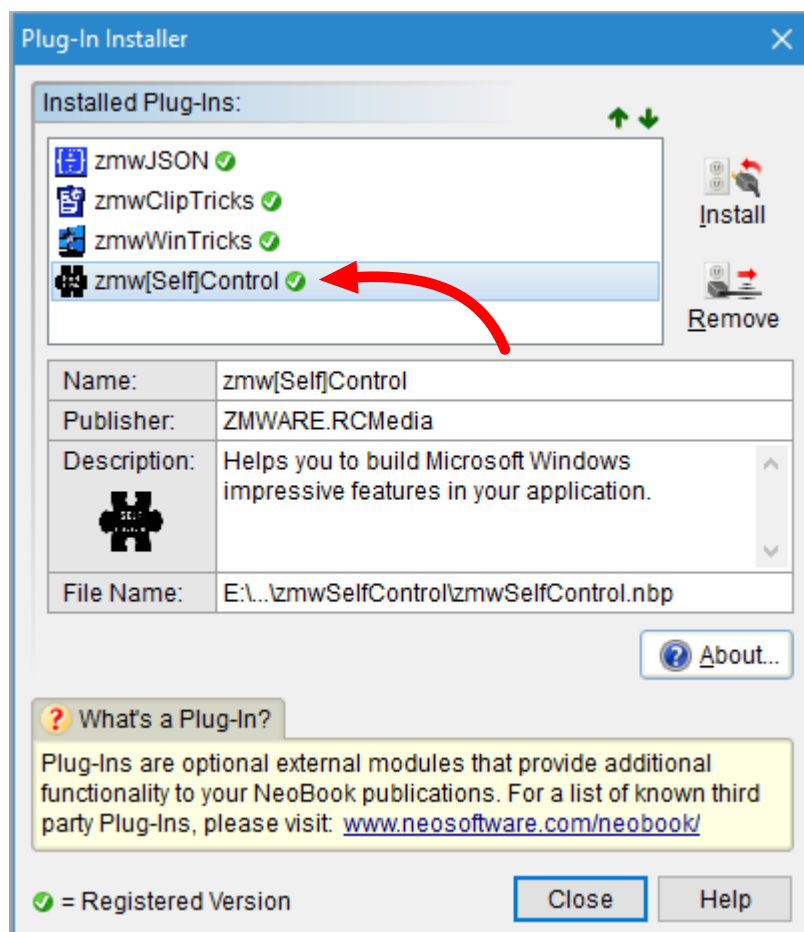
Step 1- In Plug-Ins Installer window. Select zmw[Self]Control from Installed Plug-Ins list

Step 2- Click on Register this Plug-In button and the Registration Dialog will appear which contains Registration ID.

Step 3- Send us this Registration ID with Plugin Info to contact@zmware.rcmediaservices.net.

Step 4- You will receive an email containing your Unlock Code within two business days. Please know that this is a manual process, so please be patient

Step 5- Type your Unlock Code into the Registration Dialog's Unlock Code field and click the Register button. A green checkmark will display indicating that you are registered.



Action Overview:

zmwAddOpenWith:

Purpose: Adds your defined application in Open With Dialog.

Syntax: `zmwAddOpenWith "Extension" "Application" "Description" "Icon" "Tag Name"`

Extension: The file extension (i.e. txt) of which OpenWithDialog, you will add your app in

Application: The application you want to add in OpenWithDialog (Leave empty for your application).

Description: Description is the type of file in file properties, It replaces the type of file after your added app is set to default.

Icon: Changes the icon to your given icon after your added app is set as default.

Tag Name: Tag name is the identifier (Assigned by you) of defined application.

zmwRemoveOpenWith:

Purpose: Removes application by it's tag name from Open With Dialog.

Syntax: `zmwRemoveOpenWith "Extension" "Tag Name"`

Extension: The file extension (i.e. txt) of which OpenWithDialog, you want to remove your app from

Tag Name: The tag name of your added application you want to remove from OpenWithDialog,

zmwFileAssociate:

Purpose: Sets your defined app that can act as the default for that file type.

Syntax: `zmwFileAssociate "File extension" "Associate with" "Description" "Icon"`

File Extension: Extension you want to associate your application with.

Associate With: The Application you want to associate with your defined extension (Leave empty for your application).

Description: The description of file type, Appears in the file properties.

Icon: The icon of file type, Icon will be changed to your given icon (Leave empty for your application icon).

zmwFileDisassociate:

Purpose: Disassociates file type.

Note: Some extensions are used by windows. So in that case Window default program(For that extension) will automatically be associated.

Syntax: `zmwFileDisassociate "File extension"`

File Extension: Extension you want to disassociate.

zmwCreateShortcut:

Purpose: Creates a link at various directories to points to your defined file.

Syntax: `zmwCreateShortcut "File" "Location"`

File: Create shortcut of file.

Location: Directory in which you want to create shortcut.

zmwAddContextMenu:

Purpose: Adds menu item in file explorer context menu also allows to add submenu in that created menu.

Note: Use [CommandLine] variable to parse file name and menu text.

Syntax: `zmwAddContextMenu "Extension" "Application to attach" "Menu Text" "Menu Icon" "Menu Name" "Add Sub Menu" "Sub Menus"`

Extension: Extension (i.e. txt) on which you want this menu to appear (Use * to appear on every extension)

Application to Attach: Application you want to attach to the menu (Leave empty for your application).

Menu Text: Text of the menu.

Menu Icon: Icon of the menu (Leave empty for your application icon).

Menu Name: Name of the menu.

Add Sub Menu: Allows to create sub menu (Boolean).

Sub Menus: List of sub menus you want to add (Note: Must be edited using wizard).

zmwRemoveContextMenu:

Purpose: Removes menu item with submenus from file explorer context menu.

Syntax: `zmwRemoveContextMenu "Extension" "Menu name" "Remove Sub Menu" "Sub menus names"`

Extension: Extension (i.e. txt) from which you want this menu to be removed (Use * to remove from every extension (Only for menus created with *)).

Menu Name: Name of the menu to remove.

Remove Sub Menus only: Allows you to remove only sub menus (Boolean).

Sub Menus Name: Name of the sub menus you want to remove.

zmwRunAtWindowsStartup:

Purpose: Runs defined application at windows startup.

Syntax: `zmwRunAtWindowsStartup "Applicaton" "State" "Variable"`

Application: Application to run at windows startup.

State: Change the state of the app that runs at the startup. (0= Enable, 1=Disable, 2=None).

Note: "None" will not change the state it simply stores the state into your variable.

Variable: Variable name to store the state.

zmwRunDosCommand:

Purpose: Executes DOS command and stores the command result into defined variable.

Syntax: `zmwRunDosCommand "Command" "Work directory" "Variable"`

Command: Command to be executed through DOS.

Work Directory: Sets DOS working directory.

Variable: Stores the command's result.

zmwShellExecute:

Purpose: Runs defined application as administrative rights.

Syntax: `zmwShellExecute "Application to Execute" "Parameters" "Run As Admin" "Hidden"`

Application to Execute: Application to run.

Parameters: Optional items to be passed to the application (a file name, command switches, etc.) options

Run As Admin: Run Application as Administrative Rights (Boolean)

Hidden: Hide the application's main window. Some types of applications will not allow their main window to be hidden and will ignore this option. Hidden works well when running most DOS command line utilities, batch files, etc. that do not require interaction from the user. (Boolean)

zmwGetHostFileLocation:

Purpose: Stores host file location and name into defined variable.

Syntax: `zmwGetHostFileLocation "Variable" "Open Host File in Notepad"`

Variable: Stores the host file location with name.

Open Host File in Notepad: Check to open host file in a notepad (Boolean)

zmwGetPCInfo:

Purpose: Gets Operating System, PC, BIOS and Network information and stores into defined variable.

Syntax: `zmwGetPCInfo "System Info" "Size" "Variable"`

System Info: Type of system information to obtain. Select from combo box.

Size: Gets size in selected data storage unit (KB, MB, GB).

Note: Only for items contains memory.

Variable: Stores the obtained system info.

zmwFlashApp:

Purpose: Flashes your application (one time).

Syntax: `zmwFlashApp`

This command has no parameters.

Technical Support

This help file contains just about everything you will need to know in order to use zmw[Self]Control Plugin effectively. If you should encounter a technical problem or question not covered here, you may send us email at support@zmware.rcmediaservices.net

When contacting technical support, please include your VisualNeo Win version number, Windows version, zmw[Self]Control Plugin version number and a detailed description of the problem and how to reproduce it.

**We intend to add features to this project.
If you have suggestions, please send them our way!**

Thank you for your support.

zmware.rcmediaservices.net