

InternetExplorer object

63 out of 153 rated this helpful

Controls an instance of Windows Internet Explorer through automation.

Members

The **InternetExplorer** object has these types of members:

- [Events](#)
- [Methods](#)
- [Properties](#)

Events

The **InternetExplorer** object has these events.

Event	Description
BeforeNavigate	Fires before navigation occurs in the given object (on either a window or frameset element).
CommandStateChange	Fires when the enabled state of a command changes.
DocumentComplete	Fires when a document is completely loaded and initialized.
DownloadBegin	Fires when a navigation operation begins.
DownloadComplete	Fires when a navigation operation finishes, is halted, or fails.
NavigateComplete	Fires after a navigation to a link is completed on either a window element or a frameSet element.
NavigateComplete2	Fires after a navigation to a link is completed on a window element or a frameSet element.
NewProcess	Creates a new process to handle the navigation.
NewWindow	Fires when a new window is to be created.
NewWindow2	Fires when a new window is to be created.
NewWindow3	Raised when a new window is to be created. Extends NewWindow2 with additional information about the new window.
OnFullScreen	Fires when the FullScreen property is changed.
OnMenuBar	Fires when the MenuBar property is changed.
OnQuit	Fires before the Internet Explorer application quits.
OnStatusBar	Fires when the StatusBar property is changed.
OnTheaterMode	Fires when the TheaterMode property is changed.

OnToolBar	Fires when the ToolBar property is changed.
OnVisible	Fires when the Visible property of the object is changed.
PrintTemplateInstantiation	Fires when a print template is instantiated.
PrintTemplateTeardown	Fires when a print template is destroyed.
ProgressChange	Fires when the progress of a download operation is updated on the object.
PropertyChange	Fires when the PutProperty method of the object changes the value of a property.
StatusTextChange	Fires when the status bar text of the object has changed.
TitleChange	Fires when the title of a document in the object becomes available or changes.
UpdatePageStatus	
WindowActivate	
WindowMove	
WindowResize	
WindowStateChanged	Fires when the visibility state of a content window, such as the browser window or a tab, changes.

Methods

The **InternetExplorer** object has these methods.

Method	Description
ExecWB	Executes a command and returns the status of the command execution using the IOleCommandTarget interface.
GetProperty	Gets the value associated with a user-defined property name.
GoBack	Navigates backward one item in the history list.
GoForward	Navigates forward one item in the history list.
GoHome	Navigates to the current home or start page.
GoSearch	Navigates to the current search page.
Navigate	Navigates to a resource identified by a URL or to a file identified by a full path.
Navigate2	Navigates the browser to a location that might not be expressed as a URL, such as a PIDL for an entity in the Windows Shell namespace.
PutProperty	Associates a user-defined name/value pair with the object.
QueryStatusWB	Queries the object for the status of commands using the IOleCommandTarget interface.
Refresh	Reloads the file that is currently displayed in the object.

Refresh2	Reloads the file that is currently displayed with the specified refresh level.
Stop	Cancels a pending navigation or download, and stops dynamic page elements, such as background sounds and animations.

Properties

The **InternetExplorer** object has these properties.

Property	Description
AddressBar	Sets or gets a value that indicates whether the address bar of the object is visible or hidden.
Application	Gets the automation object for the application that is hosting the WebBrowser Control .
Busy	Gets a value that indicates whether the object is engaged in a navigation or downloading operation.
Container	Gets an object reference to a container.
Document	Gets the automation object of the active document, if any.
FullName	Note FullName may be altered or unavailable in subsequent versions of the operating system or product. Retrieves the fully qualified path of the Internet Explorer executable.
FullScreen	Sets or gets a value that indicates whether Internet Explorer is in full-screen mode or normal window mode.
Left	Sets or gets the coordinate of the left edge of the object.
LocationName	Retrieves the path or title of the resource that is currently displayed.
LocationURL	Gets the URL of the resource that is currently displayed.
MenuBar	Sets or gets a value that indicates whether the Internet Explorer menu bar is visible.
Offline	Sets or gets a value that indicates whether the object is operating in offline mode.
Parent	Gets the parent of the object.
Path	Note Path may be altered or unavailable in subsequent versions of the operating system or product. Retrieves the system folder of the Internet Explorer executable.
ReadyState	Gets the ready state of the object.
RegisterAsBrowser	Sets or gets a value that indicates whether the object is registered as a top-level browser window.
RegisterAsDropTarget	Sets or gets a value that indicates whether the object is registered as a drop target for navigation.
Silent	Sets or gets a value that indicates whether the object can display dialog boxes.
StatusBar	Sets or gets a value that indicates whether the status bar for the object is visible.

Statusbar	Sets or gets a value that indicates whether the status bar for the object is visible.
TheaterMode	Sets or gets whether the object is in theater mode.
ToolBar	Sets or gets whether toolbars for the object are visible.
TopLevelContainer	Gets a value that indicates whether the object is a top-level container.
Type	Gets the user type name of the contained document object.
Visible	Sets or gets a value that indicates whether the object is visible or hidden.

Remarks

Windows Internet Explorer 8. On Windows Vista, to create an instance of Internet Explorer running at a medium integrity level, pass `CLSID_InternetExplorerMedium` (defined in `exdisp.idl`) to **CoCreateInstance**. The resulting **InternetExplorerMedium** object supports the same events, methods, and properties as the **InternetExplorer** object.

Examples

The following example uses **CreateObject** in Visual Basic to launch an instance of Internet Explorer.

VB

```
Dim IE As SHDocVw.InternetExplorer

Set IE = CreateObject("InternetExplorer.Application")
```

The following C# example launches an instance of Internet Explorer and navigates to a Web page. The code also demonstrates how to create an event handler to listen for the **BeforeNavigate2** event. The project requires a reference to the Microsoft Internet Controls (SHDocVw) type library.

C#

```
using System;
using System.Collections.Generic;
using System.Text;

namespace ConsoleApplication1
{
    class EventHandlers
    {
        public void OnBeforeNavigate2(object sender, ref object URL,
                                     ref object Flags, ref object Target,
                                     ref object postData, ref object Headers,
                                     ref bool Cancel)
        {
            Console.WriteLine("BeforeNavigate2 fired!");
        }
    }

    class Program
    {
        static void Main(string[] args)
        {
            EventHandlers e = new EventHandlers();
            SHDocVw.InternetExplorer IE = new SHDocVw.InternetExplorer();
            object Empty = 0;
            object URL = "http://www.live.com";
```

```
// override BeforeNavigate2 event
IE.BeforeNavigate2 += new
    SHDocVw.DWebBrowserEvents2_BeforeNavigate2EventHandler(
        e.OnBeforeNavigate2);

IE.Visible = true;
IE.Navigate2(ref URL, ref Empty, ref Empty, ref Empty, ref Empty);

System.Threading.Thread.Sleep(5000);

IE.Quit();
    }
}
}
```

Community Additions

© 2014 Microsoft. All rights reserved.