CC BY-NC-SA 4.0

ECMA Script in Windows Script Host

Quick Reference

Key

Object

Collection

JScript/ WSH Specifics

Enumerator

Implement of For... next style loop over an object collection:

```
var en = new
Enumerator(sourceFolder.subFolders);
var folderLoop;
for (; !en.atEnd(); en.moveNext()) {
  folderLoop = en.item();
  //Do stuff with folderLoop
}
```

Referencing External Files

Refer to external EcmaScript files using an opening and closing script element NOT an empty script element.

```
// Do this
<script type="text/javascript"
src="C:/Data/Sda/dateTime.js"></script>

// Not this
<script type="text/javascript"
src="C:/Data/Sda/dateTime.js" />
```

Although either work in windows script hosts, in web browsers only the top one works. Using the top convention facilitates copying code between WSH scripts and Web pages.

Refer to external EcmaScript files either by using a forward slash (my preference) or double back slash.

```
// Prefered
<script type="text/javascript"
src="C:/Data/Sda/dateTime.js"></script>

// Alternative
<script type="text/javascript"
src="C:\\Data\\Sda\\dateTime.js"></script>
```

Bentley > Experiment

Path and File Name Terms

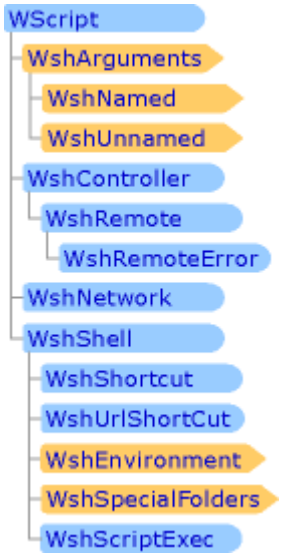
See Every Application > Method > Conventions > Path and File Name Terms.doc

Recursion

```
function ListFolderAndSub( sourceFolder ) {

   // Do Stuff in Folder

  var arrFolders = new Array();
  subFolders = new
Enumerator(sourceFolder.subFolders);
  var i = 0;
  for (; !subFolders.atEnd();
subFolders.moveNext()) {
    arrFolders[i++] = subFolders.item();
  }

  for (var j = 0; j < arrFolders.length;
j++) {
    // Don't copy over the xwmEngine itself
    if (arrFolders[j].Name != "xwmEngine") {
      ListFolderAndSub( arrFolders[j] )
    }
  } // For (j
  return true;
}
```

CC BY-NC-SA 4.0

WSH Object Model



.NET or COM?:
COM

Assembly or Library Name:
IwshRuntimeLibrary

File And Path Name:
C:\WINNT\System32\wshom.ocx

COM Library Long Name:
Windows Script Host Object Model.

WScript

Properties	Methods
Arguments BuildVersion FullName Interactive Name Path ScriptFullName ScriptName StdErr StdIn StdOut Version	ConnectObject CreateObject DisconnectObject Echo GetObject Quit Sleep

```
var ws = WScript;
ws.Echo("Hello World Again");
```

FullName: C:\WINNT\system32\cscript.exe
ScriptFullName: C:\Data\WshExplore.wsf
ScriptName: WshExplore.wsf

WshArguments

Properties	Methods
Item Length Named Unnamed	Count ShowUsage

```
var args = WScript.Arguments;
for (i = 0; i < args.Length; i++) {
  WScript.Echo("arg " + i + ": " + args(i));
}
```

WshNamed

Properties	Methods
Item Length	Count Exists

```
var namedArgs = WScript.Arguments.Named;
WScript.Echo("Named Arg /colour: " +
  namedArgs("colour"));
WScript.Echo("Named Arg /fullOutput Exist?: " +
  namedArgs.Exists("fullOutput"));
```

WshUnnamed

Properties	Methods
Item Length	Count

```
var unnamedArgs = WScript.Arguments.Unnamed;
for (i = 0; i < unnamedArgs.Length; i++) {
  WScript.Echo("Unnamed Arg " + i + ": " +
  unnamedArgs(i));
}
```

WshController

Properties	Methods
	CreateScript

WshRemote

Properties	Methods	Events
Error Status	Execute Terminate	End Error Start

WshRemoteError

Properties	Methods
Character Description Line Number SourceText Source	

WshNetwork

Properties	Methods
ComputerName UserDomain UserName	AddPrinterConnection AddWindowsPrinterConnection EnumNetworkDrives EnumPrinterConnections MapNetworkDrive RemoveNetworkDrive RemovePrinterConnection SetDefaultPrinter

WshShell

Properties	Methods
CurrentDirectory Environment SpecialFolders	AppActivate CreateShortcut Exec ExpandEnvironmentStrings LogEvent Popup RegDelete RegRead RegWrite Run SendKeys

```
var WshShell =
WScript.CreateObject("WScript.Shell");
```

WshShortcut

Properties	Methods
Arguments Description FullName Hotkey IconLocation RelativePath TargetPath WindowStyle WorkingDirectory	Save

```
// Where the shorcut is stored
var WshShortcut =
WshShell.CreateShortcut("C:\\Data\\John
Monkey.lnk");
// What is it a shorcut to? ...
WshShortcut.TargetPath = "C:\\Data\\Temp";
WshShortcut.Save();
```

WshUrlShortcut

Properties	Methods
FullName TargetPath	Save

WshEnvironment

Properties	Methods
Item Length	Count Remove

WshSpecialFolders

Properties	Methods
Item Length	Count
Special Folders	
AllUsersDesktop AllUsersStartMenu AllUsersPrograms AllUsersStartup Desktop Favorites Fonts MyDocuments NetHood PrintHood Programs Recent SendTo StartMenu Startup Templates	

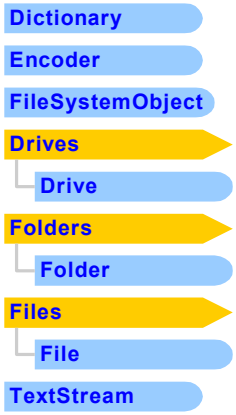
```
var WshSpecialFolder =
 WshShell.SpecialFolders("Programs");
WScript.Echo("Start Menu Programs Path is: "
 + WshSpecialFolder);
```

WshScriptExec

Properties	Methods
ExitCode ProcessID Status StdErr StdIn StdOut	Terminate

CC BY-NC-SA 4.0

Scripting Runtime



.NET or COM?:
COM

Assembly or Library Name:
Scripting

File And Path Name:
C:\WINNT\System32\scrrun.dll

COM Library Long Name:
Microsoft Scripting Runtime

Dictionary

Properties	Methods
CompareMode Count HashVal Item Key	Add Exists GetEnumerator Items Keys Remove RemoveAll

Encoder

Properties	Methods
	EncodeScriptFile

FileSystemObject

Properties	Methods
Drives	BuildPath CopyFile CopyFolder CreateFolder

Properties	Methods
	CreateTextFile DeleteFile DeleteFolder DriveExists FileExists FolderExists GetAbsolutePathName GetBaseName GetDrive GetDriveName GetExtensionName GetFile GetFileName GetFileVersion GetFolder GetParentFolderName GetSpecialFolder GetStandardStream GetTempName MoveFile MoveFolder OpenTextFile

```
var fso = new
ActiveXObject("Scripting.FileSystemObject");
var a = fso.CreateTextFile("c:\\testfile.txt", true);
a.WriteLine("This is a test.");
a.Close();
```

Drive

Properties	Methods
AvailableSpace DriveLetter DriveType FileSystem FreeSpace IsReady Path RootFolder SerialNumber ShareName TotalSize VolumeName	

Drives

Properties	Methods
Count Item	GetEnumerator

Folder

Properties	Methods
Attributes DateCreated DateLastAccessed DateLastModified Drive	Copy CreateTextFile Delete Move

Properties	Methods
Files IsRootFolder Name ParentFolder Path ShortName ShortPath Size SubFolders Type	

Folders

Properties	Methods
Count Item	Add GetEnumerator

File

Properties	Methods
Attributes DateCreated DateLastAccessed DateLastModified Drive Name ParentFolder Path ShortName ShortPath Size Type	Copy Delete Move OpenAsTextStream

```
function ShowFileInfo(filespec)
{
  var fso, f, s;
  fso = new
ActiveXObject("Scripting.FileSystemObject");
  f = fso.GetFile(filespec);
  s = f.DateCreated;
  return(s);
}
```

Files

Properties	Methods
Count Item	GetEnumerator

TextStream

Properties	Methods
AtEndOfLine AtEndOfStream Column Line	Close Read ReadAll ReadLine Skip SkipLine Write

Properties	Methods
	WriteBlankLines WriteLine

```
var fso = new
ActiveXObject("Scripting.FileSystemObject");
// Second Arg: Overwrite?
var ts =
fso.CreateTextFile("C:\\Data\\Temp\\WshOutputExample.
txt", true);
ts.WriteLine("This is a test.");
ts.Close();
```

Source

John Bentley formatted into Word
johnny_bentley@yahoo.com.au

Microsoft > MSDN > Inside Microsoft Office 2000 > September 2001 > An Introduction to the Scripting Runtime Object Library

Microsoft > MSDN > Scripting > SDK Documentation > Windows Script Technologies

CC BY-NC-SA 4.0

Template

Example

```
<?xml version="1.0" encoding="utf-8" ?>
<!-- Windows Script Host Template -->
<!--**************************************************************************70**************87-->
<package xmlns="urn:johnbentley:wshMetaAsInMsdnDocumentation">
  <job id="Job1">
    <?job debug="false" error="true"?>

    <runtime>
      <description>
        Performs various operations with the Windows Script Host
        Object Model.
        Serves to demonstrate programming techniques.
      </description>
      <named
        name="tool"
        helpstring="Dummy named string argument."
        required="true"
        type="string" />
      <named
        name="colour"
        helpstring="Dummy named boolean argument."
        type="boolean" />
      <named
        name="fullOutput"
        helpstring="Dummy named simple arguement."
        type="simple" />
      <unnamed
        name="anything"
        helpstring="Anything you like"
        many="false" />
      <example>
        Example:
        cscript WshExplore.wsf /colour+ /tool:hammer /fullOutput mouse
      </example>
    </runtime>

    <!-- Reference External Library 1: <reference> markup
        Just refer to any programmatic identifier of a COM class.
        Use this for refering to constants_in the type library.
        Eg This makes the ADODB Type Library Available -->
    <reference object="ADODB.connection" />

    <object id="rst" progid="ADODB.Recordset" />
    <!--<object id="WshShell" progid="WScript.Shell" />-->

    <script
src="C:\\Data\\Dev\\Code\\EcmaScript\\Libraries\\StandardLibrary\\WindowsScriptHostLibrary\\PopupConstants.js" />

    <script type="text/ecmascript" >
      <![CDATA[
        // ******************
        //  Windows Script Host Object Model
        // ******************

// Event handler.
// Fires when field value is set (not when rst.update called)
// "rst_" defined in CreateObject Method bellow.
function rst_FieldChangeComplete() {
  WScript.Echo("Field Change Complete");
}

// Object Referencing
// Object Use
// WScript already exists.
WScript.Echo("Hello World");
// WshShell needs to be created before using the
// WshShell.Exec Method.
WScript.Echo("Current Directory: "
  + WScript.CreateObject("WScript.Shell").CurrentDirectory);

// Object Assignment
// Programmatic Id: ActiveXObject
//var rst = new ActiveXObject("ADODB.Recordset");
//var WshShell = new ActiveXObject("WScript.Shell");

// Programmatic Id: CreateObject
var rst = WScript.CreateObject("ADODB.Recordset", "rst_");
var WshShell = WScript.CreateObject("WScript.Shell");

// Non Programmatic Id: Direct
var ws = WScript;

// Non Programmatic Id: Property
var namedArgs = WScript.Arguments.Named;

// Non Programmatic Id: Ad Hoc Pseudo Constructor
var WshShortcut
  = WshShell.CreateShortcut("C:\\Data\\CoolCut.lnk");
WshShortcut.TargetPath = "C:\\Data\\Temp";
WshShortcut.Save();

// Proof of object assignment working.
// ...WScript
ws.Echo("Hello World");

// ... Named Arguments
WScript.Echo("Tool: " + namedArgs("tool"));

// ... Using Shell
WScript.Echo("Current Directory: "
  + WshShell.CurrentDirectory);

// ... Using ADO

// Change this path to your copy of Northwind.mdb
// (Installed with MS Access)
var accessDBPath
  = "C:\\Data\\Dev\\Code\\Access\\Northwind\\Northwind.mdb";

var cnn = WScript.CreateObject("ADODB.Connection");
cnn.Open("Provider=Microsoft.Jet.OLEDB.4.0;Data Source="
  + accessDBPath
```

```
    + ";Persist Security Info=False");
  rst.Open("Employees", cnn, adOpenKeyset, adLockOptimistic,
    adCmdTable);
  rst.MoveFirst;
  WScript.Echo("The Second Field Value is "  + rst.Fields(1));
  rst.close;
  cnn.close;

  // WScript
  WScript.Echo("Hello World Again");

  // WshArguments
  WScript.Echo("");
  var args = WScript.Arguments;
  for (i = 0; i < args.Length; i++) {
    WScript.Echo("arg " + i + ": " + args(i));
  }

  // WshNamed
  WScript.Echo("");
  var namedArgs = WScript.Arguments.Named;
  WScript.Echo("Named Arg /tool: " + namedArgs("tool"));
  WScript.Echo("Named Arg /colour: " + namedArgs("colour"));
  WScript.Echo("Named Arg /fullOutput Exist?: "
    + namedArgs.Exists("fullOutput"));

  // WshUnnamed
  WScript.Echo("");
  var unnamedArgs = WScript.Arguments.Unnamed;
  for (i = 0; i < unnamedArgs.Length; i++) {
    WScript.Echo("Unnamed Arg " + i + ": " + unnamedArgs(i));
  }

  // WshShell
  WScript.Echo("");
  WScript.Echo("Shell Registry Read");
  //var WshShell = WScript.CreateObject("WScript.Shell");
  //var WshShell = new ActiveXObject("WScript.Shell");
  WScript.Echo(WshShell.RegRead("HKEY_LOCAL_MACHINE\\SOFTWARE\\Microsoft\\Windows Scripting
Host\\Locations\\CScript"));
  //WshShell.Exec("calc");


  //WshSpecialFolders
  WScript.Echo("");
  var WshSpecialFolders = WshShell.SpecialFolders;
  WScript.Echo("My Documents Path is: "
    + WshSpecialFolders("MyDocuments"));
  for (i = 0; i < WshSpecialFolders.Length; i++) {
    WScript.Echo("Special Folder " + i + ": "
      + WshSpecialFolders(i));
  }
  WScript.Echo("");
  var WshSpecialFolder = WshShell.SpecialFolders("Programs");
  WScript.Echo("Start Menu Programs Path is: "
    + WshSpecialFolder);

  // Poup box
  WshShell.Popup("All Done", 7, WScript.ScriptName,
    PopupTypeButtonOK + PopupTypeIconInformation);

  // ******************
  // Scripting Runtime
  // ******************
  // Text Stream
  var fso = new ActiveXObject("Scripting.FileSystemObject");
  // Second Arg: Overwrite?
  var ts = fso.CreateTextFile("C:\\Data\\Temp\\WshOutputExample.txt", true);
  ts.WriteLine("This is a test.");
  ts.Close();

 ]]>
</script>
</job>
</package>
```

Document Licence

ECMA Script - In Windows Script Host - Quick Reference © 2021 by John Bentley is licensed under Attribution-NonCommercial-ShareAlike 4.0 International