JScript Cheat Sheet

A quick reference guide for JScript, Microsoft's implementation of ECMAScript, used for Windows scripting. This cheat sheet covers essential syntax, objects, and techniques for automating tasks in Windows environments.

JScript Fundamentals

Basic Syntax

Variables	`var variableName = value;` Declares a variable. JScript is loosely typed.
Data Types	Number, String, Boolean, Object, Null, Undefined
Comments	`//` Single-line comment `/* ... */` Multi-line comment
Operators	Arithmetic (+, -, *, /, %), Assignment (=, +=, -=), Comparison (==, !=, >, <, >=, <=), Logical (&&, ||, !)

Conditional Statements	`if (condition) { ... } else { ... }`
	`switch (expression) { case value: ... break; default: ... }`

Loops	`for (initialization; condition; increment) { ... }`
	`while (condition) { ... }`
	`do { ... } while (condition);`

Functions

Function Declaration	`function functionName(parameter1, parameter2) { ... return value; }`
Function Call	`functionName(argument1, argument2);`
Anonymous Function	`var myFunction = function(parameter) { ... };`
Return Statement	`return value;` Returns a value from the function.
Arguments Object	Access function arguments using the `arguments` object.

Working with Objects

Object Creation

Object Literal	`var myObject = { property1: value1, property2: value2 };`

Constructor Function	`function MyObject(property1, property2) {`
	`  this.property1 = property1;`
	`  this.property2 = property2;`
	`}`
	`var myObject = new MyObject(value1, value2);`

Adding Properties	`myObject.newProperty = value;`
Accessing Properties	`myObject.property1` or `myObject["property1"]`

Built-in Objects

WScript

Provides methods and properties for interacting with the Windows Script Host.

- `WScript.Echo(message)` - Displays a message box.
- `WScript.CreateObject(progID)` - Creates an instance of a COM object.
- `WScript.Arguments` - Collection of command-line arguments.

FileSystemObject (FSO)

Provides access to the file system.

- `FSO.CreateTextFile(filename, overwrite)` - Creates a text file.
- `FSO.OpenTextFile(filename, iomode, create, format)` - Opens a text file.
- `FSO.FolderExists(folderpath)` - Checks if a folder exists.
- `FSO.FileExists(filepath)` - Checks if a file exists.

Shell Object

Provides methods for interacting with the Windows shell.

- `Shell.Run(command, windowStyle, waitOnReturn)` - Executes a command.
- `Shell.ExpandEnvironmentStrings(string)` - Expands environment variables.
- `Shell.RegWrite(key, value, type)` - Writes to the registry.

Advanced JScript Techniques

Error Handling

Try...Catch...Finally	`try {`
	`  // Code that might throw an error`
	`} catch (error) {`
	`  // Handle the error`
	`  WScript.Echo(error.message);`
	`} finally {`
	`  // Code that executes regardless of error`
	`}`

Error Object	The `error` object in the `catch` block contains information about the error. • `error.number` - Error code. • `error.description` - Error message.

Working with Arrays

Array Creation	`var myArray = [value1, value2, value3];` or `var myArray = new Array(value1, value2, value3);`
Accessing Elements	`myArray[index]` (index starts at 0)
Array Methods	• `myArray.push(value)` - Adds an element to the end of the array. • `myArray.pop()` - Removes the last element from the array. • `myArray.length` - Returns the number of elements in the array.

Regular Expressions

Creating a RegExp Object	`var myRegExp = /pattern/flags;` or `var myRegExp = new RegExp("pattern", "flags");`
RegExp Methods	• `myRegExp.test(string)` - Returns true if the string matches the pattern. • `myRegExp.exec(string)` - Returns an array containing the matched text.
Common Flags	• `i` - Case-insensitive. • `g` - Global match (find all occurrences). • `m` - Multiline.

JScript and WSH

WSH Scripting

JScript files are executed by the Windows Script Host (WSH). WSH provides a runtime environment for executing scripts written in various scripting languages, including JScript and VBScript.

Scripts are typically saved with a `.js` extension and can be executed by double-clicking them or by using the `wscript.exe` or `cscript.exe` command-line interpreters.

Example Scripts

Displaying a Message Box

```
WScript.Echo("Hello, World!");
```

Reading a Text File

```
var FSO = WScript.CreateObject("Scripting.FileSystemObject");
var file = FSO.OpenTextFile("C:\\example.txt", 1); // 1 = ForReading
var content = file.ReadAll();
file.Close();
WScript.Echo(content);
```

Creating a Folder

```
var FSO = WScript.CreateObject("Scripting.FileSystemObject");
var folderPath = "C:\\NewFolder";
if (!FSO.FolderExists(folderPath)) {
  FSO.CreateFolder(folderPath);
  WScript.Echo("Folder created successfully!");
} else {
  WScript.Echo("Folder already exists.");
}
```

Running an External Program

```
var Shell = WScript.CreateObject("WScript.Shell");
Shell.Run("notepad.exe");
```