#

# Configuration File for JavaScript Lint

#

# This configuration file can be used to lint a collection of scripts, or to enable

# or disable warnings for scripts that are linted via the command line.

#

### Warnings

# Enable or disable warnings based on requirements.

# Use "+WarningName" to display or "-WarningName" to suppress.

#

+ambiguous\_else\_stmt # the else statement could be matched with one of multiple if statements (use curly braces to indicate intent

+ambiguous\_nested\_stmt # block statements containing block statements should use curly braces to resolve ambiguity

+ambiguous\_newline # unexpected end of line; it is ambiguous whether these lines are part of the same statement

+anon\_no\_return\_value # anonymous function does not always return value

+assign\_to\_function\_call # assignment to a function call

-block\_without\_braces # block statement without curly braces

+comma\_separated\_stmts # multiple statements separated by commas (use semicolons?)

+comparison\_type\_conv # comparisons against null, 0, true, false, or an empty string allowing implicit type conversion (use === or !==)

+default\_not\_at\_end # the default case is not at the end of the switch statement

+dup\_option\_explicit # duplicate "option explicit" control comment

+duplicate\_case\_in\_switch # duplicate case in switch statement

+duplicate\_formal # duplicate formal argument {name}

+empty\_statement # empty statement or extra semicolon

+identifier\_hides\_another # identifer {name} hides an identifier in a parent scope

-inc\_dec\_within\_stmt # increment (++) and decrement (--) operators used as part of greater statement

+incorrect\_version # Expected /\*jsl:content-type\*/ control comment. The script was parsed with the wrong version.

+invalid\_fallthru # unexpected "fallthru" control comment

+invalid\_pass # unexpected "pass" control comment

+jsl\_cc\_not\_understood # couldn't understand control comment using /\*jsl:keyword\*/ syntax

+leading\_decimal\_point # leading decimal point may indicate a number or an object member

+legacy\_cc\_not\_understood # couldn't understand control comment using /\*@keyword@\*/ syntax

+meaningless\_block # meaningless block; curly braces have no impact

+mismatch\_ctrl\_comments # mismatched control comment; "ignore" and "end" control comments must have a one-to-one correspondence

+misplaced\_regex # regular expressions should be preceded by a left parenthesis, assignment, colon, or comma

+missing\_break # missing break statement

+missing\_break\_for\_last\_case # missing break statement for last case in switch

+missing\_default\_case # missing default case in switch statement

+missing\_option\_explicit # the "option explicit" control comment is missing

+missing\_semicolon # missing semicolon

+missing\_semicolon\_for\_lambda # missing semicolon for lambda assignment

+multiple\_plus\_minus # unknown order of operations for successive plus (e.g. x+++y) or minus (e.g. x---y) signs

+nested\_comment # nested comment

+no\_return\_value # function {name} does not always return a value

+octal\_number # leading zeros make an octal number

+parseint\_missing\_radix # parseInt missing radix parameter

+partial\_option\_explicit # the "option explicit" control comment, if used, must be in the first script tag

+redeclared\_var # redeclaration of {name}

+trailing\_comma\_in\_array # extra comma is not recommended in array initializers

+trailing\_decimal\_point # trailing decimal point may indicate a number or an object member

+undeclared\_identifier # undeclared identifier: {name}

+unreachable\_code # unreachable code

-unreferenced\_argument # argument declared but never referenced: {name}

-unreferenced\_function # function is declared but never referenced: {name}

+unreferenced\_variable # variable is declared but never referenced: {name}

+unsupported\_version # JavaScript {version} is not supported

+use\_of\_label # use of label

+useless\_assign # useless assignment

+useless\_comparison # useless comparison; comparing identical expressions

-useless\_quotes # the quotation marks are unnecessary

+useless\_void # use of the void type may be unnecessary (void is always undefined)

+var\_hides\_arg # variable {name} hides argument

+want\_assign\_or\_call # expected an assignment or function call

+with\_statement # with statement hides undeclared variables; use temporary variable instead

### Output format

# Customize the format of the error message.

# \_\_FILE\_\_ indicates current file path

# \_\_FILENAME\_\_ indicates current file name

# \_\_LINE\_\_ indicates current line

# \_\_COL\_\_ indicates current column

# \_\_ERROR\_\_ indicates error message (\_\_ERROR\_PREFIX\_\_: \_\_ERROR\_MSG\_\_)

# \_\_ERROR\_NAME\_\_ indicates error name (used in configuration file)

# \_\_ERROR\_PREFIX\_\_ indicates error prefix

# \_\_ERROR\_MSG\_\_ indicates error message

#

# For machine-friendly output, the output format can be prefixed with

# "encode:". If specified, all items will be encoded with C-slashes.

#

# Visual Studio syntax (default):

+output-format \_\_FILE\_\_(\_\_LINE\_\_): \_\_ERROR\_\_

# Alternative syntax:

#+output-format \_\_FILE\_\_:\_\_LINE\_\_: \_\_ERROR\_\_

### Context

# Show the in-line position of the error.

# Use "+context" to display or "-context" to suppress.

#

+context

### Control Comments

# Both JavaScript Lint and the JScript interpreter confuse each other with the syntax for

# the /\*@keyword@\*/ control comments and JScript conditional comments. (The latter is

# enabled in JScript with @cc\_on@). The /\*jsl:keyword\*/ syntax is preferred for this reason,

# although legacy control comments are enabled by default for backward compatibility.

#

-legacy\_control\_comments

### Defining identifiers

# By default, "option explicit" is enabled on a per-file basis.

# To enable this for all files, use "+always\_use\_option\_explicit"

-always\_use\_option\_explicit

# Define certain identifiers of which the lint is not aware.

# (Use this in conjunction with the "undeclared identifier" warning.)

#

# Common uses for webpages might be:

+define \_\_dirname

+define clearInterval

+define clearTimeout

+define console

+define exports

+define global

+define process

+define require

+define setInterval

+define setTimeout

+define Buffer

+define JSON

+define Math

### JavaScript Version

# To change the default JavaScript version:

#+default-type text/javascript;version=1.5

#+default-type text/javascript;e4x=1

### Files

# Specify which files to lint

# Use "+recurse" to enable recursion (disabled by default).

# To add a set of files, use "+process FileName", "+process Folder\Path\\*.js",

# or "+process Folder\Path\\*.htm".

#