

xsl:stylesheet / xsl:transform

version="version" id="name" extension-element-prefixes="list" exclude-result-prefixes="list"

import

href="URI"

Imports the contents of one style sheet into another.
Note: An imported style sheet has lower precedence than the importing style sheet

include

href="URI"

Includes the contents of one style sheet into another.
Note: An included style sheet has the same precedence as the including style sheet

param

name="name" select="expression"

Declares a local or global parameter

output

method="xml|html|text|name" version="string" encoding="string" omit-xml-declaration="yes|no"
standalone="yes|no" doctype-public="string" doctype-system="string" cdata-section-elements="namelist"
indent="yes|no" media-type="string"

Defines the format of the output document

variable

name="name" select="expression"

Declares a local or global variable

decimal-format

name="name" decimal-separator="char" grouping-separator="char"
infinity="string" minus-string="char" NaN="string" percent="char"
per-mille="char" zero-digit="char" digit="char" pattern-separator="char"

Defines the characters and symbols to be used when converting numbers into strings, with the format-number() function

namespace-alias

stylesheet-prefix="prefix" (# default)
result-prefix="prefix" (# default)

Replaces a namespace in the style sheet to a different namespace in the output

attribute-set

name="name" use-attribute-sets="name-list"

Defines a named set of attributes

attribute

name="attributename" namespace="URI"

Adds an attribute

key

name="name" match="pattern" use="expression"

Declares a named key that can be used in the style sheet with the key() function

preserve-space

elements="list-of-element-names"

Defines the elements for which white space should be preserved

strip-space

elements="list-of-element-names"

Defines the elements for which white space should be removed

xsl:stylesheet / xsl:transform

version="version" id="name" extension-element-prefixes="list" exclude-result-prefixes="list"

template

name="name" match="pattern" mode="mode" priority="number"

Rules to apply when a specified node is matched

param

name="name" select="expression"

Declares a local or global parameter

copy-of

select="expression"

Creates a copy of the current node (with child nodes and attributes)

comment

-

Creates a comment node in the result tree

apply-imports

-

Applies a template rule from an imported style sheet

copy

use-attribute-sets="name-list"

Creates a copy of the current node (without child nodes and attributes)

variable

name="name" select="expression"

Declares a local or global variable

message

terminate="yes|no"

Writes a message to the output (used to report errors)

text

disable-output-escaping="yes|no"

Writes literal text to the output

fallback

-

Specifies an alternate code to run if the processor does not support an XSLT element

element

name="name" namespace="URI"

use-attribute-sets="use-attribute-sets"

Creates an element node in the output document

processing-instruction

name="process-name"

Writes a processing instruction to the output

if

test="expression"

Contains a template that will be applied only if a specified condition is true

template

xsl:stylesheet / xsl:transform

version="version" id="name" extension-element-prefixes="list" exclude-result-prefixes="list"

template

name="name" match="pattern" mode="mode" priority="number"

Rules to apply when a specified node is matched

choose

-
Used in conjunction with <when>
And <otherwise> to express
Multiple conditional tests

when

test="boolean-expression"

Specifies an action for the
<choose> element

otherwise

-

Specifies a default action
For the <choose> element

apply-templates

select="expression" mode="name"

Applies a template rule to the current element
or to the current element's child nodes

with-param

name="name" select="expression"

Defines the value of a parameter
to be passed into a template

sort

select="xpath-expression" lang="language-code"
data-type="text/number/qname" order="ascending/descending"
case-order="upper-first/lower-first"

Sorts the output

call-template

name="templatename"

Calls a named template

with-param

name="name" select="expression"

Defines the value of a parameter
to be passed into a template

value-of

select="expression" disable-output-escaping="yes/no"

Extracts the value of a selected node

for-each

select="expression"

Loops through each node in a specified node set

sort

select="xpath-expression" lang="language-code"
data-type="text/number/qname" order="ascending/descending"
case-order="upper-first/lower-first"

Sorts the output